October 9 to 11, 2016
Centro de Informática
Universidade Federal de Pernambuco
Recife, PE, Brazil

PROCEEDINGS OF THE 4TH SYMPOSIUM ON KNOWLEDGE DISCOVERY, MINING AND LEARNING

Renato Vimieiro, Leandro Balby Marinho, Luiz Merschmann (Eds.)

4th SYMPOSIUM ON KNOWLEDGE DISCOVERY, MINING AND LEARNING

October 9 to 11, 2016
Recife – PE – Brazil

PROCEEDINGS

Organization
Centro de Informática – CIn-UFPE
Universidade Federal de Pernambuco– UFPE

Local Organization Chair
Renato Vimieiro, UFPE

Program Committee Chairs
Leandro Balby Marinho, UFCG
Luiz Merschmann, UFOP

Steering Committee Chair
Alexandre Plastino, UFF

Support
Brazilian Computer Society – SBC

ISSN: 2318-1060

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

S989p Symposium on Knowledge Discovery, Mining and Learning (4.: 2016: Recife, PE)

Proceedings [recurso eletrônico] / 4th Symposium on Knowledge Discovery, Mining and Learning; edited by Renato Vimieiro, Leandro Balby Marinho, Luiz Merschmann – Recife: CIn, UFPE, 2016.

248 f.: il., fig.

Realizado de 09 a 11 de outubro de 2016 em Recife, PE.

Disponível em: http://cin.ufpe.br/~rv2/kdmile2016/anais-kdmile-2016.pdf

Inclui referências.

1. Mineração de dados. 2. Aprendizagem de máquina. I. Vimieiro, Renato (editor). II. Marinho, Leandro Balby (editor). III. Merschmann, Luiz (editor). IV. Título.

006.312 CDD (23. ed.) UFPE- MEI 2017

Editorial

The Symposium on Knowledge Discovery, Mining and Learning (KDMiLe) aims at integrating researchers, practitioners, developers, students and users to present theirs research results, to discuss ideas, and to exchange techniques, tools, and practical experiences – related to Data Mining and Machine Learning areas.

KDMiLe is organized alternatively in conjunction with the Brazilian Conference on Intelligent Systems (BRACIS) and the Brazilian Symposium on Databases (SBBD). In its fourth edition, KDMiLe is held in Recife, Pernambuco, from October 9th to 11th in conjunction with BRACIS.

This year's edition KDMiLe features one tutorial and one invited talk. The tutorial, titled "Graph analytics using Spark", is presented by two experts in the topic: Ana Paula Appel and Renan Souza, from IBM Research – Brazil. We invited Prof. Miguel Couceiro (CNRS – Inria Nancy Grand Est – University of Lorraine, France) to present a talk on "Extracting decision rules using version spaces in a qualitative approach to multi-attribute decision aid".

The event received in 2016 a total of 68 manuscripts, of which 31 were selected for oral presentation after a rigorous reviewing process. This corresponds to an acceptance rate of 45%. The papers are distributed into seven technical sessions, where authors will present and discuss their work with the audience.

We thank BRACIS Organization Committee for hosting KDMiLe at CIn-UFPE and also our sponsors for their valuable support. We are also grateful to the Program Committee members for carefully evaluating the submitted papers. Finally, we give our special thanks to all the authors who submitted their research work to KDMiLe and contributed to a yet another high quality edition of this ever growing event in Data Mining and Machine Learning.

Recife, October 9, 2016

Renato Vimieiro, UFPE
KDMiLe 2016 Local Organization Chair

Leandro Balby Marinho, UFCG
KDMiLe 2016 Program Committee Chair

Luiz Merschmann, UFOP
KDMiLe 2016 Program Committee Co-Chair

4th Symposium on Knowledge Discovery, Mining and Learning

October 9-11, 2016
Recife – PE – Brazil

Organization

Centro de Informática – CIn-UFPE
Universidade Federal de Pernambuco – UFPE

Support

Brazilian Computer Society – SBC

KDMiLe Steering Committee

Alexandre Plastino, UFF
André Ponce de Leon F. de Carvalho, ICMC-USP
Wagner Meira Jr., UFMG

KDMiLe 2016 Committee

Local Organization Chair
Renato Vimieiro, UFPE

Program Committee Chairs
Leandro Balby Marinho, UFCG
Luiz Merschmann, UFOP

Steering Committee Chair
Alexandre Plastino, UFF

KDMiLe Program Committee

Leandro Balby Marinho (Federal University of Campina Grande)
Wagner Meira Jr. (UFMG)
Sandra de Amo (Universidade Federal de Uberlândia)
Fabio Cozman (Universidade de Sao Paulo)
Cícero Nogueira Dos Santos (IBM Research)
Ana Paula Appel (IBM Research Brazil)
Maria Camila Nardini Barioni (Universidade Federal de Uberlândia)
Luis Zárate (PUC-MG)
Maira Gatti (IBM Research)
Ricardo Prudencio (Informatics Center, UFPE)
Rui Camacho (LIACC/FEUP University of Porto)
Solange Rezende (Universidade de São Paulo)
Leonardo Rocha (Federal University of São João Del Rei)
Ronaldo Prati (Universidade Federal do ABC - UFABC)
Andre Carvalho (ICMC - USP)
Marcilio De Souto (LIFO/University of Orleans)
Francisco De A. T. De Carvalho (Centro de Informatica - CIn/UFPE)
Nuno C. Marques (DI - FCT/UNL)
Aline Paes (Institute of Computing, Universidade Federal Fluminense)
Luiz Merschmann (Federal University of Ouro Preto)
Aurora Pozo (Federal University of Paraná)
Humberto Luiz Razente (Universidade Federal de Uberlandia - UFU)
Angelo Ciarlini (EMC Brazil R&D Center)
Julio Cesar Nievola (Pontifícia Universidade Católica do Paraná - PUCPR Programa de Pós Graduação em Informática Aplicada)
Ana L. C. Bazzan (Universidade Federal do Rio Grande do Sul)
Flavia Bernardini (Universidade Federal Fluminense (UFF) - Instituto de Ciência e Tecnologia - Departamento de Computação)
Elaine Sousa (University of Sao Paulo - ICMC/USP)
Marcio Basgalupp (ICT-UNIFESP)
Fernando Otero (University of Kent)
Marcelino Pereira (Universidade do Estado do Rio Grande do Norte - UERN)
Jose Alfredo Ferreira Costa (UFRN – Universidade Federal do Rio Grande do Norte)
Marcelo Albertini (Federal University of Uberlandia)
Alexandre Plastino (Universidade Federal Fluminense)
Edson Matsubara (UFMS)
Elaine Faria (Federal University of Uberlandia)
Marcelo Ladeira (Universidade de Brasília)
Herman Gomes (Universidade Federal de Campina Grande)
Carlos Eduardo Pires (UFCG)
Kate Revoredo (UNIRIO)
Adriana Bechara Prado (EMC Brazil R&D Center)
Adriano Veloso (UFMG)
Vasco Furtado (UNIFOR)
Gisele Pappa (UFMG)

Carlos Soares (University of Porto)
Bianca Zadrozny (IBM Research)
José Viterbo Filho (UFF)
Karin Becker (UFRGS)

External Reviewers

Christian Cesar Bones
Carlos Caminha
Eduardo Corrêa
Thiago Ferreira Covões
Tiago Cunha
Jose Gildo de Araujo Junior
Francisca Raquel De Vasconcelos Silveira
Ernani Melo
Bruno M. Nogueira
Caio Nóbrega
Ricardo Oliveira
Leandro Pasa
Rafael Pereira
Vladia Pinheiro
Rômulo Pinho
Fabio Procopio
Allan Sales
Pablo N. Da Silva
Gustavo Souza
Pedro Strecht
Mariana Tasca

Table of Contents

Um nova metodologia não-linear supervisionada para redução de dimensionalidade e visualização de observações

Vinicius Layter Xavier[1], Nelson Maculan[2]

[1] Universidade do Estado do Rio de Janeiro , Brazil
viniciuslx@gmail.com
[2] Universidade Federal do Rio de Janeiro, Brazil
maculan@cos.ufrj.br

Abstract. Este artigo descreve uma nova metodologia para redução de dimensionalidade não-linear supervisionada fundamentada em protótipos representativos de cada classe. Essa metodologia tem a vantajosa característica de gerar problemas de otimização de dimensão muito baixa, que independem do número de observações. Como os problemas de otimização são não-diferenciáveis, é proposta a substituição por alternativas aproximadas completamente diferenciáveis. Devido às características de diferenciabilidade e de baixa dimensão, torna-se viável a resolução de problemas de grande porte. Para ilustrar o funcionamento do algoritmo, resultados computacionais obtidos usando duas instâncias são apresentados. A primeira considera uma aplicação em redes sociais, contendo dados sobre ataques terroristas. A segunda ilustra a aplicação em uma grande base de dados de imagens de dígitos.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning; I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding; H.2.8 [**Database Management**]: Database Applications; I.4.2 [**IMAGE PROCESSING AND COMPUTER VISION**]: Compression (Coding)

Keywords: Nonlinear Dimensionality Reduction, Sammon mapping, Hyperbolic Smoothing, parallel computing, R project

1. INTRODUÇÃO

O principal objetivo dos métodos de redução de dimensionalidade é encontrar um conjunto menor de atributos, calculado a partir dos originais, buscando uma mínima perda de informação em relação aos dados originais [Cox and Cox 2000]. A redução no número de atributos, possibilita, entre outras vantagens, o uso de ferramentas de mineração de dados de um modo mais eficiente, viabilizando uma melhor visualização dos dados, em especial quando a redução é feita para os espaços $\mathbb{R}^2$ e $\mathbb{R}^3$.

Esse artigo propõe uma nova metodologia para redução de dimensionalidade não-linear supervisionada fundamentada em protótipos representativos de cada classe. A metodologia proposta engloba três conceitos principais que se articulam: redução de dimensionalidade, supervisão através da utilização da informação da classe à qual a observação pertence e generalização para observações novas.

A maioria dos métodos de redução de dimensionalidade processa os dados em uma única etapa, *batch process*, de modo que não possuem a capacidade de generalização para uma observação nova. Alguns métodos lineares possuem um modelo ou função explícita para o mapeamento entre os espaços de alta e baixa dimensão, permitindo uma generalização direta para uma nova observação. A extensão para o tratamento de observações novas não é uma operação direta no uso de métodos não-lineares [Lee and Verleysen 2007]. Esses fazem uso de funções auxiliares dependentes de parâmetros livres, cujas especificações podem ser não-triviais.

Dentre a variedade de métodos de redução de dimensionalidade existentes na literatura, nesse artigo aborda-se um método não-linear, método esse que não é fundamentado em uma combinação linear das variáveis originais. Em particular, faz-se uso do método de Sammon [Sammon 1969]. Devo ser enfatizado, todavia, que a metodologia proposta pode ser aplicada para outros métodos, como o escalonamento multidimensional métrico. O método de Sammon pertence a uma classe de métodos de redução de dimensionalidade cuja métrica é o critério de preservação de distâncias, pois busca minimizar as distorções entre as distâncias medidas com as observações no espaço original de alta dimensão e as distâncias medidas no espaço de baixa dimensão, preservando assim a informação da estrutura dos dados, bem como relações de vizinhanças entre as observações.

O método de Sammon é um método de redução de dimensionalidade que não utiliza qualquer informação da classe de cada observação, sendo assim é um método não supervisionado. A metodologia proposta inova ao introduzir o conceito de supervisão nesse método não-supervisionado. A informação da classe da observação é incorporada no modelo através da utilização de protótipos. Por protótipo se entende um elemento representativo de cada classe ou grupo, contendo alguma informação geométrica do grupo. Por exemplo, pode-se utilizar como protótipo: o centro de gravidade, ou seja, a média, a mediana geométrica ou uma particular observação que esteja no centro do grupo. O processo supervisionado ocorre no conjunto treinamento, composto pelas observações utilizadas no cálculo dos protótipos. Todas as observações que não foram utilizadas no conjunto de treinamento são chamadas de observações do conjunto de teste, podendo essas possuir ou não a identificação da classe na qual a observação é associada.

Na presente metodologia, inicialmente o método de Sammon é aplicado nos protótipos, gerando uma configuração de pontos em baixa dimensão contendo uma informação representativa de cada uma das classes nesse espaço reduzido. Com exatamente o mesmo princípio de preservação de distâncias e sem a necessidade de qualquer ajuste de parâmetros, uma nova formulação para redução de dimensionalidade de observações do conjunto de teste é proposta nesse artigo.

O método de Sammon possui a característica de ser não diferenciável. O método de resolução proposto nesse artigo adota uma estratégia de suavização usando uma função diferenciável de classe C^{∞} . A utilização dessa técnica, denominada suavização hiperbólica (SH), permite a utilização dos métodos mais poderosos de otimização não-linear, como por exemplo: gradiente conjugado, quase-Newton ou o método de Newton.

2. MAPEAMENTO DE SAMMON

Para a descrição formal do problema, a seguinte notação é apresentada. Suponha que haja uma matriz de distância ou de dissimilaridade simétrica, D_{nn} , calculada a partir de um conjunto de n observações, $\mathbf{z}_i, i = 1, \ldots, n$, cada uma com S atributos, ou seja, cada observação pertence a um espaço de dimensão $\mathbb{R}^S$. O processo de redução de dimensionalidade busca a representação de cada observação em um espaço de dimensão menor s , onde $s < S$.

O método de redução de dimensionalidade de Sammon em sua formulação ortodoxa consiste em obter um novo conjunto de observações, $\mathbf{x}_i, i = 1, \ldots, n$, pertencentes ao espaço de menor dimensão s , ou seja, $x_i \in R^s$ de modo que gere uma matriz de distância $\hat{D}_{nn}$, que se aproxime ao máximo da matriz de dissimilaridade D_{nn}, tendo com base a função de escalonamento a ser minimizada:

$$minimize \;\; f(\mathbf{x}) = \frac{1}{c} \sum_{i<j}^{n} \frac{(D_{ij} - \|\mathbf{x}_i - \mathbf{x}_j\|)^2}{D_{ij}} \tag{1}$$

onde é uma constante normalizadora, usualmente igual a soma dos termos da matriz de dissimilaridade $c = \sum_{i<j} D_{ij}$.

A função de escalonamento de Sammon (1968) dá mais importância à preservação da estrutura local dos dados, pois para cada par de observações há uma ponderação inversamente proporcional à distância desse par medida no espaço de alta dimensão. Dessa forma, Sammon dá uma importância relativa maior para os valores de erro associados a pequenas distâncias D_{ij} e menos importância a valores altos de D_{ij} .

A formulação de Sammon, dada pela equação (1), é definida em um espaço de dimensão ns , onde n é o número de observações e s é a dimensão do espaço reduzido. Por essa característica, o problema pode assumir um número muito grande de variáveis. Para ilustrar tal fato, considere um problema de tamanho médio com 5000 observações, a serem reduzidas para espaço com 3 dimensões. Nesse caso, tem-se 15000 variáveis a serem calculas pela resolução de um problema não-linear e não- diferenciável, tarefa que já apresenta grande dificuldade. Com o aumento do número de observações, chega-se facilmente à impossibilidade prática de resolução de grandes problemas.

3. UTILIZAÇÃO DE PROTÓTIPOS NA FORMULAÇÃO DE SAMMON

Suponha que o conjunto de observações contenha n diferentes classes. Assim, cada observação contém uma classe associada com um respectivo rótulo representado por $y_i, i = 1, \ldots, n$, onde y_i assume um dos valores $1, \ldots, k$ referentes a sua classe. Esse conjunto de observações é denominado de conjunto de treinamento. Para cada classe, um protótipo $\bar{\mathbf{z}}_\mathbf{k}$, pertencente ao espaço original de dimensão S, é calculado ou escolhido. A matriz das distâncias entre os protótipos é representada por DP_{kk} . Os protótipos no espaço de baixa dimensão são obtidos através da aplicação direta do método de Sammon, e são representados por $\bar{\mathbf{x}}$.

$$minimize \;\; f(\bar{\mathbf{x}}) = \frac{1}{c} \sum_{i<j}^{k} \frac{(\, DP_{ij} \; - \|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|)^2}{DP_{ij}} \tag{2}$$

A formulação (2), usada para a redução de dimensionalidade dos protótipos, é definida em um espaço de dimensão ks , onde k é o número de protótipos e s é a dimensão do espaço reduzido. Em contraste à anterior, essa formulação, em geral, define um problema com poucas variáveis. Esse fato pode ser ilustrado por um exemplo com um número expressivo de classes, $k = 50$, e com redução de dimensionalidade para espaço com 3 dimensões, cujo problema tem somente 150 variáveis. É importante frisar que o número de variáveis para a formulação (2) é independente do número de observações. Assim, essa relevante propriedade da formulação (2) pode viabilizar a resolução de problemas de redução de dimensionalidade de bases de dados com milhões de registros, que surgem de forma cada vez mais frequente em aplicações práticas reais.

4. REDUÇÃO DE DIMENSIONALIDADE PARA NOVAS OBSERVAÇÕES

Dada uma observação nova ou pertencente ao conjunto de teste, independentemente do conhecimento ou não de sua classe de pertinência, deseja-se reduzir a dimensionalidade utilizando o mesmo princípio de preservação de distância aplicado no mapeamento prévio dos protótipos. Nesse processo nenhuma informação de sua classe é utilizada. Para isto é adotada a notação: O índice $n + 1$ faz referência a uma observação nova genérica, sendo assim, a observação nova é representada por $\mathbf{z}_{n+1}$. O vetor de distâncias entre $\mathbf{z}_{n+1}$ e os protótipos $\bar{\mathbf{z}}_k$, ambos no espaço de alta dimensão, é representado por $\mathbf{DZ}_{n+1} = [DZ_{n+1,1}, DZ_{n+1,2}, \ldots, DZ_{n+1,k}]^T$. Com os valores calculados das distâncias e dos protótipos no espaço reduzido, $\bar{\mathbf{x}}_i, i = 1, \ldots, k$, propõe-se a seguinte função para a determinação da posição $\mathbf{x}_{n+1}$ de uma observação fora da amostra:

$$minimize \;\; FSnova(\mathbf{x}_{n+1}) \;=\; \frac{1}{c}\;\sum_{i=1}^{k}\;\frac{(\;DZ_{n+1,i}\;-\|\bar{\mathbf{x}}_i-\mathbf{x}_{n+1}\|)^2}{DZ_{n+1,i}} \tag{3}$$

As coordenadas da observação nova no espaço de baixa dimensão são calculadas pela minimização da função $FSnova(\mathbf{x}_{n+1})$. Deve ser enfatizado que a redução de dimensionalidade de cada observação é completamente independente de outra. Assim, para cada observação, tem-se um problema de otimização de baixa dimensão, com somente s variáveis, usualmente duas ou três quando o objetivo da redução de dimensionalidade é a visualização dos dados.

5. REDUÇÃO DE DIMENSIONALIDADE DAS OBSERVAÇÕES DO CONJUNTO DE TREINAMENTO

O mesmo procedimento descrito na seção acima é utilizado para a redução de dimensionalidade de cada observação do conjunto de treinamento, $\mathbf{x}_i, i = 1,\ldots,n$. É importante ressaltar que a redução de dimensionalidade de cada observação é completamente independente, seja essa observação do conjunto de treinamento ou do conjunto de teste, observação nova. Dessa forma, o processo pode ser paralelizado de uma forma muito direta: n problemas de baixa dimensão, com somente s variáveis em cada, problemas estes absolutamente independentes entre si.

É importante destacar essas duas características das formulações (2) e (3) componentes da proposta inovadora, pois viabilizam a resolução de problemas de redução de dimensionalidade com um número extremamente grande de observações, que são absolutamente intratáveis, sob ponto de vista numérico, pelos métodos de redução de dimensionalidade não-lineares congêneros.

6. SUAVIZAÇÃO HIPERBÓLICA

Nessa seção é apresentada a abordagem da Suavização Hiperbólica (SH), com o objetivo de superar a dificuldade de não-diferenciabilidade intrínseca às formulações (2) - (3). A abordagem SH tem sido aplicada com sucesso num grande número de problemas difíceis da classe NP-hard, incluindo problemas não diferenciáveis e não convexos com um grande número de mínimos locais, por exemplo: recobrimento de uma região planar [Xavier and De Oliveira 2005] ou de um corpo sólido [Venceslau et al. 2015], geometria das distâncias [Souza et al. 2011], clustering [Xavier 2010] [Xavier and Xavier 2011], Multisource Fermat-Weber [Xavier et al. 2014] e hub location [Xavier et al. 2015]. Uma revisão bibliográfica sobre aplicações bem sucedidas pode ser encontrada em [Xavier and Xavier 2013]. Dados dois pontos $\mathbf{x}_i$ e $\mathbf{x}_j$ no $\mathbb{R}^s$ e sendo u a distância euclidiana entre $\mathbf{x}_i$ e $\mathbf{x}_j$, $u = \|\mathbf{x}_i - \mathbf{x}_j\|$, u é não diferenciavel quando $\mathbf{x}_i = \mathbf{x}_j$. Para se obter diferenciabilidade é usada a função :

$$\theta(\,u\,,\,\gamma\,) \;=\; \sqrt{u^2\,+\,\gamma^2}\,. \tag{4}$$

A função θ possui muitas propriedades [Xavier and Xavier 2013], dentre elas destacam-se duas:

(*a*) $\lim\limits_{\gamma\to 0}\;\theta(\,u\,,\,\gamma\,) \;=\; |\,u\,|\,;$

(*b*) θ pertence a classe C^∞ de funções diferenciáveis.

Note que a propriedade (a) implica que θ é uma aproximação assintótica da função u . A propriedade (b) permite a utilização de métodos de otimização poderosos baseados em aproximação por série de Taylor de primeira ou de segunda ordens.

Adicionalmente a essas propriedades fundamentais, a Suavização Hiperbólica (SH) tem a propriedade de eliminar pequenos mínimos locais viabilizando a obtenção de mínimos locais profundos. Essa ocorrência foi verificada empiricamete na resolução de problemas de recobrimento, clustering, Multisource Fermat-Weber e hub location, vide refências bibliográficas supra-citadas. No caso particular do problema de geometria de distâncias, a SH tem a extraordinária propriedade de convexificação do problema original, conforme provado analiticamente em [Xavier 2003] e [Souza et al. 2011].

7. APLICANDO SUAVIZAÇÃO HIPERBÓLICA EM SAMMON

Considerando as duas formulações de MDS expressas pelas funções (2) - (3) nota-se que a não diferenciabilidade é uma propriedade comum nessas duas formulações. A proposta consiste em aplicar SH e substituir essas funções por um conjunto de problemas sucedâneos bem-comportados, completamente diferenciáveis. Com esse propósito em mente, problemas intrinsecamente não diferenciáveis são aproximados por alternativas diferenciáveis.

Aplicando a suavização hiperbólica às funções (2) e (3), se obtem:

$$minimize \;\; f(\bar{\mathbf{x}}) \;=\; \frac{1}{c}\sum_{i<j}^{k} \frac{(\,DP_{ij} \;- \theta(\|\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j\|, \gamma))^2}{DP_{ij}} \tag{5}$$

$$minimize \;\; FSnova(\mathbf{x}_{n+1}) \;=\; \frac{1}{c}\sum_{i=1}^{k} \frac{(\,DZ_{n+1,i} \;- \theta(\|\bar{\mathbf{x}}_i - \mathbf{x}_{n+1}\|, \gamma))^2}{DZ_{n+1,i}} \tag{6}$$

As formulações suavizadas (5) e (6) são definidas em espaços com as mesmas dimensões associadas as formulações (2) e (3). Assim, a implementação da proposta inovadora envolve a resolução de um primeiro problema com ks dimensões e a subsequente, resolução de N problemas com s dimensões, onde N é a soma do número de observações do conjunto de treinamento com as do conjunto de teste. Ademais, para a formulação (6) foi verificada empiricamente a propriedade de eliminação de pequenos mínimos locais.

8. RESULTADOS COMPUTACIONAIS

O método proposto foi implementado utilizando a linguagem estatística R [Team et al. 2013]. As tarefas de otimização foram realizadas por meio do método de Gradiente Conjugado, através da rotina *optim* da biblioteca *stats*. O processo de redução de dimensionalidade das observações do conjunto de treinamento e do conjunto de teste foi implementado utilizando processamento paralelo por meio da biblioteca *doSNOW*, com o processador Intel Core i7 3632QM.

Para mostrar o desempenho da metodologia proposta em redes de relacionamento, foi realizado um primeiro experimento computacional aplicando o método proposto no conjunto de dados *Terrorist Attacks*, disponível em http://linqs.umiacs.umd.edu/projects//projects/lbc/, no qual atentados terroristas são representados em um Grafo.

Este conjunto de dados consiste de 1293 ataques terroristas, classificados em 6 tipos de ataques, $k = 1, \ldots, 6$, com os seguintes rótulos de classes e seus respectivos percentuais de observações em cada classe: *Arson* (2.4%), *Bombing* (43.5%), *Kidnapping* (13.8%), *Nuclear, Biology, Chemistry and Radiology - NBCR* (0.6%), *Other attack* (1%) e *Weapon attack* (38.5%). Cada ataque é representado por um vetor com 106 atributos. Cada atributo assume um valor binário, indicando a ausência ou presença de uma particular característica. Assim, desde que o problema é definido no espaço dos

reais, temos $\mathbb{R}^{106}$. Maiores detalhes sobre esse banco de dados podem ser encontrados no artigo: Event Classification and Relationship Labeling in Affiliation Networks [Zhao et al. 2006]

Utilizando a média de cada classe como protótipo, as observações foram reduzidas para um espaço de dimensão 3. Essa redução viabiliza, em geral, uma boa visualização da rede. A Figura 1 mostra o posicionamento das observações da instância considerada após a redução segundo uma rotações no espaço. na figura, a classe*Arson* é representada em azul,*Bombing* em verde,*Kidnapping* em vermelho, *NBCR* em preto,*Other attack* em marrom e *Weapon attack* em amarelo.

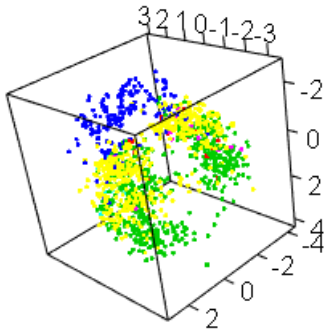


Fig. 1. Redução de Dimensionalidade, $\mathbb{R}^{106} \rightarrow \mathbb{R}^3$, para *Terrorist Attacks*.

Como mostrado pela Figura 1, a aplicação da metodologia proposta, de modo geral, produziu uma clara separação das observações segundo suas classes no espaço reduzido. As classes mais densas Bombing (verde), Kidnapping (vermelho), e Weapon Attack (amarelo) podem ser visualizadas de uma forma bem separadas, com algumas sobreposições mostrando as intersecções entre as mesmas. Nas demais três classes não é possível identificar o mesmo comportamento, pois estão de forma difusa e totalizam um percentual muito pequeno dos dados (4,18%). A Figura 2, pode ser visualizada de forma interativa permitindo rotações, no site: `https://dl.dropboxusercontent.com/u/110322274/TerrorAttack/index.html`

O segundo experimento computacional foi realizado com a base de dados MNIST [LeCun et al. 1998], base essa comumente utilizada na literatura em diversos artigos sobre métodos de reconhecimento de padrões e redução de dimensionalidade. Essa base de dados é formada por imagens de dígitos escritos à mão, contendo 60000 no conjunto de treinamento e 10000 no conjunto de teste. Cada imagem é representada por um vetor com 784 atributos, $\mathbf{z}_i \in \mathbb{R}^{784}, i = 1, \ldots, 60000$ e possui uma classe correspondente a um dígito de 0 até 9. A metodologia proposta foi aplicada em toda a base de dados, utilizando como o protótipo a média, calculada utilizando o conjunto de treinamento, e reduzindo a dimensão para o $\mathbb{R}^3$. Entretanto, no contexto de redução de dimensionalidade e visualização, por motivos computacionais, amostras são adotadas ao invés de se utilizar o conjunto inteiro da instância MNIST. [Maaten and Hinton 2008] e [Lee et al. 2013] utilizam amostras com 6000 observações.

A propriedade de escalabilidade da metodologia proposta é a seguir ilustrada com este banco de dados. Considerando somente o conjunto de treinamento, a utilização do método de Sammon para essa mesma instância compreenderia a resolução de um problema de otimização não-linear com $3 \times 60000 = 180000$ variáveis. Assumindo, para efeito de um exercício, a hipótese otimista de o problema não-linear ter a mesma complexidade da resolução de um sistema de equações lineares, em que o número de operações aritméticas é proporcional ao número de variáveis ao cubo. Assim , para esse exemplo, tem-se uma complexidade da ordem de $180000^3 = 18^3 \times 10^{12} = 5832 \times 10^{12}$ operações. Em contrapartida, o uso da metodologia proposta compreende a resolução de um primeiro problema não-linear com 10

(número de dígitos) $\times$ 3 = 30 variáveis e subsequentes 60000 problemas não-lineares com 3 variáveis. Para esse exemplo, a metodolgia inovadora compreenderia $30 \times 3 + 60000 \times 3^3 = 1647000$ operações.

Em suma, esse exercício mostra que o número de operações aritméticas da proposta inovadora é da ordem de $3,5 \times 10^9$ menor do que a aplicação do método de Sammon. Isso sem tomar em consideração as vantagens oferecidas pela resolução de um problema de otimização completamente diferenciável em comparação ao difícil problema não-diferenciável de Sammon.

A Figura 2 mostra a projeção das 70000 observações da instância completa no espaço de dimensão 3. É possível notar conjuntos densos de observações de uma mesma classe indicando uma separação adequada das classes. A Figura 2, pode ser visualizada de forma interativa permitindo rotações, no site: `https://dl.dropboxusercontent.com/u/110322274/MNIST/index.html` A posição de uma

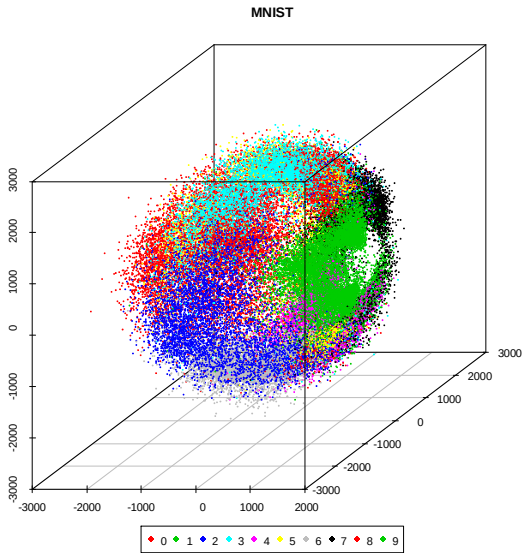


Fig. 2. Redução de Dimensionalidade, $\mathbb{R}^{784} \rightarrow \mathbb{R}^3$, para MNIST com 70000 observações

nova observação, sem a informação de classe, produzida pela aplicação da redução de dimensionalidade pode ser usada como indicador de sua classe de pertinência. Vários tipos de indicações podem ocorrer: pertinência a uma classe, quando a observação está localizada em uma região do espaço que contenha somente uma classe sendo assim uma região pura; pertinência a uma região conflituosa entre classes quando a observação está localizada entre duas classes ou mais classes; não-pertinência a uma específica classe, quando a observação está localizada em uma região do espaço que não contenha nenhuma observação da classe específica. A visualização pode ser utilizada em conjunto com um algoritmo de classificação supervisionada, permitindo uma avaliação visual do resultado da classificação feita pelo algoritmo.

9. CONCLUSÕES

Neste artigo propõe-se uma nova metodologia baseada em protótipos para problemas não-linear de redução de dimensionalidade. Utiliza-se o método de Sammon para a redução de dimensionalidade dos protótipos. O método de Sammon pertence à classe de métodos não-lineares de preservação de distâncias. Uma generalização direta para essa metodologia pode ser feita para o método de Escalonamento Multidimensional Métrico, método este pertencente a mesma classe do método de Sammon.

Considerando as coordenadas dos protótipos no espaço de baixa dimensão, propõe-se a generalização do método de Sammon para redução de dimensionalidade de uma nova observação. A regra de redução é empregada exatamente da mesma forma para as observações dos conjuntos de treinamento e de teste. Assim, a metodologia tem a importante característica de possuir unicidade no tratamento de todas observações.

A minimização das distorções entre a projeção no espaço reduzido e no espaço original dos dados não é uma tarefa trivial. A metodologia proposta tem a diferenciada característica de gerar um conjunto de N problemas separáveis de otimização, onde N faz referência ao número total de observações. Cada problema possui dimensão muito baixa, com somente s variáveis, onde s faz referencia a dimensão do espaço reduzido. A propriedade de separabilidade, por sua vez, permite a utilização de computação paralela. Na aplicação direta do método de Sammon, o número de variáveis é extremamente maior, Ns, inviabilizando sua aplicação prática em problemas de grande porte.

Ademais, como os problemas de otimização são não-diferenciáveis, com a utilização da SH é proposta a substituição por alternativas aproximadas completamente diferenciáveis. Devido às características de baixa dimensão, de paralelismo no tratamento de cada observação e de suavização, torna-se possível resolver com eficiência e precisão problemas intratáveis por outras alternativas.

A resolução do problema teste NMIST com 70000 observações, é um evento inaudito na literatura, que mostra de forma experimental uma medida do alcance da metodologia inovadora proposta nesse artigo.

REFERENCES

Cox, T. F. and Cox, M. A. *Multidimensional scaling*. CRC press, 2000.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86 (11): 2278–2324, 1998.

Lee, J. A., Renard, E., Bernard, G., Dupont, P., and Verleysen, M. Type 1 and 2 mixtures of kullback–leibler divergences as cost functions in dimensionality reduction based on similarity preservation. *Neurocomputing* vol. 112, pp. 92–108, 2013.

Lee, J. A. and Verleysen, M. *Nonlinear dimensionality reduction*. Springer Science & Business Media, 2007.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of Machine Learning Research* 9 (Nov): 2579–2605, 2008.

Sammon, J. W. A nonlinear mapping for data structure analysis. *IEEE Transactions on computers* 18 (5): 401–409, 1969.

Souza, M., Xavier, A. E., Lavor, C., and Maculan, N. Hyperbolic smoothing and penalty techniques applied to molecular structure determination. *Operations Research Letters* 39 (6): 461–465, 2011.

Team, R. C. et al. R: A language and environment for statistical computing, 2013.

Venceslau, H. M., Lubke, D. C., and Xavier, A. E. Optimal covering of solid bodies by spheres via the hyperbolic smoothing technique. *Optimization Methods and Software* 30 (2): 391–403, 2015.

Xavier, A. E. Convexificação do problema de distância geométrica através da técnica de suavização hiperbólica. In *Workshop em Biociências COPPE UFRJ*, 2003.

Xavier, A. E. The hyperbolic smoothing clustering method. *Pattern Recognition* 43 (3): 731–737, 2010.

Xavier, A. E. and De Oliveira, A. A. F. Optimal covering of plane domains by circles via hyperbolic smoothing. *Journal of Global Optimization* 31 (3): 493–504, 2005.

Xavier, A. E., Gesteira, C. M., and Xavier, V. L. Solving the continuous multiple allocation p-hub median problem by the hyperbolic smoothing approach. *Optimization* 64 (12): 2631–2647, 2015.

Xavier, A. E. and Xavier, V. L. Solving the minimum sum-of-squares clustering problem by hyperbolic smoothing and partition into boundary and gravitational regions. *Pattern Recognition* 44 (1): 70–77, 2011.

Xavier, A. E. and Xavier, V. L. Flying elephants: a general method for solving non-differentiable problems. *Journal of Heuristics*, 2013.

Xavier, V. L., França, F. M., Xavier, A. E., and Lima, P. M. A hyperbolic smoothing approach to the multisource weber problem. *Journal of Global Optimization* 60 (1): 49–58, 2014.

Zhao, B., Sen, P., and Getoor, L. Event classification and relationship labeling in affiliation networks. In *Proceedings of the Workshop on Statistical Network Analysis (SNA) at the 23rd International Conference on Machine Learning (ICML)*, 2006.

Exploiting Brazilian Economic News to Predict BM&FBOVESPA

J. G. de Araújo Jr, L. B. Marinho

Federal University of Campina Grande, Brazil
zegildo@copin.ufcg.edu.br,lbmarinho@dsc.ufcg.edu.br

Abstract. In this paper we investigate the impact of economic news for predicting, by means of machine learning algorithms, the BM&FBOVESPA index, a benchmark for the Brazilian stock exchange. We collected economic news published in high circulation newspapers in Brazil between 2000 and 2015 as well as reader comments and the number of shares on Twitter, Facebook, LinkedIn and GooglePlus. We extracted several quantitative features from the news collected and fed them as input for machine learning prediction models with the aim of predicting the BM&FBOVESPA index for the next 15 minutes. Through the experiments conducted we concluded that news carry an interesting signal for predicting the BM&FBOVESPA trends where we achieved up to 20% improvement over a random model.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: BM&FBOVESPA, Brazilian newspapers, social medias, trend detection, sentiment analysis, stock market

1. INTRODUCTION

The Stock Exchange, Commodities and São Paulo Futures Exchange (BM&FBOVESPA) is one of the most influential stock exchanges in the world, having moved 1.8 trillion dollars in 2014 and getting in the first half 2015 a net income of R$ 392 million associated to an increase of 17% compared to 2011[1]. Although smaller than other stock exchanges such as New York and the Nasdaq, many factors make BM&FBOVESPA attractive, including: steady growth trend of the market, high volatility of stock prices, indexes and dollar. Although potentially profitable, this market is scarcely explored by Brazilian citizens, being dominated by foreign investment, banks and high-frequency trading [Hagströmer and Norden 2013]. In fact, only about 10% of the Brazilian population invests in the domestic stock market[2].

Recent studies argue that through the analysis news data and social medias it is possible to elaborate smarter investment strategies and thus obtain strategic advantage in generating wealth [Schumaker and Chen 2010; Tetlock et al. 2008]. Nevertheless, many investors do not take information about economic facts into account to make their decisions, relying only on technical and graphical analysis [Bulkowski 2011]. For them, the prices of assets summarise by itself all known information about the assets, as postulated by the well known Efficient-market hypothesis (EMH) [Malkiel and Fama 1970]. The advantage of this approach is in to use only simple technical features like the price volatility and trend indicators which are easily provided by home brokers. Moreover, many home brokers are already empowered with many analysis tools to support decision making based on this kind of information.

[1] http://www.world-exchanges.org/home/index.php/statistics/monthly-reports
[2] http://www.bmfbovespa.com.br/pt_br/servicos/market-data/consultas/mercado-a-vista

On the other hand, this approach is restricted to these technical features that many times are not sufficient to anticipate facts about the stock market. When EMH was initially postulated back in 1970, it did not consider certain events that were ahead in the future such as the increasing processing power of computers, the development of machine learning techniques, the online news published in near real time and exposure of opinions of investors in social medias. These elements bring new opportunities for investigating the impact of information in the form of news and user reactions for understanding stock market dynamics.

Concerned with the potential disadvantages of graphical analysis both the scientific and business communities (e.g. Winton Capital[3]) started to investigate the impact that economic daily news published in newspapers and the opinion of their readers has on investors' decisions. They began to incorporate this information into a new segment of stock market prediction models worldwide. Figure 1, for example, illustrates the price estimation of weapons companies *Raytheon Company*[4] and *Lockheed Corporation*[5] assets on the next trading day after the attack of Paris on 11/13/2015. Techniques that take economic news analysis into account have more chance to predict such an abrupt change of price fluctuation like this than technical approaches because their models are monitoring news being published in real time while other techniques are limited to the price change which will only occur after the news. Although there is still no conclusive evidences about the importance of news for stock market prediction, many journalistic services profit from the sale of real-time economic news, e.g, Reuter[6], Bloomberg[7], Folha de São Paulo, among others, which indicates that many investors use these services on the belief that they can obtain competitive advantage through a more subjective kind of knowledge about the stock market.

In this paper we shed some light on the impact that economic news coming from Brazilian public domain newspapers have on the Brazilian stock market (BM&FBOVESPA). For that, we collected economic news published in high circulation newspapers in Brazil between 2000 and 2015 as well as reader comments and the repercussions of the news (i.e. shares) on social medias like Twitter, Facebook, LinkedIn and GooglePlus. We begin with a quantitative analysis where we try to understand the correlation between quantities such as the amount of published news as well as the sentiment of each news (i.e. positive or negative) and IBOVE (a benchmark of BM&FBOVESPA performance)[8]. Next, we developed a prediction model purely based on information extracted from the news. Our approach overcame the random model and a heuristic that predicts IBOVE based on the current IBOVE trend, thus providing a strong evidence about the correlation between economic Brazilian news and IBOVE performance.

2. PROBLEM STATEMENT

As mentioned in the previous section, in this paper we investigate the impact of economic Brazilian news on the BM&FBOVESPA performance. As a proxy of BM&FBOVESPA performance, we used IBOVE, which indicates the average performance of the most important and negotiated assets of the Brazilian stock market. IBOVE can be analysed based on four main features, which will be used as our prediction targets: *average price*, *amount of trades*, *amount of contracts traded* and *financial volume*. Notice that although these are numeric features, which would lead to a regression problem, in practice, investors often only need to know about IBOVE changes of growth. Therefore, we transformed each numeric feature into a three level categorical feature where the levels are: *decrease*, *unchanged*, and *increase*. Now, for each target class, and given a timestamp t, we want to predict whether it will

[3] https://www.wintoncapital.com
[4] http://www.raytheon.com/
[5] http://www.lockheedmartin.com/
[6] http://www.reuters.com/
[7] http://www.bloomberg.com/
[8] http://www.bmfbovespa.com.br/pt_br/produtos/indices/indices-amplos/

Fig. 1. Price time series of some weapon branch companies after the attacks in Paris 11/13/2015.

increase, stay the same, or increase on timestamp $t+1$. The prediction models must be solely based on news features and its effectiveness will be measured based on its capacity to make better predictions than a random model, i.e., a model where the outcomes have the same probability of being predicted.

More formally, we have a multi-class classification problem. Multi-class classification typically considers a set of m-dimensional feature vectors $X \in R^m$, a set of classes Y, and a training set of the form $D^{train} = \{(\vec{x}_1,y_1),\ldots,(\vec{x}_n,y_n)\}$ where $\vec{x} \in X$ is a vector of attributes and $y_i \in Y$ represents the class which $\vec{x}_i$ is associated with. The idea is to find a classification function $\hat{y} : X \rightarrow Y$ that minimises the error in the test set $D^{test} = \{(\vec{x}_1,y_1),\ldots,(\vec{x}_p,y_p)\}$, that is unavailable during training, i.e., $D^{test} \cap D^{train} = \emptyset$. More formally, the goal is to minimise:

$$err(\hat{y}; D^{test}) = \frac{1}{|D^{test}|} \sum_{(\vec{x},y) \in D^{test}} l(y, \hat{y}(\vec{x})) \quad (1)$$

where $l : Y \times Y \rightarrow R$ is a loss function measuring, for any test instance $(\vec{x}, y) \in D^{test}$, the misfit between the true y and the predicted value $\hat{y}(\vec{x})$. We instantiate this setting as follows. We have four target sets, i.e., $Y_1, Y_2, \ldots, Y_4$ where each $Y_i = \{0, 1, 2\}$ represents the i-th target feature with the corresponding levels (i.e., decrease (0), unchanged (1) and increase(2)). Thus, notice that now we have to build a separate multi-class classifier for each different target variable.

3. RELATED WORK

Several works have already studied the interplay between news and the stock market dynamics. The work of [Al Nasseri et al. 2015] describes *The StockTwiter*[9], an intelligent system that combines text mining techniques and decision tree algorithms to extract sentiment information from Twitter and use it to make predictions about the stock market. This system uses semantic analysis in tweets to infer specific feelings about the market related to the concepts *buy*, *sell* or *hold*. This approach was used to predict trends of the Dow Jones index and presented better results than random models. Although we do not explicitly exploit twitter data in this work, we investigated yet unexploited news features from many different yet unexploited social media sites such as like Facebook, LinkedIn, GooglePlus.

The authors of [Feldman et al. 2011] proposed *The StockSonar*[10], a system that employs sentiment

[9]http://stocktwits.com/
[10]www.thestocksonar.com

analysis on news raw text to understand and predict the stock market dynamics. The system is able to recognise news about dealing agreements, lawsuits, new products descriptions, and other economic events and automatically suggest updated price forecast about specific assets. While this approach predicts prices only about specific assets, our work predicts tendencies about the stock market as a whole.

The main differences between this work and the aforementioned ones are as follows. We extracted features from several previously unexploited sources, such as Facebook, LinkedIn and GooglePlus. We used the repercussion of news (e.g. number of shares), which is another previously unexploited feature, in each of the aforementioned social media sites as predictors of classification models. While the main goal of the cited papers was to develop systems based on the assumption that news are useful for understanding the financial American market, our main goal is to verify whether economic Brazilian news impacts on the BM&FBOVESPA.

4. DATA ANALYSIS

For assembling the experimental data set we collected 471, 430 news from the following Brazilian news portals: G1 (185, 733), Folha de São Paulo (113, 671) and Estadão (172, 026). This data was collected between 2000 and 2015 and these portals are among the most accessed in Brazil. For each news we collected the *timestamp* of the publication, *title* and *subtitle*, *textual content*, the corresponding *newspaper name*, the amount of *repercussion* on Facebook, Twitter, LinkedIn, GooglePlus, and the amount of *comments* on the page of the news. We define *repercussion* as the amount of shares of the news link on any of social media analysed as well as the amount of comments received on the page itself. The IBOVE target variables information were obtained directly from the BM&FBOVESPA portal [11].

4.1 Descriptive Analysis

The right hand side of Figure 2 depicts the probability density of the amount of news per day during the data collection time period for all the newspapers considered. Note that while Folha de São Paulo and Estadão present a similar behaviour in terms of the number of news published everyday, G1 presents a different distribution where an extreme amount of news (too many or too few) is published everyday. In general, G1 publish four times more daily news than Folha de São Paulo and Estadão.

For all newspapers it was verified that the number of published news has been decreasing over the years while the amount of shares on social media has been increasing (see left hand side of Figure 2). This fact emphasises that readers of these newspapers are used to share economic news many times a day. This is a strong evidence that the amount of shares of a given news may represent an important signal about the importance of this news and its potential impact on the stock market dynamics.

4.2 Sentiment Analysis

While many sentiment analysis methods are concerned with detecting the feelings associated with a given text such as: love, hate, happiness, anguish, among others, this work uses sentiment analysis to classify a text as being positive or negative (or neutral when it is not possible to detect its polarity). To some extent, the classification of the polarity of a text document is subjective and depend on its context. For example, consider the following headline: *"Dollar rises 1% and closes worth R$3,943, keep an eye in China and the political scene"*. Read by a dollar investor this news title is surely positive since his/her investment has increased in value. On the other hand, in a general economic context, a rising dollar comes accompanied by an increase in bread price, fuel, bus tickets, medicines

[11] `http://www.bmfbovespa.com.br/pt\_br/servicos/market-data/historico/`

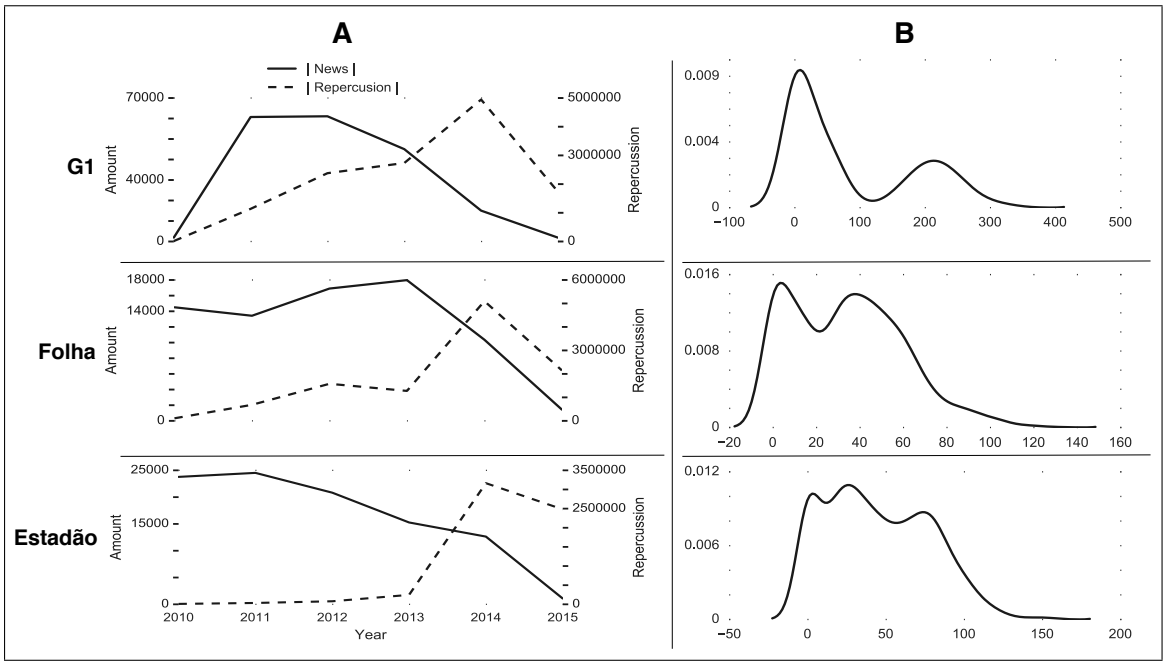


Fig. 2. Part **A** represents the behavior between the amount of published news and the amount of repercussion of each newspaper analysed. Part **B** represents the probability density of the amount of daily published news of each newspaper.

and all imported elements, being also possible to regard it as a negative news. In this perspective, we understood that the ideal classification news should take into consideration what its information reflects in a general economic context of Brazilian reality.

We used machine learning-based sentiment analyses methods for inferring the polarity of the news in our experimental data set. We first needed to label the news as positive, negative or neutral. For doing that we invited a group of 20 Computer Science students from the Federal university of Campina Grande enrolled at the Data Analysis course to label a sample of 200 news. All of them were instructed to label news based on their interpretation about the polarity of the news with respect to the Brazilian economy. This sample was then divided in chunks of 10/15 news and given to each student for labelling. The same chunk was purposely delivered to different students. News classified differently by different students were discarded as recommended by the Delphi Analysis [Linstone et al. 1975]. After this process we ended up with 165 manually labelled news. Around 82% of the students classified the same news equally which indicates that the polarity of the news to a great extent consensual among the labellers.

After that we submitted all labelled news to be evaluated by 19 sentiment analysis state-of-the-art classifiers present in the iFeel[12] tool. Given that many of the methods are designed for the English language, all news in our data set had to be translated to English. According to the results presented by [Araújo et al. 2016] this is an efficient strategy because many sentiment analysis methods are based on sentiment lexicons and these lexicons do not change much after translation. Among the evaluated methods, the Vader [Hutto and Gilbert 2014] method was the one that showed better results and performance with 73% of accuracy. Then we used this method to classify all the other news in the data set.

We found that there is ~ 4.5 times more positive economic news than negative in all considered time granularities (year, month, week) in all newspapers. Part A of Figure 4 depicts the proportion

[12]http://blackbird.dcc.ufmg.br:1210/

6 ·

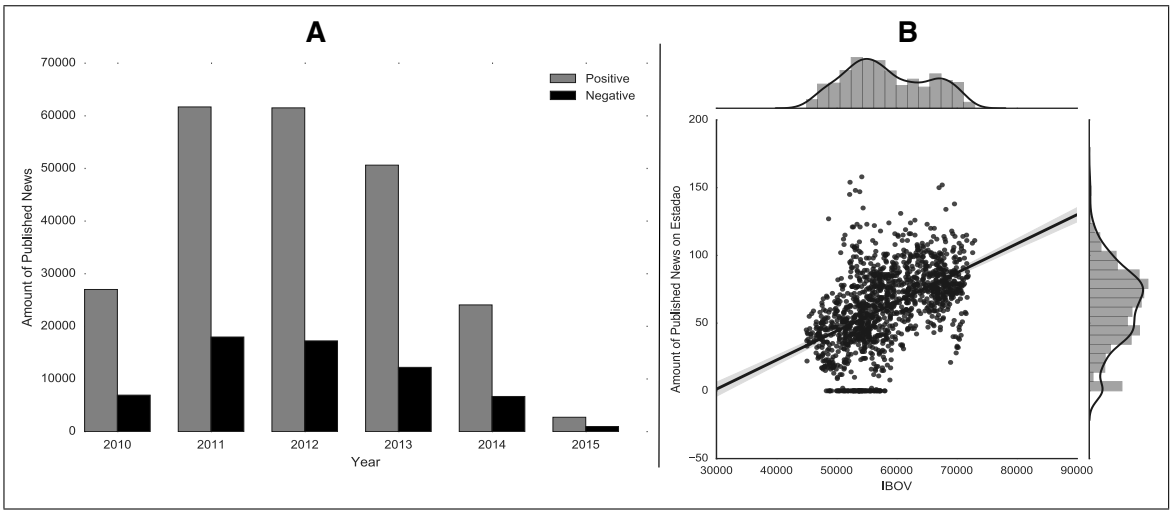


Fig. 3. Part **A** compares the amount of positive and negative news over the years. Part **B** represents the correlation analysis between IBOVE and the amount of published news day-to-day.

between positive and negative news over the years. We also find that negative news are more shared than positive news in all social medias. Finally, we find that in years of presidential elections (2002, 2006, 2010 e 2014) all newspapers decrease the amount of negative economic news. G1 and Folha de São Paulo published approximately 50% less negative news while Estadão 25%.

4.3 Feature Analysis

We selected seven quantitative features (in terms of counts) as candidates of predictors, i.e., amounts of: published news, comments made about each news, shares on Facebook, Twitter, LinkedIn, Google-Plus, positive and negative news and the balance of polarity (the difference between the amounts of good and bad news). We then analysed the correlation between each of the aforementioned features with the IBOVE value over the years. The days when there were no news and non-operating days (Saturdays, Sundays and holidays) were discarded. This analysis helped us to detect the most promising features to use in the predictive models.

Each cell of Table I shows the Kendall correlation values (τ) between IBOVE time series and each feature collected. Unlike to Pearson and Spearman correlations, the Kendall correlation is not affected by how far from each other ranks are but only by whether the ranks between observations are equal or not. We can see that Estadão is the newspaper with the highest correlation with IBOVE for all features considered. Part B of Figure 4 shows the correlation between the amount of daily published news by Estadão and IBOVE values. In contrast, G1 had the lowest correlations with many results close to zero. This means that economic news published on G1 are not correlated with the performance of IBOVE. Notice that the correlation between IBOVE values and shared news on social medias was inversely proportional in all newspapers. That is, when the index value falls, the amount of economic news shared increases (and vice-versa). Observing the Table I it is also possible to conclude that Facebook is the social media with the highest correlation value with IBOVE performance.

5. PREDICTION MODELS

In order to explore new features we expanded our initial features set with the mean, median, standard deviation, variance and the sum of each feature. Next, it was necessary to find to which granularity of time the features showed the best correlation with respect to the target variables. For this, we

	Amount of								
	News		*Social Medias*				*Polarity*		
Newspapers	Published	Comments	Twitter	Facebook	LinkedIn	Google Plus	Positive	Negative	Positive - Negative
G1	-0.03	-	-0.23	**-0.42**	-0.16	-	-0.001	-0.08	0.01
Folha	0.12	0.17	-0.25	-0.39	-0.24	-	0.15	0.013	0.16
Estadão	**0.37**	-	**-0.42**	-0.41	**-0.34**	**-0.46**	**0.38**	**0.24**	**0.34**
All	0.33	0.17	0.16	0.14	0.12	-	0.11	0.021	0.13

Table I. Correlation values between IBOVE historical time series and candidate features.

measured the correlation between each target and the features in three different granularities of time: 15 minutes, 1 hour and 1 day. Each granularity contains the aggregation of all features extracted from the news that occurred in the corresponding time period. For instance, the feature *amount of repercussion on Facebook* at the granularity of 15 minutes will have the sum of all shares on Facebook for all news published for each 15 minutes time window. We measured the correlation between each summarised feature in each granularity with IBOVE performance and we found that the best correlation values were found for 15 minutes. Thus, we seek to build models for predicting IBOVE for the next 15 minutes of operation.

The training data is now composed as follows. Each instance corresponds to the aggregated values of each feature and the target is the IBOVE value for the next 15 minutes (for each target). Figure 4 illustrates this process.

After aggregating all feature values in periods of 15 minutes we divided all data set (423,000 instances) in 10 groups of 42,300 instances each ordered in time. We did this in order to measure the variability of the methods evaluated. We submitted each group to the following classifiers: Gaussian Naive Bayes, Decision Tree, Random Forest, Extra Trees, Adaptive Boosting and Gradient Boosting in several different configurations. For all classifiers we checked three different train and test proportions: 60/40, 70/30 and 75/25. The proportion of 70/30 presented the best results in all predictions. For the classifiers Random Forest, Extra Trees, Adaptive Boosting and Gradient Boosting depend on the amount of trees used. Hence we evaluated each one with 100, 150, 500 and 1000 trees and compared them with *Random Model* and *Keep Trend*. The *Keep Trend* method assumes that the market will maintain the current trend in the next 15 minutes, i.e., it predicts that IBOVE will maintain the current trend in the next 15 minutes.

Finally, we evaluated each classifier on the test partitions. We also applied T-test to assess the

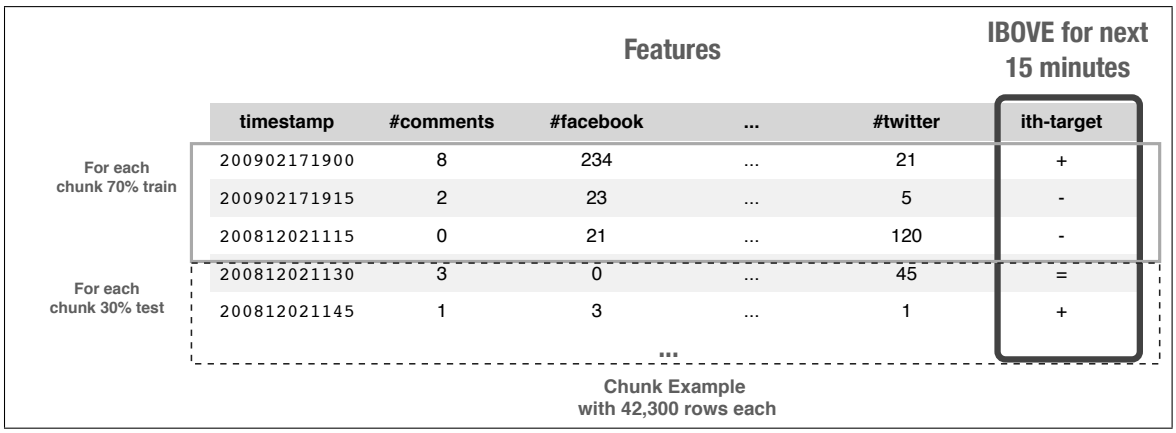


Fig. 4. Example of a train and test perform in a chunk.

8 ·

	Attributes of IBOV			
Methods	*Average*	*Amount of Business*	*Amount of Contracts*	*Financial Volume*
Random Walk	0.33	0.33	0.33	0.33
Keep Trend	0.43	0.38	0.37	0.38
Our Approach	**0.48**	**0.52**	**0.53**	**0.53**
	Extra Trees trees: 150	**Random Forest trees: 100**	**Gradient Boosting trees: 100**	**Adaptive Boost trees: 100**

Table II. Comparison of the methods and presentation of the best classifiers and their settings.

differences between each pair of means between the compared approaches. All results were statistically significant for $\alpha = 0.95$. Table II summarises the performance of the methods. Notice that our approach overcame *Random Model* and *Keep Trend* for all IBOVE targets.

6. CONCLUSIONS AND FUTURE WORK

This paper investigated the impact of Brazilian economic news on the performance of BM&FBOVESPA by means of machine learning prediction methods. Our results show evidences that contradicts the market efficiency hypothesis for the Brazilian stock market confirming which has been verified in other markets such as the United States, Hong Kong, China, Turkey and Tehan. Through the experiments conducted we achieved up to 20% improvement over the *Random Model* and 15% over the *Keep Trend* baseline. In all cases, for each target on IBOVE, we found the best settings of the prediction models used. For future work we plan to build an online system to help small investors to make profitable decisions and popularise the BM&FBOVESPA among Brazilians. All data and code produced on this work is made publicly available online[13].

REFERENCES

Al Nasseri, A., Tucker, A., and de Cesare, S. Quantifying stocktwits semantic termsâĂŹ trading behavior in financial markets: An effective application of decision tree algorithms. *Expert Systems with Applications* 42 (23): 9192–9210, 2015.

Araújo, M., Reis, J. C., Pereira, A. C., and Benevenuto, F. An evaluation of machine translation for multilingual sentence-level sentiment analysis. In *Proceedings of the 31th Annual ACM Symposium on Applied Computing*. ACM, 2016.

Bulkowski, T. N. *Encyclopedia of chart patterns*. Vol. 225. John Wiley & Sons, 2011.

Feldman, R., Rosenfeld, B., Bar-Haim, R., and Fresko, M. The stock sonarâĂŤsentiment analysis of stocks based on a hybrid approach. In *Twenty-Third IAAI Conference*, 2011.

Hagströmer, B. and Norden, L. The diversity of high-frequency traders. *Journal of Financial Markets* 16 (4): 741–770, 2013.

Hutto, C. and Gilbert, E. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth International AAAI Conference on Weblogs and Social Media*, 2014.

Linstone, H. A., Turoff, M., et al. *The Delphi method: Techniques and applications*. Vol. 29. Addison-Wesley Reading, MA, 1975.

Malkiel, B. G. and Fama, E. F. Efficient capital markets: A review of theory and empirical work. *The journal of Finance* 25 (2): 383–417, 1970.

Schumaker, R. P. and Chen, H. A discrete stock price prediction engine based on financial news. *Computer* (1): 51–56, 2010.

Tetlock, P. C., SAAR-TSECHANSKY, M., and Macskassy, S. More than words: Quantifying language to measure firms fundamentals. *The Journal of Finance* 63 (3): 1437–1467, 2008.

[13]https://github.com/zegildo/PhD

A Linked Open Data Approach for Feature Based Diversification in Music Recommendation Systems

C. Nobrega, R. Oliveira, N. Leite, L. B. Marinho, N. Andrade and C. E. Pires

Federal University of Campina Grande, Brazil
[caionobrega, ricardooliveira, nailsonleite]@copin.ufcg.edu.br, [lbmarinho, nazarenoandrade, cesp]@dsc.ufcg.edu.br

Abstract. Diversification is an important concept in Recommender Systems given that it may increase users' satisfaction over recommendations that are solely based on accuracy. The reason is that diversification may recommend items that are not necessarily familiar to the users but are nevertheless interesting, thus causing a positive surprise. In this work we propose to exploit Linked Open Data (LOD) representing different kinds of artists' relationships, such as genre and artist's origin, for music recommendation. The idea is to use a LOD traversal algorithm for re-ranking items provided by state-of-the-art recommender algorithms built for improving accuracy. Differently from related works, the diversification of our approach is driven by a given feature, e.g., genre. We conduct experiments on a large data set collected from Last.fm and DBPedia and show that while our approach is competitive to state-of-the-art diversification methods it provides better diversification concerning the specific features of interest.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning - Knowledge Acquisition; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.2.8 [**Database Applications**]: Data Mining

Keywords: Diversification; Linked Open Data; Recommender systems

1. INTRODUCTION

In scenarios of vast availability of heterogeneous content, such as online music streaming services, users typically face the so called information overload problem which may difficult the discovery of relevant and interesting items on their own. Recommender Systems are software tools that are commonly applied to address this issue, since they learn the preferences of users and anticipate their information needs by providing personalized recommendations.

Recommender Systems is a very active area of research with a large number of algorithms available in the literature. Most of these algorithms, including the most successful ones, first build users' profiles based on users' consumption history, and then learn recommendation models with the aim of recommending yet unknown items that match the users preferences. It is remarkable that in most of these works the loss functions of the recommendation models are mostly defined in terms of the misfit between the predicted items and the items in the users' profiles (i.e., the items used for test). Although effective in many scenarios, this approach may deliver recommendations that are relevant but uninteresting. For example, recommendations about previously unheard albums of *The Beatles* to a user that has only heard *The Beatles* albums is extremely accurate, although eventually tedious and obvious since the user could have find these items by himself. Since this user appears to be a fan of *The Beatles*, he would be maybe more surprised to receive recommendations about the other bands that *Paul McCartney* played in.

In order to mitigate this problem, many approaches have appeared with the aim of not only improv-

ing accuracy but also diversity. This is usually achieved by mechanisms that avoid recommendation lists where items are very similar to each other. Such approaches can potentially generate recommendation lists that increase the user's satisfaction by providing novelty and serendipity. By novelty we mean the presence of items that the user did not know, but that fits his expectation while serendipity is the sensation of finding a valuable item that could not be found by the user himself [Herlocker et al. 2004].

In this paper we propose to exploit Linked Open Data (LOD) for diversifying recommendations withou sacificing too much accuracy. LOD are data repositories that link data entities by establishing relationships between them. LOD is an initiative that has encouraged good practices of storing data, generating the so called Web of Data [Bizer et al. 2009], making possible not only the browsing in the document space, but also the use of search engines to provide query capabilities. Our method, called LOD-Diversification, consists of using the LOD graph to look for indirect relations between artists in the level of their features, e.g. genre, discovering in this way item relations not captured by traditional collaborative filtering algorithms. Initially, an user-based collaborative filtering (CF) recommendation is computed for generating a Top-N recommendation list, where N is considerably larger than usual recommendatin lists (e.g. Top-10). Next, we apply an incremental re-ranking algorithm on this initial list for generating a smaller but more diverse list.

The main difference of our approach with respect to other related approaches from the literature is that we diversify the recommendation list according to a given item feature, e.g. genre. LOD is then used to explore the items' relations with respect to the chosen feature. For example, an algorithm that recommends a *Black Metal* artist for a user who likes *Death Metal* does not represent a diversified recommendation. In summary, the contributions of our paper are:

—The development of an algorithm based on Linked Open Data to generate diverse recommendation lists;
—The diversification of recommendation lists per item feature, giving the user the possibility to indicate in which aspect he wants the recommendation to be more diverse.

We conducted experiments on a large data set collected from Last.fm[1] and DBPedia[2] and show that while our approach is competitive to state-of-the-art diversification methods it provides better diversification concerning the specific features of interest.

2. RELATED WORK

Several works have addressed diversity in recommender systems. In his seminal work, Ziegler et al. [Ziegler et al. 2005] proposed Topic Diversification an algorithm that exploits taxonomies of item categories to achieve diversification. Ziegler et al. [Ziegler et al. 2005] also defined the Intra-list Similarity (ILS) metric. This metric uses Pearson correlation or cosine similarity as a function $c_o(b_k, b_e)$ to compare two elements b_k and b_e, then sum all the comparison results in a set to produce a degree of similarity between them. Vargas and Castells [Vargas and Castells 2011] defined a framework containing several state-of-the-art metrics, like ILD (Intra-List Diversity), the average pairwise distance of the recommendation items, and nDCG (normalized Discounted Cumulative Gain), a metric that takes into account the position were the item appears in the recommendation list, privileging the better placed. Both metrics were used in this work.

Ristoski et al. [Ristoski et al. 2014] defined a heuristic to produce recommendations that are both diverse and accurate. The basic idea is to re-rank an initial recommendation list obtained by a user-based CF, by keeping the Top-m original recommendations, with m being smaller then the final

[1]http://www.last.fm/
[2]http://wiki.dbpedia.org/

recommendation list size, and completing the list with the elements that, at each step, increase the diversity among all the remaining candidates. The m items kept from original recommender have the intent to preserve accuracy, while the list is completed by items that maximize diversity.

Di Noia et al [Di Noia et al. 2012] propose a recommender system for movies based exclusively on Linked Open Data information. The similarity between items was established by the use of TF-IDF to create a score for terms coming from LOD, generating a vector for each movie. These vectors are them compared by cosine similarity. The LOD information is structured as a RDF graph, and the relation between items is stored in a multidimensional matrix, where each slice corresponds to a different property in the graph. The evaluation of the method is made measuring its accuracy by precision and recall metrics. In this work, diversity was not addressed.

Our work is built on top of the aforementioned in terms of: (i) re-ranking, we also used a heuristic for re-ordering an user-based CF recommendation; (ii) knowledge graph, we used LOD for finding similarities between items that were not noticeable without metadata.

3. LINKED OPEN DATA AND FEATURE EXTRACTION

In the user-based CF algorithm, a neighborhood is established composed of users with similar history, i.e., users that listened to several artists in common. Then the recommendation is formed by the items that were not listened by the user but were most popular among the nearest neighbors. Notice that in this approach, diversity is not taken into account since there is no guarantee that the neighbors consume diverse items. When promoting diverse recommendation lists it is important to consider the dimension/feature accross which the items are diversified. For example, users may prefer to listen to artists from different genres (diversificaton by genre) or artists from the same genre but from different nationalities (diversification by nationality). If we considered ILD measured on genres, i.e., the extent to which the recommended items differ in terms of genres, a recommendation list containing artists of *Trance* and *House* would yield a high ILD (i.e. high diversification), while we know that *Trance* and *House* are similar and thus do not represent a diverse recommendation.

LOD can help to overcome this problem by finding that *Trance* and *House* are closely related since they have the same stylistic origin. By parsing the LOD graph, we can find such connections that can be used for promoting diverse recommendations. Nodes in the LOD graph are connected with respect to some given relation. In this paper we have chosen the *StylisticOrigin* relation, which links genres to their originating genres (for example *Rock Music* originated *Pop Rock*). The nodes in the LOD can belong to different granularity levels. For example, the genres (in this case graph nodes) that directly describes artists are in the first level (depth one). In the second level we have the genres that are the stylistic origins of the genres in the first level, and so on. Figure 1 illustrates a part of the genre LOD graph used in this work.

For example, in Table I we see part of the LOD graph for the artists *The Beatles* and *The Rolling Stones*, according to their genres in depth 2. Considering only the direct genres that describe these artists, i.e., level 1 of depth in the graph, they only shared *Rock music*. However, traversing the graph towards depth 2, *Rock and roll*, a stylistic origin for both *Pop Music* and *Hard Rock*, becomes a shared genre, which leads to the realization that *The Beatles* and *The Rolling Stones* are connected. The effect of traversing the graph is that the deeper is the traversing, the higher is the possibility to find similarity between artists that would not be regarded as similar in a traditional CF setting.

To compare the artists we can use cosine similarity or Pearson correlation. In both cases, the presence of a greater number of shared genres will cause an increase of the similarity. In Figure 2 we present the comparison between *The Beatles* and *The Rolling Stones* using these two metrics to show how the similarity increases when a greater depth is taken into consideration. In this example, the artists were compared until 5 degrees of depth and it is noticeable that the similarity increases in greater depths, specially until the depth 3.

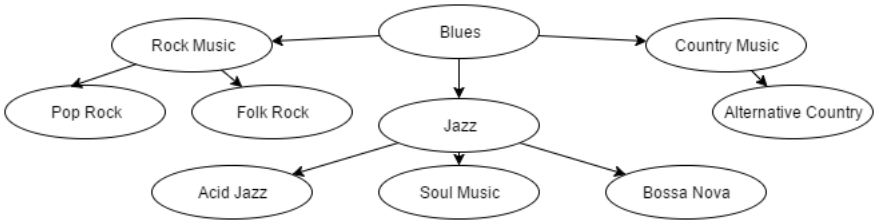


Fig. 1: Example of Genre Graph

Table I: Part of LOD graph from *The Beatles* and *The Rolling Stones*

(a) *The Beatles*

Genre	Depth	Weigth	Normalized Weight
Pop music	1	0.5	28.05941
Rock music	1	0.5	28.05941
Rock and roll	2	0.08333	4.00849
Classical music	2	0.08333	4.00849

(b) *The Rolling Stones*

Genre	Depth	Weigth	Normalized Weight
Blues	1	0.2	13.28836
Rhythm and blues	1	0.2	13.28836
Rock music	1	0.2	13.28836
Blues rock	1	0.2	13.28836
Hard rock	1	0.2	13.28836
Folk music	2	0.02857	2.65767
Rock and roll	2	0.02857	2.65767

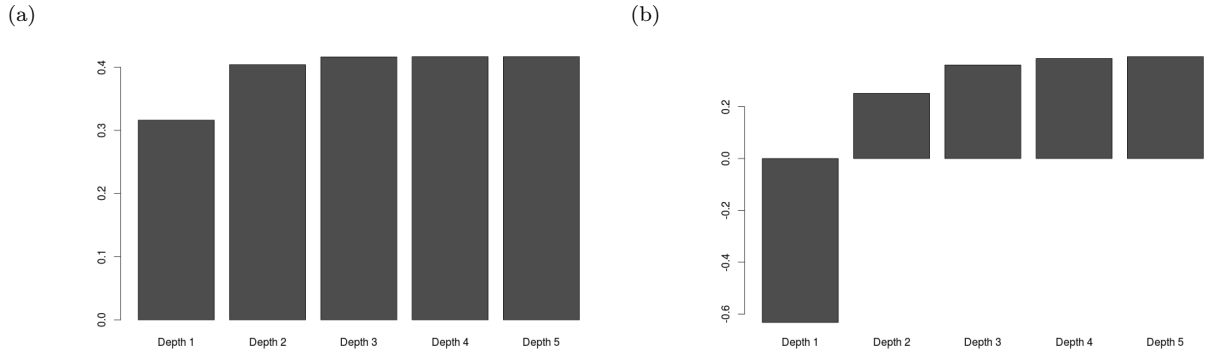


Fig. 2: Comparing *The Beatles* and *The Rolling Stones* by (a) cosine similarity and (b) Pearson correlation

3.1 Determining weights for the LOD graph

For this graph to be useful for the LOD-Diversification algorithm, we need to define the weight that each genre has for an artist. We define the LOD graph of a given artist as the genres associated to it, considering different dephts of the graph, and their corresponding weights.

For a given genre, we first need to find all the linked genres according to the chosen relation and a given depth. We call this set $C_g = \{g_1, g_2, \ldots, g_i\}$ where g is the chosen genre and $g_1, \ldots, g_i$ are the linked genres according to the LOD graph and the depth chosen. Given a graph $G = (V, R)$ where $V = C_g$ is the set of vertexes and $R = C_g \times C_g$ is a set of relations between the vertexes, the weight distribution of each genre in the LOD graph is computed by Equation 1 below,

$$weight(g_i) = \begin{cases} \frac{1}{|adj(g_i)|} & if level(g_i) = 1, \\ \frac{weight(parent(g_i))}{level(parent(g_i)) + |adj(parent(g_i))|} & otherwise \end{cases} \quad (1)$$

where $weigth(g_i)$ represents the current weight genre g_i; $|adj(g_i)|$, $parent(g_i)$ represents node g_j that

participates in the relation $(g_j, g_i) \in R$; $adj(g_i)$ returns the set of vertexes adjacent to g_i.

For example, let's consider *The Beatles* as the artist of interest and generate its LOD graph for genre as feature. In DBpedia, *The Beatles* are associated to two genres, $C = \{PopMusic, RockMusic\}$. As $|C| = 2$, each genre is assigned the weight 1/2. The stylistic origin for *Pop_ music* is the set *{Folk music, Rock and roll, Classical music, Traditional Pop Music, Popular music}*. These, besides the genres that form the stylistic origin of *Rock music*, form the LOD graph at depth 2 for *The Beatles*. To determine the weight of these genres we have $weight(parent(g_i)) = 0.5$, $level = (parent(g_i)) = 1$, $|adj(parent(g_i))| = 5$, so $0.5/(1 + 5) = 0.083$.

4. LOD-DIVERSIFICATION

The LOD Diversification algorithm is a greedy strategy that iteratively selects, among the elements of the base recommendation list, the one that presents the higher dissimilarity to the already selected elements. This heuristic leads to a list that promotes diversity.

The inputs to the LOD Diversification algorithm are the set:

—$R_u = \{i_1, i_2, ..., i_n\}$: the base recommendation list computed by user-based CF;
—the *depth* of the LOD graph;
—$cutoff$: number of items in the final recommendation list;
—α: balances accuracy/diversity trade-off.

It is important that the size of the input list is considerably greater than the size of the final recommendation. This will give the algorithm more options to maximize the objective function and achieve a greater diversity. Finally, LOD-Diversification is described in Algorithm 1.

Algorithm 1 LOD Diversification algorithm

```
1:  procedure LOD-DIVERSIFICATION(R_u, depth, cutoff, α)
2:      B_u ← R_u
3:      for each item i in R_u do
4:          B_{u,i} ← lod_graph(i, depth)
5:      end for
6:      S_u(1) ← B_u(1)
7:      for z = 2 : cutoff do
8:          B_u ← B_u − S_u
9:          for each item i in B_u do
10:             v_i ← dist(i, S_u)
11:         end for
12:         B_u ← order(B_u, v)
13:         for each item b in B_u do
14:             scores(b) ← (position(R_u, b)/size(R_u))×
        (1 − α) + (position(B_u, b)/size(B_u)) × α
15:         end for
16:         S_u(z) ← B_u(argmin(scores(B_u)))
17:     end for
18:     return S_u
19: end procedure
```

In line 4, the *lod_graph*() function is generated for each artist as explained in Section 3. Each item in the base recommendation list must have its LOD graph extracted from Linked Open Data. A new recommendation list, S_u, receives initially the first element in the base recommendation list (line 6). This is done to preserve the accuracy brought by the first element in a accuracy-based list. At each iteration, one element is selected to compose the list S_u. The element that minimizes the objective function in line 14 is selected to compose the final recommendation. Each element in the received list that have not already been selected for the final list is a candidate to compose it. In line 10, the *dist*() function used to compare a candidate artist to the artists already included in the final recommendation list can be the cosine similarity or Pearson correlation. The candidate list is then increasingly ordered, the objective function is applied and the next element in the final recommendation is selected.

5. EVALUATION

In this section, we evaluate the effectiveness of our approach to accomplish music recommendation with diversity. In Section 5.1, we describe the setup that supports our evaluation, including the Last.fm dataset description, the evaluation protocol and metrics, and the recommendation baselines used in our experiments. In Section 5.2, we discuss and analyze our experimental results.

5.1 Experimental Setup

5.1.1 *DBPedia and Last.fm datasets.* We used two datasets to set up the experimentation in this work: DBpedia and Last.fm. DBpedia is a structuralization of data available in Wikipedia and is one of the most famous LOD dataset. Data is accessed using SPARQL, a SQL-like query language for RDF data (commom format of LOD datasets). However, DBpedia is also available in table format, which was used in this work. We used specifically the tables "MusicalArtist" and "Band" to compose our Artist database, besides the "MusicGenre".Last.fm is a social network and a scrobbling system, which allows users keep a history of musics played on media players. We used the Last.fm dataset collected by Celma and Lamere, which used the user.getTopArtists() method from Lasf.fm API [Celma and Lamere 2011].

The data cleaning procedure consisted of first only use artists with genre information available. Next, users with less than 10 artists and potential outlier users were removed. In the later case, we used the threshold calculated by the boxplot of number of artists. So, we merged these datasets by the name of the artists [3].

5.1.2 *Evaluation Protocol.* In order to evaluate the artists recommendation, we split the dataset into training and test partitions, in a ratio of 80%/20%, respectively. So, we followed a common evaluation procedure [Cremonesi et al. 2010] in which for each target user two steps are required: (*i*) a default recommender outputs list with top 50 artists. In this step, only relevance criteria should be taken into account; and (*ii*) a re-ranker algorithm selects a subset of 10 relevant items from previous step, with $\alpha = 1$, maximizing the importance of diversity in the generation of the final recommendation lists. The default recommender was the classic kNN user-based [Ricci et al. 2010] and the re-ranker algorithms are listed in the next subsection.

Finally, we used three kind of metrics. nDCG to evaluate the relevance criteria solely, ILD to evaluate the diversity solely and α-nDCG to combine these two criteria simultaneously.

5.1.3 *Compared Algorithms.* As compared algorithms we selected *user-based CF* (UB), MMR, an implementation of the Ziegler et al. [Ziegler et al. 2005] Topic Diversification and ESWC14 [Ristoski et al. 2014], both described in section 2. In the ESWC14 we used $m = 4$ in our experiments, as suggested in their work.

5.2 Results and Discussion

In particular, we aim to answer the following research questions:

Q1. Is the LOD Diversification algorithm capable to build a recommendation list with higher diversity rates than algorithms only built for improving accuracy, without a significant loss of accuracy?
Q2. Can a LOD graph achieve higher diversification metrics when traversed in a greater depth?

In order to analyze the of LOD-Diversification algorithm, we used the Top1000 users of Last.fm database, i.e., we selected the top 1,000 users in terms of the number of listened distinct artists.

[3]The merged dataset is available at https://goo.gl/7LdijG

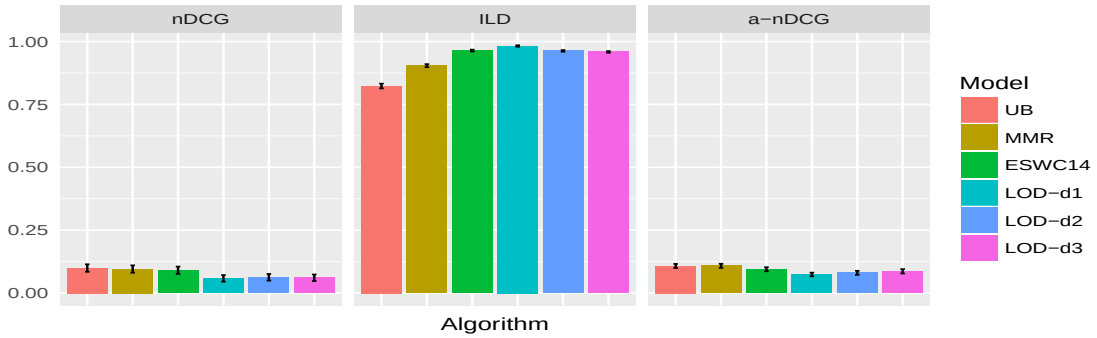


Fig. 3: Results for cut-off 10 for different recommendation algorithms and their diversified re-rankings applied to the Last.fm dataset - Feature music genre.

Moreover, with respect to LOD graph construction, we used artist's genre as feature and tested different levels of diversification, defining three degrees of depth.

To address Q1, we first analyzed individually the accuracy (nDCG) and diversity (ILD) metrics. The baseline algorithm UB achieved the best accuracy performance but the worst in diversity. It was an expected result since this algorithm does not consider diversity at all. Our results show that LOD-Diversification with one degree of depth (LOD-d1) achieved best diversity performance with minor but statistically significant difference compared to ESWC14[4]. Considering the α-nDCG metric, all versions of LOD-Diversification and the baseline ESWC14 have worse performance than the baseline UB. This is a counterintuitive result and could indicate that this is not the right metric to balance accuracy and diversity.

To answer Q2, we analyzed the performance in terms of the diversity of LOD-Diversification in different levels of depths. Taking into account ILD, increments of depth caused slight loss of diversity, which needs further investigation.

5.3 Qualitative Analisys

In Table II we selected an user from the Last.fm database to examplify the diversification process. The first column presents the 10 artists most listened by the user, and the second column shows the recommendation provided by LOD-Diversification with 2 degrees of depth.

Table II: An example of recommendation by LOD-Diversification with $depth = 2$

History (Top 10 artists)	Recommendation
The Beatles	The Rolling Stones
Red Hot Chili Peppers	Depeche Mode
Miles Davis	Buckethead
Metallica	Charles Mingus
Johnny Cash	MxPx
The Mars Volta	The Band
Buddy Guy	A Tribe Called Quest
Howlin' Wolf	Belle and Sebastian
Bob Dylan	Charley Patton
B.B. King	Slayer

The user history is mainly formed by *rock'n'roll* artists from the 60s to the 80s and *blues* artists. Remembering that the LOD-Diversification recommendation is made on top of a 50 recommendation list

[4]friedman test with $\alpha = 5\%$ and p-value = 1.137e-11

generated by user-based CF, which aims at accuracy. It is natural that we do not see recommendations of genres like *samba* or *salsa*, that probably would dissatisfy the user. The first recommendation, *The Rolling Stones*, was kept from the collaborative filtering list and is a very appropriate recommendation for an user with a strong history of *rock* and *blues*. The search for diversity brings to the recommendation different genres like *new wave* (*Depeche Mode*), *progressive metal* (*Buckethead*), *indie rock* (*Belle and Sebastian*), *thrash metal* (*Slayer*), *punk rock* (*MxPx*) and others. Notice that in spite of bringing diversity to recommendation, the cited artists are closely related to rock music, which increases the chances of satisfying the user.

6. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a method for re-ranking user-based collaborative filtering recommendation by using Linked Open Data to increase the diversity of recommended items. Our proposed LOD-Diversification algorithm proved competitive among several algorithms, without a significant loss in accuracy.

Besides the diversity achieved, taking LOD into account allows to generate exaplainable recommendations to the user.

In future works we intend to use databases from other domains, like movies, to verify if the method is still applicable. Another work is to find or develop a metric to analyze both the diversity and accuracy as a single problem, instead of treating them as a binary problem. Finally, we intend to consider other features, taken in isolation or collectivelly, and use online evaluation protocols to measure user satisfaction.

REFERENCES

Bizer, C., Heath, T., and Berners-Lee, T. Linked data-the story so far. *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, 2009.

Celma, O. and Lamere, P. Music recommendation and discovery revisited. In *Proceedings of the Fifth ACM Conference on Recommender Systems*. RecSys '11. ACM, New York, NY, USA, pp. 7–8, 2011.

Cremonesi, P., Koren, Y., and Turrin, R. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the Fourth ACM Conference on Recommender Systems*. RecSys '10. ACM, New York, NY, USA, pp. 39–46, 2010.

Di Noia, T., Mirizzi, R., Ostuni, V. C., and Romito, D. Exploiting the web of data in model-based recommender systems. *6th ACM conference on Recommender systems - RecSys '12*, 2012.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)* 22 (1): 5–53, 2004.

Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B. *Recommender Systems Handbook*. Springer-Verlag New York, Inc., New York, NY, USA, 2010.

Ristoski, P., Loza Mencía, E., and Paulheim, H. A hybrid multi-strategy recommender system using linked open data. In *Semantic Web Evaluation Challenge, Proceedings (ESWC 2014)*. Communications in Computer and Information Science, vol. 475. Springer, pp. 150–156, 2014.

Vargas, S. and Castells, P. Rank and relevance in novelty and diversity metrics for recommender systems. *Proceedings of the 5th ACM conference on Recommender systems - RecSys '11*, 2011.

Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*. ACM, pp. 22–32, 2005.

Recomendação não personalizada baseada em Cobertura Máxima

Nícollas Silva[1], Adriano César[1], Fernando Mourão[2], Leonardo Rocha[2]

[1] Universidade Federal de Minas Gerais, Brazil
ncsilvaa@gmail.com, adrianoc@dcc.ufmg.br
[2] Universidade Federal de São João del-Rei, Brazil
fhmourao@ufsj.edu.br, lcrocha@ufsj.edu.br

Abstract. Atualmente, os Sistemas de Recomendação (SsR) estão tão focados em agradar os usuários já existentes no domínio, que acabam dando pouca relevância a uma fase talvez mais importante sob a lógica de negócio: a aquisição de novos usuários. Estratégias de recomendação não personalizadas acabam cada vez mais negligenciadas na literatura. Normalmente, estas estratégias focam em recomendar itens mais populares, mais recentes, mais recorrentes ou mais bem avaliados. Tais abordagens partem da premissa que o consumo dos usuários é, em geral, enviesado por uma dessas dimensões nos cenários em que SsR são aplicáveis. Entretanto, tais premissas não são válidas para o consumo de nichos, em que usuários se interessam por itens diferentes do gosto comum de uma população. Neste intuito, este trabalho tem por objetivo validar a aplicabilidade da Cobertura Máxima, um problema NP-Completo, em SsR sobre os níveis de novidade, diversidade e serendipidade. O pressuposto dessa abordagem é que a probabilidade de um usuário qualquer encontrar um item de sua preferência aumenta devido a diversidade de gostos dos usuários do sistema, que leva a recomendações inusitadas. De fato, os resultados encontrados apresentam em média um ganho de 5,5% em novidade, 18% em diversidade e 60% em serendipidade quando comparada as estratégias de consumo de massa (i.e., itens mais populares ou mais bem avaliados). Além disso, os resultados indicam que os itens recomendados por Cobertura Máxima pertencem a um grupo de itens não populares, o que mostra que a estratégia é capaz de fugir da bolha de itens sempre recomendados. O ganho obtido ao aplicar a estratégia de Cobertura Máxima mostra-se relevante para cenários específicos como *e-commerce*, ou mesmo para solucionar problemas como o de usuários *cold-start* e o efeito da *long-tail*.

Categories and Subject Descriptors: H.3.3 [**Retrieval tasks and goals**]: Recommender Systems

Keywords: Sistemas de Recomendação, Cobertura Máxima, Novidade, Diversidade, Serendipidade

1. INTRODUÇÃO

Sistemas de Recomendação (SsR) constituem uma das mais importantes ferramentas utilizadas para auxiliar usuários na tomada de decisão em variados cenários reais [Koren et al. 2009]. Atualmente, grande parte dos SsR se preocupam em atender adequadamente aos gostos dos usuários já existentes em um dado sistema através da recomendação personalizada. Entretanto, pouca relevância é dada a uma fase igualmente importante sob a lógica de negócio: a aquisição de novos usuários. Neste caso, tornam-se pertinentes perguntas tais como: *Como apresentar os itens disponíveis no sistema a potenciais usuários quando não se sabe nada sobre estes?*; e *Quais itens de um catálogo de produtos apresentam maior potencial para atrair novos usuários?*. As técnicas de recomendação mais utilizadas para abordar essas questões se baseiam em estratégias não personalizadas, considerando assim o contexto geral do cenário estudado ao invés do passado de cada usuário específico.

As principais estratégias de recomendação não personalizadas utilizadas atualmente exploram informações simples como popularidade dos itens, avaliações positivas ou recência de lançamento [Iskold 2007]. Tais abordagens partem da premissa que o consumo é, em geral, enviesado por uma dessas dimensões nos cenários em que SsR são aplicáveis, permitindo-se atingir alta eficácia de predições

Este trabalho foi parcialmente patrocinado pelo Instituto Nacional de Ciência e Tecnologia para a Web (CNPq no. 573871/2008-6), MASWeb (grant FAPEMIG/PRONEX APQ-01400-14), EUBra-BIGSEA (H2020-EU.2.1.1 690116, Brazil/MCTI/RNP GA-000650/04), CAPES, CNPq e Fapemig.

através de estratégias simples e não supervisionadas. Apesar dessa premissa ser válida para o denominado "consumo de massa", em que usuários consomem itens que despertam interesse em grande parte de uma população, o mesmo não pode ser dito sobre "consumo de nichos", em que usuários se interessam por itens diferentes do gosto comum de uma população. Dessa forma, apresentar apenas itens pertencentes ao consumo de massa a potenciais consumidores de nicho não representaria uma boa estratégia para a aquisição deste tipo de consumidores. Atender adequadamente aos gostos deste tipo de usuário, porém, é de suma importância para diversos cenários, uma vez que o mercado de nicho pode representar mais da metade do lucro em cenários de e-commerce, definindo-se o que se conhece como cauda longa do consumo [Anderson 2006].

Este trabalho tem por objetivo avaliar uma nova estratégia de recomendação não personalizada que visa apresentar a potenciais usuários tanto itens pertencentes ao consumo de massa quanto ao consumo de nichos. Nossa proposta baseia-se na aplicação do famoso problema de Cobertura Máxima a cenários de recomendação [Michael and David 1979]. A estratégia de Cobertura Máxima visa identificar um subconjunto de itens do domínio que são potencialmente relevantes para o maior número de usuários distintos. A premissa dessa abordagem é que recomendar os itens que satisfaçam ao maior número de usuários traz mais diversidade e surpresa a todos os usuários do sistema quando comparada a estratégias de recomendar os itens mais populares ou mais bem avaliados. Além disso, a probabilidade de um usuário específico encontrar pelo menos um item próximo de sua preferência pessoal aumenta devido a estratégia tentar cobrir uma grande diversidade de gostos distintos. Cabe salientar que recomendar itens que atendam ao consumo de massa é considerada uma tarefa fácil, dado o grande volume de informações de consumo disponíveis sobre tais itens. Por outro lado, a recomendação de itens pertencentes ao consumo de nichos é uma tarefa difícil e com grandes implicações para variados cenários, mas ainda em aberto na literatura.

De forma a avaliar a relevância prática das recomendações geradas pela estratégia de Cobertura Máxima, calcula-se os níveis de novidade, diversidade e serendipidade sobre as coleções de filmes do *MovieLens*, através de métricas tradicionais propostas em [Vargas and Castells 2011] e [Zhang et al. 2012]. De fato, os resultados encontrados ao compararmos essas estratégias de recomendação evidenciam o pressuposto dessa nova abordagem. Aplicar a estratégia de Cobertura Máxima tende a trazer mais supresa aos usuários do domínio, satisfazendo suas distintas preferências. As recomendações geradas apresentam em média um ganho de 5,5% em novidade, 18% em diversidade e 60% em serendipidade quando comparada as estratégias de consumo de massa que recomendam itens mais populares ou mais bem avaliados. Nota-se também, que as recomendações geradas por Cobertura Máxima consistem em itens não populares, contendo em média 18% de itens distintos do gosto comum da população geral. Tais itens são capazes de satisfazer ao consumo de nichos de alguns usuários. Além disso, pode-se notar que mesmo com esses ganhos, a recomendação por Cobertura Máxima não se distancia das demais estratégias quando comparada as métricas clássicas de acurácia. Ressaltamos que as contribuições deste trabalho são particularmente relevantes para estudos aplicados, como na aplicação em cenários no qual o consumo dos usuários está centrado em itens populares. Cabe ainda salientar que não encontramos na literatura trabalhos que abordem os aspectos levantados sobre a aplicabilidade do problema de Cobertura Máxima para recomendações não personalizadas.

2. REFERENCIAL TEÓRICO

Em variados cenários reais de recomendação, como de *e-commerce* [Hinz and Eckert 2010], turismo [Batet et al. 2012] ou filmes [Herlocker et al. 1999], surgiu a necessidade de avaliar não apenas a recomendação personalizada ao usuário. Em sistemas *e-commerce*, por exemplo, os proprietários estão interessados em maximizar o lucro final, aderindo a estratégias de recomendação não personalizada, e não se preocupando com a fidelidade do usuário [Hammar et al. 2013]. Nestes, as técnicas de SsR utilizam estratégias de recorrência, recência e itens mais bem avaliados. De maneira geral, recomendam-se itens que estão relacionados entre si (i.e., quem comprou este, comprou também), ou mesmo os itens mais vendidos recentemente. Em [Hammar et al. 2013], os autores se preocuparam em maximizar a probabilidade dos clientes realizarem compras e utilizaram a estratégia de cobertura

máxima para aumentar a diversidade de produtos aos clientes. Por sua vez, em cenários de filmes, uma dificuldade comum encontra-se em recomendar itens a usuários novos no sistema (i.e., *cold-start*), visto que estes não possuem nenhum histórico de atividades [Schein et al. 2002]. Neste contexto, estratégias de popularidade estão sendo cada vez mais aplicadas na tentativa de apresentar um contexto global do sistema a um usuário que não se tem um histórico de consumo definido.

Em geral, estratégias não personalizadas em SsR, referem-se a recomendadores que não visam somente os usuários finais. As recomendações geradas são independentes dos usuários, de modo que cada usuário recebe as mesmas recomendações [Schafer et al. 1999]. Basicamente, estratégias não personalizadas visam recomendar itens aos usuários com base na opinião geral destes sobre os conteúdos disponíveis. Técnicas não personalizadas são comuns em cenários onde é necessário configurar uma exibição que é vista sem alterações específicas por cada usuário. Por exemplo, as recomendações apresentadas pela *amazon.com*, quando um usuário específico não está logado no sistema, são recomendações não personalizadas que representam um contexto geral, sendo completamente independentes de um usuário específico.

2.1 Popularidade

Recomendadores baseados em popularidade consistem em uma estratégia simples e intuitiva, que sempre recomenda os itens mais populares na coleção de dados, independente do usuário alvo. A popularidade de um item é estimada pelo número de usuários distintos que já consumiram este item no passado. No contexto de filmes, por exemplo, para cada filme do domínio calcula-se a quantidade de usuários distintos que o assistiu. Esta abordagem, embora seja muito simples, tem se mostrado eficiente em algumas aplicações [Bobadilla et al. 2013] e constitui uma das principais técnicas de recomendação não personalizadas alcançando bons resultados.

2.2 Itens mais bem avaliados

Recomendadores baseados em itens mais bem avaliados consistem em outra estratégia simples, que sempre recomenda os mesmos itens independente do usuário alvo. Basicamente, gera um *ranking* de itens ordenados decrescentemente pela média dos *ratings* recebidos por cada item do sistema. O pressuposto dessa abordagem é que os itens mais bem avaliados tendem a interessar vários usuários.

2.3 Cobertura Máxima

No cenário de recomendação, podemos formalizar o problema de Cobertura Máxima da seguinte maneira: dado $U = \{u_1, u_2, ..., u_m\}$, como o conjunto de usuários do sistema e dado $F = \{S_1, S_2, ..., S_n\}$ como uma coleção de conjuntos S_i de usuários que consumiram o item i, o objetivo é determinar um subconjunto $F^* = \{S_{i1}, S_{i2}, ..., S_{ik}\}$ que contém o maior número de usuários distintos possíveis.

Maximum k-Coverage

Instância: Conjunto de elementos $U = \{u_1, ..., u_m\}$, um valor inteiro k e uma coleção de conjuntos $F = \{S_1, ..., S_n\}$, onde cada conjunto S_i é um subconjunto de U.

Objetivo: Encontrar o subconjunto $F^* \subseteq F$ tal que $|F^*| \leq k$ e o número de elementos cobertos $|\bigcup_{S_i \in F^*} S_i|$ seja maximizado.

De maneira geral, podemos dizer que o objetivo da Cobertura Máxima é encontrar um subconjunto F^* de itens, tal que $|F^*| \leq k$, que maximize o número de usuários distintos atingidos. Em outras palavras, pretendemos encontrar os k filmes que interessaram ao maior número de usuários distintos e recomendá-los aos usuários finais. Note que este problema, não está interessado no passado de consumo de um usuário específico, mas sim de todos os usuários da coleção de dados.

O problema de Cobertura Máxima é um variante do famoso problema de Cobertura de Vértices, bem estudado pela literatura [Alon et al. 2003]. Infelizmente, estes são problemas da classe NP-Completo, e portanto não existe uma solução ótima que possa ser resolvida em tempo polinomial.

Entretanto, usando uma simples heurística gulosa, conforme mostra o Algoritmo 1 que obtém uma aproximação de 63% do número máximo de usuários que podem ser cobertos, conforme mostrado em [Chvatal 1979]. Em alguns cenários menores o algoritmo aproximativo de tempo polinomial, possui um bom desempenho [Feige 1995], conseguindo atingir as soluções ótimas do problema. Este algoritmo é implementado com complexidade temporal de $O(knm)$, sendo k o número de itens encontrados pelo algoritmo, m o número de usuários e n o número de itens.

Algorithm 1 GREEDY-MAX-COVERAGE(U,k,F)

$R \leftarrow U$
$F^* \leftarrow \emptyset$
for i from 1 to k **do**
  $S \leftarrow max_{S \in (F \setminus F^*)} |S \cap R|$
  $F^* \leftarrow F^* \cup \{S\}$
  $R \leftarrow R \setminus S$
  if $|R| = 0$ **then**
    break
  end if
end for
return F^*

3. PROJETO EXPERIMENTAL

Motivados em analisar a aplicabilidade do problema de Cobertura Máxima em SsR, o projeto de análise experimental está relacionado às dimensões de análise de novidade, diversidade e serendipidade. Primeiramente, são apresentadas as coleções de dados utilizadas na análise. Em seguida, define-se as métricas básicas de novidade, diversidade e serendipidade a serem utilizadas por nossas análises. Por fim, é apresentada a metodologia de avaliação utilizada para analisar os resultados encontrados.

3.1 Coleções de Dados

Dada a relevância destes cenários, selecionamos os conjuntos de dados *MovieLens* 100k e *MovieLens* 1M[1] que foram reunidos pelo *GroupLens* e contém, respectivamente, 100 mil e 1 milhão de *ratings* atribuídos por usuários a filmes de diversas categorias. Ambos são conjuntos de dados explícitos, onde existem *ratings* atribuídos pelos usuários no intervalo de 1 a 5. Em cada conjunto, existem pelo menos vinte *ratings* atribuídos por cada usuário a filmes de seu interesse. Tal abordagem resulta em conjuntos de dados altamente esparsos, conforme mostra a tabela I. Para este trabalho, as bases de dados foram divididas em treino e teste, mantendo a premissa da ordenação temporal do consumo dos itens, com uma distribuição de 70% para treino.

Table I. Bases de dados utilizadas.

Base	**Usuários**	**Itens**	**Esparsidade**
MovieLens 100k	943	1.676	93,67%
MovieLens 1M	6.040	3.952	95,82%

3.2 Métricas de Avaliação

A novidade de uma informação geralmente se refere a quão diferente esta informação é com relação a tudo que tinha sido previamente observado, por um específico usuário, ou por uma comunidade como um todo [Ricci et al. 2011]. Neste trabalho, o conceito de novidade é mensurado por meio da distância entre os itens recomendados e os itens do perfil de cada usuário, definido como distância esperada do perfil (EPD), como proposto no *framework* de [Vargas and Castells 2011]. Vale destacar que a fórmula 1 utilizada considera o conceito de novidade intrinsecamente ligado ao conceito de utilidade (relevância), uma vez que recomendar algo novo que não seja útil ao usuário é uma tarefa fácil. Um item é considerado relevante se a média das avaliações atribuídas a ele for maior ou igual a um limiar minímo, que pode ser a nota média do usuário ao qual ele está sendo recomendado.

[1]Disponíveis em: <http://www.grouplens.org/node/12>

$$nov(R|u) = EPD = C' \sum_{i_k \in R, j \in U} disc(k) p(rel|i_n, u) p(rel|j, u) d(i_k, j) \quad (1)$$

Por sua vez, diversidade geralmente se aplica a um conjunto de itens, e está relacionada com o quão diferente os itens são com relação uns aos outros [Ricci et al. 2011]. Em [Vargas and Castells 2011], diversidade é mensurada como a distância média esperada de um item para uma lista de itens (ILD), calculada como o complemento da similaridade dos itens recomendados, como mostra a equação 2.

$$div(R|u) = ILD = \frac{2}{|R|(|R|-1)} \sum_{i_k \in R, l<k} d(i_k, i_l) \quad (2)$$

Por outro lado, o conceito de serendipidade relacionado a recomendação, é uma forma de mensurar o quão surpreso o usuário ficou com o sucesso das recomendações. Neste trabalho, utiliza-se o *framework* proposto em [Zhang et al. 2012], que calcula serendipidade com base na equação 3. Basicamente, utiliza-se a similaridade de cosseno para medir a similaridade média entre os itens presentes no histórico de um usuário e novas recomendações geradas. Os valores mais baixos indicam que as recomendações desviam do comportamento tradicional de um usuário, e, portanto, trazem maior surpresa.

$$ser = 1 - \sum_{u \in S} \frac{1}{|S||H_u|} \sum_{h \in H_u} \sum_{i \in R_{u,20}} \frac{CosSim(i, h)}{20} \quad (3)$$

Por fim, calcula-se também o número absoluto de acertos (*hits*) gerados pelas estratégias não personalizadas. Para tal, simplesmente verifica-se quantos itens da lista Top-k, gerada pelo recomendador, foram de fato *consumidos* pelos usuários de acordo com as informações retidas no conjunto teste.

3.3 Metodologia de Avaliação

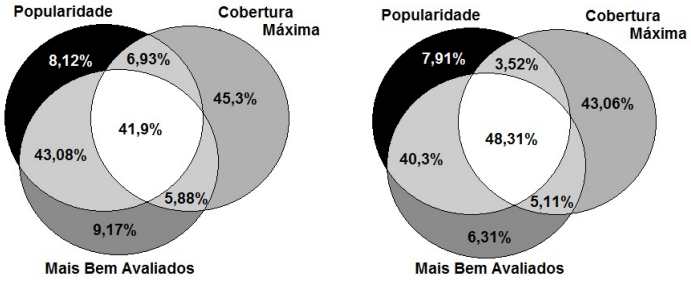
As estratégias de recomendação não personalizadas são aplicadas sobre cada conjunto de dados e são geradas listas de recomendação: *C-list*, com os itens da cobertura máxima; *P-list*, com os itens da popularidade; e *A-list*, com os itens mais bem avaliados. Posteriormente, para cada usuário, são recomendados os k primeiros itens das listas geradas, uma vez que o foco deste trabalho está na tarefa *Top-k*. Para as análises, varia-se o valor de k entre $[5, 10, 20, 50, 100]$ a fim de avaliar o desempenho da recomendação em cenários reais. A metodologia é definida nos seguintes passos:

(1) Avaliar a semelhança entre os *Top-k* itens presentes nas listas *P-list*, *A-list* e *C-list*.
(2) Avaliar a acurácia de cada estratégia implementada, no intuito de determinar se os itens recomendados são potencialmente relevantes para os usuários. Calcula-se a taxa de acerto de cada recomendador e encontra-se a taxa de acertos média gerada.
(3) Avaliar o nível de novidade, diversidade e serendipidade para cada uma das estratégias. Para cada um dos *ranking* obtidos, calcula-se o valor médio de surpresa obtido em cada recomendação *Top-k*, bem como a área sobre a curva (AUC).
(4) Avaliar o impacto de cobertura máxima nos cenários, verificando os itens recomendados por cada técnica. Procura-se analisar em qual parte da distribuição dos dados os itens recomendados pertencem, a fim de descobrir se a técnica é efetiva para atenuar o problema da *long-tail*.

4. ESTUDO DE CASO

Com base na metodologia de avaliação proposta, primeiramente, calcula-se a porcentagem de itens iguais recomendados pelas estratégias de recomendação não personalizadas. Com o diagrama de Venn, mostrado na Figura 1, pode-se notar que mais de 40% dos itens recomendados são semelhantes em todas as estratégias. Nota-se também que, para ambos os cenários, a estratégia de recomendar os itens mais bem avaliados se assemelha mais de 80% da estratégia de popularidade. Essa observação refere-se ao fato que os itens populares tendem a ser bem avaliados. Por sua vez, nota-se que 45% dos itens recomendados por Cobertura Máxima são diferentes das demais estratégias. Tal fato indica que as técnicas estudadas apresentam resultados distintos que podem ser relevantes para cenários reais.

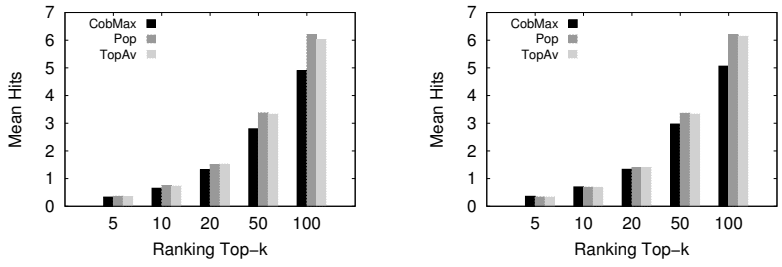
Fig. 1. Diagrama de Venn para representar a semelhança dos itens recomendados na lista *top-100*.



(a) ML-100k (b) ML-1M

Em seguida, calcula-se a taxa média de acertos gerada por cada uma das estratégias para cada cenário estudado, considerando as distintas listas *Top-k* propostas. Pode-se notar, com base no resultado mostrado na Figura 2, que a estratégia de popularidade possui uma maior taxa de acertos. Conforme o esperado, tais resultados referem-se ao fato de que recomendar itens populares é uma tarefa fácil, pois os usuários tendem a gostar dos itens mais assistidos pelos demais. Nota-se também que o resultado da estratégia de recomendar itens mais bem avaliados se assemelha muito a de popularidade, visto que os itens recomendados são 80% semelhantes.

Fig. 2. Taxa média de acertos obtidos por ambas as estratégias para os dois cenários.



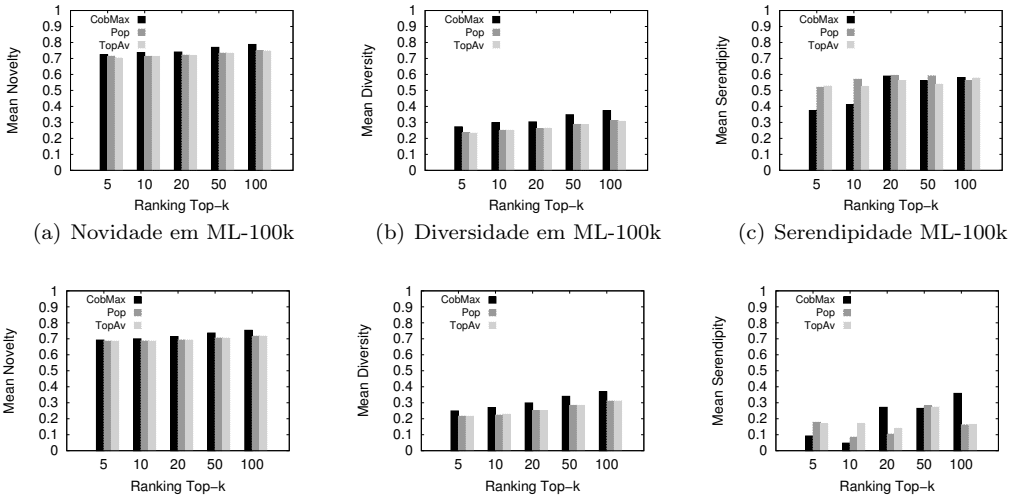
(a) ML-100k (b) ML-1M

Por fim, foi avaliado as métricas de novidade, diversidade e serendipidade, para as estratégias não personalizadas. A Figura 3 mostra a média dos valores obtidos por cada métrica, criando um contexto global de análise sobre todos as listas de recomendação propostas. Pode-se notar que, em ambos conjuntos de dados, a estratégia de Cobertura Máxima obteve resultados estatisticamente melhores que as demais estratégias, com uma confiança de 95% e um *p-value* = 0,01 utilizando o teste de *Wilcoxon* para distribuições não normais. A diferença entre as técnicas pode ser observada ao analisarmos a AUC para os *rankings* gerados pelas métricas na recomendação *top-100*. Conforme mostra a Tabela II, nota-se que utilizar a estratégia de Cobertura Máxima tende a trazer mais surpresa aos usuários finais. Para ambos os cenários avaliados, as recomendações geradas obtiveram um ganho considerável, apresentando em média um ganho de 5,5% em novidade, 18% em diversidade e 60% em serendipidade quando comparada as estratégias de consumo de massa. Vale ressaltar que a pequena porcentagem de ganho de novidade está relacionada ao conceito de utilidade presente na métrica, uma vez que os itens populares e mais bem avaliados são também relevantes para os usuários. Tais resultados mostram que, mesmo não sendo uma estratégia muito utilizada na literatura, Cobertura Máxima apresenta resultados potencialmente relevantes para o arcabouço de recomendações não personalizadas.

Table II. AUC de novidade, diversidade e serendipidade, na recomendação *top-100*.

	Coleção	Novidade	Diversidade	Serendipidade
Pop	*MovieLens 100k*	0.748535	0.311146	0.5605
	MovieLens 1M	0.716321	0.310218	0.162252
Av	*MovieLens 100k*	0.747797	0.306348	0.578043
	MovieLens 1M	0.717488	0.312109	0.164391
CM	*MovieLens 100k*	0.786936	0.373318	0.579967
	MovieLens 1M	0.753135	0.369072	0.358536

Fig. 3. Média dos valores de novidade, diversidade e serendipidade para ambas as estratégias.



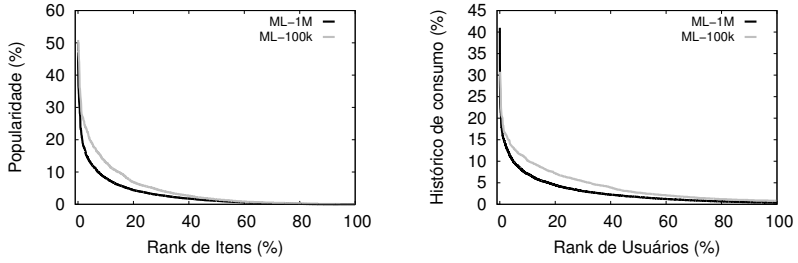
(a) Novidade em ML-100k (b) Diversidade em ML-100k (c) Serendipidade ML-100k

(d) Novidade em ML-1M (e) Diversidade em ML-1M (f) Serendipidade em ML-1M

4.1 Impacto da estratégia de Cobertura Máxima

O impacto de aplicar a estratégia de Cobertura Máxima se torna evidente em cenários cujo consumo dos usuários está fortemente relacionado a itens que despertam interesse em grande parte da população (i.e., consumo de massa). Ao relacionar a popularidade dos itens e o histórico de consumo dos usuários no domínio estudado, como mostram as Figuras 4(a) e 4(b), nota-se um efeito similar a *long-tail*[2] causado pelo consumo de massa. Em geral, a grande maioria dos usuários do domínio consomem poucos itens e estes itens consumidos são os populares. Neste contexto, recomendações relacionadas ao consumo de massa não representariam uma boa estratégia para a aquisição de potenciais consumidores de nichos, visto que estes recomendadores ficariam presos a recomendar itens da *cabeça* da distribuição (i.e., itens muito consumidos).

Fig. 4. Distribuições geradas pelas coleções de dados utilizadas.



(a) Popularidade dos itens (b) Consumo dos usuários

Entretanto, ao avaliar se os itens recomendados por Cobertura Máxima pertencem a *cabeça* ou a *cauda* da distribuição de popularidade, nota-se que tal estratégia é de fato válida para o consumo de nichos. Por meio do cálculo da derivada segunda sobre a distribuição de popularidade encontra-se o *joelho* da distribuição na qual pode-se dividir o conjunto de itens em 12% para a *cabeça* e 88% para a *cauda*. Desse modo, verifica-se que em torno de 21% e 15% dos itens recomendados por Cobertura Máxima, para os cenários do ML-100k e ML-1M, respectivamente, pertencem a *cauda* da distribuição. Em outras palavras, recomendações geradas por Cobertura Máxima tendem a atenuar o problema da *long-tail*, trazendo benefícios para diversas aplicações reais por satisfazer o consumo de nichos ao recomendar itens não populares.

[2]Uma distribuição de potência relativa ao consumo dos usuários, que tendem a consumir muitos itens populares.

5. CONCLUSÕES & TRABALHOS FUTUROS

Este trabalho teve por objetivo avaliar a aplicabilidade da estratégia de Cobertura Máxima quando aplicada a tarefa de recomendação não personalizada, a fim de atender adequadamente as preferências de usuários que se interessam por itens diferentes do gosto comum da população (i.e., consumo de nichos). No intuito de avaliar a relevância prática da estratégia, foi calculado os níveis de novidade, diversidade e serendipidade sobre as coleções de filmes do *MovieLens* e comparado os resultados encontrados com os das estratégias de recomendar itens mais populares e mais bem avaliados (i.e., consumo de massa). De maneira geral, as recomendações geradas apresentam em média um ganho de 5,5% em novidade, 18% em diversidade e 60% em serendipidade quando comparada as estratégias de consumo de massa que recomendam itens mais populares ou mais bem avaliados. Nota-se que as recomendações geradas por Cobertura Máxima consistem também em itens não populares, contendo em média 18% de itens distintos do gosto comum da população geral. Tais itens devem ser recomendados para cenários onde almeja-se satisfazer as distintas preferências de potencias usuários do domínio, satisfazendo o consumo de nichos. Além disso, pode-se notar que mesmo com esses ganhos, a recomendação por Cobertura Máxima não se distancia das demais estratégias quando comparada as métricas clássicas de acurácia.

Dessa forma, pode-se dizer que estes resultados respondem as questões levantadas por este trabalho, ressaltando a relevância de se apresentar itens distintos ao gosto comum da população. Os resultados também evidenciam a necessidade de melhor analisar a estratégia de Cobertura Máxima dentro do contexto de SsR. Pretendemos, futuramente, combiná-la a outras estratégias comuns na literatura, ou mesmo adicionar restrições que sejam capaz de melhorar a taxa de acertos dessa estratégia, bem como os índices de surpresa.

REFERENCES

Alon, N., Awerbuch, B., and Azar, Y. The online set cover problem. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*. ACM, pp. 100–105, 2003.

Anderson, C. *The long tail: Why the future of business is selling less of more*. Hyperion, 2006.

Batet, M., Moreno, A., Sánchez, D., Isern, D., and Valls, A. Turist@: Agent-based personalised recommendation of tourist activities. *Expert Systems with Applications* 39 (8): 7319–7329, 2012.

Bobadilla, J., Ortega, F., Hernando, A., and Gutiérrez, A. Recommender systems survey. *Knowledge-Based Systems* vol. 46, pp. 109–132, 2013.

Chvatal, V. A greedy heuristic for the set-covering problem. *Mathematics of operations research* 4 (3): 233–235, 1979.

Feige, U. A threshold of ln n for approximating set cover. *Journal of the ACM*, 1995.

Hammar, M., Karlsson, R., and Nilsson, B. J. Using maximum coverage to optimize recommendation systems in e-commerce. In *Proceedings of the 7th ACM conference on Recommender systems*. ACM, pp. 265–272, 2013.

Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 230–237, 1999.

Hinz, J. D. O. and Eckert, D.-K. J. The impact of search and recommendation systems on sales in electronic commerce. *Business & Information Systems Engineering* 2 (2): 67–77, 2010.

Iskold, A. The art, science and business of recommendation engines. *Retrieved April* vol. 5, pp. 2012, 2007.

Koren, Y., Bell, R., and Volinsky, C. Matrix factorization techniques for recommender systems. *Computer* (8): 30–37, 2009.

Michael, R. G. and David, S. J. Computers and intractability: a guide to the theory of np-completeness. *WH Free. Co., San Fr*, 1979.

Ricci, F., Rokach, L., and Shapira, B. *Introduction to recommender systems handbook*. Springer, 2011.

Schafer, J. B., Konstan, J., and Riedl, J. Recommender systems in e-commerce. In *Proceedings of the 1st ACM conference on Electronic commerce*. ACM, pp. 158–166, 1999.

Schein, A. I., Popescul, A., Ungar, L. H., and Pennock, D. M. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 253–260, 2002.

Vargas, S. and Castells, P. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*. ACM, pp. 109–116, 2011.

Zhang, Y. C., Séaghdha, D. Ó., Quercia, D., and Jambor, T. Auralist: introducing serendipity into music recommendation. In *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, pp. 13–22, 2012.

Image segmentation via superpixels self-organized motion

Roberto Alves Gueleri[1], Liang Zhao[1]

University of São Paulo, Brazil
gueleri@icmc.usp.br

Abstract. We present a segmentation technique based on patterns that emerge from a system of moving particles. It takes inspiration from the flocking formation that can be seen in nature, e.g. in large groups of birds and fish, which display a complex, coordinated motion without the guidance of any leader. Firstly, we downsample the image by dividing it into superpixels, each of which will be a moving particle. After assigning random directions of motion for every particle, we let the dynamics begin and, after some time, it is expected that the new particle arrangement reflects useful image features, i.e., the segments we search for. We present a set of experimental results, which enables us to discuss some pros and cons of this approach.

Categories and Subject Descriptors: I.4.6 [**Image Processing and Computer Vision**]: Segmentation; I.5.3 [**Pattern Recognition**]: Clustering

Keywords: clustering, collective motion, flocking, image segmentation, machine learning, self-organizing systems

1. INTRODUCTION

Image segmentation is the process of partitioning an image into a set of segments, by grouping pixels that share similar properties. It plays a fundamental role in the fields of computer and machine vision. The goal of image segmentation is thus to provide a meaningful separation of different objects in a scene, e.g. isolating an object from its background.

Several different approaches have been proposed in the image-segmentation literature. For example, histogram-based methods consider peaks and valleys of histograms to decide where to split the image [Ohlander et al. 1978; Tobias and Seara 2002]. They usually employ pixel color or intensity as the source for the histogram, so essentially they do not consider individual pixel xy-location. Those methods can be very fast, even when requiring more than one pass by recursive refinement of segmented regions [Ohlander et al. 1978]. Another example are region-growing methods, which start with small regions — or single pixels — and iteratively merge neighboring regions according to some criteria [Adams and Bischof 1994; Nock and Nielsen 2004; Shih and Cheng 2005; Alpert et al. 2012]. Another common approach for image segmentation is found in graph-based methods [Grady 2006; Couprie et al. 2011; Peng et al. 2013; Wang et al. 2015]. They model the image as a graph, by making every pixel — or small region of pixels — be a vertex of the graph, and making every edge connect neighboring pixels and carry a weight value that defines the similarity between a pair of pixels.

In this paper we propose an image-segmentation technique based on superpixels motion in a multidimensional space. This technique takes inspiration from the flocking formation that can be seen in nature, specially in large groups of birds and fish [Vicsek and Zafeiris 2012]. Flocks are groups of individuals that move in a coordinated fashion, i.e., individuals that move in the same direction and close to each other. This coordinated motion emerges even in the absence of a leader, what makes it a self-organized phenomenon. Each superpixel is treated as an individual, denoted here as a *particle*.

This work has been supported by The State of São Paulo Research Foundation (FAPESP) (Project 2013/08666-8) and by the Brazilian National Research Council (CNPq).

Starting from random directions, each particle moves in accordance with its neighborhood, taking the same direction of motion. The closer two particles are, the more intense will be the influence over each other. The magnitude of velocity is held the same for all particles, i.e., we model a sort of self-propelled motion, such that no particle is faster than any other. At the end of the process, groups — the segments — are established among those superpixels that take the same direction of motion, consequently keeping themselves close to each other.

This paper is organized as follows. In Section 2, we formally describe the flocking dynamic model proposed here. In Section 3, we present a set of experiments and the results achieved by the flocking segmentation technique. Also, we make a comparison to an well-known algorithm, namely Potts Segmentation [Storath and Weinmann 2014; Storath et al. 2015], so we can discuss some pros and cons of applying our proposed technique. Finally, some conclusions are drawn in Section 4.

2. MODEL DESCRIPTION

A *superpixel* represents a small group of pixels, usually pixels that are close to each other and share similar properties, like color, intensity, etc. Forcing every single pixel to be a moving particle would make the system computationally expensive. This is the main reason of employing superpixels: a dynamic system that contains a moderate number of particles, such that each particle represents a small portion of image well enough.

The overall segmentation process proposed in this paper makes use of *k-means*, one of the most popular clustering algorithms [MacQueen 1967; Lloyd 1982]. Every cluster is denoted by its centroid, i.e., by the vector that indicates the average of its objects. Then the algorithm aims at finding k centroids that minimizes the squared error of the resultant partition. In our technique, k-means is applied in two distinct stages: (1) in order to group the pixels and result in superpixels; and (2) after the superpixels' motion by the dynamic system — and flocks being emerged —, in order to group the superpixels and result in the final image segments. It is worth noting that the dynamic process does not actually group the particles, i.e, does not assign cluster IDs, but instead it changes the *arrangement* of those particles in space, highlighting useful patterns and making it easier for an actual clustering algorithm to correctly group those particles into image segments. Different algorithms could also be employed in the first stage to obtain the superpixels, e.g., the SLIC algorithm[1] [Achanta et al. 2012]. Many others could be applied to the second stage too, because the literature of clustering algorithms is extensive. The reason for choosing k-means for both stages lies on its simplicity, efficiency, and good results that can be achieved, although the usage of different algorithms may lead to some other interesting results as well.

The space through which the particles move is composed of five or more dimensions:

—The first two are the horizontal and vertical image dimensions.
—The following three describe the image color attributes. Two color models were considered here: RGB e CIE L*a*b*.
—Additional dimensions reduce the risk of collision between different groups of particles, but make the process more expensive. Three additional dimensions were considered here, summing a total of eight dimensions.

The overall segmentation process consists of the following stages:

(1) Given the maximum desired number of superpixels — sampling —, the initial placement of their centroids forms a regular grid over the image. The horizontal and vertical distances between

[1]Actually, the SLIC algorithm is based on k-means too and its approach is very similar to the one described here, but SLIC is more sophisticated.

adjacent centroids should be as similar as possible. The definitive number of superpixels will be the largest that still makes a regular grid, without exceeding the maximum desired number. All the centroids have zero initial coordinate, except for the first two dimensions. The data are normalized such that the color components vary in the interval $[-0.5; +0.5]$. Also, the horizontal and vertical distances between adjacent centroids are made to be as close as possible to 1. Since those distances are not necessarily identical, we just make the average of both be 1.

(2) The k-means algorithm is applied in order to optimize the centroid positions. By doing so, each color dimension ends up having non-zero values. This is a fundamental step, because it defines the initial arrangement of the dynamical system, thus having a strong influence on the final segmentation result as well. Remember that each centroid represents a superpixel, which in turn represents a group of pixels. Therefore, until the end of the process, every centroid will be bound to all the pixels it represents. Every centroid will be a moving particle $\mathbf{x}_i(t)$, where the iteration index t (time) starts being 0, what denotes the initial system configuration.

(3) The particles motion, i.e., the dynamic process itself. But just before that, we need a data renormalization so as to expand the color components interval, which were $[-0.5; +0.5]$ and now have to be the image size, in terms of number of superpixels. Similarly to the normalization done before, here we consider the average of both the horizontal and vertical sizes. This renormalization assigns the same level of "importance" for color components and xy-image-coordinates. The mathematical expressions that precisely define the particles motion is presented just ahead in this section. We also need a stop criterion, so we define the concept of *radius*, which is the Euclidean distance of the farthest particle from the space origin. In order to do so, it is important that the set of particles is centralized around the origin before the dynamics begin. In the experiments presented in this paper, the process is stopped when it reaches a radius 16 times bigger than the initial one. It is expected that useful patterns have already been emerged by this moment.

(4) The k-means algorithm is applied again, but this time to group the particles in their final arrangement, just after stopping the motion. Since every particle is a superpixel, this final grouping results in the image segments we want. K-means can be asked to come up with different number k of segments. Values between 2 and 6 have been used in this paper. Unlike the first application of k-means, in this final stage it assigns random initial positions for the centroids. Therefore, different executions may lead to different results, so we run k-means 50 times and select the best partition based on the quadratic error it results.

Precisely, the motion of every particle $\mathbf{x}_i$ is governed by the following system:

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \alpha_1(t) \cdot \hat{\mathbf{v}}_i(t) \tag{1}$$

$$\mathbf{v}_i(t+1) = \hat{\mathbf{v}}_i(t) + \alpha_2 \cdot \hat{\mathbf{v}}_i^{\text{chg}}(t) \tag{2}$$

$$\mathbf{v}_i^{\text{chg}} = \sum_{\mathbf{x}_j \in \text{nei}(\mathbf{x}_i)} \left(\hat{\mathbf{v}}_j - \hat{\mathbf{v}}_i\right) \cdot \frac{1}{\|\mathbf{x}_i - \mathbf{x}_j\|^{\beta}}. \tag{3}$$

The initial velocity $\mathbf{v}_i(0)$ of all particles is randomly assigned. The initial placement $\mathbf{x}_i(0)$ of particles is the arrangement of centroids — superpixels — that forms a grid over the image. Those centroids have their location optimized by the k-means algorithm, as discussed before.

The symbols $\hat{\mathbf{v}}_i$ and $\hat{\mathbf{v}}_i^{\text{chg}}$ denote unit vectors, i.e., $\hat{\mathbf{v}}_i = \mathbf{v}_i/\|\mathbf{v}_i\|$ and $\hat{\mathbf{v}}_i^{\text{chg}} = \mathbf{v}_i^{\text{chg}}/\|\mathbf{v}_i^{\text{chg}}\|$. Specially, in Equation 1, $\hat{\mathbf{v}}_i$ enforces that all the particles move at the same speed. $\|\mathbf{x}_i - \mathbf{x}_j\|$ is the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$. The Euclidean distance was employed in this paper.

The symbol $\text{nei}(\mathbf{x}_i)$ denotes the neighborhood of $\mathbf{x}_i$. That neighborhood has to be updated regularly as the particles move. Instead of making all the particles interact with every other, restricting the interaction to take place only via neighborhood is necessary in order to keep a feasible computational cost. In this paper, the 8 nearest neighbors were considered. It is worth noting that

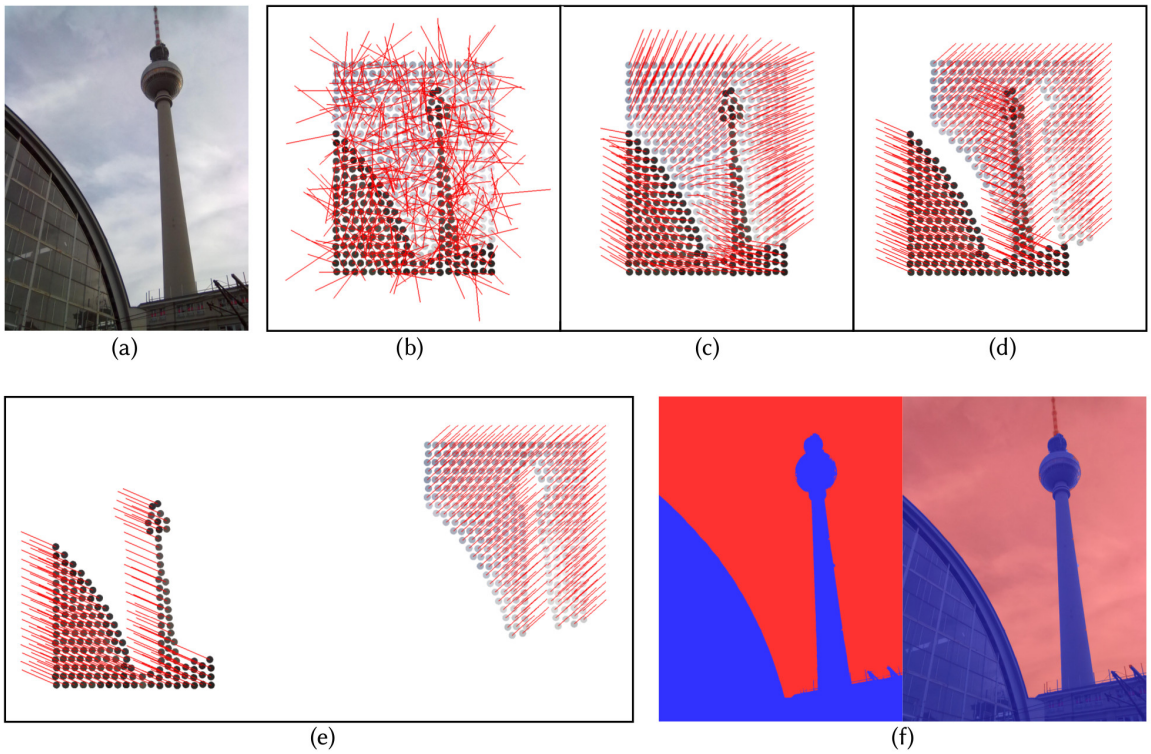


Fig. 1. Illustration of the segmentation process. (a): Source image. (b): Initial configuration of the dynamic system. Each particle is a superpixel, which is denoted by its centroid. All the velocity vectors point to random directions (red line segments). Specifically for this example, six dimensions was employed: horizontal and vertical image dimensions, plus three color components (RGB in this case), and finally one additional dimension to avoid collisions. However, only the projection on the first two dimensions is displayed in this figure. (c, d): After some time. (e): The instant when the dynamical process is stopped and the particles are grouped. (f): The result of obtaining two segments, by using the k-means algorithm.

a space-partitioning technique was employed in order to avoid a quadratic time cost of finding the nearest neighbors. A detailed description of that technique is presented in [Gueleri et al. 2014]. The neighborhood updating takes place at every 50 iterations.

α_1 and α_2 are parameters that define the intensity by which the position and the velocity of particles vary, respectively. Notice that α_1 is a function of t whereas α_2 is constant throughout the process. Since all velocities point to random directions at the beginning, it is reasonable to let those velocities accommodate first, and only after that to allow the particles to perform a more intensive motion. In order to achieve such behavior, α_1 is progressively increased as t advances. Precisely,

$$\alpha_1 = 10^{\text{exp}} \cdot t^3, \tag{4}$$

such that different values of exponent exp lead to distinct behaviors, therefore they are studied in this paper. Smaller exponent gives more time for the particles to get aligned, so the motion becomes visible only after that alignment. We set $\alpha_2 = 0.2$ in our experiments. β is also a parameter and was set to 2. An illustration of the segmentation process proposed in this paper is presented in Figure 1.

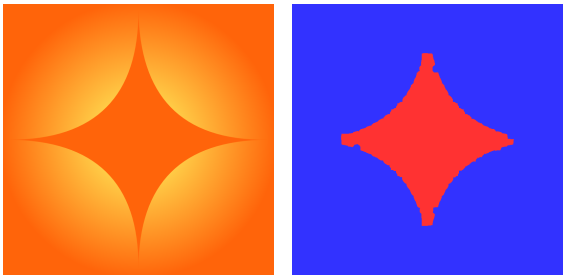


Fig. 2. [Left]: An artificial image used in the experiments. [Right]: One of the good results achieved. Parameter setting: RGB color mode, 2000 samples, exp $= -12$, and $k = 2$ (2 final segments).

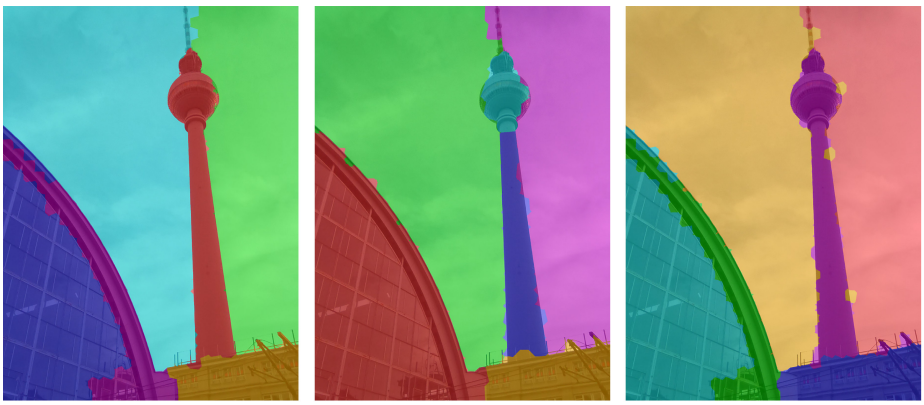


Fig. 3. Some interesting results for 6 segments.

3. EXPERIMENTAL RESULTS

Experiments that have been performed on both an artificial image and a real photograph are presented in this section. First, let us consider the results on the artificial image, shown in Figure 2. That image imposes some challenges, specially at the four tips of the star, which tend to have the same color of the gradient background. Despite some irregularities on the edges, we can see that the proposed technique is able to separate the star from the background. Furthermore, all the background ends up as a single segment, not being broken into smaller ones, although different parameters may lead to different results, as discussed in this paper.

Now we present the results on a real photograph, the same one shown in Figure 1. Just like the previous artificial image, this photograph imposes some challenges too. The sky background is easily distinguishable from the front building elements. However, those building elements have similar colors and intensities, what makes it difficult to correctly isolate those elements. On the other hand, we want all the background to result in a single element. However, the sky is a little clouded and spatially separated by the tower, what tends to break the background sky into more than one segment. Some interesting results achieved by the proposed technique are presented in Figure 3. By using the CIE L*a*b* color mode, we notice that the borders between different segments appear to be more irregular. That was a common observation through our set of experiments. For that reason, we focus on the RGB results in this paper.

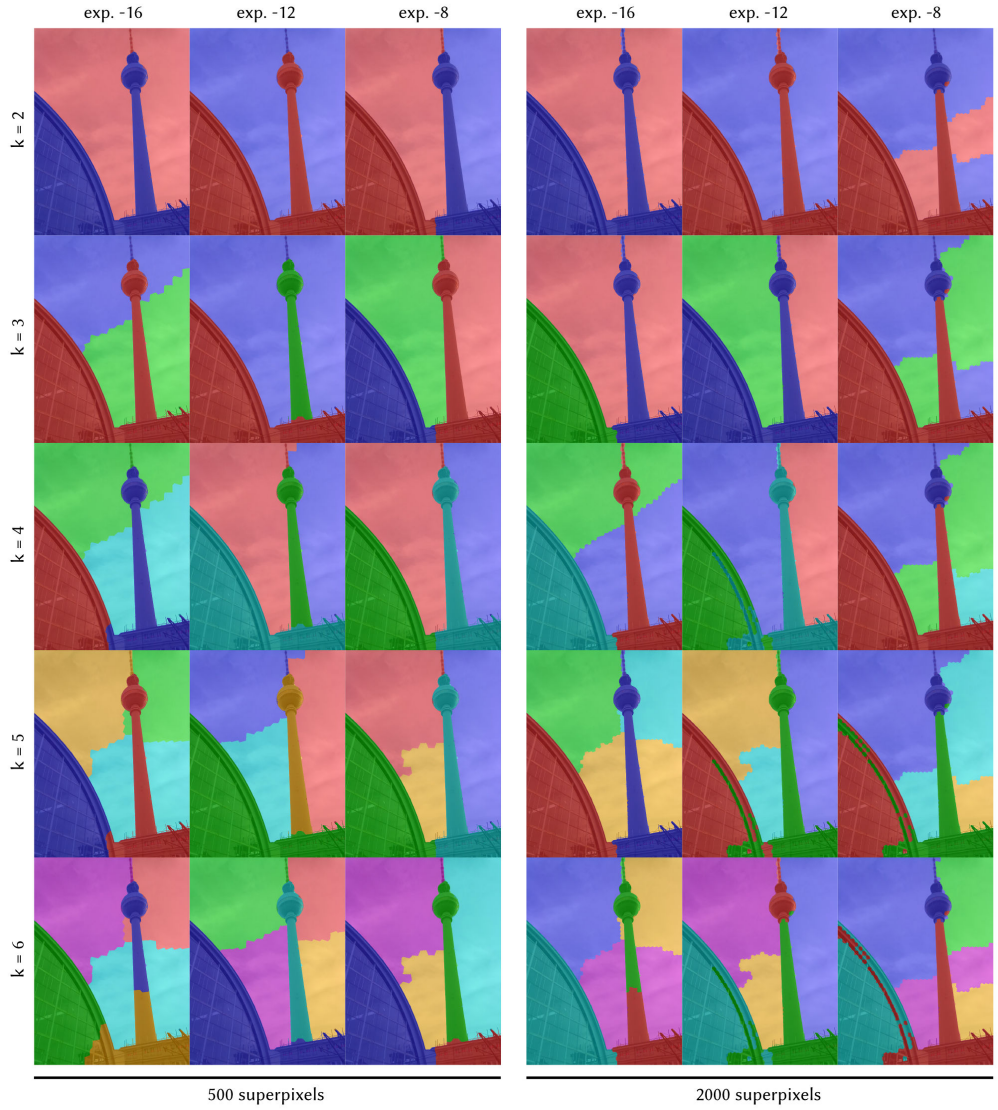


Fig. 4. Results of different parameter settings: sampling size (500 or 2000) vs. exponent value (-16, -12, or -8) vs. k (2, 3, 4, 5, or 6). Only the results of RGB color mode is shown.

Results of different parameter settings are presented in Figure 4. Due to the randomness of assigning the particles' initial velocity, different executions may lead to slightly different results. However, the overall system behavior is the same for different executions that employ the same parameter setting, despite the random initial configuration.

Some observations can be drawn from those experiments, concerning the number of segments k, the exponent value, and the sampling size. Smaller exponent (more negative) leads to better solutions for small k (e.g., $k = 2$, front buildings vs. sky background). The particles are given more time to get aligned, before starting a more intense motion. Big scene objects are able to become entirely aligned (e.g., all the sky). However, it is not good for larger k, because those big objects become overly mixed, thus hard to properly break them into their smaller components. It seems that there is a correlation here, because large exponent, in turn, presents poor solutions for small k and good solutions for large k. For instance, for large k, notice that the tower is broken by using small exponent values, whilst it

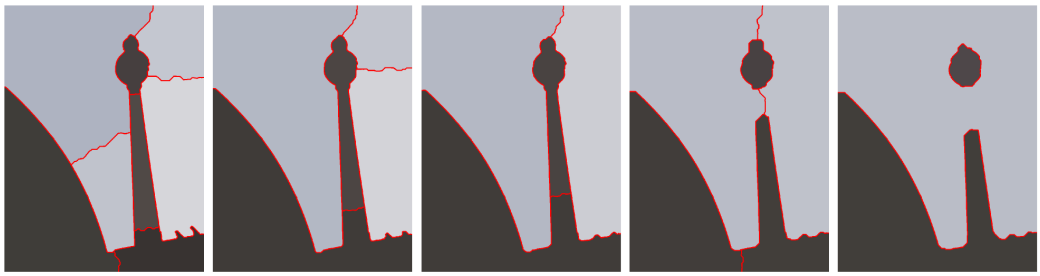


Fig. 5. Results of applying the Potts Segmentation algorithm. The source-image size is 900px × 1200px. From left to right: $\gamma = 6$, 12, 24, 48, and 96.

remains intact for large exponent. We can somewhat conclude that a larger exponent (less negative, larger α_1) should be employed when we want more image segments. In fact, larger exponents lead to a more "segmented" behavior of the moving particles.

As we would expect, large sampling size better defines small scene objects. For instance, the small antenna on the top of the tower remains isolated from the sky only by using large sampling. Also, it seems like large sampling works better with small exponent, otherwise large sampling may introduce some undesirable effects. Small sampling size, on the other hand, although unable to capture small details, it usually leads to more consistent segments when considering the image as a whole, specially for large exponent values.

Finally, in order to compare our results to those from another well-know technique, we have also applied the Potts Segmentation algorithm [Storath and Weinmann 2014; Storath et al. 2015], which presents similar goals and properties. The results were obtained with the aid of Icy software[2], by using its Potts Segmentation plugin[3]. Different values of the scale parameter γ were used. The remaining parameters were kept on defaults: 0.001 for initial value of μ, 2 for step value of μ, and isotropism enabled. Potts Segmentation also offers further merging or refinement of the partition obtained in the first step, but it is a human-interactive procedure and was not considered here. Only the non-interactive step was performed. The results of employing different values of γ are displayed in Figure 5. By using those values of γ, we achieve 3, 4, 5, and 8 segments, what is comparable to the range from 2 to 6 that we used for our technique. Smaller values of γ usually lead to more segments.

The results from Potts Segmentation look consistent, although they still display some undesirable effects. For instance, as we increase γ in order to obtain just two segments — the front building elements apart from the sky background —, it fails by losing part of the tower. Also, even for larger number of segments, the border at the base of the tower shifts from its ideal position as we increase γ — that base border moves upward the middle of the tower.

The sky background is challenging for both techniques. However, by applying our technique, we are able to achieve a correct separation by asking for 2 or 3 segments, i.e., keep all the sky as one single segment isolated from the buildings, without breaking the tower at some undesirable part. We do not achieve such correct result for all the parameter settings, but for many of them, specially by asking for 2 segments.

Concerning the base of the tower, our technique may either separate at the middle of the tower, separate at the correct position, or not separate at all, depending on the sampling size and the number of segments k. However, unlike Potts Segmentation, when separating at the correct position it does not display that "shifting" effect.

[2] `http://icy.bioimageanalysis.org`
[3] `http://icy.bioimageanalysis.org/plugin/Potts_Segmentation`

4. CONCLUSIONS

Through the set of experiments presented in Section 3, we see that the technique based on particle motion displays interesting properties when applied to image segmentation. Consistent results have been obtained for a broad range of parameters. Furthermore, we even expect that the decentralized, self-organized flocking model tends to be useful and robust not only for image segmentation, but also for other applications where patterns have to be highlighted from a large set of vector data. With this paper, we therefore expect not just to bring a new segmentation algorithm, but more importantly to bring ideas that may motivate new studies and applications of self-organized dynamical systems to solve complex problems that involve large datasets. New dynamical models, different from that one presented in Section 2, may probably lead to different results as well. More specifically, in Section 2 we described a normalization procedure that gives the same importance level for both xy-location and color dimensions. Unbalancing it by means of different normalizations may result in segments that prioritize either the spatial distance (in terms of xy-coordinates) or the pixel color.

REFERENCES

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (11): 2274–2281, 2012. DOI: 10.1109/TPAMI.2012.120.

Adams, R. and Bischof, L. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16 (6): 641–647, 1994. DOI: 10.1109/34.295913.

Alpert, S., Galun, M., Brandt, A., and Basri, R. Image segmentation by probabilistic bottom-up aggregation and cue integration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34 (2): 315–327, 2012. DOI: 10.1109/TPAMI.2011.130.

Couprie, C., Grady, L., Najman, L., and Talbot, H. Power watershed: a unifying graph-based optimization framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (7): 1384–1399, 2011. DOI: 10.1109/TPAMI.2010.200.

Grady, L. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (11): 1768–1783, 2006. DOI: 10.1109/TPAMI.2006.233.

Gueleri, R. A., Cupertino, T. H., de Carvalho, A. C., and Zhao, L. A flocking-like technique to perform semi-supervised learning. In *Proceedings of the "2014 International Joint Conference on Neural Networks – World Congress on Computational Intelligence (IJCNN/WCCI 2014), Beijing, China"*. pp. 1579–1586, 2014. DOI: 10.1109/IJCNN.2014.6889434.

Lloyd, S. P. Least squares quantization in pcm. *IEEE Transactions on Information Theory* 28 (2): 129–137, 1982. DOI: 10.1109/TIT.1982.1056489.

MacQueen, J. B. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*. pp. 281–297, 1967.

Nock, R. and Nielsen, F. Statistical region merging. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (11): 1452–1458, 2004. DOI: 10.1109/tpami.2004.110.

Ohlander, R., Price, K., and Reddy, D. R. Picture segmentation using a recursive region splitting method. *Computer Graphics and Image Processing* 8 (3): 313–333, 1978. DOI: 10.1016/0146-664X(78)90060-6.

Peng, B., Zhang, L., and Zhang, D. A survey of graph theoretical approaches to image segmentation. *Pattern Recognition* 46 (3): 1020–1038, 2013. DOI: 10.1016/j.patcog.2012.09.015.

Shih, F. Y. and Cheng, S. Automatic seeded region growing for color image segmentation. *Image and Vision Computing* 23 (10): 877–886, 2005. DOI: 10.1016/j.imavis.2005.05.015.

Storath, M. and Weinmann, A. Fast partitioning of vector-valued images. *SIAM Journal on Imaging Sciences* 7 (3): 1826–1852, 2014. DOI: 10.1137/130950367.

Storath, M., Weinmann, A., Frikel, J., and Unser, M. Joint image reconstruction and segmentation using the potts model. *Inverse Problems* 31 (2), 2015. DOI: 10.1088/0266-5611/31/2/025003.

Tobias, O. J. and Seara, R. Image segmentation by histogram thresholding using fuzzy sets. *IEEE Transactions on Image Processing* 11 (12): 1457–1465, 2002. DOI: 10.1109/TIP.2002.806231.

Vicsek, T. and Zafeiris, A. Collective motion. *Physics Reports* vol. 517, pp. 71–140, 2012. DOI: 10.1016/j.physrep.2012.03.004.

Wang, X., Tang, Y., Masnou, S., and Chen, L. A global/local affinity graph for image segmentation. *IEEE Transactions on Image Processing* 24 (4): 1399–1411, 2015. DOI: 10.1109/TIP.2015.2397313.

Quando a Amazônia Encontra a Mata Atlântica: Empilhamento de Florestas para Classificação Efetiva de Texto

Raphael R. Campos, Marcos A. Gonçalves, Thiago Salles

Dep. de Ciência da Computação - Universidade Federal de Minas Gerais (UFMG)
Av. Antônio Carlos 6627 - ICEx - 31270-010 Belo Horizonte , Brasil
{rcampos, mgoncalv, tsalles}@dcc.ufmg.br

Abstract. Floresta Aleatória (FA) tem sido um dos classificadores mais bem-sucedidos em uma enorme variedade de tarefas de classificação automática, comparável, e algumas vezes superior, ao Support Vector Machine (SVM). Portanto, é natural que haja várias tentativas de melhorias da mesma. Nesse trabalho nós estudamos o papel desempenhado pelas derivações da FA encontradas na literatura e propostas por nós - Árvores Extremamente Aleatórias (AEA), BROOF, BERT, Lazy Random Forest e Lazy Extra-Trees - na efetividade e diversidade de sistemas multi-classificadores baseados em empilhamento. Nossos experimentos mostram, que a combinação desses métodos baseados em Floresta Aleatória tem papel importante no aumento do poder de generalização de métodos de comitês de classificadores em tarefas de classificação textual, havendo melhora nos resultados em todos os conjuntos de dados testados quando comparados aos melhores classificadores de base estado-da-arte, obtendo-se ganhos de até 8% na métrica MacroF1 e 16% na métrica MicroF1. Além disso, foram obtidos ganhos de até 5% e 5,5% em MacroF1 e MicroF1 quando comparado ao comitê de classificadores estados-da-arte em classificação de texto (SVM, KNN, Naive Bayes).

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: Aprendizado de Máquina, Comitê de Classificadores, Classificação Automática de Texto, Empilhamento, Florestas Aleatórias, Sistema Multi-Classificadores

1. INTRODUÇÃO

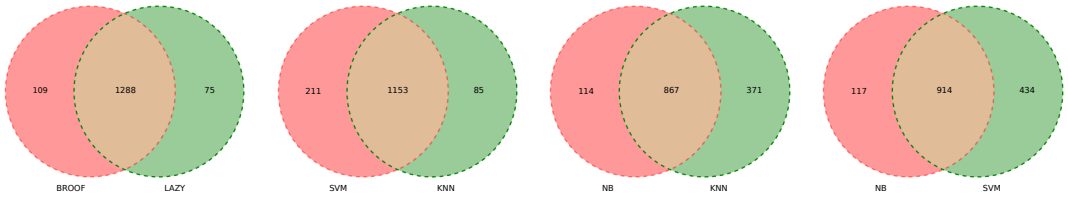
Desde o advento da *Web* o volume de dados disponível tem crescido de modo impressionante. Uma forma de organizar e lidar com essa quantidade de informação é o uso de métodos automáticos de classificação textual. Diferentemente de domínios convencionais de aprendizagem de máquina, é comum em várias tarefas de classificação de texto haver milhares de atributos a serem considerados (alta dimensionalidade), enquanto muitos deles são esparsos na coleção de documentos. Além disso, muitos atributos nesse tipo de aplicação são ruidosos ou irrelevantes[Khan et al. 2010], o que reduz a eficácia de alguns métodos de classificação. Inspirado no sucesso em diferentes tarefas de classificação, trabalhos recentes [Salles et al. 2015; Danesh et al. 2007; Dong and Han 2004] exploram diferentes técnicas de aprendizado de máquina no contexto de classificação textual, tais como *Support Vector Machine* (SVM), *k-vizinhos mais próximos* (kNN), *Naive Bayes* (NB) e *Floresta Aleatória* (FA).

Todos os métodos supracitados pertencem a diferentes famílias de classificadores. Devido às características distintas dessas famílias, é intuitivo que esses métodos sejam complementares, e por isso geralmente são escolhidos para compor sistemas multi-classificadores[Ting and Witten 1999; Danesh

Esse trabalho foi parcialmente financiado pelos projetos InWeb (bolsa MCT/CNPq 573871/2008- 6) e MASWeb (bolsa FAPEMIG/PRONEX APQ-01400-14), e pelas bolsas individuais dos autores concedidas por CNPq , CAPES e FAPEMIG.

et al. 2007]. Além das características complementares das diferentes famílias de classificadores, pode haver também complementariedade entre variações de classificadores baseados em um mesmo paradigma. Mais especificamente, as derivações da Floresta Aleatória podem também apresentar tais características complementares.

A Floresta Aleatória (FA) tem sido um dos classificadores mais bem-sucedidos em uma enorme variedade de tarefas de classificação[Fernández-Delgado et al. 2014]. Apesar de ser um classificador com grande capacidade de generalização, tem sido mostrado que modelos FA sofrem do problema de sobre-ajuste[Segal 2004], tendo sua eficácia degradada e prejudicada na presença de muitos atributos irrelevantes e ruidosos, o que é característico em tarefas de classificação de texto. Nos últimos anos, alguns classificadores baseados em FA têm produzidos resultados estado-da-arte utilizando estratégias distintas para amenizar o problema de sobre-ajuste sofrido por ela. Esses métodos aprendem modelos de classificação focando em sub-regiões específicas do espaço de entrada, na esperança de filtrar atributos e dados irrelevantes. Esses métodos são a Floresta *Lazy* [Salles et al. 2015] e o BROOF [Salles et al. 2015]. O primeiro treina uma FA para cada instância de teste utilizando seus respectivos k vizinhos mais próximos como conjunto de treino. O segundo, por sua vez, combina suavemente *boosting* e *florestas aleatórias*, usando o erro *Out-Of-Bag (OOB)* como uma estimativa menos enviesada de erro. Bem como os métodos clássicos, esses novos métodos baseados em FA abordam estratégias distintas para mitigar o problema de sobre-ajuste desse modo nos levando a intuição de terem características complementares. Nossa intuição pode ser empiricamente evidenciada pela Figura 1 que mostra um diagrama de Venn que representa a interseção entre os exemplos de teste corretamente classificados por cada método. A partir do diagrama, pode-se calcular os graus de discordância[1][Kuncheva and Whitaker 2003] dos pares de classificadores $Dis_{<BROOF,Lazy>}$, $Dis_{<SVM,kNN>}$, $Dis_{<NB,kNN>}$ e $Dis_{<NB,SVM>}$ que são respectivamente: 0.32, 0.29, 0.33 e 0.37. Pode-se notar que os pares têm graus semelhantes, isso nos da forte evidência que os métodos Lazy e BROOF são tão complementares quantos os métodos clássicos.



(a) BROOF/LAZY (b) SVM/KNN (c) NB/KNN (d) NB/SVM

Fig. 1: Diagrama de Venn do conjunto de dados 4UNI - representando a interseção entre as instâncias de teste corretamente classificadas por cada estratégia.

Nesse contexto, esse trabalho almeja preencher duas lacunas na literatura[2]. Primeiro, investigar o papel dos recém-desenvolvidos métodos baseados em FA na geração de sistemas multi-classificadores robustos e com alta efetividade em tarefas de classificação quando estes são combinados usando *empilhamento* [Wolpert 1992]. Segundo, analisar como os métodos baseados em FA se comportam e relacionam com os métodos clássicos de classificação de texto (SVM, NB e kNN) em sistemas multi-classificadores. Nossos resultados experimentais mostram, que a combinação desses métodos baseados em floresta aleatória tem papel importante no aumento do poder de generalização de métodos de comitês de classificadores em tarefas de classificação textual, havendo melhora nos resultados em todos os

[1]Métrica de complementariedade(diversidade) entre pares de métodos. Calculada pela equação $\frac{|C|}{|D_{teste}|}$, onde D_{teste} é o conjunto de teste e $C \subset D_{teste}$ é o conjunto de exemplos corretamente classificados por apenas um dos métodos.
[2]Até onde sabemos não há estudo envolvendo Floresta Aleatórias e suas derivações no contexto de empilhamento e sistemas multi-classificadores para classificação de texto.

conjuntos de dados testados quando comparados aos melhores classificadores de base estado-da-arte, obtendo-se ganhos de até 8% na métrica MacroF_1 e 15% na métrica MicroF_1. Ademais, foram obtidos ganhos de até 5,5% em MacroF_1 quando comparado à combinação de classificadores estados-da-arte em classificação de texto.

O restante desse artigo está organizado do seguinte modo. Seção 2 revisa conceitos importantes para entendimento desse trabalho, tais como Floresta Aleatória e *Empilhamento*. Seção 3 apresenta uma visão geral sobre as extensões da Floresta Aleatória utilizadas nesse trabalho. Seção 4 apresenta nossa avaliação experimental, resultados e análises. Seção 5 conclui o artigo.

2. REVISÃO LITERÁRIA

2.1 Florestas Aleatórias

Proposto em [Breiman 2001], Floresta Aleatória (FA) tem sido um dos classificadores mais bem-sucedidos em uma enorme variedade de tarefas de classificação automática, comparável, e algumas vezes superior, ao Support Vector Machine (SVM).

Floresta Aleatória é um comitê de árvores de decisão, cada uma das quais é construída da seguinte forma: Primeiramente, o procedimento de *bagging (bootstrap aggregating)* é executado. Introduzido por [Breiman 1996], *bagging* é um método que almeja controlar variância criando várias versões do classificador e tirando a média deles. Dado o conjunto de treino D_{treino}, um procedimento de *bagging* produz M conjuntos de treino $D^i_{train}|_{i=1}^M$ amostrando com substituição do conjunto de dados original. Segundo, cada árvore $h_i|_{i=1}^M$ é treinada considerando os recém-criados conjuntos de treino e um subconjunto de atributos F_i é amostrado do conjunto de atributos original F sem substituição, onde $|F_i| << |F|$. Finalmente, a predição final é dada pela média de cada h_i, $1 \leq i \leq M$.

Essa série de procedimentos aleatórios, tais como *bagging* do conjunto de treino e seleção aleatória de atributos, produzem um comitê de árvores de decisão com baixa correlação. Árvores com correlação reduzida é um dos aspectos fundamentais que garantem a alta efetividade do classificador FA[Breiman 2001]. A fim de alcançar outro aspecto fundamental necessário para construir árvores com melhores capacidades de predição do que aleatório, as árvores são tipicamente construídas até sua profundidade máxima. Considerando a média de várias árvores descorrelacionadas como a predição final do modelo, pode-se mostrar que o classificador FA reduz a variância enquanto mantém o viés intocado.

2.2 Empilhamento

Empilhamento, concebido originalmente como *"Stacked Generalization"* [Wolpert 1992], é um método para combinar múltiplos classificadores usando algoritmos de aprendizados heterogêneos $L_1, ..., L_K$ sobre um único conjunto de dados D, que consiste de exemplos $e_i = (x_i, y_i)$, onde x_i é o vetor de atributos e y_i sua classificação. Nesse trabalho focaremos em empilhamento de dois níveis que pode ser dividido em duas fases: Na primeira fase, um conjunto de classificadores do nível base $C_1, C_2, ..., C_K$ é gerado, onde $C_i = L_i(D)$. Na segunda fase um classificador do meta-nível aprende a combinar as saídas dos classificadores do nível base.

Para gerar o conjunto de treino para o aprendizado do classificador do meta-nível, pode-se aplicar o procedimento **leave-one-out** ou **validação cruzada** (nesse trabalho usamos **validação cruzada *5-fold***). Assim, cada classificador do nível base aprende usando $D - F_t$ deixando o t-ésimo *fold* para teste: $\forall i = 1, ..., N : \forall t = 1, ..., 5 : C^t_i = L_i(D - F_t)$. Então, cada classificador recém-aprendido C^t_i gera uma distribuição de probabilidade (DP) sobre todas as possíveis classes $\forall x_j \in F_t : p_i(x_j) = (p_i(c_1|x_j), p_i(c_2|x_j), ..., p_i(c_M|x_j)) = C^t_i(x_j)$, onde $\{c_1, ...c_M\}$ é o conjunto de possíveis classes e $p_i(c_m|x)$ descreve a probabilidade de uma instância x ser classificada como a classe c_m pelo i-ésimo classificador. O conjunto de treino do meta-nível consiste de instâncias da forma

$((p_i^1(x_i), ..., p_i^K(x_i)), y_i)$. O número total de atributos no conjunto de treino do meta-nível será MK, M atributos para cada classificador do nível base.

Usar probabilidade para gerar o conjunto de treino do meta-nível é mais vantajoso do que utilizar somente a classe predita já que disponibiliza mais informação acerca das predições feitas pelos classificadores do nível base. Essa informação adicional permite que não seja usado somente a predição, mas também a confiança de cada classificador do nível base [Ting and Witten 1999] .

3. EXTENSÕES DA FLORESTA ALEATÓRIA

3.1 Árvores Extremamente Aleatórias

As Árvores Extremamente Aleatórias(AEA) [Geurts et al. 2006] diferem das FAs em dois aspectos: (1) elas não aplicam o procedimento de *bagging* ao construir o conjunto de exemplos de treino para cada árvore. Portanto, o mesmo conjunto de treino é usado para treinar todas as árvores; (2) elas escolhem o nodo para divisão de forma extrema (tanto o índice do atributo quanto o valor do limiar são escolhidos aleatoriamente), por outro lado a FA encontra o melhor ponto de corte (i.e., o valor ótimo tanto para o índice quanto para valor de corte) dentre subconjunto de atributos escolhidos aleatoriamente. Essas mudanças tornam o algoritmo competitivo e em alguns casos superior à FA.

3.2 BROOF

Em [Salles et al. 2015] os autores combinam *boosting* e *bagging* explorando FAs como aprendizes fracos no processo de *boosting*. Em contraste com algoritmo tradicional do AdaBoost [Freund and Schapire 1997], a regra de ponderação foi alterada para usar o erro Out-of-Bag(OOB), provido pela Floresta Aleatória durante o treino, além de "suavemente" atualizar o peso apenas das instâncias OOB. Essas duas abordagens juntas previnem o Boosting de convergir rapidamente e mantêm as Florestas Aleatórias focadas nas regiões do espaço de entrada de difícil classificação. Assim, o sistema se torna mais robusto ao problema de sobre-ajuste e a questão do ruído, gerando modelos com maior capacidade de generalização.

3.3 Boosted Extremely Randomized Trees

BERT (*Boosted Extremely Randomized Trees*) é uma proposta original nossa que combina *boosting* e Árvores Extremamente Aleatórias, explorando a ideia do BROOF de usar o erro *Out-Of-Bag* (OOB) como uma estimativa de erro mais robusta para o processo de ponderação do *boosting* e apenas atualizar os pesos das instâncias OOB. [Salles et al. 2015] mostra que o uso de uma estimativa de erro menos enviesada, como erro OOB, juntamente com a ponderação apenas das instâncias deixadas de fora (OOB), melhoram drasticamente o poder de generalização do *boosting* e FAs, sobretudo quando aplicados a tarefas com dados textuais. Tendo isso em vista, nós introduzimos o procedimento de *bagging* nos modelos de AEA de modo que possamos estimar melhor o erro Out-of-Bag (OOB) no processo do BROOF. Como pode ser observado nos experimentos seguintes, o BERT está dentre os classificadores com melhores desempenhos em todos os conjuntos de dados testados, saindo-se melhor que o BROOF em vários casos.

3.4 Floresta Lazy

Florestas *Lazy* [Salles et al. 2015] tentam resolver o problema de sobre-ajuste sofrido por Florestas Aleatórias, quando aplicadas a dados de alta dimensionalidade com atributos irrelevantes e ruidoso, projetando o conjunto de treino às k instâncias de treino mais próximas aos dados de teste. Em outras palavras, dado um conjunto de treino D e uma instância de teste x, a vizinhança da instância x em D formada pelos k exemplos de treino mais próximos a x, compõem o conjunto de treino $D^{'}$ que é dado

a Floresta Aleatória para aprendizagem. Subsequentemente, a FA treinada é usada para predizer a classe do exemplo de teste x. A projeção do conjunto de treino da FA para a vizinhança dos dados de teste pode mitigar o problema de sobre-ajuste sofrido pela Floresta Aleatória já que a projeção filtra atributos irrelevantes e ruidosos que não tinham muito a ver com a instância de teste em questão, assim provendo maior poder de generalização as FAs.

4. PROJETO EXPERIMENTAL - CLASSIFICAÇÃO TEXTUAL

Para avaliação dos métodos para categorização de tópicos, nós consideramos quatro conjuntos de dados textuais reais, conhecidos como: *20 Newsgroups* (Notícias), *Four Universities* (Páginas web), *Reuters* (Notícias) e *ACM Digital Library* (Artigos Científicos em Computação).Para todos os conjuntos de dados, nós executamos pré-processamento padrão: removemos *stopwords* usando a lista padrão SMART e aplicamos uma simples remoção de atributos removendo termos com baixa "frequência nos documentos (FD)"[3]. Em relação à ponderação de termos, nós utilizamos os esquemas de ponderação mais adequados para se obter a melhor efetividade de cada método, portanto, nós utilizamos TF para todos os classificadores baseados em Floresta Aleatória e Naïve Bayes, e usamos TDIDF com normalização L2 para os classificadores KNN e SVM. Todos os conjuntos de dados são de rótulo único.

Conjunto	Classes	# atrib	# docs	Densidade	Tamanho
4UNI	7	40,194	8,274	140.325	14MB
20NG	20	61,049	18,766	130.780	30MB
ACM	11	59,990	24,897	38.805	8.5MB
REUT90	90	19,589	13,327	78.164	13MB

Table I: Informações gerais sobre os conjuntos de dados.

4.1 Experimentação

Nós conduzimos experimentos controlados para analisar e comparar a efetividade dos métodos em estudo. Os métodos foram comparados utilizando duas medidas padrões de recuperação de informação: *micro averaged* F_1 (MicroF_1) e *macro averaged* F_1 (MacroF_1). Para compararmos a média dos resultados usando nosso experimento de validação cruzada 5-*folds*, nós avaliamos a significância estatística dos nossos resultados com um teste t pareado, com 95% de confiança e correção de Bonferroni. Este teste garante que os melhores resultados, marcados em **negrito**, são estatisticamente superiores aos outros. Foram avaliados nove algoritmos de aprendizado: (1) **SVM com *kernel* linear**; (2) **k-vizinhos mais próximos (KNN)**; (3) **Naïve Bayes Multinomial (NB)**; (4)**Floresta Aleatória (FA)** ;(5)**Árvores Extremamente Aleatórias (AEA)**, todos esses algoritmos estão disponíveis em *Python* pela biblioteca **Scikit-learn** [4][Pedregosa et al. 2011]; (6) **BROOF** [Salles et al. 2015]; (7) **Boosted Extremely Randomized Trees (BERT)**; (8) **LazyNN_RF (LAZY)**, versão *Lazy* da FA que utiliza as k instâncias de treino mais próximas do exemplo de teste para treinar uma FA e predizer a classe da instância e (9) **Lazy Extra-Trees (LXT)**, similar ao algoritmo LAZY, porém utiliza AEA ao invés de FA; esses quatro últimos foram implementados por nós em *Python*. Os hiperparâmetros foram escolhidos usando validação cruzada *5-fold* no conjunto de treino para todos os classificadores, exceto BROOF e BERT. Para esses fixamos o número de árvores e iterações em 8 e 200, respectivamente, e utilizamos os melhores hiperparâmetros encontrados para FA e AEA para os respectivos conjuntos de dados. Para os sistemas de *empilhamento* foram utilizados como classificadores de base subconjuntos dos nove classificadores listados anteriormente e como classificador do meta-nível foi utilizado uma FA utilizando $\sqrt{|F|}$ atributos e 200 árvores.

Gostaríamos de salientar que alguns dos resultados obtidos podem diferir dos reportados em outros trabalhos para os mesmos conjuntos de dados. Tais discrepâncias podem ser devido a vários fatores

[3]Nós removemos todos os termos que ocorrem em menos de seis documentos (i.e., FD<6).
[4]Disponível em `http://scikit-learn.org/`

tais como diferença na preparação dos conjuntos de dados[5], o uso de diferentes divisões do conjunto de dados (e.g., alguns conjuntos de dados têm "divisões padrões" tal como REUT90[6]). Além disso, nós rodamos todas as alternativas sob as mesmas condições em todos os conjuntos de dados, usando o esquema de ponderação mais adequado para cada método, um padronizado e bem aceito procedimento de validação cruzada para otimizar os parâmetros para cada uma das alternativas e aplicamos o apropriado ferramental estatístico para análise dos resultados.

4.2 Resultados e Discussões

		20NG	4UNI	ACM	REUT90
Todos	microF1	**91.67 ± 0.44**	**86.74 ± 1.17**	**78.46 ± 0.72**	**80.02 ± 1.24**
	macroF1	**91.43 ± 0.42**	**79.45 ± 2.23**	**63.72 ± 1.01**	**37.84 ± 3.14**
BERT+BROOF+LXT+LAZY	microF1	90.63 ± 0.57 †	**86.79 ± 0.86** †	**77.34 ± 0.6** †	**79 ± 1.14** †
	macroF1	90.4 ± 0.57 †	**79.63 ± 1.91** †	**62.91 ± 0.92** †	**33.93 ± 2.97**†
BERT+LXT	microF1	90.2 ± 0.51†	**86.54 ± 1.06** †	76.88 ± 0.55 †	**78.25 ± 1.17** †
	macroF1	89.95 ± 0.52 †	**79.41 ± 1.63** †	**62.66 ± 0.81** †	**32.86 ± 2.23** †
SVM+NB+KNN+RF	microF1	90.65 ± 0.45 †	84.95 ± 1.15	**77.78 ± 0.73** †	74.63 ± 1.00
	macroF1	90.42 ± 0.44†	75.86 ± 1.48	**63.04 ± 0.85** †	32.00 ± 0.88 †
BROOF+LAZY	microF1	89.32 ± 0.42	**86.52 ± 1.18** †	76.74 ± 0.73 †	77.22 ± 1.14 †
	macroF1	89.01 ± 0.44	**78.66 ± 1.9** †	**62.2 ± 1.01** †	31.71 ± 2.7 †
BERT	microF1	89.45 ± 0.46 *	84.61 ± 0.98*	74.8 ± 0.59*	67.33 ± 0.72*
	macroF1	89.13 ± 0.58*	73.61 ± 1.85 *	**62.1 ± 0.99** *†	29.24 ± 1.40 *
SVM	microF1	90.06 ± 0.43 *†	83.48 ± 1.08 *	75.4 ± 0.66*	68.19 ± 1.15 *
	macroF1	89.93 ± 0.43 *†	73.39 ± 2.17 *	**63.84 ± 0.55** *†	31.95 ± 2.59 *†
BROOF	microF1	87.96 ± 0.24	84.41 ± 1.07 *	73.35 ± 0.79	66.79 ± 0.97*
	macroF1	87.44 ± 0.28	73.23 ± 1.10 *	60.76 ± 0.8	28.48 ± 2.17*
LAZY	microF1	87.96 ± 0.37	82.34 ± 0.61 *	74.02 ± 0.79 *	66.3 ± 1.07 *
	macroF1	87.39 ± 0.37	68.33 ± 1.6	59.46 ± 1.35	26.61 ± 2.12
NB	microF1	88.99 ± 0.54*	62.63 ± 1.7	73.54 ± 0.71	65.32 ± 1.13
	macroF1	88.68 ± 0.55*	51.38 ± 3.19	58.03 ± 0.85	27.86 ± 0.79*
KNN	microF1	87.53 ± 0.69	75.63 ± 0.94	70.99 ± 0.96	68.07 ± 1.07 *
	macroF1	87.22 ± 0.66	60.34 ± 1.36	55.85 ± 0.97	29.93 ± 2.48 *
AEA	microF1	87.03 ± 0.41	82.87 ± 1.00 *	73.08 ± 0.55	64.87 ± 0.81
	macroF1	86.65 ± 0.56	68.54 ± 2.60	58.71 ± 0.89	26.18 ± 2.55
LXT	microF1	88.39 ± 0.51	81.24 ± 0.71	69.63 ± 0.91	65.92 ± 0.82
	macroF1	88.05 ± 0.44	66.89 ± 1.23	57.33 ± 1.48	26.71 ± 2.53
FA	microF1	83.64 ± 0.29	81.52 ± 1	71.05 ± 0.31	63.92 ± 0.81
	macroF1	83.08 ± 0.35	65.44 ± 1.91	56.56 ± 0.45	24.36 ± 1.98

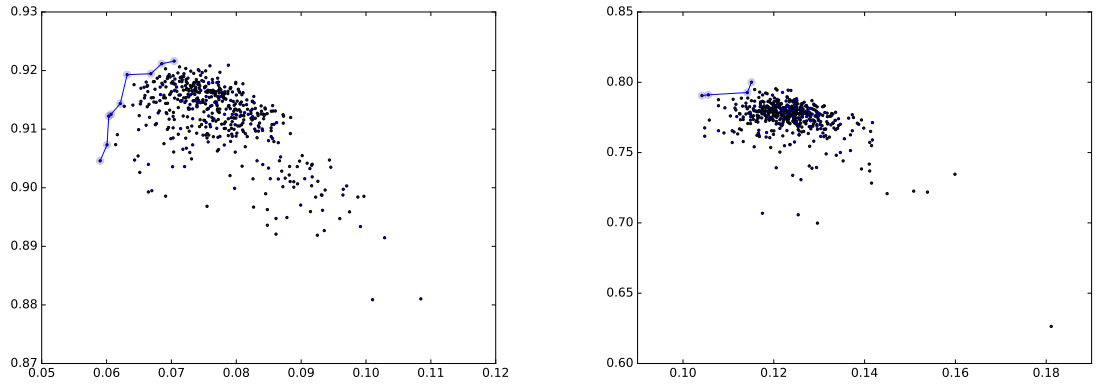
Table II: Comparação entres os métodos de base e empilhamentos. † e * indicam os melhores métodos considerando todos os métodos exceto o empilhamento de *Todos* e apenas os métodos base, respectivamente.

A Tabela II mostra a eficácia dos métodos clássicos (SVM, KNN, NB), métodos baseados em florestas (FA, AEA, BROOF, BERT, LXT, LAZY), e os comitês por empilhamento. Os comitês de métodos baseados em FA obtiveram ganhos estatisticamente superiores em MicroF_1 e MacroF_1 sobre todos métodos de base em todos os conjuntos de dados testados, havendo apenas um empate com SVM no conjunto de dados 20NG, e nos conjuntos ACM e REUT90 em termos da métrica MacroF_1. Os melhores resultados foram alcançados no 4UNI e REUT90 com ganhos de até 8% em MacroF_1 e 16% em MicroF_1, respectivamente. Em geral, os resultados mostraram que a combinação de métodos baseados em FA é efetiva em aumentar o poder de generalização do comitê, isso corrobora a intuição que os recém-propostos métodos baseado em FA são complementares e efetivos em sistemas multi-classificadores. Além disso, os comitês de métodos baseados em FA foram superiores (ou pelos empataram com) o comitê formado pelos métodos clássicos em todos os conjuntos de dados, obtendo ganhos de até 5% em MacroF_1 (4UNI) e 5,8% em MicroF_1 (REUT90).

Um indicativo do papel dos novos métodos baseados em FA em sistemas multi-classificadores é suas respectivas capacidades individuais de generalização, como pode-se notar na Tabela II três dos cinco

[5]Por exemplo, alguns trabalhos exploram complexos e sofisticados esquemas de ponderação para os atributos ou mecanismos de seleção de atributos que favorecem alguns algoritmos em detrimento de outros.
[6]Nós acreditamos que rodar experimentos somente nas divisões padrões não é um bom procedimento experimental já que não nos permite um tratamento estatístico apropriado dos resultados.

classificadores de base mais eficazes são extensões da FA (BERT, BROOF e LAZY). Ao mesclarmos o métodos baseados em FA com os métodos clássicos no comitê *Todos* podemos notar na Tabela II que houve ganhos em todos os conjuntos de dados, porém estatisticamente significante somente no 20NG com ganhos de 1,14% e 1,13% (em MicroF_1 e MacroF_1) isso evidencia que os métodos baseados em FA também podem complementar os métodos clássicos e prover melhoramento a sistemas multi-classificadores estado-da-arte. Em geral, podemos obter resultados estado-da-arte apenas utilizando os métodos baseados em FA (BERT+LAZY+BROOF+LXT) que empatam em três conjuntos de dados com a combinação de todos os métodos de base, usando um número reduzido de métodos.



(a) Conjunto de dados: 20NG

(b) Conjunto de dados: 4UNI

Fig. 2: MacroF_1 vs. *Double-Fault* - cada ponto representa um *empilhamento* das possíveis combinações dos 9 classificadores de base. A linha ligando os pontos em destaque é a fronteira de *Pareto*.

A Figura 2 mostra gráficos de dispersão para vários conjuntos de dados, onde cada ponto representa um comitê por empilhamento do conjunto dos possíveis comitês gerados pela combinação dos 9 classificadores de base (número de classificadores por comitê variando de 2 até 9). O eixo x do gráfico representa a métrica de diversidade *Double-Fault*, que foi escolhida por ser a métrica pareada que melhor representa a relação entre acurácia e diversidade de um sistema de comitê [Kuncheva and Whitaker 2003]. A métrica foi usada originalmente por [Giancinto and Roli 2001] para criar uma matriz dependências par a par usada para selecionar os classificadores menos relacionados. A métrica estima a probabilidade dos erros do par de classificadores coincidirem (menor o valor da métrica maior a diversidade/complementariedade). Já o eixo y é a métrica de eficácia MacroF_1. Para esses pontos foi calculada a fronteira de *Pareto* que tem como objetivo maximizar a métrica de eficácia e minimizar a métrica *Double-Fault* (aumentar diversidade), que são características desejáveis de qualquer método de *comitê* (alta diversidade e eficácia). Com a fronteira de *Pareto* queremos confirmar que os métodos recém-propostos baseados em FA são importantes para geração de comitês de classificadores com alto poder de generalização e com diversidade. Os comitês na fronteiras de *Pareto* das Figuras 2 (a) e (b) são os seguintes: $\{(LXT, NB), (LAZY, NB), (SVM, KNN), (LXT, SVM, NB), (LAZY, SVM, NB), (LXT, SVM, NB, KNN), (LXT, SVM, NB, KNN, AEA), (LAZY, BERT, LXT, SVM, NB, KNN), (LAZY, LXT, SVM, NB, KNN, FA)\}$, $\{(BERT, SVM), (BERT, LXT, SVM), (BROOF, BERT, SVM, KNN), (BERT, SVM, NB)\}$. Podemos notar que à medida que percorremos os pontos na fronteira de *Pareto* da esquerda para direita há um aumento na eficácia e redução na diversidade do sistema. Isso implica que não necessariamente um sistema muito diverso vá resultar no sistema mais acurado, mas que há um "*trade-off*" entre efetividade dos classificadores de base que compõem o sistema e a diversidade deles. Analisando as fronteiras é nítido que as extensões da FA tem um papel importante em prover um balanceamento entre diversidade e poder de generalização aos comitês mais eficazes. É possível notar que há uma perda de diversidade à medida que são

inseridos métodos mais similares (métodos que possuem raízes comuns, e.g baseados em FA), mas a diminuição da diversidade é compensada pelo alto poder de generalização que esses métodos possuem gerando métodos mais eficazes. Esses resultados juntamento com a análise de efetividade corroboram nossa hipótese: o uso de algoritmos baseados em floresta aleatória em sistemas multi-classificadores podem alavancar o poder de generalização do classificador resultante.

5. CONCLUSÃO E TRABALHOS FUTUROS

Nesse trabalho, nós propusemos combinar derivações da FA, tais como Árvores Extremamente Aleatórias (AEA) [Geurts et al. 2006], BROOF [Salles et al. 2015], *Boosted Extremely Randomized Tree*, *Lazy Random Forest*[Salles et al. 2015] e *Lazy Extra-Trees*, e estudar seu impacto em sistemas de *empilhamento* no contexto de classificação textual. Nossos experimentos mostraram, que a combinação desses métodos baseados em floresta aleatória tem papel importante no aumento do poder de generalização de métodos de comitês de classificadores em tarefas de classificação textual, havendo melhora em termos de MicroF_1 e Macro$F1$ (com significância estatística) em todos os conjuntos de dados testados quando comparados aos melhores classificadores estado-da-arte individuais. Além disso, foram obtidos ganhos de até 5% em MacroF_1 e 5,8% em MicroF_1 quando comparado à combinação de classificadores estados-da-arte em classificação de texto. Como trabalho futuro pretendemos estudar o comportamento dos recém-propostos algoritmos baseados em FA no meta-nível quando comparados a meta-classificadores estado-da-arte.

REFERENCES

Breiman, L. Bagging predictors. *Mach. Learn.* 24 (2): 123–140, Aug., 1996.

Breiman, L. Random forests. *Machine Learning* 45 (1): 5–32, 2001.

Danesh, A., Moshiri, B., and Fatemi, O. Improve text classification accuracy based on classifier fusion methods. In *Information Fusion, 2007 10th International Conference on*. IEEE, Montreal, pp. 1–6, 2007.

Dong, Y.-S. and Han, K.-S. A comparison of several ensemble methods for text categorization. In *Services Computing, 2004. (SCC 2004). Proceedings. 2004 IEEE International Conference on*. IEEE, Shanghai, China, pp. 419–422, 2004.

Fernández-Delgado, M., Cernadas, E., Barro, S., and Amorim, D. Do we need hundreds of classifiers to solve real world classification problems? *J. Mach. Learn. Res.* 15 (1): 3133–3181, Jan., 2014.

Freund, Y. and Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* 55 (1): 119–139, Aug., 1997.

Geurts, P., Ernst, D., and Wehenkel, L. Extremely randomized trees. *Machine Learning* 63 (1): 3–42, 2006.

Giancinto, G. and Roli, F. Design of effective neural network ensembles for image classification purposes. *IMAGE VISION AND COMPUTING JOURNAL* vol. 19, pp. 699–707, 2001.

Khan, A., Baharudin, B., Lee, L. H., Khan, K., and Tronoh, U. T. P. A review of machine learning algorithms for text-documents classification. In *Journal of Advances In Information Technology*. Academy Publisher, Bangkok, Thailand, 2010.

Kuncheva, L. I. and Whitaker, C. J. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Mach. Learn.* 51 (2): 181–207, May, 2003.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* vol. 12, pp. 2825–2830, 2011.

Salles, T., Gonçalves, M., and Rocha, L. Phd proposal: Random forest based classifiers for classification tasks with noisy data (approved), 2015. Universidade Federal de Minas Gerais.

Salles, T., Gonçalves, M., Rodrigues, V., and Rocha, L. Broof: Exploiting out-of-bag errors, boosting and random forests for effective automated classification. In *Proc. of the 38th International ACM SIGIR Conference on Inf. Retrieval*. ACM, Santiago, Chile, pp. 353–362, 2015.

Segal, M. R. Machine learning benchmarks and random forest regression. Tech. rep., University of California, 2004.

Ting, K. M. and Witten, I. H. Issues in stacked generalization. *J. Artif. Int. Res.* 10 (1): 271–289, May, 1999.

Wolpert, D. H. Stacked generalization. *Neural Networks* vol. 5, pp. 241–259, 1992.

Using LSTM and Technical Indicators to Predict Price Movements

David M. Q. Nelson, Adriano C. M. Pereira

Universidade Federal de Minas Gerais (UFMG)
Departamento de Ciência da Computação (DCC)
Belo Horizonte – MG – Brasil
email: davidmn@ufmg.br
adrianoc@dcc.ufmg.br

Abstract. The financial market has been object of various studies that seek to model or predict its behavior. However, this prediction is a great challenge, since this market is a complex and chaotic environment. The usage of Machine Learning has become a popular prediction method for transactions on the stock market and has been offering satisfactory results in many cases for its capacity of learning from history to infer future trends. This project proposes the creation of a model using a LSTM neural network, describe and predict the behavior of assets on the stock market from a big variety of information, such as historical price data and technical analysis indicators. There were executed a series of experiments and various metrics were collected from the results in order to confirm that the proposed approach is promising, getting up to 57.5% of precision on determining the price direction for some assets, which leads to believe that it can result in a model capable of predicting price movement with satisfactory precision and consistency.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning

Keywords: lstm, machine learning, recurrent neural networks, stock markets, supervisioned learning, technical analysis

1. INTRODUCTION

This article aims to explore the usage of recurrent neural networks for predicting price movement on the stock market. The idea is to take advantage of the short term memory capabilities of such networks to make them use historic price data in order to determine whether the price is going to go up or not in the near future.

Stock markets are a very dynamic, complex and chaotic environment which makes it incredibly hard to foresee its future behaviour. There are so many different variables and sources of information that can impact and determine a stock price that causes all existent and well known models to fail to precisely predict stocks trends in a general sense. Nevertheless, there is a vast amount of content in the literature, from many disciplines, aiming to take on that challenge and research different ways to perform better predictions.

The algorithm chosen for this purpose in this project is the LSTM network, a network that is capable to give different importance to recent events compared to the distant ones, and eventually forget when it is helpful. And the source of information for feeding that network is the price/candlestick database for a few different stocks from the Brazilian stock exchange along side with a large amount technical indicators generated from those same prices.

The objective of this project is to study the applicability of recurrent neural networks, in particular the LSTM networks, on price movement prediction for stock markets. Assess their performance in

Symposium on Knowledge Discovery, Mining and Learning, KDMILE 2015.

terms of accuracy and other metrics through experiments on top of real life data and analyze if they present any sort of gain in comparison to more traditional machine learning algorithms.

The main contributions of this work are the following: (1) a new price movement prediction model for stock markets using deep learning based technique; (2) the validation of the model using real data from Brazilian stock exchange; (3) evaluation of the model by comparing and analysing it against some typical baselines.

The remainder of this paper is organized as follows: Section 2 presents an overview of the theoretical concepts that are base of this article, both in the context of stock markets and of machine learning. It also mentions some of the related work to this subject that can be found in the literature and highlights the novelty this project aims to bring; Section 3 details the model proposed in this article, in Section 4 the experimental results are presented and discussed, and finally Section 5 concludes the article.

2. BACKGROUND AND RELATED WORK

When it comes to stock markets, in addition to its innate complexity and dynamism, there has been a constant debate on the predictability of stock returns. Fama and Malkiel (1970) introduced the Efficient-market hypothesis that defines that a price of an asset always reflects all information available for it instantly. There is also the Random-walk hypothesis (Malkiel, 1973) which claims that a stock price changes independently of its history, in other words, tomorrow's price will only depend on tomorrow information regardless of today's price. Those two hypothesis determine that there are no means to accurately predict a stock price. On top of that, Biondo et al. (2013) has performed a series of experiments showing that a random strategy can outperform some of the most classic methods of technical trading, like Moving Average Convergence Divergence (MACD) and Relative Strength Index (RSI).

On another hand, there are other authors who claim that, in fact, stock prices can be predicted at least to some degree (Lo and MacKinlay, 1999). And a variety of methods for predicting and modeling stock behavior have been object of study of many different disciplines, such as economics, statistics, physics and computer science. And as of 2012 it was estimated that approximately 85% of trades within the United States' stock markets were performed by algorithms (Glantz and Kissell, 2013).

A popular method of modeling and predicting the stock market is technical analysis, which is a predicting method based on historic data from the market, mainly price and volume. It follows some assumptions: (1) prices are defined exclusively by the supply-demand relation; (2) prices change following tendencies; (3) changes on supply and demand cause tendencies to reverse; (4) changes on supply and demand can be identified on charts; And (5) patterns on charts tend to repeat (Kirkpatrick and Dahlquist, 2006). In other words, technical analysis do not take into account any external factors like political, social or macro-economical.

In regards to computational intelligence there are plenty of studies assessing different methods in order to accomplish accurate predictions on the stock market. They go from evolutionary computation through genetic algorithms as exemplified in Allen and Karjalainen (1999), statistical learning by using algorithms like Support Vector Machines (SVM) (Kim, 2003) and a variety of others including neural networks, component modeling, textual analysis based on news data, that are discussed by Melo (2012) which also proposed a new approach based on collective intelligence.

Taking a closer look into works related to deep learning in stock markets there are some examples like Batres-Estrada (2015) where a study is made on the usage of a Deep Belief Network (DBN), which is composed of stacked Restricted Boltzmann Machines, coupled to a Multi-level Perceptron (MLP) and using long range log returns from stock prices to predict above-median returns for each day. Sharang and Rao (2015) also use of DBN, but this time using price history in addition to technical

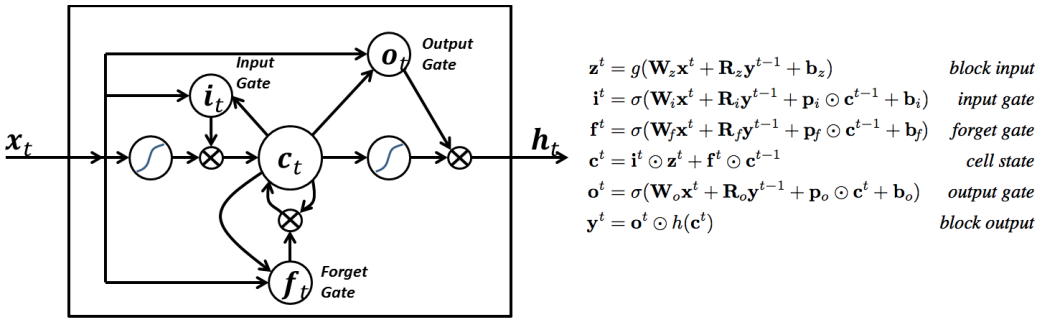


Fig. 1. Long short term memory neural network (Greff et al., 2015)

indicators as input, in a similar approach to this project. Both of those works present improved results compared to their baselines, as well as in Heaton et al. (2016) where a survey in deep learning methods applied to finance is made and their improvements discussed.

Long Short Term Memory (LSTM) networks (Figure 1), which are used in this project are a deep and recurrent model of neural networks. Recurrent networks differ from the traditional feed-forward networks in the sense that they don't only have neural connections on a single direction, in other words, neurons can pass data to a previous or the same layer. In which case, data doesn't flow on a single way, and the practical effects for that is the existence of short term memory, in addition to long term memory that neural networks already have in consequence of training. LSTM were introduced by Hochreiter and Schmidhuber (1997) and it aimed for a better performance by tackling the vanishing gradient issue that recurrent networks would suffer when dealing with long data sequences. It keeps the error flow constant through special units called "gates", then uses it to able adjust the weights and it is able to truncate the gradient when it does not harm.

These networks have been widely used and been able to accomplish some of the best results when compared to other methods (Graves, 2012), especially on Natural Language Processing (NLP), and in particular for handwriting recognition it is considered the state-of-the-art (Graves et al., 2009). And since its inception it has been branched into a number of variations which were assessed against their original version by Greff et al. (2015) but do not seem to present any considerable improvements so far.

For stock prices prediction, given the notorious performance LSTM networks have shown in NLP, it has been mostly used taking news text data as input to predict price trends. But there is also some work using price data to foresee price movement, Chen et al. (2015) uses historic price data in addition to stock indexes to determine if a stock price is going up, down or stay the same on the next day. Luca Di Persio (2016) compares the performance of LSTM and MLP to their own proposed method based on a combination of wavelets and Convolutional Neural Networks, which outperforms both but has very close results to the LSTM network.

We propose a model based on LSTM to predict price movements using an input that is not based on text, which as mentioned before, is not something that has been widely explored. We plan to use a wide range of technical indicators to do so, and the intention is to assess the usage of such method that is something commonly used on investment strategies. Additionally, we want to test the hypothesis that the short term memory capability can present better results compared to traditional feed forward networks.

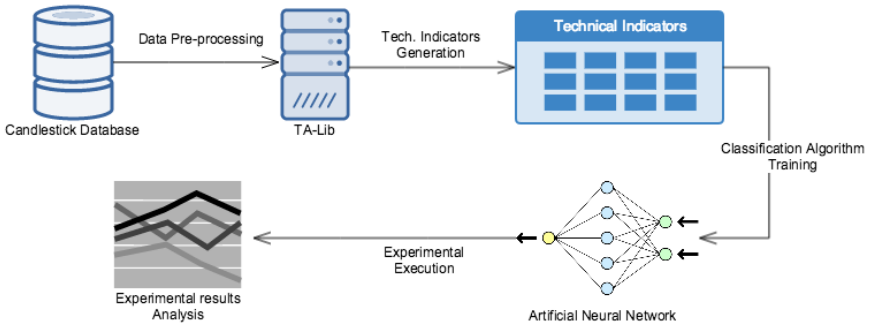


Fig. 2. Methodology

3. METHODOLOGY & DEVELOPMENT

For this project we designed a classification model (Figure 2) using LSTM networks that aims to predict price movements for individual stocks in the near future. It will attempt to determine if the price of a particular stock will be higher or not than the current price in 15 minutes in the future.

3.1 Data processing

The input for this model consists of historic stock price data, as a time series of candles (open, close, high, low and volume) in a granularity of 15 minutes. We collected data from 2008 to 2015 for the stocks that are part of the IBovespa index from the BM&F Bovespa stock exchange.

In addition to the price data, a set of technical indicators is generated using the TA-Lib [1] library. Such indicators are mathematical calculations intended to determine or predict characteristics from stocks based on their historic data. A total of 175 values are generated for each period, and they are intended to represent or predict a very diverse set of characteristics of the stock, like the future price, volume to be traded, the intensity of the current movement tendency, visual graphical patterns, among others.

On top of the historic price data, before generating the technical indicators, exponential smoothing was performed through exponentially weighted moving averages (1) in order to reduce random variation and noise on the pricing series as indicated by Khaidem et al. (2016) to cause improvements on the prediction capability.

$$z_i = \lambda \bar{x}_i + (1 - \lambda) z_{i-1} \tag{1}$$

For each entry of the dataset, a binary class y is attributed, "1" will indicate that the price will go up on the following time step, and "0" that it won't. So, given that i is the current moment and j is the following, then $j = i + timestep$, and for this project $timestep$ is equivalent to 15 minutes. In case the output is "1", a "buy" operation will be triggered at i.

For determining the class, the policy to set the value will be based on whether or not the closing price of the next period will be higher than the current one:

$$y = \begin{cases} 1 & \text{if } close_j > close_i \\ 0 & \text{if } close_j \leq close_i \end{cases} \tag{2}$$

[1] http://ta-lib.org

The neural network will take k instances of X as input $(X_{i-k}, ..., X_{i-2}, X_{i-1}, X_i)$, where X consists in tuples of price data along with technical indicators, and that will be used to predict y_j.

4. EVALUATION

This model was built to work on a sliding window fashion. A new neural network is generated at the end on each trading day, meaning that a new set of weights is defined using a new set of training and validation data. For training it is used the last 10 months of trading prior to the current day, and the model performance is validated by using the data of the current day itself. On the following day, all the predictions will be done using the most recent model.

Given that we are working with a time series, the supervised learning algorithm that was chosen was the LSTM neural network (Long short term memory), which is a recurrent neural network capable o classifying input data taking into account the previous instances.

A LSTM layer will take both technical indicators and pricing data as input and will feed an output layer through a linear function and that will output the class prediction using a sigmoid function.

We executed a series of experiments using the proposed model using 2014 pricing data for a few different stocks (BOVA11, BBDC4, CIEL3, ITUB4 and PETR4) from the Brazilian stock market.

Metrics around the algorithm performance and the financial results were collected and compared with the baselines selected for this project. The metrics for evaluating the network performance were the accuracy (3), precision (4), recall (5) and F1, a harmonic mean between precision and recall (6).

$$A = \frac{tp + tn}{tp + fp + tn + fn} \tag{3}$$

$$P = \frac{tp}{tp + fp} \tag{4}$$

$$R = \frac{tp}{tp + fn} \tag{5}$$

$$F1 = 2\frac{P * R}{P + R} \tag{6}$$

4.1 Trading operations

When the predicted class is "1", in other words, in the case that the network predicts that the stock price will go up, then the strategy is to open a "buy" position on the current moment (i) and close it on j. In that case, profit was defined as $close_j - close_i$.

The financial results were calculated based on hypothetical trades of sets of hundred stocks per operation disregarding costs and taxes.

4.2 Baselines

The baselines chosen for this project are based on other machine learning algorithms in addition to other simplistic investment methods.

The machine learning methods consist on approaches that are very traditional and widely used but less complex than the one of this project. Using the exact same input, trying to do the same predictions. The models chosen were Multi-Layer Perceptron and Random Forest.

The other baselines are the following investment strategies:

—Buy and hold: Buy at the first time step and sell at the latest ($profit = close_n - close_1$)

—Optimistic: If prices went up on the previous time step ($close_i > close_{i-timestep}$), then perform a buying operation and sell it on the following step.

—Random (following class distribution): Decides whether or not to perform a trading operation based on probabilities according to the class distribution.

4.3 Empirical Results

There were performed experiments to predict price movement during December 2014 for the aforementioned stocks and following the methodology described on this article. The results presented below are an average of all executions per each stock.

Table I presents a characterization of the data used for the experiments, with the price evolution during the period along with the class distribution within the test data.

Stock	Staring price	Ending price	Price difference	% times price goes up
BOVA11	55.53	55.73	0.20	0.471875
BBDC4	15.09	13.26	-1.83	0.435937
ITUB4	13.81	14.73	0.92	0.442188
CIEL3	20.42	21.97	1.55	0.453125
PETR4	21.71	18.52	-3.19	0.479687

Table I. Experimental data

Stock	Accuracy	Precision	Recall	F1
BOVA11	0.564063	0.560847	0.350993	0.431772
BBDC4	0.575000	0.553846	0.129032	0.209302
ITUB4	0.551562	0.475000	0.134276	0.209366
CIEL3	0.540625	0.476190	0.137931	0.213904
PETR4	0.545312	0.563492	0.231270	0.327945

Table II. Results - Metrics

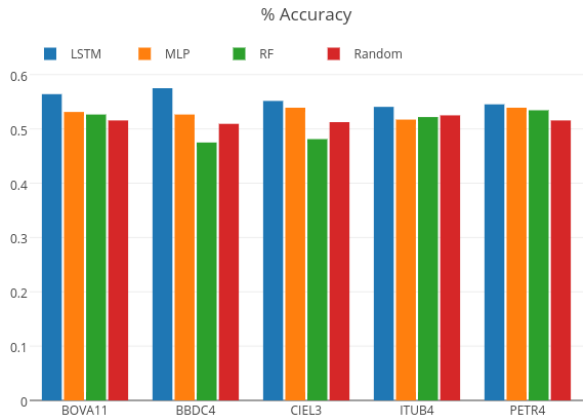


Fig. 3. Accuracy compared to baselines

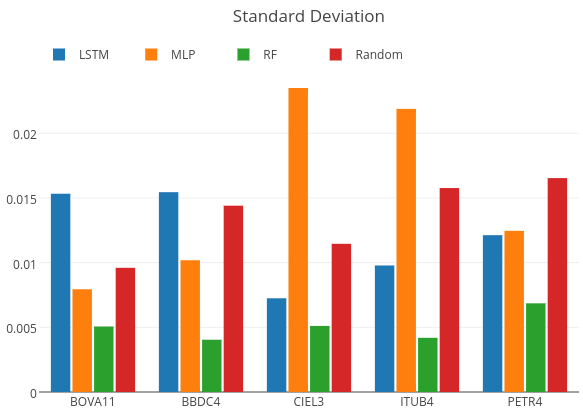


Fig. 4. Accuracy standard deviation

On Table II the metrics of the algorithm prediction performance are shown for each of the stocks, that tells how well it does as a prediction model. Figure 3 compares it to MLP and Random Forest baselines in terms of accuracy, showing that it outperforms both. And Figure 4 presents the standard

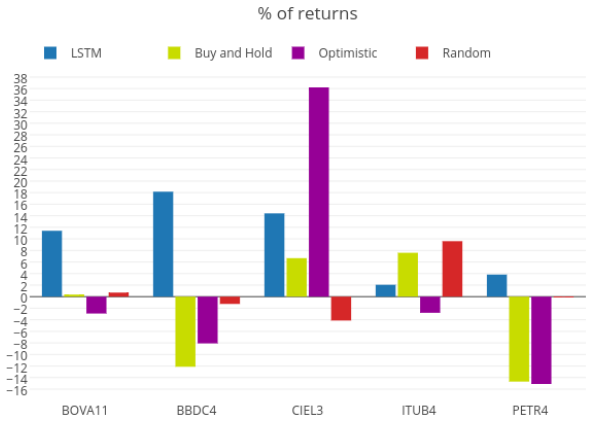


Fig. 5. Returns

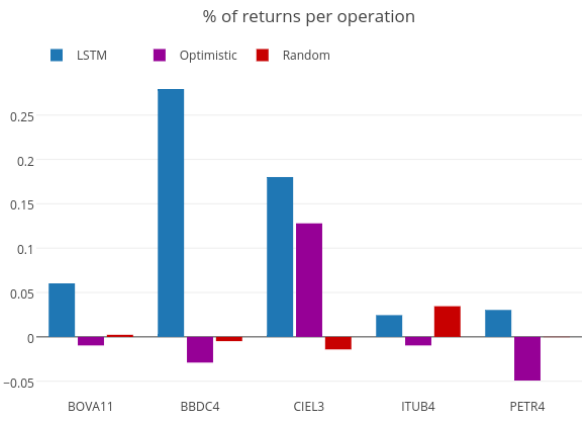


Fig. 6. Average return per operation

deviation for a number of experiments on each of the models to give an understanding of the variance and consistency of each model.

Figure 5 displays the average financial returns for the experiments per each stock, compared to each of the investment baselines that were used for this work, while Figure 6 shows the average earnings per individual buy-and-sell operation.

5. CONCLUSIONS

The results were, in general, better than the baselines with few exceptions, which is very promising since this model has proven itself capable of accomplish reasonable results compared to other approaches we can find on the literature already.

The algorithm displays capability to learn from complex input sets without requiring any sort of dimensionality reduction, like feature selection for example.

It displayed noticeable gains in terms of accuracy when compared to the other machine learning models, but in another hand we believe that variance could be lower and that would contribute for a more reliable model.

When it comes to the financial results it's important to note that it was able to keep it positive for all stocks, even though it didn't necessarily had the best results when compared to the baselines. Another positive aspect is that it had a high return ratio per operation, meaning that it had more success on detecting high variations and that becomes extremely important when taking into account transaction costs and taxes.

5.1 Forthcoming Research

We will keep investigating ways to improve the results of the predictions, studying changes on the neural network model and different approaches for pre-processing the data and possibly new features.

In addition to that, we also plan to explore sentiment analysis on news related to the stock market and combine that to the existing predictions and then evaluate if that presents any gains to the results.

We also intend to evaluate the model using different and more realistic trading strategies, instead of simply buying and selling after a fixed amount of time. And also take into account intrinsicalities of stock markets, like timing, queue of execution, and costs.

Acknowledgments

This work was partially funded by Instituto Nacional de Ciência e Tecnologia para a Web (CNPq no. 573871/2008-6), MASWeb (grant FAPEMIG/PRONEX APQ-01400-14), EUBra-BIGSEA (H2020-EU.2.1.1 690116, Brazil/MCTI/RNP GA-000650/04), CAPES, CNPq and Fapemig.

REFERENCES

Allen, F. and Karjalainen, R. Using genetic algorithms to find technical trading rules. *Journal of Financial Economics* 51 (2): 245 – 271, 1999.

Batres-Estrada, B. Deep learning for multivariate financial time series, 2015.

Biondo, A. E., Pluchino, A., Rapisarda, A., and Helbing, D. Are Random Trading Strategies More Successful than Technical Ones? *PLoS ONE* vol. 8, pp. e68344, July, 2013.

Chen, K., Zhou, Y., and Dai, F. A lstm-based method for stock returns prediction: A case study of china stock market. In *Big Data (Big Data), 2015 IEEE International Conference on*. pp. 2823–2824, 2015.

Fama, E. F. and Malkiel, B. G. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance* 25 (2): 383–417, 1970.

Glantz, M. and Kissell, R. *Multi-asset Risk Modeling*. Academic Press, 2013.

Graves, A. *Supervised Sequence Labelling with Recurrent Neural Networks*. Studies in Computational Intelligence, vol. 385. Springer, 2012.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., and Schmidhuber, J. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31 (5): 855–868, May, 2009.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. LSTM: A search space odyssey. *arXiv preprint arXiv:1503.04069*, 2015.

Heaton, J. B., Polson, N. G., and Witte, J. H. Deep learning in finance. *CoRR* vol. abs/1602.06561, 2016.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Comput.* 9 (8): 1735–1780, Nov., 1997.

Khaidem, L., Saha, S., and Dey, S. R. Predicting the direction of stock market prices using random forest. *CoRR* vol. abs/1605.00003, 2016.

Kim, K. Financial time series forecasting using support vector machines. *Neurocomputing* 55 (1–2): 307 – 319, 2003. Support Vector Machines.

Kirkpatrick, C. and Dahlquist, J. *Technical Analysis: The Complete Resource for Financial Market Technicians*. FT Press, 2006.

Lo, A. and MacKinlay, A. *A non-random walk down Wall Street*. Princeton Univ. Press, Princeton, NJ [u.a.], 1999.

Luca Di Persio, O. H. Artificial neural networks approach to the forecast of stock market price movements. *nternational Journal of Economics and Management Systems* vol. 1, pp. 158–162, 2016.

Malkiel, B. G. *A Random Walk Down Wall Street*. Norton, New York, 1973.

Melo, B. Considerações cognitivas nas técnicas de previsão no mercado financeiro. *Universidade Estadual de Campinas*, 2012.

Sharang, A. and Rao, C. Using machine learning for medium frequency derivative portfolio trading. *CoRR* vol. abs/1512.06228, 2015.

Minimum Classification Error Principal Component Analysis

T. B. A. de Carvalho[1], M. A. A. Sibaldo[1], Tsang I. R.[2], G. D. C. Cavalcanti[2]

[1] Universidade Federal Rural de Pernambuco, Brasil
{tiago.buarque, mariaaparecida.sibaldo}@ufrpe.br
[2] Universidade Federal de Pernambuco, Brasil
{tir,gdcc}@cin.ufpe.br

Abstract. We present an alternative method to use Principal Component Analysis (PCA) for supervised learning. The proposed method extract features similarly to PCA but the features are selected by minimizing the Bayes error rate for classification. Experiments using two real datasets shows that the recognition accuracy of the proposed technique is improved compared to PCA.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning

Keywords: Principal component analysis, Dimensionality reduction and manifold learning, Supervised learning by classification, Data mining

1. INTRODUCTION

Principal Component Analysis (PCA) is a technique used to reduce data dimensionality. It projects the points into the directions of maximal variance within data space. These directions are the eigenvectors of data covariance matrix. In most of the cases, only some few eigenvectors are selected, normally the ones that have the highest eigenvalues. The eigenvalue is equivalent to the variance of a new variable, that is obtained by projecting the data into an eigenvector. The new variables not only have maximal variance, but they are also uncorrelated [Bishop 2006]. PCA is a very well-known technique that is used in several different applications such as face recognition [Turk and Pentland 1991] and text classification [Alencar et al. 2014].

From the perspective of machine learning, PCA is an unsupervised feature extraction technique. Nonetheless, it is also used in supervised tasks such as in classification and regression. Some versions of supervised PCA have been proposed, for example, Barshan et al. [Barshan et al. 2011] proposed a version of supervised PCA for classification. The method defines class representatives and computes PCA for these points. Directions with maximal variances for those points are also the directions that best separate the classes. Another version of supervised PCA was proposed by Bair et al. [Bair et al. 2006] for regression. The technique selects features that have high predictive power and compute PCA using only those features. Therefore, avoiding the interference of features that have high variance but low predictive power.

The Bayesian approach for classification is very robust and, similar to PCA it depends on the data covariance matrix [Duda et al. 2000]. Here, we propose a supervised version of PCA that minimizes the Bayes error rate for classification. The method projects the same features as PCA but selects the ones that minimize the Bayes error rate, while PCA select the features with maximal variance. Therefore, it can be more suitable for classification task than standard PCA. Since projections of

This work was partially supported by CAPES, CNPq and FACEPE.

maximal variance might not be the best way to separate (discriminate) data form different classes [Bishop 2006].

2. FEATURE EXTRACTION WITH PCA

Suppose a dataset matrix $\mathbf{X}'_{n\times d}$ with n points and d features. Each row of $\mathbf{X}'$ is a point and each column is a feature.

$$\mathbf{X}' = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{bmatrix}. \tag{1}$$

The j-th point is defined as a d dimensional column vector $\mathbf{x}_j$,

$$\mathbf{x}_j = \begin{bmatrix} x_{j1} \\ x_{j2} \\ \vdots \\ x_{jd} \end{bmatrix}, \tag{2}$$

for $j = 1, \ldots, n$ and the data mean vector is

$$\bar{\mathbf{x}} \;=\; n^{-1} \sum_{j=1}^{n} \mathbf{x}_j. \tag{3}$$

The centered matrix is $\mathbf{X}$ having the j-th row equal to $(\mathbf{x}_j - \bar{\mathbf{x}})^T$:

$$\mathbf{X} = \begin{bmatrix} (\mathbf{x}_1 - \bar{\mathbf{x}})^T \\ (\mathbf{x}_2 - \bar{\mathbf{x}})^T \\ \vdots \\ (\mathbf{x}_n - \bar{\mathbf{x}})^T \end{bmatrix}. \tag{4}$$

The covariance matrix of $\mathbf{X}$ is defined as

$$\mathbf{\Sigma}_{\mathbf{X}} = \frac{1}{n}\mathbf{X}^T\mathbf{X}. \tag{5}$$

Each column $\boldsymbol{\xi}_i$, for $i = 1, \ldots, k$, of the matrix

$$\mathbf{E}_k = [\boldsymbol{\xi}_1 \ldots \boldsymbol{\xi}_k], \tag{6}$$

is an eigenvector of $\mathbf{\Sigma}_{\mathbf{X}}$. $\mathbf{E}_k$ have up to d eigenvectors, for $k = 1, \ldots, d$. Each eigenvector $\boldsymbol{\xi}_i$ have an associated eigenvalue λ_i, which is the variance of the extracted feature

$$\mathbf{f}_i = \mathbf{X}\boldsymbol{\xi}_i. \tag{7}$$

The value of the i-th extracted feature for the j-th point is w_{ij}, where $\mathbf{f}_i = [w_{1i} \ldots w_{ni}]^T$.

The projection of the point $\mathbf{x}_j^T = [x_{j1} \ldots x_{jd}]$ for the space of projected features is $\mathbf{w}_j^T = [w_{j1} \ldots w_{jk}]$, given by

$$\mathbf{w}_j^T = \mathbf{x}_j^T\mathbf{E}_k. \tag{8}$$

The eigenvectors in $\mathbf{E}_k$ are sorted, so that $\lambda_1 > \ldots > \lambda_k$. In PCA, the points are projected in the directions of maximal variances, these directions are the eigenvectors of the covariance matrix that has the greatest eigenvalues. The new data matrix $\mathbf{W}_{n\times k}$ is defined as:

$$\mathbf{W} = \mathbf{X}\mathbf{E}_k, \tag{9}$$

Each row of this matrix is a point and each column an extracted feature.

The covariance matrix of $\mathbf{W}$ is $\mathbf{\Sigma_W} = n^{-1}\mathbf{W}^T\mathbf{W}$, so that $\mathbf{\Sigma_W} = diag(\lambda_1, \ldots, \lambda_k)$. The variables are uncorrelated since the off-diagonal elements of $\mathbf{\Sigma_W}$ are equal to 0. This property is very relevant for supervised learning, because it allows the selection of any subset of the projected variables by ignoring their interaction. However, selecting eigenvectors of highest eigenvalues may not be the best strategy for classification problems. Therefore, we propose a method of selecting the eigenvectors by minimizing the Bayes error rate for classification.

3. BAYES ERROR RATE

The Bayes error rate for classification is defined as the probability of the classification error, *i.e.*, the expected error rate. This error estimation can have a simplified form by imposing some restrictions [Duda et al. 2000]. Here, we consider the following five restrictions. (1) The data presents a multivariate normal distribution. (2) The problem has only two classes. (3) The prior probabilities of both classes are equal. (4) Both classes have the same covariance matrix, the same assumption is used for PCA. Finally, (5) the features are statistically independent, similarly to PCA. Then the Bayes error rate is given by

$$P(e) = \frac{1}{\sqrt{2\pi}} \int_{r/2}^{\infty} e^{-u^2/2} du. \tag{10}$$

The Bayes error rate decreases as r increases. We define r^2 as the Mahalanobis distance between the mean vectors of the classes ($\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$):

$$r^2 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)\Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2). \tag{11}$$

For independent features, the covariance matrix is diagonal. The off-diagonal elements are the features covariances, which have values equal to zero. This means that each feature is uncorrelated so r have a special form:

$$r = \sqrt{\sum_{i=1}^{d} \left(\frac{\mu_{1i} - \mu_{2i}}{\sigma_i} \right)^2}, \tag{12}$$

where $i = 1, \ldots, d$ are the indexes for the features. The variables μ_{1i} and μ_{2i} are the mean of the feature i for classes 1 and 2, respectively. And σ_i is the variance of the feature i that is the same for both classes.

We emphasize that the probability of classification error decreases as r increases. From Equation 12, we can conclude that each feature contributes for minimizing the probability of classification error. In fact, some feature contribute more than others. The larger the difference between the means of the two classes related to the feature variance, the higher is the contribution of this feature to minimize Bayes error.

4. PROPOSED METHOD

Since PCA generate uncorrelated features and considering that the covariance matrix is the same for every class in the dataset (because it computes direction of maximal variance for a covariance matrix estimated for all data), then the Bayes error rate can be minimized, proportionally to r as defined in (12), for features extracted using (9). The proposed method considers these equations to choose the PCA projected variables. However, instead of selecting the directions of maximal variance for the classification task, we select the directions that minimizes Bayes error rate.

The problem continues to be restricted to two classes, setting $\mathbf{W} = \mathbf{X}\mathbf{E}_d$, as in (9). However, now the features are extracted for d eigenvectors. We define w_{ij} as the value of the i-th new feature

$(i = 1, \ldots, d)$ for the j-th point $(j = 1, \ldots, n)$. The mean of the i-th feature for the c-th class $(c = 1, 2)$ is

$$\bar{w}_{ci} = \frac{\sum_{j=1}^{n} w_{ij}\delta_{jc}}{\sum_{j=1}^{n} \delta_{jc}}, \tag{13}$$

where δ_{jc} is the Dirac delta function $\delta_{jc} = 1$ if the j-point belongs to the c-th class, and $\delta_{jc} = 0$, otherwise.

We propose a score of the relevance for the classification of a feature extracted with PCA, s_i is the score for the i-th extracted feature:

$$s_i = \begin{cases} |\bar{w}_{1i} - \bar{w}_{2i}|/\lambda_i, \text{ if } \lambda_i \neq 0 \\ 0, \qquad\qquad\qquad \text{if } \lambda_i = 0 \end{cases}, \tag{14}$$

where λ_i is the eigenvalue of the eigenvector from which the i-th feature were computed, and $\bar{w}_{ci}$ is the mean of the i-th feature for the c-th class $(c = 1, 2)$. If $\lambda_i = 0$ the variance of the i-th extracted feature is zero, which means that the variable has the same value for all points. Therefore it is not useful for classification and its score is set as $s_i = 0$. Otherwise the score is positive and is defined as the absolute value of the difference between the mean of each class divided by the variance of the feature. Features selected according to this score minimize the Bayes error rate. The proposed method consists in the following steps:

(1) Project the data as $\mathbf{W} = \mathbf{X}\mathbf{E}_d$, similar to Equation (9).
(2) Compute the mean of each feature for each class $\bar{w}_{ci}$, Equation (13).
(3) Compute the score s_i of each feature, Equation (14).
(4) Select k features with the highest score.
(5) Define the projection matrix as:

$$\mathbf{S}_k = [\boldsymbol{\xi}_1 \ldots \boldsymbol{\xi}_k] \tag{15}$$

with the eigenvectors that have the highest scores s_i, such that $s_i \geq s_j$ if $\boldsymbol{\xi}_i \in \mathbf{S}$ and $\boldsymbol{\xi}_j \notin \mathbf{S}$.
(6) Project the data as:

$$\mathbf{V} = \mathbf{X}\mathbf{S}_k, \tag{16}$$

where $\mathbf{V}_{n\times k}$ is the projected data matrix with n points and k discriminant features.

The difference between standard PCA and the proposed method is that the selected features in PCA are the ones of highest eigenvalues (λ_i) and the selected features in the proposed method are the ones with the highest discriminant score (s_i).

5. EXPERIMENTS

The experiments were performed using two datasets from the UCI Machine Learning Repository [Lichman 2013]. The Climate Model Simulation Crashes Data Set that has 540 points, 18 features and the Banknote Authentication Data Set that has 1,372 points, 4 features, both datasets have two classes. Accuracy, the rate of corrected classified points, is the metric used to evaluated the methods. Each point in the plot is the average accuracy for 100 holdout experiments. In each holdout experiment, 50% of the points from each class were randomly chosen for training and the remaining points were used for testing. The training set were used for both PCA and the proposed method. Both training and test sets were projected using k selected eigenvector, $k = 1, \ldots, d$. The 1-NN (Nearest Neighbor) with Euclidean distance, Naive Bayes with normal kernel smoothing density estimate, pruned Decision Tree with Gini's diversity index and a minimum of 10 nodes per leaf, and Fisher's Linear Discriminant were used for classification. The experiment were performed using Matlab 2015b

Statistics and Machine Learning Toolbox[1]. The result are shown in Figures 1, 2, 3, and 4. The results are also detailed in Tables I, II, III, and IV.

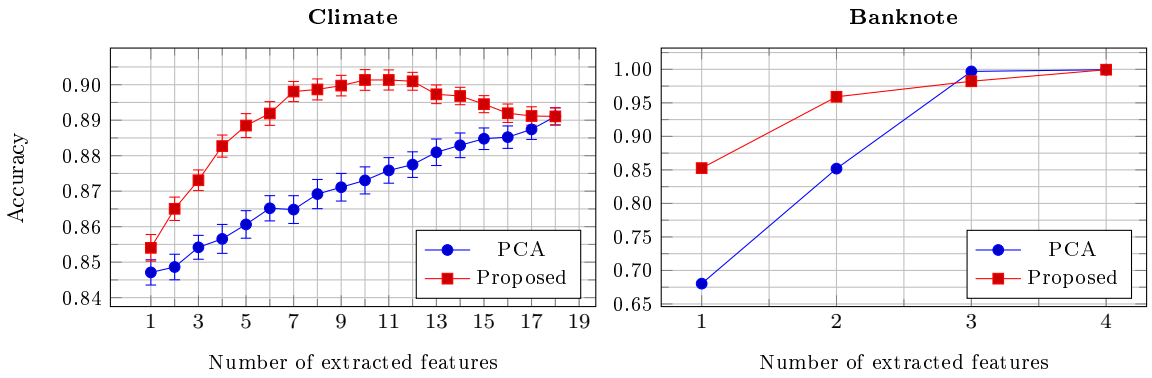


Fig. 1. Accuracy per number of extracted features for Climate and Banknote datasets using features extracted with PCA and the proposed method using **1-NN** Classifier.

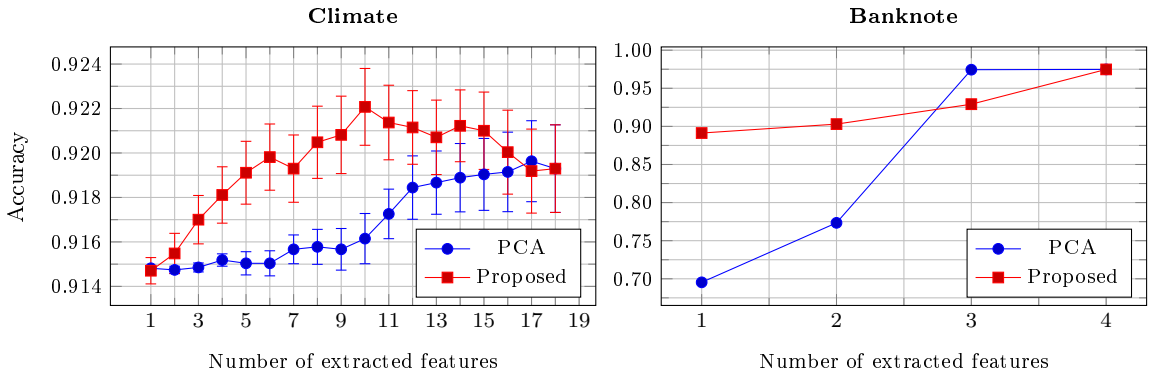


Fig. 2. Accuracy per number of extracted features for Climate and Banknote datasets using features extracted with PCA and the proposed method using **Naive Bayes** Classifier.

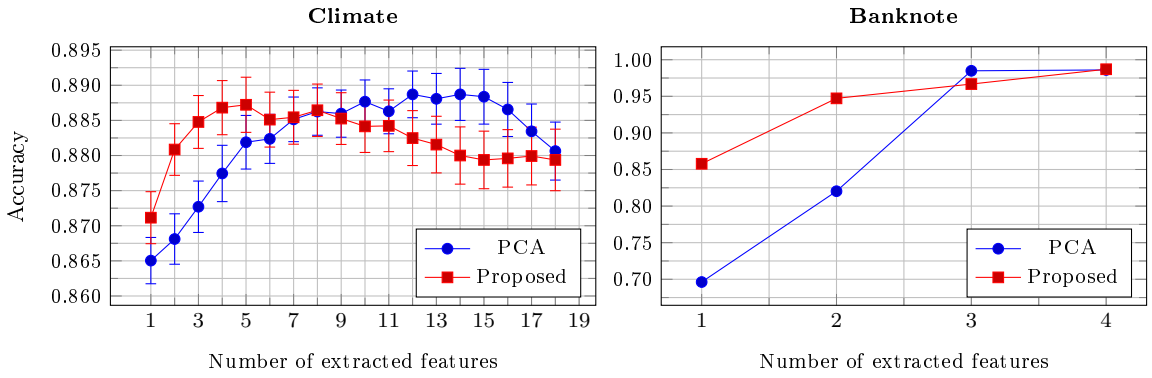


Fig. 3. Accuracy per number of extracted features for Climate and Banknote datasets using features extracted with PCA and the proposed method using **Decision Tree** Classifier.

[1] `http://www.mathworks.com/help/stats/index.html`.

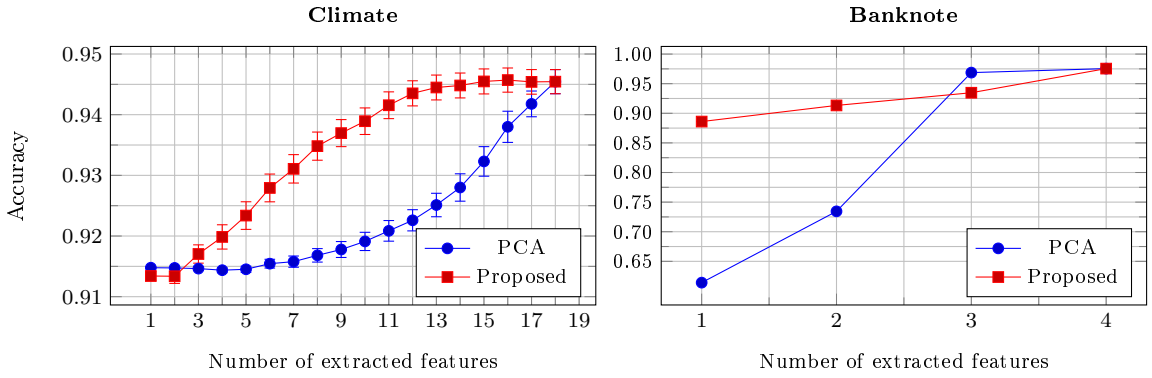


Fig. 4. Accuracy per number of extracted features for Climate and Banknote datasets using features extracted with PCA and the proposed method using **Linear Discriminant** Classifier.

We calculated confidence intervals assuming that each mean follows a Student's t distribution. For a 95% confidence level this interval is $[\bar{a} - E, \bar{a} + E]$, where $\bar{a}$ is the mean accuracy, b is the accuracy standard deviation, and $E = 1.984b/\sqrt{100}$. If there is no overlap between the confidence intervals of PCA and the proposed method the difference is considered significant [Schenker and Gentleman 2001]. The error bars shown for the Climate dataset in the Figures, represent the confidence intervals. For Banknote dataset, the values are too small to appear in the plots. The results for each classifier are discussed in the following subsections.

Analysis for the 1-NN classifier. The results show that the proposed method present accuracy significantly higher than PCA from 2 to 16 extracted features, for the Climate dataset; and for 1 and 2 extracted features for the Banknote dataset. For the Climate dataset (Table I), the maximum mean accuracy obtained using the proposed method was 0.901, for 10 extracted features. PCA presented a smaller mean accuracy 0.873 for the same number of features. The maximum mean accuracy obtained using PCA was 0.891 with 18 extracted features. For the Banknote dataset (Table III), for 1 and 2 extracted features the difference are quite significant. The obtained values were 0.680 (PCA), 0.852 (proposed) and 0.851 (PCA), 0.959 (proposed) respectively.

Analysis for the Naive Bayes classifier. The results show that the proposed method present accuracy significantly higher than PCA from 3 to 11 extracted features, for the Climate dataset; and for 1 and 2 extracted features for the Banknote dataset. For the Climate dataset (Table I), the maximum mean accuracy obtained using the proposed method was 0.922, for 10 extracted features. PCA presented a smaller mean accuracy 0.916 for the same number of features. The maximum mean accuracy obtained using PCA was 0.920 with 17 extracted features. For the Banknote dataset (Table III), for 1 and 2 extracted features the difference are quite significant. The obtained values were 0.695 (PCA), 0.891 (proposed) and 0.773 (PCA), 0.903 (proposed) respectively.

Analysis for the Decision Tree classifier. The results show that the proposed method present accuracy significantly higher than PCA from 2 to 4 extracted features, for the Climate dataset; and for 1 and 2 extracted features for the Banknote dataset. For the Climate dataset (Table II), the maximum mean accuracy obtained using the proposed method was 0.887, for 4 extracted features. PCA presented a smaller mean accuracy 0.877 for the same number of features. The maximum mean accuracy obtained using PCA was 0.889 with 12 extracted features. for 1 and 2 extracted features the difference are quite significant. The obtained values were 0.696 (PCA), 0.858 (proposed) and 0.820 (PCA), 0.947 (proposed) respectively.

Analysis for the Linear Discriminant classifier. The results show that the proposed method present accuracy significantly higher than PCA from 4 to 16 extracted features, for the Climate dataset; and for 1 and 2 extracted features for the Banknote dataset. For the Climate dataset (Table

Table I. The results of the **Climate** database showing the Mean Accuracy (M.A.), Standard Deviation (S.D.) and the number of Extracted Features (E.F.) for each method using 1-NN and Naive Bayes classifiers.

	1-NN		**Naive Bayes**	
	PCA	Proposed	PCA	Proposed
E.F.	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)
1	0.847 (0.021)	0.854 (0.022)	0.915 (0.000)	0.915 (0.003)
2	0.849 (0.021)	0.865 (0.019)	0.915 (0.001)	0.915 (0.005)
3	0.854 (0.020)	0.873 (0.017)	0.915 (0.001)	0.917 (0.006)
4	0.857 (0.024)	0.883 (0.018)	0.915 (0.002)	0.918 (0.007)
5	0.861 (0.023)	0.888 (0.019)	0.915 (0.003)	0.919 (0.008)
6	0.865 (0.021)	0.892 (0.019)	0.915 (0.003)	0.920 (0.008)
7	0.865 (0.023)	0.898 (0.016)	0.916 (0.004)	0.919 (0.008)
8	0.869 (0.024)	0.899 (0.017)	0.916 (0.004)	0.920 (0.009)
9	0.871 (0.023)	0.900 (0.016)	0.916 (0.005)	0.921 (0.010)
10	0.873 (0.022)	0.901 (0.016)	0.916 (0.006)	0.922 (0.009)
11	0.876 (0.021)	0.901 (0.016)	0.917 (0.006)	0.921 (0.009)
12	0.877 (0.021)	0.901 (0.014)	0.918 (0.008)	0.921 (0.009)
13	0.881 (0.021)	0.897 (0.015)	0.919 (0.008)	0.921 (0.009)
14	0.883 (0.020)	0.897 (0.014)	0.919 (0.008)	0.921 (0.009)
15	0.885 (0.017)	0.895 (0.014)	0.919 (0.009)	0.921 (0.010)
16	0.885 (0.018)	0.892 (0.015)	0.919 (0.010)	0.920 (0.010)
17	0.887 (0.016)	0.891 (0.015)	0.920 (0.010)	0.919 (0.010)
18	0.891 (0.014)	0.891 (0.014)	0.919 (0.011)	0.919 (0.011)

II) with 11 extracted features the proposed method have accuracy 0.942, and PCA 0.92. For the Banknote dataset (Table IV), for 1 and 2 extracted features the difference are quite significant. The obtained values were 0.614 (PCA), 0.886 (proposed) and 0.734 (PCA), 0.913 (proposed) respectively.

Classifier independent analysis. The proposed method presents greater accuracy for fewer extracted features if compared to PCA. For the Climate dataset and Decision Tree classifier, the maximum accuracy is achieve using only 5 of 18 features, using other classifiers more than 10 features are needed. For the Banknote dataset, if the accuracy for 1 extracted feature is about 0.9 the accuracy is similar for 2 extracted features (Naive Bayes and Linear Discriminant). If the accuracy for 1 extracted feature is about 0.85 the accuracy is about 0.95 for 2 extracted features (1-NN and Decision Tree).

6. CONCLUSION

We proposed a feature extraction technique that is similar to PCA but selects features that minimizes the Bayes error rate instead of features that maximizes the variance. The method presented a higher mean accuracy compared to PCA in two datasets using a small number of features. For future work, we will evaluate more databases, extend the proposed method to problems with more than two classes and to test in other PCA-based techniques [de Carvalho et al. 2015], [de Carvalho et al. 2014].

REFERENCES

Alencar, A. S. C., Gomes, J. P. P., Souza, A. H., Freire, L. A. M., Silva, J. W. F., Andrade, R. M. C., and Castro, M. F. Regularized supervised distance preserving projections for short-text classification. In *Intelligent Systems (BRACIS), 2014 Brazilian Conference on*. pp. 216–221, 2014.

Bair, E., Hastie, T., Paul, D., and Tibshirani, R. Prediction by supervised principal components. *Journal of the American Statistical Association* vol. 101, pp. 119–137, 2006.

Barshan, E., Ghodsi, A., Azimifar, Z., and Jahromi, M. Z. Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds. *Pattern Recognition* 44 (7): 1357 – 1371, 2011.

Bishop, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

Table II. The results of the **Climate** database showing the Mean Accuracy (M.A.), Standard Deviation (S.D.) and the number of Extracted Features (E.F.) for each method using 1-NN and Naive Bayes classifiers.

	Decision Tree		**Linear Discriminant**	
	PCA	Proposed	PCA	Proposed
E.F.	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)
1	0.865 (0.0192)	0.871 (0.0214)	0.915 (0.000)	0.913 (0.004)
2	0.868 (0.0208)	0.881 (0.0210)	0.915 (0.001)	0.913 (0.007)
3	0.873 (0.0211)	0.885 (0.0214)	0.915 (0.002)	0.917 (0.008)
4	0.877 (0.0230)	0.887 (0.0219)	0.914 (0.002)	0.920 (0.011)
5	0.882 (0.0218)	0.887 (0.0223)	0.915 (0.003)	0.923 (0.012)
6	0.882 (0.0199)	0.885 (0.0223)	0.915 (0.004)	0.928 (0.012)
7	0.885 (0.0181)	0.885 (0.0218)	0.916 (0.005)	0.931 (0.013)
8	0.886 (0.0192)	0.886 (0.0212)	0.917 (0.006)	0.935 (0.013)
9	0.886 (0.0191)	0.885 (0.0210)	0.918 (0.007)	0.937 (0.012)
10	0.888 (0.0176)	0.884 (0.0211)	0.919 (0.008)	0.939 (0.012)
11	0.886 (0.0182)	0.884 (0.0209)	0.921 (0.009)	0.942 (0.012)
12	0.889 (0.0189)	0.882 (0.0223)	0.923 (0.010)	0.944 (0.011)
13	0.888 (0.0205)	0.882 (0.0230)	0.925 (0.011)	0.944 (0.011)
14	0.889 (0.0210)	0.880 (0.0233)	0.928 (0.012)	0.945 (0.011)
15	0.888 (0.0221)	0.879 (0.0235)	0.932 (0.013)	0.945 (0.011)
16	0.887 (0.0219)	0.880 (0.0235)	0.938 (0.014)	0.946 (0.011)
17	0.883 (0.0222)	0.880 (0.0236)	0.942 (0.011)	0.945 (0.011)
18	0.881 (0.0236)	0.879 (0.0251)	0.945 (0.011)	0.945 (0.011)

Table III. The results of the **Banknote** database showing the Mean Accuracy (M.A.), Standard Deviation (S.D.) and the number of Extracted Features (E.F.) for each method.

	1-NN		**Naive Bayes**	
	PCA	Proposed	PCA	Proposed
E.F.	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)
1	0.680 (0.015)	0.853 (0.014)	0.695 (0.015)	0.891 (0.013)
2	0.852 (0.012)	0.959 (0.008)	0.773 (0.013)	0.903 (0.011)
3	0.997 (0.002)	0.982 (0.013)	0.974 (0.006)	0.929 (0.036)
4	0.999 (0.001)	0.999 (0.001)	0.975 (0.006)	0.975 (0.006)

Table IV. The results of the **Banknote** database showing the Mean Accuracy (M.A.), Standard Deviation (S.D.) and the number of Extracted Features (E.F.) for each method.

	Decision Tree		**Linear Discriminant**	
	PCA	Proposed	PCA	Proposed
E.F.	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)	M.A. (S.D.)
1	0.696 (0.017)	0.858 (0.017)	0.614 (0.011)	0.886 (0.012)
2	0.820 (0.017)	0.947 (0.011)	0.734 (0.011)	0.913 (0.009)
3	0.985 (0.006)	0.967 (0.019)	0.969 (0.006)	0.935 (0.029)
4	0.986 (0.006)	0.987 (0.007)	0.975 (0.005)	0.975 (0.005)

de Carvalho, T. B. A., Costa, A. M., Sibaldo, M. A. A., Tsang, I. R., and Cavalcanti, G. D. C. Supervised fractional eigenfaces. In *Image Processing (ICIP), 2015 IEEE International Conference on*. pp. 552–555, 2015.

de Carvalho, T. B. A., Sibaldo, M. A. A., Tsang, I. R., Cavalcanti, G. D. C., Tsang, I. J., and Sijbers, J. Fractional eigenfaces. In *2014 IEEE International Conference on Image Processing (ICIP)*. pp. 258–262, 2014.

Duda, R. O., Hart, P. E., and Stork, D. G. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000.

Lichman, M. UCI machine learning repository, 2013.

Schenker, N. and Gentleman, J. F. Statistical practice: On judging the significance of differences by examining the overlap between confidence intervals. 55 (3): 182–186, Aug., 2001.

Turk, M. and Pentland, A. Eigenfaces for recognition. *J. Cognitive Neuroscience* 3 (1): 71–86, Jan., 1991.

Caracterizando a dinâmica de evolução temporal de mensagens em mídias sociais

Bruno Conde Kind, Victor B.R. Jorge, Denise E.F. de Brito,
Roberto C.S.N.P. Souza, Wagner Meira Jr.

Departamento de Ciência da Computação
Universidade Federal de Minas Gerais
{condekind,victorbrjorge,denise.brit,nalon,meira}@dcc.ufmg.br

Abstract. Nos últimos anos, dados de mídias sociais *online* têm sido usados em uma vasta gama de aplicações como fontes informais de dados em tempo real. No entanto, a dinamicidade dessas fontes pode levar à degradação dos modelos utilizados, se eles não forem apropriadamente adaptados. Neste trabalho, é investigada a evolução temporal de uma base de dados reais através da análise das distribuições de classes e termos e da acurácia para uma tarefa de classificação de análise de sentimento. Os dados usados consistem em uma coleção do Twitter de 2010 a 2016 oriunda do Brasil, relacionada à dengue. É mostrada a importância da atualização do conjunto de treino, especialmente quando é usada a classificação em *batches*.

Categories and Subject Descriptors: H.2.8 [**Database Applications**]: Data Mining

Keywords: concept drift, mídias sociais, classificação, análise de sentimento

1. INTRODUÇÃO

A crescente popularidade de dispositivos portáteis como *tablets* e *smartphones* e a facilidade, cada vez maior, de acesso à Internet têm proporcionado uma explosão na quantidade de dados gerados e consumidos a todo tempo na Web. Em particular, mídias sociais como Twitter e Facebook são responsáveis por grande parte desse volume de dados. Atualmente, esses canais constituem um dos principais meios pelos quais as pessoas se comunicam e emitem opiniões. Assim, essa massa de dados gerados captura, de forma instantânea, as reações dos usuários em relação a acontecimentos cotidianos, refletindo a dinâmica dessa interação em tempo real e suas possíveis tendências.

A relativa facilidade na obtenção desses dados e sua característica de fluxo contínuo atraiu a atenção de diversas comunidades científicas. Dentre os interesses mais comuns no uso dessas informações estão o monitoramento e previsão de eventos reais [Sakaki et al. 2010; Gomide et al. 2011; Souza et al. 2014] e a compreensão do comportamento dos usuários [Silva et al. 2014; Lima et al. 2016]. Nesse processo, classificar as mensagens de acordo com o conteúdo textual é uma tarefa fundamental. Por outro lado, a qualidade dessa classificação é frequentemente afetada pelo caráter dinâmico das mensagens, que mudam suas características ao longo do tempo e de formas variadas.

Nesse fluxo contínuo de dados, a mudança do vocabulário é constante. Termos surgem e desaparecem a todo momento, seja de forma abrupta ou em padrões recorrentes. Além disso, suas propriedades mudam ao longo do tempo. Esse problema, chamado de *concept drift* [Klinkenberg and Joachims 2000; Gama et al. 2013] tem sido amplamente estudado com objetivo de se desenvolver classificadores que obtenham resultados mais robustos e acurados nesses cenários [Nishida et al. 2012; Forman 2006].

Neste trabalho, apresenta-se um estudo detalhado da evolução temporal de mensagens em mídias sociais *online* e como essa evolução pode impactar a acurácia da classificação. Para tanto, analisa-se três fatores principais: (i) como a acurácia da classificação se comporta em cenários quando dados de treino e teste estão temporalmente distantes; (ii) o desbalanceamento na distribuição das classes e como essa distribuição muda ao longo do tempo de diferentes formas; (iii) o surgimento de novos termos, desaparecimento e migração de termos através das classes ao longo do tempo. Observar essas características é fundamental para construir modelos de classificação que obtenham melhores resultados em cenários de fluxo de dados.

2. TRABALHOS RELACIONADOS

Em tarefas de classificação, assume-se que o treino seja representativo da base completa, ou seja, que o significado e a distribuição do treino e do teste sejam os mesmos. Com o armazenamento de dados durante vários anos, um problema que pode surgir é uma mudança significativa na base ao longo do tempo, chamada de *concept drift*, o que pode comprometer a qualidade da classificação. Diversos estudos foram feitos com o objetivo de caracterizar esse problema em bases de dados reais [Mourão et al. 2008; Salles et al. 2016], estabelecer critérios para a detecção automática do *concept drift* [Lanquillon and Renz 1999], propor soluções para ajustar a classificação na presença de *concept drift* [Klinkenberg and Renz 1998; Salles et al. 2010; Rocha et al. 2008], além de caracterizar de forma ampla os tipos e características desse problema [Tsymbal 2004].

Em [Lanquillon and Renz 1999], os autores descrevem dois tipos de atributos para detecção de mudanças: atributos do texto (distribuição das classes e frequência dos termos) e atributos da classificação (medidas de qualidade). Klinkenberg e Renz [1998] utilizam um terceiro tipo de atributo: os atributos do modelo de classificação, tais como as regras geradas. Em uma abordagem mais prática, Mourão et al. [2008] expõem o impacto temporal na classificação de textos científicos, ao passo que Nishida et al. [2012] caracterizam essas flutuações usando três conjuntos de dados coletados do Twitter, no contexto de esportes e televisão.

A principal diferença do presente trabalho para os demais está na fonte de dados. A base utilizada foi extraída de uma coleta longitudinal de uma mídia social, em um contexto de evolução lenta, que é o de vigilância epidemiológica, diferentemente do trabalho de Nishida et al., que utiliza uma base coletada em 14 dias. Nos últimos anos, o Twitter tem sido usado como fonte informal de dados em tempo real em cenários variados. Contudo, pouco se sabe sobre como os usuários se comportam durante um período longo de tempo nessa rede, se a forma como as pessoas se expressam sobre determinado assunto se mantém, e principalmente, se os modelos utilizados para classificação continuam adequados.

3. BASE DE DADOS

3.1 Coleta e pré-processamento

A base de dados utilizada neste trabalho foi coletada do Twitter através de sua *Streaming Application Programming Interface*[1] (API). As palavras-chave utilizadas para coleta são: dengue, *aedes* e *aegypti*. Trabalhos anteriores utilizaram os dados coletados através dos mesmos termos para estimar a incidência de dengue nas cidades brasileiras [Gomide et al. 2011; Souza et al. 2014].

O período de coleta de dados vai de Novembro de 2010 até Junho de 2016, compreendendo aproximadamente 7 anos. Esse período favorece o estudo feito neste trabalho, pois é possível observar como as mensagens relacionadas aos termos anteriores variaram no tempo, em diferentes cenários, tais como a emergência de outras doenças transmitidas pelo mesmo mosquito. Além disso, todas as mensagens foram filtradas pela localização e aquelas provenientes de fora do Brasil foram descartadas.

[1]https://dev.twitter.com/streaming/overview. Acessado em agosto de 2016.

3.2 Caracterização dos dados

3.2.1 *Volume.* Além de sua característica de tempo real, é interessante observar o volume de mensagens e de usuários no cenário analisado. A Tabela I apresenta um sumário do volume de mensagens coletadas. Naturalmente, como o período da coleta começa no final do ano de 2010, o volume de mensagens desse ano é consideravelmente menor. A partir de 2011, observa-se que o número de mensagens é considerável, sendo 2014 o ano com o menor volume. Esse volume reduzido em 2014 comparado aos demais anos pode ser justificado pela queda de 59% no número de casos reais de dengue no Brasil no mesmo ano[2]. Gomide et al. [2011] e Souza et al. [2014] observaram que a correlação entre a quantidade de mensagens relacionadas aos termos de coleta e a incidência da doença fornecida pelo Ministério da Saúde era alta em diversas cidades. A partir do ano de 2015, observa-se que o número de mensagens mais do que dobrou e continua crescendo em 2016. Esse último efeito pode estar relacionado à emergência de *Chikungunya* e *Zika*, doenças transmitidas pelo mesmo mosquito que a dengue. Além disso, é interessante notar que o crescimento do volume de usuários únicos acompanha o do volume de mensagens.

Tabela I. Características gerais da base de dados.

Dados	2010	2011	2012	2013	2014	2015	2016	Total
#msg	26472	505964	303102	285823	177093	417882	475461	2191797
#usuários únicos	17621	230830	147447	128260	81644	145435	150911	702464
%msg localizáveis	60	64	63	72	74	74	74	69
#cidades	1034	3162	2826	2991	2598	3131	3168	4597

3.2.2 *Localização geográfica.* A localização dos usuários no Twitter pode ser fornecida de três formas distintas: (i) através do GPS, no caso de indivíduos enviando mensagens a partir de dispositivos móveis; (ii) sugerida a partir do endereço IP; (iii) definida pelo próprio usuário em um campo livre. Cada forma de localização tem também um nível de confiança, sendo a localização pelo GPS a mais confiável. Com o objetivo de recuperar a localização para o maior número de usuários possível, as informações de localização disponíveis nas mensagens foram exaustivamente processadas. A Tabela I mostra o percentual de mensagens cuja localização foi possível resolver. É interessante notar que esse percentual aumenta ao longo do tempo. Esse fato pode estar associado com a crescente utilização de dispositivos móveis com a localização ativada. Outro fator interessante é que o número de cidades

Tabela II. Percentual de mensagens por região em cada ano.

Região	2010	2011	2012	2013	2014	2015	2016
Sudeste	54,11	50,38	48,02	59,16	55,69	57,83	52,26
Sul	10,29	11,93	12,53	12,44	13,10	12,10	15,15
Centro-Oeste	8,84	6,24	7,71	9,90	9,33	7,65	8,47
Norte	8,25	9,57	6,45	5,07	5,25	4,35	5,39
Nordeste	18,51	21,88	25,29	13,43	16,63	18,07	18,73

de onde são emitidas as mensagens é bastante representativo, embora o volume para algumas cidades seja muito reduzido, que pode estar relacionado à taxa de acesso à Internet naquele local. Para observar melhor a distribuição dessas mensagens, a Tabela II apresenta o percentual de mensagens por região do Brasil. Naturalmente, a maior parcela das mensagens vem da região Sudeste, onde está concentrada também a maior parte da população. Além disso, observa-se uma elevação no percentual

[2]http://portalsaude.saude.gov.br/index.php/cidadao/principal/agencia-saude/16191-casos-de-dengue-caem-59-e-obitos-40-em-2014. Acessado em agosto de 2016.

de mensagens da região nordeste nos últimos anos, onde foram notificados os primeiros casos de *Zika* no país.

4. ANÁLISE EXPERIMENTAL

Nesta seção, apresenta-se o estudo da dinâmica temporal das mensagens coletadas do Twitter. Esse estudo foi dividido em três partes principais: efeito temporal na classificação, distribuição das categorias de mensagens e evolução dos termos relevantes nas classes.

4.1 Classificação das mensagens

Como mencionado anteriormente, o objetivo deste estudo é entender a dinâmica temporal das mensagens e como ela pode impactar na acurácia da classificação dessas mensagens. Neste trabalho, o algoritmo utilizado é o Classificador Associativo sob Demanda (LAC) [Veloso et al. 2006]. Esse algoritmo constrói regras de associação, atribuindo padrões textuais a classes pré-definidas. Cada regra equivale a um voto cujo peso é dado pela confiança da regra. Uma instância de teste é então assinalada à classe com a maior quantidade de votos.

Para treinar o classificador, dados manualmente rotulados são requeridos. Similar a [Gomide et al. 2011] e [Souza et al. 2014], as mensagens foram classificadas em 5 categorias: experiência pessoal, informação, opinião, campanha e piada/ironia. Para cada ano do período de análise, um conjunto de aproximadamente 2 mil mensagens selecionadas aleatoriamente foram rotuladas manualmente para serem usadas como conjunto de treinamento para o classificador. Além disso, essas mensagens serviram de base na comparação da acurácia do classificador no cenário de variação das características do texto ao longo do tempo. O número de mensagens rotuladas para cada ano (2010 a 2016) foi, respectivamente: 2142, 2000, 2000, 2000, 2000, 1999 e 1999.

4.2 Efeito temporal

O primeiro experimento realizado tem como objetivo verificar a dinâmica temporal das mensagens. Espera-se que o vocabulário dos conteúdos emitidos em redes sociais mude constantemente. Dessa forma, quanto mais distante temporalmente o dado de treinamento encontra-se do dado que se deseja classificar, pior espera-se que a acurácia seja. Para verificar isso, este experimento foi dividido em duas etapas: (i) a primeira etapa consiste em usar as mensagens do ano de 2010 como conjunto de treinamento. Assim, essas mensagens foram utilizadas para construir o modelo de classificação e todas as mensagens também rotuladas manualmente dos anos seguintes foram classificadas a partir desse modelo, de forma que a acurácia foi obtida para cada ano; (ii) na segunda etapa, utilizou-se um protocolo de janela deslizante. Assim, cada ano foi utilizado como conjunto de treinamento para classificar as mensagens do ano seguinte. Essa estratégia busca aproximar temporalmente os conjuntos de treino e teste com vista a elevar a acurácia do resultado. Os resultados são apresentados no gráfico mais à esquerda da Figura 1.

A primeira observação é que, quando a classificação é feita utilizando a janela deslizante, a acurácia em geral é melhor, exceto pelos dois últimos anos em que ambos os resultados são muito semelhantes. É importante lembrar que os dois últimos possuem uma característica mais complexa ainda, com *burst* de mensagens sobre *Zika* e *Chikungunya*, levando ambas estratégias a um declínio na acurácia. Além disso, mesmo com o treino fixado no ano de 2010, observa-se uma leve melhora na acurácia em alguns anos. Esse resultado pode ser um efeito direto do funcionamento do LAC. Como o algoritmo funciona de forma *lazy*, criando uma projeção da base de acordo com o conjunto de teste, ele consegue adaptar o conjunto de treinamento da melhor forma possível para o conjunto de teste. Ainda assim, é possível observar que atualizar o conjunto de treinamento para um contexto temporal mais próximo é uma estratégia interessante. Contudo, o tamanho da janela de tempo pode ser difícil de se prever. Nos

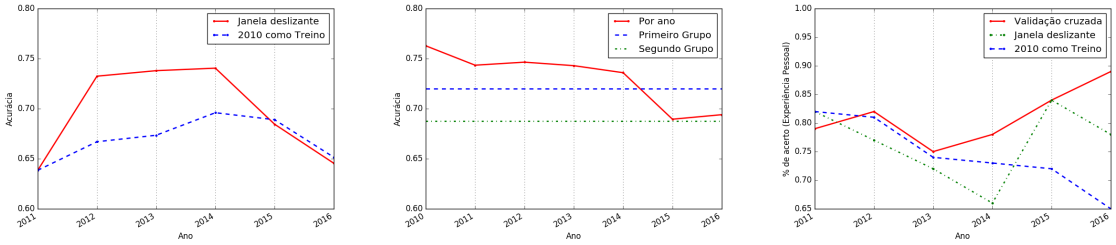


Figura 1. Efeito temporal. *Esquerda:* Acurácia do classificador usando o ano de 2010 como conjunto de treinamento fixo e também com o protocolo de janela deslizante. *Meio:* Acurácia do classificador usando um protocolo de validação cruzada. *Direita:* Precisão na classe experiência pessoal.

resultados obtidos, a janela de um ano parece não ser suficiente para solucionar o problema. Isso reflete a dinâmica das informações difundidas no Twitter, cuja granularidade temporal pode ser muito menor do que um ano, muitas vezes semanas ou dias.

Para melhor caracterizar esse efeito temporal na classificação das mensagens, um segundo experimento foi realizado. Esse segundo experimento foi dividido em três etapas e em todas foi utilizado um protocolo de validação cruzada com três conjuntos: (i) na primeira etapa, as mensagens manualmente rotuladas de cada ano, individualmente, foram separadas de forma aleatória em três conjuntos do mesmo tamanho. Em seguida, a validação cruzada foi aplicada, ou seja, cada combinação de dois dos conjuntos é utilizada para construir o modelo, que é então aplicado ao terceiro conjunto usado como teste. Nesse caso, os conjuntos de treinamento e teste pertencem sempre ao mesmo ano, e são portanto temporalmente mais próximos ainda do que no protocolo da janela deslizante do experimento anterior. A acurácia média para as três execuções é então calculada; (ii) na segunda etapa, o mesmo protocolo de validação cruzada é utilizado, porém o dado consiste em um único conjunto de 2002 mensagens selecionadas aleatoriamente dentre todas as mensagens rotuladas, independentemente do ano. Para evitar um efeito no viés da seleção, esse processo é repetido de forma independente por 10 vezes. A acurácia final é obtida através da média de cada uma das execuções independentes; (iii) a terceira etapa é similar à anterior, porém, no processo de seleção do conjunto de 2002 mensagens, um número constante é selecionado para cada ano da análise. Assim, 286 mensagens são selecionadas aleatoriamente em cada ano. Esse processo também é repetido de forma independente por 10 vezes e a acurácia média de cada execução é utilizada para a acurácia média final.

Os resultados desse conjunto de experimentos são apresentados no gráfico central da Figura 1. Pode-se notar que os resultados quando o ano do conjunto de treinamento é o mesmo do teste são melhores na maioria dos anos. Novamente, os dois últimos anos sofrem uma degradação na acurácia. Esse efeito reforça a hipótese da dinâmica temporal mencionada anteriormente. Novos termos surgem de forma abrupta (como no caso da *Zika*) e mesmo com um conjunto de treinamento temporalmente próximo, o classificador obteve resultados piores que nos anos anteriores e do que na etapa (ii). Além disso, os resultados da etapa (iii) também reforçam essa hipótese. Quando o número de mensagens foi fixado igualmente entre os anos, a pior acurácia foi observada. Isso aponta que um algoritmo de classificação deve idealmente ser capaz de lidar com padrões recorrentes, mas também com mudanças abruptas nos dados em um contexto de *concept drift*.

4.3 Avaliando a acurácia na categoria experiência pessoal

No contexto dos sistemas de alarme para vigilância epidemiológica, a categoria de experiência pessoal se destaca dentre as outras por estar relacionada a indivíduos reportando uma experiência com a doença. Grande parte das previsões feitas a partir dos dados informais obtidos nas redes sociais consideram essas mensagens como indicativos mais fortes [Gomide et al. 2011; Souza et al. 2014].

Para avaliar o efeito temporal na acurácia do classificador para essa categoria em especial, observou-se o resultado da precisão obtida na classificação dessa classe em alguns dos experimentos descritos na seção 4.2. O gráfico mais à direita da Figura 1 apresenta o percentual de acerto da classe experiência pessoal em relação às mensagens dessa mesma categoria.

Os resultados mostram claramente que, quando o conjunto de treinamento é mantido fixo no ano de 2010, a precisão ao prever a classe experiência pessoal sofre uma degradação consistente. Isso mostra que olhando para a classe experiência pessoal, de forma particular, quanto mais distante temporalmente estão os conjuntos de treino e teste, pior é o resultado. Os resultados obtidos com a janela deslizante seguem um caráter mais imprevisível. Como mencionado anteriormente, essa janela de um ano não é suficiente para resolver o problema dada a dinâmica do contexto e, portanto, o resultado acaba sofrendo grande variação. O melhor resultado é obtido na validação cruzada, em que os dados de treino e teste são do mesmo ano de análise. Esse resultados apontam uma forte evidência de que o efeito temporal é um fator importante a ser considerado e que pode causar degradação na performance dos classificadores.

4.4 Distribuição de classes

Nesta seção, é avaliado com mais detalhes o efeito temporal na distribuição das classes. A Figura 2 ilustra a variação na distribuição das cinco classes de forma mensal. No gráfico mais à esquerda da Figura 2, essa distribuição de classes é computada para o conjunto das mensagens rotuladas manualmente, como descrito na seção 4.1. No gráfico mais à direita, a distribuição de classes é obtida a partir dos dados classificados com o LAC, fixando o conjunto de treinamento com as mensagens do ano de 2010.

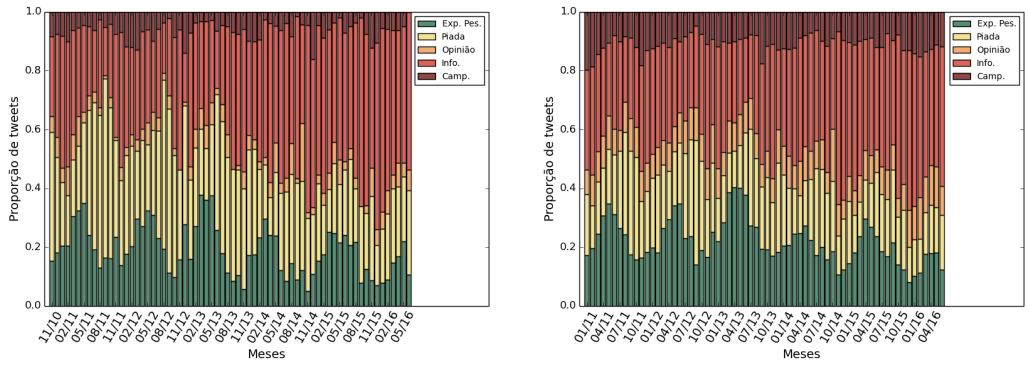


Figura 2. *Esquerda:* Mensagens classificadas manualmente; *Direita:* Mensagens classificadas pelo LAC.

É interessante notar que a proporção das classes de mensagens varia amplamente ao longo dos meses em cada ano. Essa variação reflete a representatividade de cada classe no tempo. Algumas dessas variações podem ser claramente associadas a padrões temporais. Por exemplo, a classe de experiência pessoal notoriamente aumenta sua representatividade em períodos coincidentes com o aumento de incidência da doença. Da mesma forma, algumas classes podem ganhar representatividade de forma repentina, como acontece com a classe de informação a partir de 2015. Nesse caso, o número de mensagens aumenta de forma abrupta. Esse efeito de variação da proporção das classes no tempo torna ainda mais desafiadora a construção de modelos de classificação que obtenham boa performance.

4.5 Evolução dos termos relevantes

Nesta seção, é feita a avaliação da evolução dos termos mais relevantes nas classes ao longo do tempo e de como isso pode impactar a tarefa de classificação. A todo momento surgem novos termos, outros

desaparecem e alguns migram através de classes se tornando mais ou menos relevantes em períodos distintos. Por exemplo, termos de mensagens atribuídas à classe de informação podem rapidamente surgir na classe de piada/ironia se houver algum fator externo que favoreça isso.

Para medir esse efeito na base de dados deste estudo, um vocabulário foi construído com os 50 termos mais relevantes de cada classe, em cada ano do período de análise. A relevância dos termos foi calculada através do ganho de informação [Yang and Pedersen 1997] após a aplicação de um corte de frequência mínima por classe (0.1%, 0.5% e 1%). Os termos obtidos podem ser considerados como os mais discriminativos de cada classe no determinado ano. Esses termos são representados em um vetor para cada ano em que cada posição contém um termo. A seguir, computa-se a união de todos os vetores da mesma classe para todos os anos. Essa estratégia permite avaliar o quanto o vocabulário de cada classe variou ao longo dos anos. No cenário extremo em que o vocabulário se mantivesse constante ao longo de todos os anos, o esperado é que o vetor união obtivesse exatamente os mesmos 50 termos para cada classe. No outro extremo, se os termos mais relevantes de cada ano fossem completamente diferentes, o vetor união final teria o tamanho equivalente à soma dos tamanhos dos vetores de cada ano. A Tabela III apresenta os resultados dessa união.

Tabela III. Quantidade de termos no vetor união de cada classe.

Frequência	Experiência Pessoal	Campanha	Informação	Opinião	Piada/Ironia
0.1%	161	161	160	174	188
0.5%	135	161	139	174	196
1.0%	125	168	138	176	199

Os resultados mostram que o surgimento de novos termos acontece em proporções diferentes em cada classe, ou seja, algumas categorias são mais dinâmicas do que outras. Por exemplo, as classes de piada/ironia e opinião são as que obtiveram a maior quantidade de termos no vetor união, o que condiz com a semântica relacionada a essas classes. Por outro lado, as classes de experiência pessoal e informação geraram a menor quantidade de novos termos. Os termos mais relevantes para a classe informação intuitivamente variam menos, estando muitas vezes relacionados a novos relatórios de registros de casos. O mesmo vale para a classe de experiência pessoal, em que as pessoas muitas vezes reportam os sintomas da doença. Nota-se que quanto mais distantes no tempo entre si estão as mensagens, maior a chance de que elas tenham poucos termos relevantes em comum. Esse efeito pode prejudicar amplamente a acurácia do classificador e deve ser levado em conta na construção de modelos de classificação para contextos de evolução temporal.

5. CONCLUSÕES

Neste trabalho foi feito um estudo da dinâmica temporal de mensagens em mídias sociais. O objetivo do estudo foi mostrar como a performance de algoritmos de classificação pode ser profundamente afetada em um cenário que está em constante mudança. O estudo se baseou em três fatores principais: o efeito da distância temporal entre conjuntos de treinamento e teste, a distribuição de classes e a variação de cada uma em diferentes períodos e, finalmente, como a movimentação de termos relevantes se comporta de forma diferente através da classes.

Os resultados obtidos mostraram claramente como a acurácia da classificação pode ser afetada por cada um dos cenários analisados. A variação na distribuição das classes mostra que o modelo ideal de classificador precisa ser capaz de responder a padrões recorrentes no tempo, mas também ao surgimento de eventos abruptos. A degradação observada na qualidade do classificador quando os conjuntos de treinamento e teste estão distantes temporalmente sugere que o conjunto de treino deve ser constantemente atualizado, mas sem perder as informações anteriores. Nesse caso, uma estratégia de ponderação pode ser utilizada. Finalmente, observou-se que o surgimento e desaparecimento de

termos relevantes no tempo acontece de forma variada de acordo com a categoria avaliada. Assim, o classificador precisa também ser capaz de responder a essa característica.

Todas essas conclusões impõem compromissos muitas vezes distintos no processo de construção de um classificador. Levar em consideração todas elas pode inviabilizar a geração de um modelo que seja capaz também de fornecer respostas na velocidade que a dinâmica da aplicação requer. Contudo, ainda há espaço para melhorar a acurácia dos modelos levando em consideração esses aspectos. Na sequência deste trabalho, pretende-se compreender também como a localidade espacial, além da temporal, influencia na mudança de conceito e os respectivos impactos nos resultados do classificador, aplicando também em outros cenários além da vigilância de saúde. Ainda, deseja-se avaliar melhor como esses compromissos mencionados anteriormente podem afetar a construção do modelo de classificação, com vista no desenvolvimento de uma estratégia que seja capaz de contornar os efeitos apresentados.

REFERÊNCIAS

Forman, G. Tackling concept drift by temporal inductive transfer. In *SIGIR* (2006-08-30), E. N. Efthimiadis, S. T. Dumais, D. Hawking, and K. Järvelin (Eds.). ACM, pp. 252–259, 2006.

Gama, J., Sebastião, R., and Rodrigues, P. P. On evaluating stream learning algorithms. *Machine Learning* 90 (3): 317–346, 2013.

Gomide, J., Veloso, A., Meira Jr., W., Almeida, V., Benevenuto, F., Ferraz, F., and Teixeira, M. Dengue surveillance based on a computational model of spatio-temporal locality of twitter. In *Proc. of the ACM WebSci Conference*, 2011.

Klinkenberg, R. and Joachims, T. Detecting concept drift with support vector machines. In *Proceedings of the Seventeenth International Conference on Machine Learning*. ICML '00. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 487–494, 2000.

Klinkenberg, R. and Renz, I. Adaptive information filtering: Learning in the presence of concept drifts. In *Workshop Notes of the ICML/AAAI-98 Workshop Learning for Text Categorization*. AAAI Press, pp. 33–40, 1998.

Lanquillon, C. and Renz, I. Adaptive information filtering: Detecting changes in text streams. In *Proc. of the 8th International Conference on Information and Knowledge Management*. pp. 538–544, 1999.

Lima, A., Stanojevic, R., Papagiannaki, D., Rodriguez, P., and González, M. C. Understanding individual routing behaviour. *The Royal Society Interface* 13 (116), 2016.

Mourão, F., Rocha, L., Araújo, R., Couto, T., Gonçalves, M., and Meira Jr., W. Understanding temporal aspects in document classification. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*. WSDM '08. ACM, New York, NY, USA, pp. 159–170, 2008.

Nishida, K., Hoshide, T., and Fujimura, K. Improving tweet stream classification by detecting changes in word probability. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '12. ACM, New York, NY, USA, pp. 971–980, 2012.

Rocha, L., Mourão, F., Pereira, A., Gonçalves, M. A., and Meira Jr., W. Exploiting temporal contexts in text classification. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*. CIKM '08. ACM, New York, NY, USA, pp. 243–252, 2008.

Sakaki, T., Okazaki, M., and Matsuo, Y. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*. pp. 851–860, 2010.

Salles, T., da Rocha, L. C., Gonçalves, M. A., Almeida, J. M., Mourão, F., Meira Jr., W., and Viegas, F. A quantitative analysis of the temporal effects on automatic text classification. *JASIST* 67 (7): 1639–1667, 2016.

Salles, T., Rocha, L., Pappa, G. L., Mourão, F., Meira Jr., W., and Gonçalves, M. Temporally-aware algorithms for document classification. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '10. ACM, New York, NY, USA, pp. 307–314, 2010.

Silva, T. H., de Melo, P. O. V., Almeida, J., and Loureiro, A. A. Large-scale study of city dynamics and urban social behavior using participatory sensing. *IEEE Wireless Commun.* 21 (1): 42–51, 2014.

Souza, R. C. S. N. P., de Brito, D. E. F., Cardoso, R. L., de Oliveira, D. M., Meira Jr., W., and Pappa, G. L. An evolutionary methodology for handling data scarcity and noise in monitoring real events from social media data. In *14th IBERAMIA Conference*, L. A. Bazzan and K. Pichara (Eds.). pp. 295–306, 2014.

Tsymbal, A. The problem of concept drift: Definitions and related work. Tech. rep., 2004.

Veloso, A., Meira Jr., W., and Zaki, M. J. Lazy associative classification. In *Proc. of the International Conference on Data Mining*. pp. 645–654, 2006.

Yang, Y. and Pedersen, J. O. A comparative study on feature selection in text categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning*. pp. 412–420, 1997.

Redes sociais na saúde: conectando os mundos real e virtual na investigação da obesidade

P. P. V. Brum, K. B. Enes, D. E. F. de Britto, T. O. Cunha, W. Meira Jr. e G. L. Pappa

Universidade Federal de Minas Gerais, Brasil
{pedrobrum, karen, denise.brit, tocunha, meira, glpappa}@dcc.ufmg.br

Resumo. Redes sociais *online* são grandes fontes de extração de dados sobre os mais diversos assuntos. É de particular interesse a obtenção de informações relevantes na previsão de eventos do mundo real, incluindo assuntos relacionados à saúde da população. A obesidade é uma questão de saúde pública e afeta cerca de um terço da população mundial. Nesse sentido, é importante analisar como os dados de redes sociais podem se relacionar com os indicadores de obesidade do mundo real. Por essa razão, esse trabalho apresenta um estudo comparativo entre os indicadores de obesidade da comunidade *loseit (Lose the fat)* do *Reddit* e os indicadores de um estudo americano sobre saúde pública, o BRFSS. Os resultados obtidos mostram que a comunidade é composta em sua maioria por mulheres com menos de 30 anos. Ao final do intervalo de tempo analisado, mais da metade dos usuários deixa de ser obeso e a distribuição dos usuários em categorias é bem próxima daquela reportada pelo BRFSS.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning

Keywords: redes sociais online, obesidade, dados reais *versus* virtuais

1. INTRODUÇÃO

Redes sociais *online* se tornaram muito populares por fornecerem um serviço que permite que seus usuários criem e compartilhem informações sobre os mais diversos temas, em um ambiente distribuído e interativo. Essas comunidades funcionam como meios de influência nos quais os usuários interagem expondo opiniões, sentimentos e questionamentos [Mislove et al. 2007].

A crescente quantidade de dados digitais gerados dessas interações dos usuários, cria uma nova oportunidade para estudar vários tópicos, incluindo aqueles relacionados à saúde [Dredze 2012]. Nessas comunidades usuários acabam revelando muito sobre seus hábitos de vida de forma espontânea e a anonimidade oferecida por muitas delas faz com que o pudor e a vergonha de falar sobre certos assuntos ou revelar hábitos não saudáveis fique de lado.

Nesse contexto, diversos estudos já foram realizados no ambiente *online* para entender uma doença crônica em constante crescimento: a obesidade [Turner-McGrievy and Tate 2013; Chunara et al. 2013]. Nos últimos 30 anos a taxa de obesidade cresceu em vários países do mundo, tornando-se um grande problema de saúde pública. A obesidade está relacionada a um aumento no risco de desenvolver mais de 20 doenças crônicas [Thiese et al. 2015]. Desse modo, a investigação de assuntos relacionados à obesidade em meios digitais pode ajudar profissionais no entendimento das dinâmicas nas comunidades, o efeito do suporte social recebido, bem como fornecer informações importantes para o desenvolvimento de intervenções *online*.

Porém, uma importante questão pouco analisada na literatura é como esses dados *online* se relacionam com os indicadores de obesidade do mundo real. Esses indicadores são característicasquantificá-

veis que os pesquisadores utilizam como evidência para descrever o nível de obesidade da população. Normalmente, essas informações são adquiridas por telefone, em questionários anuais para monitorar a frequência e a distribuição dos fatores de riscos. Esses esforços fornecem valiosas informações, contudo, eles necessitam de muito tempo para serem concluídos, tem um custo financeiro elevado e geralmente são limitados em amostra, frequência e granularidade geográfica.

Esse trabalho apresenta um estudo comparativo entre os indicadores de obesidade da rede social *Reddit*[1], especificamente a subcomunidade de perda de peso *Loseit (Lose the fat)* e o *Behavioral Risk Factor Surveillance System* (BRFSS), um sistema de questionários por telefone que coleta dados de saúde de residentes dos Estados Unidos. O principal objetivo do presente trabalho é a caracterização dos dados e usuários da comunidade, comparando as características dos usuários e o indicador de obesidade baseado no índice de massa corporal obtidos a partir dos dados *online* com os dados obtidos pelo BRFSS.

Ao contrário dos métodos tradicionais que se baseiam em dados de questionários, comunidades como o *loseit* fornecem dados dinâmicos e públicos, o que motiva o desenvolvimento de novas técnicas e sistemas. Em geral, aplicações em redes sociais *online* têm o potencial para se tornarem ferramentas para auxiliar os profissionais de saúde a proporcionar um tratamento direcionado e mais personalizado.

As análises mostraram que, após certo período na rede, mais da metade dos usuários deixa de ser obeso e a distribuição final em categorias de obesidade é bem próxima do panorama do BRFSS. O gênero dos usuários é consistente com aquele mostrado em estudos reais, com uma maior porcentagem de mulheres com problemas de peso. Em relação à idade, a questão das redes sociais realmente concentra a comunidade em usuário na faixa de 20 a 30 anos.

2. TRABALHOS RELACIONADOS

A ideia de correlacionar dados de comunidades virtuais com comunidades reais tem se tornado recentemente um assunto de extremo interesse na comunidade médica, mas metodologias de como isso deve ser feito ainda não foram estabelecidas. A grande maioria dos estudos procura por correlações entre variáveis dos dois mundos. Essa é a principal ideia desse trabalho, que no futuro será estendido para tentar determinar correlações entre variáveis mais complexas e causalidade.

Dos estudos já realizados para correlacionar dados de comunidades virtuais com comunidades reais, [Eichstaedt et al. 2015] usaram dados extraídos da linguagem utilizada no Twitter e, considerando diferentes regiões dos EUA, conseguiram correlacionar as taxas de mortalidade por doenças arteriais coronarianas obtidas pela rede social com os dados reais. Nessa mesma linha, [Padrez et al. 2015] obtiveram dados de pacientes em salas de emergência nos EUA e caracterizaram aqueles que consentiram o uso de seus dados e os que se recusaram a fornecê-los. Para os primeiros, eles caracterizaram suas atividade *online*, e defenderam a importância de um registro médico que também pudesse ser integrado com a vida diária dos pacientes nas redes sociais.

Já em [Braithwaite et al. 2016] os autores correlacionaram dados do Twitter a respeito de suicídio com os índices nacionais de ocorrência. Os autores em [Ranney et al. 2016] fizeram uma análise parecida, mas correlacionando apenas o volume de mensagens sobre álcool em redes sociais em uma determinada região e o número de ocorrências em uma sala de emergência. Em relação à obesidade, não foi encontrado nenhum estudo nessa linha.

3. FONTES DE DADOS

Como mencionado, esse trabalho tem como objetivo comparar dados reais com dados virtuais. Essa seção descreve as fontes de dados utilizadas para realizar essa comparação.

[1] *Reddit: The front page of the internet.* www.reddit.com

Dados *online* - *loseit*: o *Reddit*, forma reduzida das palavras "read it", é uma rede social multilíngue fundada em 2005. Atualmente, de acordo com os dados do *SimilarWeb*[2], o *Reddit* ocupa a 26^a posição no raking de acessos à sites no mundo. Entre as redes sociais, trata-se da sexta rede mais utilizada. O conteúdo do site é criado por usuários chamados de *redditors*, que podem postar textos, links, imagens e vídeos. Além disso, os *redditors* podem votar (positiva ou negativamente) nas postagens dos demais usuários. O conteúdo das postagens é dividido em várias comunidades chamadas *subreddits*.

Um dos *subreddits* de maior popularidade na rede é o *loseit*[3], uma comunidade de suporte e apoio aos usuários que estão em busca da perda de peso. Os usuários postam periodicamente comentários sobre sua rotina de exercícios, dietas e sentimentos relacionados a metas de perda de peso. Essa comunidade existe há cerca de 6 anos e conta com aproximadamente 337 mil usuários. Ela foi escolhida para esse estudo devido a sua popularidade e padrões pré-estabelecidos de postagem, que possibilitam inferências sobre características do usuário. A partir das postagens e comentários do *loseit*, uma base de dados foi construída para inferência e análise das informações contidas nos dados coletados.

Dados reais - *Behavioral Risk Factor Surveillance System*: os indicadores obtidos a partir dos dados do *loseit* são comparados aos resultados reais obtidos pelo *Behavioral Risk Factor Surveillance System*[4] (BRFSS). O BRFSS, programa criado em 1984 nos EUA, realiza, anualmente, uma coleta de dados a partir de 400 mil entrevistas telefônicas em todos os 50 estados americanos. Esse programa é considerado o maior sistema de entrevistas em saúde do mundo e tem como objetivo fazer uma análise comportamental dos americanos relacionados à saúde de risco, doenças crônicas e prevenção.

A base de dados do BRFSS conta com os seguintes atributos de interesse em relação a obesidade: sexo, idade e índice de massa corporal (IMC). A utilização dos dados provenientes do BRFSS e a comparação com as informações extraídas do *loseit* é viável e adequada, já que cerca de 43% dos acessos feitos ao *Reddit* são provenientes dos Estados Unidos e todas as mensagens coletadas foram escritas em inglês.

4. CONSTRUÇÃO DA BASE DE DADOS

Os dados utilizados nesse estudo foram extraídos de um repositório de dados do *Reddit* disponível *online*[5]. Foram coletados todas as postagens e comentários do *loseit* de janeiro de 2010 a maio de 2016, em um total de 164.762 postagens e 2.074.413 comentários. Desse total, foram selecionados apenas os usuários ainda ativos na comunidade, isto é, que não apagaram sua conta, totalizando 124.251 postagens e 1.824.144 comentários provenientes de 174.480 usuários. Desses 174.480 usuários, 62.524 escreveram pelo menos uma postagem e 161.972 escreveram pelo menos um comentário. A Tabela I resume as estatísticas iniciais sobre comentários e postagens.

Tabela I: Estatísticas iniciais sobre postagens e comentários em relação ao número de usuários

	Usuários com P e C $\geq$1	Média	Mediana	Média de palavras
Postagens (P)	62524 (35,8%)	1	1	122
Comentários (C)	161972 (92,8%)	11	2	45

A partir da tabela, observa-se que a grande maioria dos usuários do *loseit* comentam na rede, porém, menos da metade fazem postagens. É importante notar que os comentários podem servir como forma de incentivo para os usuários que criam as publicações. Dessa forma, a participação ativa dos usuários da comunidade pode ser fundamental para aqueles que fazem postagens. Outro fator relevante a ser observado é que a média do número de palavras por postagem é quase 3 vezes maior do que para

[2] *SimilarWeb.* https://www.similarweb.com/website/reddit.com, acesso: 15 de agosto de 2016
[3] *Loseit – Lose the fat.* https://www.reddit.com/r/loseit/
[4] *BRFSS: Behavioral Risk Factor Surveillance System.* http://www.cdc.gov/BRFSS/about/index.htm
[5] *Directory Contents:* http://files.pushshift.io/reddit/

os comentários. O fato do número de comentários ser muito maior do que o número de postagens é esperado, uma vez que a rede é muito mais centrada nas mensagens que nos usuários [Buntain and Golbeck 2014], o que difere significantemente de outras redes socias populares, como o Facebook.

A partir das postagens e comentários dos usuário do *loseit*, foram utilizadas expressões regulares para extrair características físicas e de peso do usuário, incluindo gênero, idade, altura e relatos de perda de peso. Note que, no futuro, informações de atividades dos usuários na rede podem ser de grande valia para entender as relações entre seu comportamento e perda de peso. Porém, nesse artigo é de interesse investigar se essa amostra de usuários reflete a população de usuários obesos no mundo real, a partir dos dados coletados do BRFSS.

As expressões regulares buscam determinados padrões no texto capazes de auxiliar na inferência das características de interesse. Dessa forma, ao buscar por informações sobre o atributo "peso", a expressão regular procura no texto palavras possivelmente relacionadas ao peso atual, a perda ou ganho de peso e a alimentação do indivíduo. Portanto, se um texto possui, por exemplo, as palavras "perdeu", "calorias", "libras", "peso" ou algum número seguido de uma unidade de peso (libras/quilos), é provável que esse texto possua o peso atual do autor. A extração da idade, altura e sexo são realizados de forma análoga, isto é, se o texto possui as palavras "anos", "idade" e um número, é provável que esse número seja a idade atual do usuário. Essa extração foi validada de forma manual para uma amostra dos usuários, mostrando-se confiável.

Caso a altura e o peso de um determinado usuário sejam conhecidos, pode-se calcular o índice de massa corporal (IMC) e classificar o usuário de acordo com o seu grau de obesidade. O IMC é a razão entre o peso em quilos (kg) e o quadrado da altura em metros (m). Os diferentes graus de obesidade são definidos pela Organização Mundial de Saúde[6] (OMS) e divididos em 5 categorias: saudável ($18,5 \leq \text{IMC} \leq 24,9$), sobrepeso ($25,0 \leq \text{IMC} \leq 29,9$), obesidade grau 1 ($30,0 \leq \text{IMC} \leq 34,9$), obesidade grau 2 ($35,0 \leq \text{IMC} \leq 39,9$) e obesidade grau 3 ($\text{IMC} \geq 40,0$).

Além das informações extraídas a partir do texto do usuário, a comunidade também oferece aos usuários uma forma fácil de reportar perdas ou ganhos de peso por meio do campo *user flairs*, que é de preenchimento não obrigatório e que aparece ao lado do nome do usuário em suas postagens. Para as postagens e comentários que têm o *user flair* do usuário, foi associada também a data da publicação. A partir dessas informações é possível recuperar, por exemplo, o número de vezes em que o usuário perdeu ou ganhou peso, o intervalo médio entre perdas ou ganhos de peso e o intervalo entre submissões com *user flairs*. Pode ser possível ainda recuperar o peso atual ou final do usuário.

A Tabela II apresenta algumas observações em relação aos usuários e informações de seus atributos extraídos dos textos publicados e *user flairs*. É importante observar que, para a grande maioria dos usuários, não foi possível encontrar os atributos relacionados a características físicas. Por outro lado, uma porcentagem um pouco maior dos usuários reporta informações sobre peso inicial e perda de peso. Um outro fator observado a partir da base de dados montada é que, dos 14.690 usuários que reportaram o sexo, 46,8% são homens e 53,2% são mulheres.

Tabela II: Informações Básicas: usuários por atributo, considerando o total inicial de 174480 usuários.

Atributo	Nº usuários	%
Altura	11.614	6,7
Idade	12.212	7,0
Sexo	14.690	8,4
Peso Inicial	25.239	14,5
IMC	5.256	3,0
User flair≥3	26.804	15,4
Peso Inicial + User flair≥3	10714	6,1

[6] *World Health Organization.* http://www.who.int/mediacentre/factsheets/fs311/en/

Depois da extração dos dados acima, apenas os usuários ainda ativos na comunidade e que contém os atributos sexo, idade, altura, peso inicial e pelo menos 3 *user flairs* foram selecionadas para as etapas seguintes. Os atributos altura e peso inicial são parâmetros fundamentais para o cálculo do IMC. O atributo sexo é selecionado para medir as diferenças entre homens e mulheres durante o processo de perda de peso, fator muito conhecido na literatura [Yaemsiri et al. 2011]. Por sua vez, é desejável que o usuário tenha pelo menos 3 *user flairs* para que seja possível obter uma estimativa do peso final dele, baseado em suas diferenças de peso. Ao considerar a quantidade de usuários que reportaram todos os atributos mencionados, úteis para a comparação com o BRFSS, o tamanho da amostra é reduzido para 1.066 indivíduos, isto é, cerca de 0,6 % do total de usuários ativos, mas um número muito maior que o utilizado em estudos anteriormente reportados na literatura [Turner-McGrievy and Tate 2013; Chunara et al. 2013].

5. ANÁLISE COMPARATIVA: *LOSEIT VERSUS* BRFSS

Antes de iniciarmos a análise comparativa das amostras populacionais no *loseit* e BRFSS, é importante ressaltar que a base *online* foi coletada de uma comunidade com objetivo específico: perda de peso. Enquanto isso, o BRFSS foca em toda a população, independente dos indivíduos estarem interessados em perder peso ou não. No entanto, a comparação feita nessa seção é interessante principalmente porque mostra qual a proporção do usuários em diferentes categorias de peso que se interessam em participar de comunidades *online*, possibilitando que médicos e agentes de saúde saibam quais indivíduos precisam de mais apoio para começar a frequentar a rede.

Para a análise comparativa entre o *Loseit* e os dados reais do BRFSS, a amostra utilizada contém 1.066 usários, dos quais 35,3% são do sexo masculino e 64,7% do sexo feminino, mostrando uma tendência das mulheres compartilharem mais informações. A base do BRFSS original conta com 433.970 indivíduos. Desses, foram selecionados os usuários com idade menor que 60 anos, restando 249.970 usuários. Essa filtragem é interessante porque sabemos que indivíduos mais velhos, que são a maioria nesse estudo, não tem contato com redes sociais. Dos usuários selecionados, 45,3% são do sexo masculino e 54,7% do sexo feminino. A prevalência da obesidade em mulheres é conhecida na literatura médica [Yaemsiri et al. 2011], juntamente com sua maior adesão a programas para perda de peso.

A Figura 1 mostra a distribuição dos usuários por categorias de peso de acordo com as faixas de classificação da OMS. Note que, para os usuários do *loseit*, esse dado foi calculado utilizando o primeiro peso reportado na comunidade (peso inicial). Inicialmente, observe que os indivíduos com peso normal e sobrepeso representam, no mundo real, 34,8% e 34,7% dos usuários, enquanto que no *loseit* eles correspondem a 17,1% e 26,1%. O fato da base real ter um número maior de indivíduos normais e com sobrepeso comprovam os dados da OMS de que 30% da população sofre de obesidade. Além disso, como mencionado, as entrevistas do BRFSS não envolvem apenas pessoas sofrendo com problemas de peso e, portanto, esse fator deve ser levado em consideração. Já a presença de pessoas de IMC normal na comunidade pode ser explicada por duas hipóteses: (i) usuários que mesmo na faixa de peso normal desejam emagrecer e (ii) usuários que entram na comunidade com o intuito de motivar os demais ou entender os problemas da obesidade para ajudar amigos e familiares na vida real.

Já em relação aos indivíduos nas categorias de obesidade 3, por exemplo, a proporção de indivíduos com grau 3 de obesidade nos dados virtuais é quase 3 vezes maior do que no cenário real. Essa característica é esperada, uma vez que trata-se de uma comunidade de perda de peso. O mesmo padrão é repetido para as outras categorias de obesidade. Posteriormente, mostra-se como esses números mudam depois do usuário fazer parte da comunidade do *loseit* por um determinado período de tempo, se aproximando da distribuição real dos dados.

Como informação complementar à discussão da Figura 1, os gráficos da Figura 2 mostram a proporção dos indivíduos dos sexos masculino e feminino separados pelas categorias de peso. Para os dados do *loseit*, é possível observar a maior parte das mulheres estão na categoria de sobrepeso, enquanto a

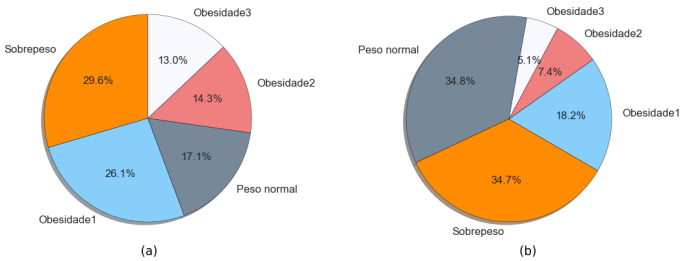


Fig. 1: Distribuição dos indivíduos por categoria de peso inicial: (a) Dados do *Reddit*, (b) Dados BRFSS.

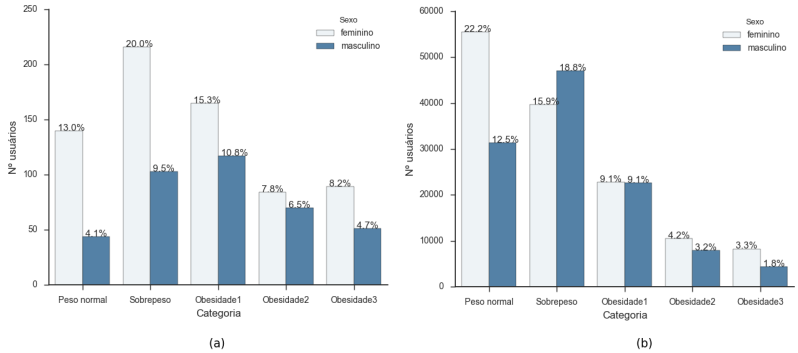


Fig. 2: Distribuição dos indivíduos por sexo e categoria de peso: (a) Dados do *Reddit*, (b) Dados BRFSS.

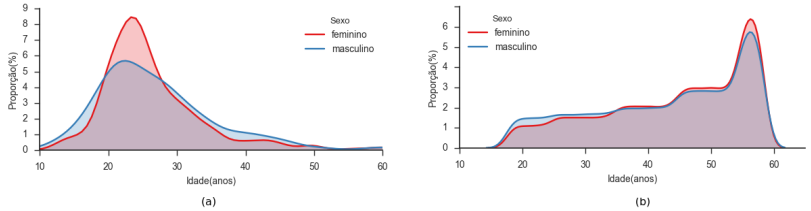


Fig. 3: Proporção de indivíduos por idade e sexo: (a) Dados do *Reddit*, (b) Dados BRFSS.

maior parte dos homens estão na categoria obesidade 1. Isso pode indicar que as mulheres começam a ser preocupar com o peso e procurar ajuda antes dos homens. Para ambos os sexos, existem mais indivíduos nas categorias sobrepeso e obesidade 1 do que nas demais. Por outro lado, no caso dos dados reais, a categoria de sobrepeso contém mais indivíduos homens do que mulheres. Assim, apesar da porcentagem de indivíduos do sexo feminino ser maior do que a do sexo masculino, em ambos os cenários, a proporção entre as classes de peso é mantida equivalente até aqui.

Já a Figura 3 compara a proporção de indivíduos dos sexos masculino e feminino em relação à idade para os dados virtuais e reais coletados. A análise dos gráficos permite identificar uma grande diferença entre os públicos atingidos pela rede social e pelo BRFSS. É possível notar que a maior parte dos indivíduos, tanto do sexo masculino quanto do feminino, estão concentrados na faixa de 20 a 30 anos para o *loseit* enquanto que para dos dados reais essa distribuição é mais uniforme. Nota-se ainda que, para os dados virtuais, a faixa etária de maior concentração de indivíduos é ainda mais restrita para mulheres. Para os dados reais, o padrão de distribuição etária para homens e mulheres é muito próximo.

As Figuras 4 e 5 apresentam resultados provenientes apenas das análises dos dados do *loseit* em relação a perda de peso. Em particular, a Figura 4(a) mostra que os usuários que se declaram do

sexo masculino perdem mais peso que os usuários do sexo feminino para qualquer categoria inicial na qual o usuário se encontrava. Esse fato também é bem conhecido da literatura, uma vez que o metabolismo dos homens, em geral, é mais acelerado. Os homens da categoria obesidade 3 foram os que mais perderam peso. Essa média de perda de peso foi calculada a partir do peso final do usuário, obtido através do peso inicial menos as perdas declaradas nos *user flairs*. Considerando apenas o sexo feminino, os usuários que perderam mais peso são aqueles que inicialmente estavam na categoria de obesidade 2.

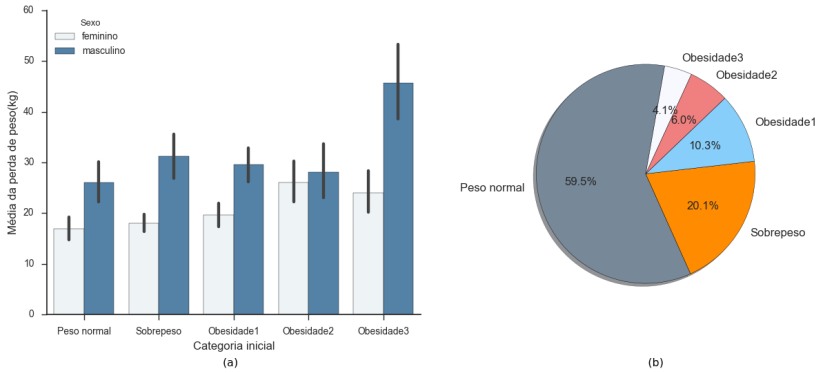


Fig. 4: Resultados finais do *loseit*: (a) Média da perda de peso para cada categoria e para cada sexo, (b) Distribuição dos usuários na comunidade por categoria de peso final.

A Figura 4(b) mostra a distribuição de usuários por categoria de obesidade após o cálculo do peso final dos indivíduos. Contrastando os panoramas inicial (Figura 1(a)) e final, nota-se que a porcentagem de usuários na categoria peso normal passa de 17,1% para 59,5%. Outro fator relevante é que o percentual de usuários em cada uma das outras categorias foi reduzido drasticamente. Ao comparar a distribuição final com a distribuição dos dados reais (Figura 1(b)), observa-se que o panorama final é mais próximo daquele mostrado pelos dados reais do que o panorama inicial.

Também foi observado a frequência de postagem de perda de peso dos usuários de diferentes sexos, medida com base no número de postagens contendo *user flairs*. A Figura 5 apresenta os resultados obtidos. A partir do gráfico nota-se que é clara a diferença entre ambos os sexos. O sexo femino é responsável pela maior parte dos *user flairs* reportados. Nota-se ainda que 9,9% dos usuários (7,1% de mulheres e 2,8% de homens) possuem pelo menos 100 postagens com *user flair*. Esse resultado pode estar associado a usuários com características de auto-monitoramento [Burke et al. 2011], situação em que o indivíduo posta constantemente sobre suas perdas de peso como meio de motivação para si próprio. Esse fator é conhecido como um dos responsáveis pela perda de peso do indivíduo.

6. CONCLUSÕES E TRABALHOS FUTUROS

Esse trabalho apresentou uma análise comparativa entre os dados extraídos da comunidade *loseit* do *Reddit* e os dados do BRFSS, um estudo sobre a saúde dos americanos. A comparação só foi possível porque foram extraídos dados sobre os usuários utilizando o texto livre postado na rede. Para a extração dos dados, foram utilizadas expressões regulares, já que padrões claros foram observados, tornando desnecessária a utilização de técnicas mais sofisticadas nesse estudo preliminar.

Observou-se que a proporção de usuários dos sexos masculino e feminino para os dados reais e virtuais é praticamente a mesma. Embora a maior parte dos usuários do *loseit* seja do sexo feminino, os usuários do sexo masculino são os que mais perdem peso na comunidade. Ao final do período avaliado, a maior parte dos usuários consegue reduzir o peso e cerca de 59,5% passam a pertencer a categoria de peso normal. Em relação a idade, os resultados mostraram que os usuários do *loseit* são em sua maioria jovens, entre 20 e 30 anos, enquanto os indivíduos do BRFSS estão distribuídos

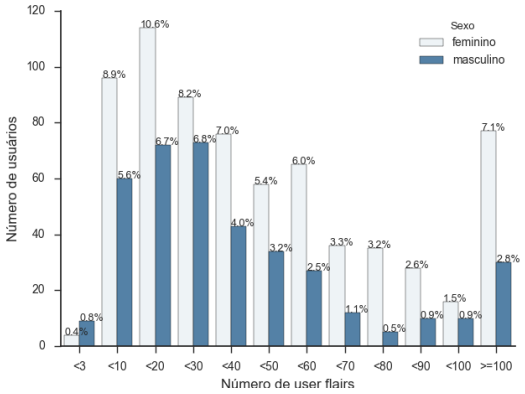


Fig. 5: Distribuição dos usuários na comunidade pelo número de *user flairs* para ambos os sexos.

de maneira mais uniforme entre as faixas etárias. Entre as aplicações possíveis para esses resultados preliminares, está a recomendação de posts entre usuários de uma mesma categoria de obesidade, por exemplo.

Como trabalhos futuros, sugere-se inicialmente o uso de técnicas de processamento de linguagem natural para aumentar o número de atributos extraídos por usuário e, consequentemente, o tamanho da base. Nessa mesma linha, métodos de imputação de dados faltantes serão explorados. Por fim, um estudo longitudinal sobre o impacto do comportamento do usuário na rede ao longo do tempo e a perda de peso obtida deve ser realizado.

REFERENCES

Braithwaite, R. S., Giraud-Carrier, C., West, J., Barnes, D. M., and Hanson, L. C. Validating machine learning algorithms for twitter data against established measures of suicidality. *JMIR Mental Health* 3 (2): e21, 2016.

Buntain, C. and Golbeck, J. Identifying social roles in reddit using network structure. In *Proceedings of the 23rd International Conference on World Wide Web*, 2014.

Burke, L. E., Wang, J., and Sevick, M. A. Self-monitoring in weight loss: a systematic review of the literature. *Journal of the American Dietetic Association*, 2011.

Chunara, R., Bouton, L., Ayers, J. W., and Brownstein, J. S. Assessing the online social environment for surveillance of obesity prevalence. *PloS one* 8 (4): e61373, 2013.

Dredze, M. How social media will change public health. *IEEE Intelligent Systems* 27 (4): 81–84, July, 2012.

Eichstaedt, J. C., Schwartz, H. A., Kern, M. L., Park, G., Labarthe, D. R., Merchant, R. M., Jha, S., Agrawal, M., Dziurzynski, L. A., Sap, M., et al. Psychological language on twitter predicts county-level heart disease mortality. *Psychological science* 26 (2): 159–169, 2015.

Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P., and Bhattacharjee, B. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet measurement*, 2007.

Padrez, K. A., Ungar, L., Schwartz, H. A., Smith, R. J., Hill, S., Antanavicius, T., Brown, D. M., Crutchley, P., Asch, D. A., and Merchant, R. M. Linking social media and medical record data: a study of adults presenting to an academic, urban emergency department. *BMJ quality & safety*, 2015.

Ranney, M. L., Chang, B., Freeman, J. R., Norris, B., Silverberg, M., and Choo, E. K. Tweet now, see you in the ed later?: Examining the association between alcohol-related tweets and emergency care visits. *Academic Emergency Medicine*, 2016.

Thiese, M. S., Moffitt, G., Hanowski, R. J., Kales, S. N., Porter, R. J., and Hegmann, K. T. Commercial driver medical examinations: prevalence of obesity, comorbidities, and certification outcomes. *Journal of Occupational and Environmental Medicine*, 2015.

Turner-McGrievy, G. M. and Tate, D. F. Weight loss social support in 140 characters or less: use of an online social network in a remotely delivered weight loss intervention. *Translational Behavioral Medicine* 3 (3): 287–294, 2013.

Yaemsiri, S., Slining, M. M., and Agarwal, S. K. Perceived weight status, overweight diagnosis, and weight control among us adults: the nhanes 2003–2008 study. *International journal of obesity* 35 (8): 1063–1070, 2011.

Transferência de Aprendizado em Contextos Semissupervisionados

Danilo C. G. de Lucena[1], Ricardo B. C. Prudêncio[1]

Universidade Federal de Pernambuco, Brazil
dcgl@cin.ufpe.br rbcp@cin.ufpe.br

Abstract. Classificação é uma das tarefas mais comuns na área de aprendizado de máquina e objetiva a criação de uma função de classificação cujo parâmetro de entrada é uma instância de dados e resulta com a atribuição de um ou mais rótulos para a instância. Uma restrição para a criação dos classificadores, supervisionados ou semissupervisionados, é a necessidade de homogeneidade para a distribuição dos dados utilizados nas fases de treinamento e de teste. Quando este requisito é violado, o problema é considerado como um caso de *dataset shift*. Como resolução para esta problemática, propomos uma correlação com o *transfer learning*, uma área de pesquisa em aprendizado de máquina que objetiva a elaboração de métodos de transferência de conhecimento adquirido na resolução de um ou vários problemas em um domínio fonte para a resolução de um problema em um domínio alvo. Apresentamos um novo método semissupervisionado para o algoritmo *TrAdaBoost*, um algoritmo de *boosting* para *transfer learning*. O método proposto é específico para o contexto de *covariate shift*, um subtópico de *dataset shift*, e utiliza mecanismos de atribuição iterativa de pesos para instâncias até que um classificador ótimo seja obtido. Os resultados preliminares obtidos são comparados com a versão original do algoritmo de *TrAdaBoost* e evidenciam uma melhoria na métrica de precisão dos classificadores usando a técnica proposta.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: transfer learning, aprendizado semissupervisionado, dataset shift, covariate shift

1. INTRODUÇÃO

Classificação é uma das tarefas mais comuns na área de aprendizado de máquina. Na formulação padrão, a tarefa objetiva a criação de uma função de predição $f(x)$ e é composta por três elementos principais: (i) um vetor x de atributos (ou covariantes); (ii) uma classe y dentre um conjunto de rótulos; (iii) uma distribuição de probabilidade conjunta entre covariantes e as classes de rótulos, $P(x, y)$. O classificador é elaborado em duas etapas distintas: treinamento e teste. A função de predição pode ser expressa em termos da distribuição de probabilidade condicional $P(y|x)$. Um requisito importante para a tarefa de classificação é referente à homogeneidade da distribuição dos dados utilizados nas fases de treinamento e de teste. Nos casos em que esse o requisito é violado, o classificador precisa ser reconstruído (e.g. seja um classificador de documentos de textos, no qual um classificador treinado para categorizar documentos com opiniões sobre a qualidade de produtos é aplicado para a classificação de documentos com críticas de filmes).

A restrição de homogeneidade exige a criação de novas formas para lidar com o problema de mudança nos dados. No contexto da tarefa de classificação, introduzimos o problema de *dataset shift* relacionado à composição dos dados em tarefas de aprendizado de máquina. Especificamente, *dataset shift* ocorre quando (i) as distribuições dos dados mudam durante o processo de execução de uma tarefa e por isso (ii) as distribuições de probabilidade conjuntas, $P(x, y)$, são diferentes nas etapas de treino e de

teste do classificador. O *covariate shift* corresponde a um tipo de mudança no qual a distribuição de probabilidade condicional $P(y|x)$ é igual nas etapas de treino e de teste mas ocorre uma mudança na distribuição das covariantes $P(x)$. Essa mudança acontece em cenários nos quais as classes de rótulos são determinadas casualmente pelos valores das covariantes, na forma $X \rightarrow Y$. Na literatura, o *covariate shift* é uma forma comum de *dataset shift* em tarefas de classificação [Moreno-Torres et al. 2012].

Neste trabalho é introduzida a problemática de *covariate shift* aplicada à área de *transfer learning*. *Transfer learning* [Pan and Yang 2010] é um tópico de pesquisa em aprendizado de máquina cuja finalidade é o desenvolvimento de métodos para a transferência de aprendizado adquirido na execução de uma determinada tarefa, em um domínio específico, para uso em uma nova tarefa em um novo domínio. Formalizando, seja T_{source} uma tarefa de aprendizado relacionado a um domínio D_{source}, objetiva-se o uso de conhecimento criado em T_{source} e D_{source} para auxiliar a tarefa de aprendizado T_{target} no domínio D_{target}. A configuração comumente estudada na literatura de *transfer learning* aplicada à tarefa de classificação considera que o domínio fonte, D_{source}, possui um conjunto de instâncias completamente rotuladas (versão supervisionada). Exemplos de trabalhos relacionados são [Dai et al. 2007], [Zhong et al. 2009], [Eaton and desJardins 2011], [Li et al. 2012]. Este trabalho considera os casos nos quais essa configuração é ampliada, permitindo o uso de dados auxiliares não rotulados em D_{source}.

Uma vez que o problema de *dataset shift* em *transfer learning* entre tarefas de classificação é identificado, analisamos o problema da composição dos dados. Em tarefas de classificação, o tipo mais comum de aprendizado é o supervisionado, no qual utiliza-se um conjunto de instâncias completamente rotuladas para a fase de treinamento. Em contrapartida, o aprendizado semissupervisionado usa um conjunto de dados composto entre rotulados e não rotulados. De acordo com [Zhu 2005], uma vantagem do aprendizado semissupervisionado é o uso, em certos casos, de apenas uma pequena proporção de dados rotulados para criar um classificador com bom desempenho utilizando os dados auxiliares não rotulados. Em tarefas de *data mining*, a coleta de uma quantidade suficiente de dados para o treinamento de classificadores pode ser bastante custosa, principalmente em tarefas que lidam com grandes volumes de dados. A proposta apresentada neste trabalho analisa o problema de *transfer learning* com a existência de dados rotulados e não rotulados no domínio fonte com o uso de contexto semissupervisionado. Reconhecemos uma lacuna neste tópico de pesquisa uma vez que poucos trabalhos na literatura propõem a abordagem com o uso de dados auxiliares em problemas de *transfer learning* (exemplos de estudos propostos são [Dai et al. 2007] e [Zhuang et al. 2010]). A Seção 2 apresenta uma análise de métodos semissupervisionados que serão utilizados no algoritmo proposto na Seção 3. A Seção 4 apresenta a avaliação do algoritmo proposto, uma versão semissupervisionada para *transfer boosting*. Na Seção 4 também serão discutidos os conjuntos de dados utilizados para a execução dos algoritmos, a metodologia de avaliação e a análise dos resultados obtidos. A Seção 5 conclui o artigo com uma breve consideração final.

2. APRENDIZADO SEMISSUPERVISIONADO

Um grupo de técnicas propostas na literatura de aprendizado semissupervisionado baseiam-se na *propagação de rótulos* que, de forma geral, aplicam métricas de similaridades entre os atributos das instâncias para criar agrupamentos de instâncias próximas. Apresentamos dois métodos da área de aprendizado semissupervisionado que implementam a propagação de rótulos: *label propagation* [Zhu and Ghahramani 2002] e *label spreading* [Zhou et al. 2004]. Os dois algoritmos executam propagação de rótulos de instâncias rotuladas para as instâncias não rotuladas, iterativamente, até que todo o conjunto de instâncias esteja em *estado de convergência*, quando todas as instâncias são rotuladas. Inicialmente, os dois métodos transformam o conjunto de instâncias em uma estrutura de grafo $G(V, E)$: (i) as instâncias do conjunto D são transformadas em nós $v \in V$ e (ii) as arestas $e \in E$ entre nos nós recebem valores de peso definidos em função da similaridade entre as instâncias (um exemplo

de cálculo de similaridade é a distância Euclideana). Ademais, os dois métodos são modulares e não-paraméricos.

Algorithm 1 Algoritmo Label Propagation.

1: **Entrada:** $D = \{D_{\{l\}} \cup D_{\{u\}}\}$
2: **Etapa I**: Inicializar o vetor de rótulos.
3: $Y^{*(0)} \leftarrow (y_1, \ldots, y_l, 0, \ldots, 0)$.
4: **Etapa II**: Definição das matrizes.
5: Calcular matriz de similaridade M e identidade I de D.
6: **Etapa III**: Processo iterativo de propagação de rótulos.
7: **repeat**
8: $\quad Y^{*(t+1)} \leftarrow I^{-1} M Y^{*(t)}$.
9: $\quad Y_l^{*(t+1)} \leftarrow Y_l$.
10: **until** estado de convergência em $Y^{*(\infty)}$.
11: **Retornar:** o vetor de rótulos Y^* para todas as instâncias de D.

Algorithm 2 Algoritmo Label Spreading.

1: **Entrada:** $D = \{D_{\{l\}} \cup D_{\{u\}}\}$
2: **Etapa I**: Inicializar o vetor de rótulos.
3: $Y^{*(0)} \leftarrow (y_1, \ldots, y_l, 0, \ldots, 0)$.
4: **Etapa II**: Definição das matrizes e variáveis.
5: Calcular matriz de similaridade M e identidade I de D.
6: Definir variável $\alpha = 1$ (o valor padrão).
7: $L \leftarrow I^{-1/2} M I^{-1/2}$.
8: **Etapa III**: Processo iterativo de propagação de rótulos.
9: **repeat**
10: $\quad Y^{*(t+1)} \leftarrow \alpha L Y^{*(t)} + (1 - \alpha) Y^{*(0)}$.
11: **until** estado de convergência em $Y^{*(\infty)}$
12: **Retornar:** o vetor de rótulos Y^* para todas as instâncias de D.

O Algoritmo 1 apresenta o pseudocódigo para o método *label propagation*. A entrada do algoritmo é conjunto $D = \{D_{\{l\}} \cup D_{\{u\}}\}$, composto por instâncias rotuladas $D_{\{l\}}$ e não rotuladas $D_{\{u\}}$. O objetivo é a obtenção do vetor final de rótulos Y^* para o conjunto de instâncias. Convenciona-se que os rótulos para as instâncias são convertidos para uma representação binária. Na *Etapa I* é criado um vetor Y^* para armazenar os rótulos atribuídos para as instâncias em cada iteração. O mecanismo de propagação de rótulos possui um fator de *clamping* (definido na *linha (9)*), para que os rótulos das instâncias de $D_{\{l\}}$ não sejam alterados. A matriz de pesos de similaridade M é definida pela função de *kernel gaussiana* na Equação 1 e aplicada para as instâncias de D. Cada elemento M_{ij} da matriz de similaridade determina o grau de proximidade entre os nós i e j. Os elementos da matriz diagonal I são definidos na forma $I_{ii} \leftarrow \sum_j M_{ij}$. Ao fim do processo iterativo os valores em $Y^{*(\infty)}$ são os rótulos finais para todas as instâncias em D.

$$M_{ij} = e^{-\frac{||x_i - x_j||^2}{2\delta^2}} \tag{1}$$

O Algoritmo 2 apresenta o pseudocódigo para o método de *label spreading*. O método é similar ao do *label propagation* uma vez que utiliza processos iterativos para propagar rótulos das instâncias rotuladas para as instâncias vizinhas não rotuladas até que o estado de convergência seja alcançado. A

matriz de pesos utiliza a Equação 1 mas restringe os valores de $M_{ii} = 0$. O processo de propagação de rótulos utiliza uma matriz em formato de grafo *Laplaciano*, definido por $L \leftarrow I^{-1/2} M I^{-1/2}$. O valor da variável auxiliar $\alpha \in [0, 1)$ serve como parâmetro de controle nas iterações. Oposto ao método de *label propagation*, o *label spreading* não usa o fator de *clamping* e as instâncias de $D_{\{l\}}$, inicialmente rotuladas, podem receber novos rótulos durante as iterações. Ao fim da execução, o vetor $Y^{*(\infty)}$, em estado de convergência, possui os rótulos finais para todas as instâncias em D.

Os dois métodos *label propagation* e *label spreading* implementam um mecanismo de funcionamento equivalente: utilizam matrizes de similaridades para representar as instâncias; são algoritmos não-paramétricos (o valor da variável α no algoritmo de *label spreading* é fixo e determinado na implementação do algoritmo); utilizam vetores de rótulos Y^* para as instâncias finais do conjunto D. A complexidade das duas abordagens é estimada, em pior caso, com $O(km^2)$. A principal diferença entre os dois algoritmos é o mecanismo de rotulação: na versão utilizada no *label propagation*, a propagação de rótulos utiliza a matriz de similaridade M combinada com a matriz diagonal I; em contrapartida, o algoritmo *label spreading* define um grafo normalizado *Laplaciano* criado com as matrizes M e I. Segundo [Bengio et al. 2006], a abordagem *Laplaciana* torna a implementação do *label spreading* mais tolerante ao ruído no conjunto de instâncias do que a implementação padrão do *label propagation*.

3. TRANSFER BOOSTING SEMISSUPERVISIONADO

Nesta seção apresentamos um novo algoritmo semissupervisionado, o $SemiSupervised_{TAB}$, como uma expansão do algoritmo *TrAdaBoost*. O *TrAdaBoost* [Dai et al. 2007] é um método de *transfer learning* que utiliza como base o *AdaBoost* [Freund and Schapire 1997], um algoritmo de aprendizado utilizado para a melhoria de um classificador "fraco" aplicando um mecanismo de ajuste de pesos das instâncias de treino para maximizar a contribuição das melhores instâncias durante a fase de treinamento. O algoritmo proposto nesta seção executa *transfer learning* no contexto de *dataset shift* e permite o uso de um conjunto de dados heterogêneos no domínio fonte com instâncias rotuladas e não rotuladas. O Algoritmo 3 apresenta o pseudocódigo para o $SemiSupervised_{TAB}$ em duas versões, uma primeira versão com o método de *label propagation* e uma segunda versão com o método de *label spreading*. No algoritmo, o domínio fonte é a união de dois subconjuntos, $D_{source} = \{D_{source(l)} \cup D_{source(u)}\}$, onde $D_{source(l)}$ são as instâncias rotuladas e $D_{source(u)}$ as instâncias não rotuladas. O conjunto de instâncias do domínio alvo é definido por D_{target}, sem alterações. Utiliza-se um algoritmo de aprendizado de base na variável *algbase* na etapa de *transfer learning* seguindo a implementação original do *TrAdaboost*. As variáveis auxiliares k e *qtdIter* são usadas para busca de instâncias k-vizinhas e para definir a quantidade de iterações, respectivamente.

A *Etapa I* define a inicialização do algoritmo. Na *Etapa II* é determinado o mecanismo de aprendizado semissupervisionado. A (*linha* 5) define o vetor de pesos para as instâncias $\mathbf{w} = (w_1, \ldots, w_{n+m})$ que serão calculados no decorrer da execução do algoritmo. Em seguida, inicia-se o processo semisupervisionado para as instâncias do domínio D_{source} (*linhas* $6 - 8$). Os métodos de aprendizado semissupervisionado *label propagation* ou *label spreading* retornam um vetor Y^* com os rótulos definidos para todos as instâncias de D_{source}. A variável auxiliar D_{aux} coleta do vetor Y^* apenas os novos rótulos atribuídos para as instâncias em $D_{source(u)}$ (*linha* 8).

A parte iterativa do algoritmo (*linhas* $9 - 22$) utiliza o resultado do processo de semi-supervisão. O conjunto geral de treino $D_{geral} = \{D_{source(l)} \cup D_{target}\}$ é definido pelo subconjunto de instâncias rotuladas do domínio fonte e o conjunto de instâncias do domínio alvo. Na *linha* 12 é definido o método de aprendizado de base e que utiliza o cojunto geral de instâncias D_{geral} com os pesos da iteração p^{iter}. A *linha* 13 especifica a função de hipótese $h_{iter} : X \rightarrow Y$ criada pelo método de aprendizado $algBase$. A *Etapa III* atribui ao conjunto ϕ as instâncias de $D_{source(l)}$ que possuem os maiores valores de peso de acordo com $\mathbf{w}$ (*linha* 15). Por padrão, a quantidade de instâncias coletadas é igual a 10% do número de instâncias em $D_{source(l)}$. O processo é definido com as etapas: (i) para cada instância de ϕ é realizada uma busca da k-instâncias mais próximas de acordo com a matriz de similaridade M

(definida pelo algoritmo semissupervisionado escolhido em $algSSL$); (ii) as k-instâncias selecionadas são adicionadas ao conjunto $D_{source(l)}$ (*linha* 16); (iii) adicionar k novas entradas no vetor de pesos $\mathbf{w}$ com o valor de peso w_ϕ (*linha* 17).

Algorithm 3 Pseudocódigo para o $SemiSupervised_{TAB}$.

1: **Entrada:** D_{source}, D_{target}, qtdeIter, algSSL, algBase, k.
2: ***Etapa I:*** *Inicialização*
3: Definir: $n = |D_{source(l)}|$, $m = |\, D_{target}\,|$.
4: ***Etapa II:*** *Aprendizado semissupervisionado.*
5: Criar o vetor de pesos $\mathbf{w} = (w_1^1, \ldots, w_{n+m}^1)$.
6: Propagação de rótulos em D_{source}.
7: Chamada de função: $algSSL(D_{source})$.
8: $D_{aux} \leftarrow ObterRotulos(D_{source(u)})$.
9: **for** $iter = 1$ to qtdeIter **do**
10: $D_{geral} = \{D_{source(l)} \cup D_{target}\}$.
11: Pesos da iteração: $p^{iter} = w^{iter}/(\sum_{i=1}^{n+m} w_i^{iter})$.
12: Utilizar $algBase$ com o conjunto D_{geral} e pesos p^{iter}.
13: Gerar a hipótese $h_{iter} : X \rightarrow Y$.
14: ***Etapa III:*** *Uso de instâncias auxiliares.*
15: Atribuir à ϕ as melhores instâncias de $D_{source(l)}$.
16: Usando ϕ, coletar as k instâncias mais próximas em D_{aux} e adicionar em $D_{source(l)}$.
17: Adicionar novas entradas em $\mathbf{w}$.
18: Atualizar $n = |D_{source(l)}|$.
19: ***Etapa IV:*** *Transfer learning.*
20: Calcular valores de erro ϵ e valores auxiliares β.
21: Atualizar $\mathbf{w}$ para a próxima iteração.
22: **end for**
23: **Retornar:** função h obtida na última iteração.

A *Etapa IV* aplica o processo de *transfer learning* de forma similar à execução do algoritmo *TrAdaBoost*. A *linha* 20 calcula a taxa de erro ϵ, definida na Equação 2, e permite mensurar a qualidade de classificação do conjunto de treino geral. A taxa de erro é calculada apenas para as instâncias do domínio alvo (instâncias de índices $n+1$ até $n+m$). Duas convenções são utilizadas nesse ponto: a função $c(x)$ retorna o rótulo real da instância x; o termo $||h_{iter}(x) - c(x)||$ retorna 1 se a predição de h estiver correta e 0 se incorreta.

$$\epsilon = \sum_{i=n+1}^{n+m} \frac{w_i^{iter}\, ||h_{iter}(x_i) - c(x_i)||}{\sum_{i=n+1}^{n+m} w_i^{iter}} \tag{2}$$

O cálculo das variáveis auxiliares β (*linha* 20), executado para todas as instâncias dos domínios, é apresentado na Equação 3 e segue as convenções especificadas em [Dai et al. 2007]. As variáveis β servem como um parâmetros de amortização das taxa de erros e são utilizadas no processo de atualização do vetor de pesos $\mathbf{w}$. O valor de β é relacionado com a complexidade de pior caso para o processo de transferência, $O(\sqrt{ln\ (n/qtdeIter)})$.

$$\beta_{iter} = \frac{\epsilon}{1-\epsilon}, \qquad \beta = \frac{1}{1 + \sqrt{2\ ln(n/qtdeIter)}} \tag{3}$$

Tabela I: Datasets dos experimentos

Dataset	Categorias	Domínio	Instâncias	Atributos	Tipo	Rótulos
20 newsgroup	comp	Texto	4931	150	Textual	5
	rec		3988	150		5
	sci		3984	150		5
Emotions	positivo	Música	350	70	Numérico	6
	negativo		142	70		6
Wine	red	Biologia	4898	12	Numérico	5
	white		1599	12		5
Scene	urbano	Imagem	1200	300	Numérico	6
	natureza		1207	300		6

Após a definição dos valores das taxas de erro ϵ e das variáveis auxiliares β, o vetor de pesos $\mathbf{w}$ é atualizado na *linha* 21 segundo a Equação 4. Intuitivamente, as instâncias do domínio fonte que contribuem positivamente para a função de classificação recebem valores de pesos maiores. No decorrer das iterações, estima-se que estas instâncias sejam as mais adequadas para o processo de transferência de instâncias do domínio fonte para o alvo. Após o processo iterativo, a *linha* 23 retorna a função h do processo de *boosting*. Na execução realizada para o algoritmo $SemiSupervised_{TAB}$, utilizam-se os valores padrões $k = 10$ e $qtdIter = 100$.

$$w_i^{t+1} = \begin{cases} w_i^{iter}\ \beta^{||h_{iter}(x_i)-c(x_i)||}, & 1 \leq i \leq n. \\ w_i^{iter}\ \beta_{iter}^{-||h_{iter}(x_i)-c(x_i)||}, & n+1 \leq i \leq n+m. \end{cases} \tag{4}$$

4. EXPERIMENTOS

Os algoritmos analisados serão avaliados com *datasets* utilizados na literatura da área de algoritmos de mineração de dados. Para simular cenários com dados configurados para uso em classificação semissupervisionada, todos os *datasets* foram pré-processados e divididos em conjuntos fonte e alvo. Adicionalmente, o conjunto fonte é dividido em instâncias rotuladas e não rotuladas. A Tabela I apresenta a descrição dos *datasets*.

20 Newsgroup dataset[1] é uma coleção de documentos de vinte tipos grupos de discussão. O conjunto de dados original é dividido em três categorias, *comp, rec, sci*. Os documentos de texto foram convertidos para o formato tipo *tf–idf* (*term frequency–inverse document frequency*). *Wine dataset*[2] reune um conjunto de observações sobre a produção de vinho tinto e branco. *Emotions dataset*[3] reune um conjunto de observações de estilos musicais de ouvintes e é categorizado pelo tipo de reação emocional associado. O *dataset* foi dividido em duas categorias principais: (i) sentimentos positivos (e.g. *amazed-suprised, happy- pleased, relaxing-clam*) e (ii) sentimentos negativos (e.g. *quiet-still, sad-lonely, angry-aggresive*). *Scene dataset*[4] reune um conjunto de observações descritivas de imagens em duas categorias principais: imagens urbanas e imagens da natureza.

4.1 Metodologia de avaliação

Para avaliar os algoritmos desenvolvidos consideramos dois tipos de cenários: (i) uso do método tradicional de *transfer learning TrAdaBoost*, utilizando uma porcentagem das instâncias do domínio fonte; (ii) uso do método proposto de *transfer learning* $SemiSupervised_{TAB}$ com as duas versões semissupervisionadas. Essa divisão permite comparar a melhoria do processo de transferência quando

[1]Dataset disponível em: `https://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups`.
[2]Dataset disponível em: `http://archive.ics.uci.edu/ml/datasets/Wine`.
[3]Dataset disponível em: `https://sourceforge.net/projects/meka/files/Datasets/Music.arff/download`.
[4]Dataset disponível em: `https://sourceforge.net/projects/meka/files/Datasets/Scene.arff/download`.

o domínio fonte muda de uma configuração supervisionada para semissupervisionada. A descrição quantitativa dos cenários é apresentada a seguir.

Tipo I - Com transfer learning; sem abordagens semissupervisionadas: nessa avaliação utiliza-se *transfer learning* para uma porção das instâncias rotuladas do domínio fonte D_{source} ignorando instâncias auxiliares não rotuladas. O algoritmo utilizado é o *TrAdaBoost*. A divisão dos conjuntos segue a regra: D_{source} utilizando 60% das instâncias (rotuladas), D_{target} utilizando 90% das instâncias (rotuladas). O teste do classificador final utiliza 10% das instâncias restantes de D_{target}.

Tipo II - Com transfer learning e com abordagens semissupervisionadas: nessa avaliação utiliza-se o algoritmo proposto $SemiSupervised_{TAB}$ com versões utilizando *label propagation* e *label spreading*. A divisão dos conjuntos segue a regra: D_{source} utilizando 100% das instâncias (com a divisão de 20% de instâncias rotuladas e 80% de instâncias não rotuladas), D_{target} utilizando 90% das instâncias (rotuladas). O teste do classificador final utiliza 10% das instâncias restantes de D_{target}.

Os algoritmos de *transfer learning* apresentados utilizam algoritmos de aprendizado de base: *j48*, *Naive Bayes* e *SVM*. As implementações dos métodos e a execução das métricas de avaliação foram executadas no *WEKA*[5], uma plataforma com uma coleção de algoritmos de aprendizado para tarefas de mineração de dados. As métricas utilizadas são a *precisão* e F_1-*score*.

5. ANÁLISE DOS RESULTADOS E CONCLUSÃO

As Tabelas II e III apresentam os resultados para os algoritmos com os *datasets* descritos anteriormente. Os resultados em negrito indicam os melhores valores nas execuções. Em cada tabela, a coluna *Dataset* identifica os domínios fonte e alvo no formato $D_{source} \rightarrow D_{target}$. A coluna *Classificador* indica o classificador base que será utilizado na variável *algBase* nos algoritmos *TrAdaBoost* e $SemiSupervised_{TAB}$. As colunas seguintes apresentam os tipos de cenários de avaliação (para o *Tipo II*, *LP* indica *label propagation* e *LS* indica *label spreading*).

De acordo com as tabelas de resultados, evidencia-se que a execução do algoritmo proposto, o $SemiSupervised_{TAB}$, apresenta um desempenho melhor do que a versão original do *TrAdaBoost* para todos os algoritmos de aprendizado de base nos dois tipos de cenários de avaliação. Concluindo a execução das avaliações, elencamos os seguintes pontos como considerações finais e trabalhos futuros. Primeiro, no algoritmo proposto o sistema de cálculo do valor de peso das instâncias segue a formulação original do *TrAdaBoost*. Um tópico para pesquisa futura é a análise de um novo sistema para definir valores de peso que possam ser mais eficientes no ponderamento das instâncias. Segundo, uma característica a ser estudada é a possibilidade de uso de técnicas de aprendizado que considerem subconjuntos de atributos de instâncias para a criação de classificadores adaptados ao *transfer learning*. Terceiro, para reforçar e ampliar os resultados apresentados, serão adicionados novos *datasets* para avaliação dos algoritmos propostos em estudos futuros. Também será investigado o uso de *datasets* artificias para simular condições específicas de cenários semissupervisionados tais como a variação na proporção entre instâncias rotuladas e não-rotuladas.

REFERENCES

Bengio, Y., Delalleau, O., and Le Roux, N. Label propagation and quadratic criterion. *Semi-supervised learning* vol. 10, 2006.

Dai, W., Yang, Q., Xue, G.-R., and Yu, Y. Boosting for transfer learning. In *Proceedings of the 24th international conference on Machine learning*. ACM, pp. 193–200, 2007.

Eaton, E. and desJardins, M. Selective transfer between learning tasks using task-based boosting. In *AAAI*. Citeseer, 2011.

Freund, Y. and Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences* 55 (1): 119–139, 1997.

[5]Disponível em: `http://www.cs.waikato.ac.nz/ml/weka/index.html`.

Tabela II: Resultados dos experimentos para o dataset de texto *20 Newsgroup*.

Dataset	Classificador	Tipo I		Tipo II - LP		Tipo II - LS	
		Precisão	F_1	Precisão	F_1	Precisão	F_1
Comp → Rec	J48	0,609	0,595	0,732	0,602	**0,763**	0,692
	Naive Bayes	0,623	0,611	**0,745**	0,638	0,711	0,663
	SVM	0,657	0,51	0,755	0,371	**0,781**	0,712
Comp → Sci	J48	0,658	0,532	**0,778**	0,501	0,746	0,532
	Naive Bayes	0,624	0,535	**0,758**	0,556	0,706	0,51
	SVM	0,82	0,261	**0,935**	0,195	0,807	0,318
Sci → Rec	J48	0,516	0,451	**0,847**	0,589	0,719	0,687
	Naive Bayes	0,493	0,441	0,624	0,549	**0,719**	0,646
	SVM	0,384	0,253	0,448	0,352	**0,651**	0,641
Sci → Comp	J48	0,523	0,412	**0,807**	0,519	0,756	0,627
	Naive Bayes	0,419	0,422	0,712	0,590	**0,781**	0,598
	SVM	0,424	0,216	0,678	0,301	**0,681**	0,601
Rec → Sci	J48	0,582	0,402	**0,754**	0,489	0,698	0,612
	Naive Bayes	0,443	0,412	0,667	0,518	**0,712**	0,632
	SVM	0,421	0,345	0,476	0,321	**0,623**	0,6
Rec → Comp	J48	0,554	0,445	**0,8**	0,511	0,757	0,613
	Naive Bayes	0,478	0,434	0,645	0,534	**0,741**	0,587
	SVM	0,41	0,246	0,621	0,321	**0,656**	0,594

Tabela III: Resultados dos experimentos para os datasets numéricos *Wine*, *Emotions* e *Scene*.

Wine dataset	Classificador	Tipo I		Tipo II - LP		Tipo II - LS	
		Precisão	F_1	Precisão	F_1	Precisão	F_1
White → Red	J48	0,659	0,663	0,678	0,413	**0,729**	0,666
	Naive Bayes	0,623	0,451	0,7	0,414	**0,726**	0,367
	SVM	0,563	0,574	0,545	0,38	**0,562**	0,541
Red → White	J48	0,631	0,6	0,642	0,445	**0,734**	0,601
	Naive Bayes	0,645	0,431	0,745	0,325	**0,787**	0,345
	SVM	0,512	0,567	**0,587**	0,524	0,523	0,586

Emotions dataset	Classificador	Tipo I		Tipo II - LP		Tipo II - LS	
		Precisão	F_1	Precisão	F_1	Precisão	F_1
Positivo → Negativo	J48	0,567	0,612	0,623	0,464	**0,712**	0,562
	Naive Bayes	0,582	0,475	0,742	0,323	**0,789**	0,34
	SVM	0,545	0,585	0,534	0,574	**0,552**	0,583
Negativo → Positivo	J48	0,522	0,663	0,675	0,473	**0,723**	0,645
	Naive Bayes	0,587	0,451	**0,763**	0,346	0,646	0,374
	SVM	0,687	0,574	0,5	0,574	**0,534**	0,534

Scene dataset	Classificador	Tipo I		Tipo II - LP		Tipo II - LS	
		Precisão	F_1	Precisão	F_1	Precisão	F_1
Urbano → Natureza	J48	0,7	0,656	0,613	0,41	**0,742**	0,623
	Naive Bayes	0,645	0,412	0,65	0,373	**0,784**	0,3
	SVM	0,566	0,562	**0,589**	0,574	0,523	0,562
Natureza → Urbano	J48	0,615	0,664	0,631	0,41	**0,735**	0,623
	Naive Bayes	0,637	0,472	**0,678**	0,323	0,67	0,41
	SVM	0,591	0,5	0,513	0,543	**0,525**	0,583

Li, L., Jin, X., and Long, M. Topic correlation analysis for cross-domain text classification. In *AAAI*, 2012.

Moreno-Torres, J. G., Raeder, T., Alaiz-RodríGuez, R., Chawla, N. V., and Herrera, F. A unifying view on dataset shift in classification. *Pattern Recognition* 45 (1): 521–530, 2012.

Pan, S. J. and Yang, Q. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on* 22 (10): 1345–1359, 2010.

Zhong, E., Fan, W., Peng, J., Zhang, K., Ren, J., Turaga, D., and Verscheure, O. Cross domain distribution adaptation via kernel mapping. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 1027–1036, 2009.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. Learning with local and global consistency. *Advances in neural information processing systems* 16 (16): 321–328, 2004.

Zhu, X. Semi-supervised learning literature survey, 2005.

Zhu, X. and Ghahramani, Z. Learning from labeled and unlabeled data with label propagation. Tech. rep., Citeseer, 2002.

Zhuang, F., Luo, P., Shen, Z., He, Q., Xiong, Y., Shi, Z., and Xiong, H. Collaborative dual-plsa: mining distinction and commonality across multiple domains for text classification. In *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, pp. 359–368, 2010.

Identificação metalográfica dos aços através de descritores de textura e ELM

Victoria Mera-Moya[1], Francisco D. S. Lima[1], Iális C. de Paula Júnior[1], Jorge I. Fajardo[2], Jarbas J. M. Sá Júnior[1]

[1] Universidade Federal de Ceará, Campus de Sobral, Brazil
vicky_memo@hotmail.com
danlima@alu.ufc.br
ialis@sobral.ufc.br
jarbas_joaci@yahoo.com.br
[2] Universidad Politécnica Salesiana, Ecuador
jfajardo@ups.edu.ec

Resumo. Este trabalho aborda um processo de identificação da microestrutura de um aço com base em uma microfotografia que pode assumir tanto uma aplicação acadêmica quanto industrial. Atualmente na disciplina de metalografia as imagens microscópicas são analisadas visualmente e sua identificação é feita com ajuda de um conjunto de imagens de referência (padrões) contidas em um atlas. Na indústria, se qualquer dos componentes de uma máquina tem uma falha, torna-se inviável realizar uma análise metalográfica pelo método tradicional na parte do componente afetado. Para solucionar ambos os problemas, propõe-se estabelecer um método automático e eficaz que possibilite aos estudantes de engenharia comprovar os efeitos de um tratamento aplicado sobre um material e que substitua os métodos tradicionais de análises; e na indústria, contribuir na inspeção *in situ* dos componentes de uma máquina ou estrutura sem alterar seu estado. Este estudo foi realizado com os cinco tipos mais comuns de aço para construção de máquinas e ferramentas: AISI 4340, AISI O1, AISI D6, AISI O1 (retificado) e AISI 1018. As imagens adotadas foram subdivididas e assumiu-se as técnicas de filtros de textura de Laws e extração de características de Haralick para formar os atributos de classificação.Os classificadores utilizados foram KNN, MLP e ELM. Os melhores resultados foram obtidos com ELM com uma taxa de acerto superior a 90%, o que evidencia a efetividade do procedimento proposto.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning; I.4.7 [**Image Processing and Computer Vision**]: Texture; I.5.2 [**Pattern Recognition**]: Design Methodology

Keywords: metalografia, aços, texture, machine learning

1. INTRODUÇÃO

O aço é um dos materiais mais utilizados na fabricação e construção de componentes estruturais em geral, devido à sua versatilidade e adaptabilidade. É basicamente uma liga ou combinação de ferro e carbono. Em alguns casos, outros elementos de ligação específicos, tais como crômio ou níquel, são adicionados para obter características próprias [Chávez 2008].

A metalografia é uma técnica amplamente utilizada e que permite conhecer a microestrutura dos metais. Esse processo deve ser realizado de acordo com a norma ASTM E3 [ASTM E3-01 2007]. O método de preparação e a qualidade das amostras obtidas desempenham um papel importante no processo metalográfico. Isso pode ser garantido com base na experiência das pessoas especializadas no campo metalográfico, como é mostrado em [Gavarito 2011]. O componente principal dos aços é o carbono que, dependendo da porcentagem de concentração (com percentual no intervalo de 0,01% a

2%) e as misturas presentes, determinarão as propriedades físicas, químicas e mecânicas do material. Aços com a concentração de carbono inferior a 0,3% são conhecidos como aços de baixo carbono. Esses aços são dúcteis, soldáveis, não se pode fazer tratamentos térmicos, possuem resistência mecânica moderada e são maquináveis. Os aços com porcentagem de concentração de carbono maior que 0,3% e menor que 0,65% têm boas propriedades de têmpera e revenido, boa resistência mecânica e moderada ductilidade. Por outro lado, todos os aços com concentração de carbono acima de 0,65% são duros e resistente ao desgaste, além de serem difíceis de soldar e pouco tenazes [Kalpakjian and Schmid 2002].

Os trabalhos realizados no campo metalográfico com apoio do processamento digital de imagens estão direcionados para avaliar as falhas por oxidação ou fissuras presentes nos aços e para analisar a qualidade da soldadura. Em [Ulyanov et al. 2013] foi realizado um estudo das fases metalográficas de ligas superficiais obtidas por técnicas de laser através da segmentação de imagens. O foco desse estudo está em detectar *in situ* as zonas afetadas por calor na microestrutura superficial. Já em [Medeiros et al. 2010] foi desenvolvida uma avaliação de características de textura e cor para a detecção de corrosão mediante um método não destrutivo. Nesse estudo o detector automático de corrosão foi realizado através do processamento de imagens digitais de aços com carbono pertencentes a uma refinaria de petróleo. As publicações citadas contêm análises metalográficas, mas o objetivo não é identificar quais são os tipos de aços. No entanto, em [de la Cruz et al. 1994] são determinados os tipos e as ligações dos componentes estruturais dos aços utilizando técnicas heurísticas. O referido estudo foi realizado para acompanhar estudantes no desenvolvimento da disciplina de metalografia. Nessa técnica é necessário que o usuário passe uma série de informações sobre as características micrográficas, em seguida, utilizando análise *fuzzy*, o algoritmo identifica o tipo de mistura dos aços e ferros. Ambas as técnicas alcançam bons resultados mas não permitem atingir a análise metalográfica assumida para este trabalho.

A metalografia tradicional usa como principal instrumento o microscópio óptico de luz refletida. Esse instrumento não requer características especiais, tornando-se uma ferramenta de fácil manipulação e bem acessível. Na classificação proposta foram utilizadas imagens com estas características. O objetivo deste artigo é identificar os tipos de aço para criar uma técnica que automatize a análise metalográfica tradicional realizada na disciplina de metalografia e constitua a base para o desenvolvimento de um módulo de verificação robusto para a indústria do aço.

Nas próximas seções explana-se o processo de obtenção de características mediante técnicas de textura e a metodologia para a classificação dos aços. Finalmente, nas Seções 4 e 5, têm a descrição da taxa de acerto alcançada nos resultados experimentais, além das conclusões e discussões alcançadas.

2. MATERIAIS E MÉTODOS

Esta seção envolve técnicas e características relevantes para o desenvolvimento da análise metalográfica proposta. A base de imagens adotada e as características a serem obtidas a partir destas são discutidas nas subseções que seguem.

2.1 Aquisição de imagens

As imagens utilizadas neste trabalho foram fornecidas pelos pesquisadores GiMaT *(Grupo de Investigación en Nuevos Materiales y Procesos de Transformación)* da Universidade Politécnica Salesiana de Cuenca-Equador. A metodologia utilizada para a obtenção de microfotografias é um processo que requer muito cuidado na sua preparação. O objetivo final desta etapa é a obtenção de uma superfície plana e lisa, semelhante a um espelho. Essas amostras são submetidas a um ataque químico com Nital 2 (solução de álcool e ácido nítrico a 2%), que revela a microestrutura do aço, tornando visível a presença de perlita na ligação. O microscópio utilizado foi de luz refletida e fornecido por Olympus com resolução de [200 ppi 600 ppi] para uso acadêmico. A única consideração especial é que o feixe de luz deve ser horizontal e não diretamente para a amostra. O ponto de partida desta análise são

imagens físicas que foram parte de um atlas padrão usado para a diferenciação visual dos componentes dos aços. Para este trabalho, foi realizada a digitalização das imagens em questão. Inicialmente, assumiu-se três imagens de cada tipo de aço (aproximadamente 900x700 pixels de dimensão), com as seguintes características: uma primeira imagem com uma resolução de 200ppi; uma segunda imagem contendo o centro da amostra com 600 ppi de resolução; uma terceira imagem contendo a periferia da amostra a 600 ppi. Com o objetivo de aumentar a quantidade de amostras e diminuir o custo computacional, cada uma das imagens foi dividida em 35 sub-imagens de 100x100, conforme mostra a Figura 1. Após esse procedimento, foram obtidas 105 imagens por classe, formando una base total de 525 imagens.

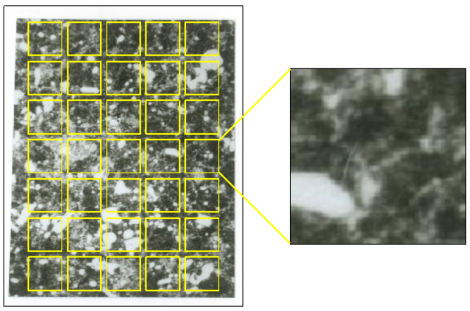


Fig. 1: Etapa inicial para obtenção das imagens de textura.

2.2 Análise de Textura

A textura em geral, é o conjunto de padrões existentes numa superfície. No caso metalográfico, o estudo destas superfícies fornece informações valiosas, tais como componentes e tratamentos aos quais o material tenha sido submetido. Robert Haralick *et al.* [Haralick et al. 1973] indicam que a textura é uma característica importante usada na identificação de objetos ou regiões de interesse em imagens microscópicas, assim como fotografias do espaço ou satélites. De acordo com [Bharati et al. 2004], existem vários métodos de extração de texturas, tais como: estatísticos, geométricos, abordagens baseadas em modelos e aquelas baseadas em uma transformação matemática. Os métodos estatísticos associam-se aos valores de intensidade da imagem no nível espacial, computando características locais. Combinar técnicas de extração de característica, em alguns casos, alcançam melhores resultados do que usando um método simples realçando características específicas do problema abordado [Bastidas-Rodriguez et al. 2016].

2.3 Filtros de Laws

O método de Laws [Laws 1980], também conhecido como energia da textura, mede a quantidade de variações sem um tamanho de janela fixa e consiste na convolução da imagem com vários filtros. Os resultados com os filtros de Laws e a matriz de co-ocorrência superam as porcentagens de classificação e diminuem o custo computacional da operação [Shapiro and Stockman 2001].

A finalidade desses filtros é destacar uma característica específica da textura. Como exemplo, uma convolução da imagem será destinada para características de bordas, outra para ondulação, rugosidade, etc. No final são obtidas nove máscaras.

As máscaras de filtro são representadas pelos vetores: $L_5(\text{nível}) = [1, 4, 6, 4, 1]$, $E_5(\text{borda}) = [-1, -2, 0, 2, 1]$, $S_5(\text{pontos}) = [-1, 0, 2, 0, -1]$, $R_5(\text{onda}) = [1, -4, 6, -4, 1]$. As máscaras de convolução $2D$ desses filtros são obtidas através da combinação dos pares de vetores, conforme mostra a Tabela I.

N.	Combinação	N.	Combinação
1	L_5E_5/E_5L_5	6	S_5R_5/R_5S_5
2	L_5R_5/R_5L_5	7	S_5S_5
3	L_5S_5/S_5L_5	8	E_5E_5
4	E_5S_5/E_5S_5	9	R_5R_5
5	E_5R_5/R_5E_5		

Table I: Combinações dos filtros de Laws.

As nove sub-imagens filtradas contêm informações específicas, de modo que este método pode ser combinado com técnicas estatísticas baseadas na matriz de co-ocorrência.

2.4 Matriz de co-ocorrência

A matriz de co-ocorrência também conhecida como GLCM (*Grey Level Co-occurrence Matrices*) é uma matriz quadrada que é composta pelo número de combinações de níveis de cinza presentes em uma imagem num determinado sentido. A distribuição dos valores da matriz depende da relação entre o ângulo (direção) e a distância dos pixels. As opções de graus de direção são as que seguem: 0°, 45°, 90°, 135°. Esta matriz aproxima a probabilidade de distribuição conjunta de um par de pixels e pode ser simétrica ou assimétrica. Uma matriz simétrica significa que os mesmos valores ocorrem na parte oposta à diagonal. Como exemplo, o valor da célula (1,2) deve ser a mesma da célula (2,1) [Pathak and Barooah 2013; Sebastian et al. 2012]. Para expressar a matriz em termos de probabilidade, o número de vezes que o evento ocorre é dividido com o número total de eventos possíveis, de acordo com a probabilidade definida que segue:

$$P_{(i,j)} = \frac{V_{(i,j)}}{\sum_{i,j}^{N-1} V_{(i,j)}}, \tag{1}$$

em que i é o número de linhas e j é o número de colunas; V corresponde ao valor da célula na matriz; $P_{(i,j)}$ é a probabilidade na célula (i, j) e N corresponde ao número de linhas ou colunas.

Na matriz resultante, os elementos da diagonal principal representam os pares de pixels que não possuem diferenças nos níveis de cinza, de modo que quanto mais alto os valores da diagonal principal então mais homogênea é a imagem. Se uma soma algébrica dos valores de cada célula da matriz de co-ocorrência é realizada, obtém-se como resposta a unidade. As matrizes de co-ocorrência contêm informações muito importantes sobre a textura da imagem, de forma que os padrões/características de textura podem ser calculados a partir dessa matriz. Para tanto, essas devem ser aplicadas nas equações descritivas, como as sugeridas por Haralick [Haralick et al. 1973]. Para o presente estudo, foram utilizadas quatro características estatísticas:

(1) **Contraste**

$$\sum_{i,j} |i-j|^2 p(i,j), \tag{2}$$

(2) **Correlação**

$$\sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)p(i,j)}{\sigma_i \sigma_j}, \tag{3}$$

(3) **Energia**

$$\sum_{i,j} p^2(i,j), \text{ e} \tag{4}$$

(4) **Homogeneidade**

$$\sum_{i,j} \frac{p^2(i,j)}{1+|i-j|}, \tag{5}$$

em que μ_i e μ_j são as médias e σ_i e σ_j correspondem ao desvio padrão das linhas e das colunas. As características com base na matriz de co-ocorrência são calculadas no domínio espacial e consideram a natureza estatística da textura.

2.5 Máquina de Aprendizado Extremo

O classificador ELM (*Extreme Learning Machine*) [Huang et al. 2006] é uma rede neural artificial do tipo *feedforward* que contém duas camadas, uma oculta e outra de saída [Cambria et al. 2013]. A essência do ELM é que os pesos da camada oculta não necessitam ser calculados, ou seja, a camada não precisa ser treinada. Os valores iniciais podem ser independentes dos dados de treinamento. Geralmente estes são valores aleatórios e os pesos de saída são calculados pelo método dos mínimos quadrados. Esta técnica proporciona maior desempenho ao processo. Quanto maior o número de neurônios na camada oculta, o sistema terá mais liberdade e pode ser adaptado de forma adequada aos dados de treinamento. Ao mesmo tempo, essa condição pode provocar a situação de *overfitting*, fazendo com que o sistema perca a capacidade de generalização.

3. METODOLOGIA

Neste trabalho foram combinados os filtros de Laws e a análise estatística para texturas de Haralick. Na Figura 2 temos o fluxograma da metodologia proposta.

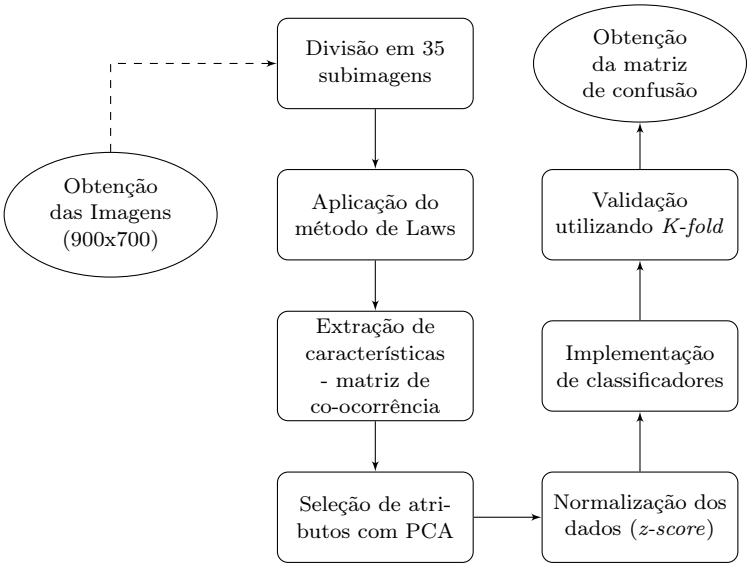


Fig. 2: Fluxograma da metologia proposta.

3.1 Extração de características

Ao aplicar o método de Laws foram geradas nove sub-imagens filtradas com informação específica de cada uma, como mostra a Figura 3. De cada uma destas nove sub-imagens filtradas foram calculadas

as matrizes de co-ocorrência utilizando a distância de um pixel em relação aos demais e combinando-a com os quatro graus de direção permissíveis: 0°, 45°, 90°, 135°. Com estes parâmetros foram obtidas quatro matrizes de co-ocorrência (uma por cada combinação) e de cada uma se extraíram as características de contraste, correlação, energia e homogeneidade de Haralick. Resultando finalmente um vetor com 144 atributos para cada imagem.

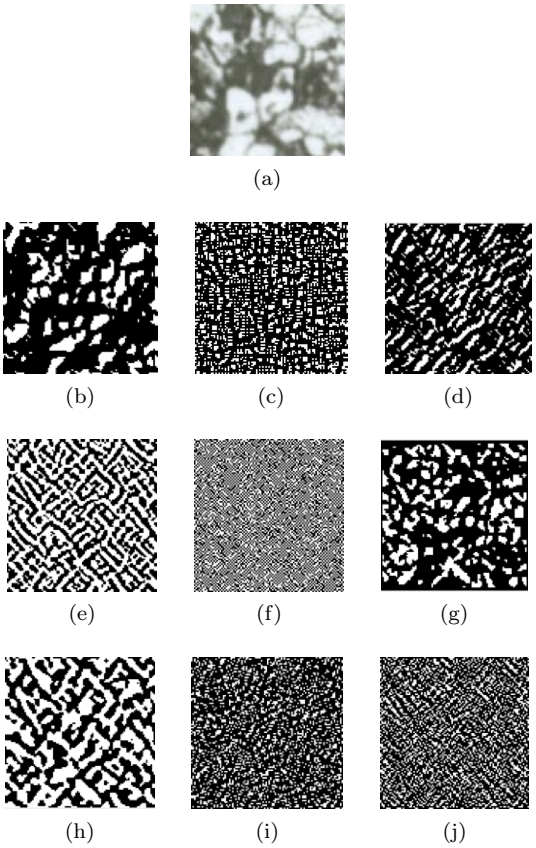


Fig. 3: Resultados da aplicação dos filtros de Laws na imagem 3a. As imagens 3b, 3c, 3d, 3e, 3f, 3g, 3h, 3i e 3j foram obtidas a partir dos filtros seguindo a ordem apresentada na Tabela I.

3.2 Tratamento de características

Com a definição das características das imagens, deve-se avaliar como elas contribuem para a classificação. Para os casos em que o vetor de atributos possui uma grande quantidade de características é aconselhável utilizar técnicas que possibilitem um melhor desempenho na utilização de tais vetores, como por exemplo, a técnica de Análise de Componentes Principais (*Principal Component Analysis – PCA*). Essa técnica converte um conjunto de variáveis correlacionadas em um subconjunto com variáveis independentes que representam a maioria da informação. Tal procedimento é realizado através de combinações lineares entre as variáveis originais assumindo que o conjunto de entrada tem componentes correlacionadas [Sandoval et al. 2015]. No vetor de atributos obtido, foi aplicada a técnica PCA para ordenar os dados correlacionados por importância. Das 144 componentes principais resultantes foram utilizadas apenas as 25 primeiras.

Para colocar todos os dados do vetor de características no mesmo intervalo, independentemente da unidade de medida, foi aplicada a normalização *z-score* que utiliza ferramentas estatísticas (média e desvio padrão) para centralizar e normalizar os dados de cada amostra:

$$z = \frac{X - \mu}{\sigma}, \tag{6}$$

em que z indica os dados normalizados, x corresponde a cada uma das características do vetor, μ é a média de cada atributo e σ representa o desvio padrão.

3.3 Implementação de Classificadores

Com os dados normalizados, foram implementados os classificadores, escolheu-se os métodos KNN e MLP por serem classificadores clássicos da literatura, e o ELM foi escolhido pelo fato de ser um classificador relativamente novo e com um treinamento muito mais rápido do que o backpropagation do MLP.

A quantidade de vizinhos mais próximos para o KNN foi de 25 dados, usando distância euclidiana. O tempo de treinamento foi de 2,31s e o teste 0,12s. A rede MLP foi treinada com 15 neurônios e uma camada oculta; a quantidade de épocas utilizadas foi de 20 e demorou 44,9 s no treino, no entanto, o tempo de teste foi de 23,8 s. Para a rede ELM os pesos aleatórios tiveram uma distribuição gaussiana e foram utilizados 93 neurônios na camada oculta. O tempo de treinamento foi de 5,3 s e o teste foi realizado em 1,8 s.

Para todos os testes com os classificadores foi usado como método de validação cruzada a técnica k-fold, assumindo o valor de k igual a 5. Os parâmetros como a quantidade de neurônios de cada camada do ELM e MLP, a quantidade de vizinhos do KNN e as épocas do MLP foram obtidos de maneira empírica. Com estes dados foi possível alcançar os melhores resultados.

4. RESULTADOS E DISCUSSÃO

A identificação dos aços proposta neste artigo é uma experimentação *in situ* que não requer ferramentas ou máquinas especializadas para obter as imagens. A taxa de acerto médio com o ELM é de 90% e o funcionamento do classificador fica evidenciado na matriz de confusão apresentada na Tabela II.

AISI 4340	AISI D6	AISI O1	AISI O1 (retificado)	AISI 1018	Taxa de acerto	Classe Real
206	0	3	1	0	98,10%	AISI 4340
1	165	12	31	11	75,00 %	AISI D6
0	0	199	1	0	99,50%	AISI 01
23	5	6	176	0	83,81%	AISI 01 (retificado)
0	6	0	1	203	96,67%	AISI 1018

Table II: Matriz de confusão obtida com a ELM e taxa de acerto.

A matriz de confusão mostra que as maiores taxas de acerto correspondem aos aços: AISI 4340 (98,10%) e AISI O1 (99,5%). A causa da variação corresponde aos acabamentos superficiais que esses tipos dos aços possuem. O aço AISI 4340 foi submetido ao tratamento térmico de bonificação, que consiste em aplicar têmple e revenido para organizar a estrutura molecular da borda. O aço AISI 01 é retificado e tem um acabamento superficial que garante a homogeneidade da estrutura molecular.

Com o KNN a taxa de acerto foi 78% com tempo de processamento de 2,43s. Aplicando o MLP o acerto foi de 90% em 68,7s. Por fim, com o classificador selecionado (rede neural ELM) o acerto foi de 91% em 7,1s. Como se pode observar, o acerto obtido com MLP e ELM foram similares mas o tempo de processamento foi menor aplicando ELM.

5. CONCLUSÕES E TRABALHOS FUTUROS

O objetivo principal do estudo, que foi identificar os tipos de aços por meio do processamento digital de imagens metalográficas, foi alcançado com sucesso geral de 90%. Essa taxa de acerto permite aplicação e solução automatizada de análise metalográfica tanto em meio acadêmico quanto para a área industrial. Deve-se ressaltar que a obtenção, o uso e o processamento das imagens originais ocasionaram uma redução da resolução dos dados de entrada, o que pode ter contribuído para não obtenção de uma maior taxa de acerto. Este trabalho pode alcançar maior eficiência se associado a uma base de imagens de melhor definição.

Mesmo com o método proposto atingindo melhor desempenho, em termos de tempo de processamento, deve-se acompanhar se essa condição se mantém com a mudança da base de imagens. Isso é relevante para acompanhar se o desempenho do algoritmo é muito dependente da base de dados adotada em comparação às demais técnicas avaliadas nos experimentos apresentados.

REFERENCES

ASTM E3-01. *Standard Practice for Preparation of Metallographic Specimens*. ASTM International, West Conshohocken, PA, 2007.

Bastidas-Rodriguez, M., Prieto-Ortiz, F., and Espejo, E. Fractographic classification in metallic materials by using computer vision. *Engineering Failure Analysis* vol. 59, pp. 237 – 252, 2016.

Bharati, M. H., Liu, J., and MacGregor, J. F. Image texture analysis: methods and comparisons. *Chemometrics and Intelligent Laboratory Systems* 72 (1): 57 – 71, 2004.

Cambria, E., Huang, G.-B., Kasun, L. L. C., Zhou, H., Vong, C. M., Lin, J., Yin, J., Cai, Z., Liu, Q., Li, K., et al. Extreme learning machines [trends & controversies]. *IEEE Intelligent Systems* 28 (6): 30–59, 2013.

Chávez, G. *El estado y la globalización en la industria siderúrgica mexicana*. Universidad Nacional Autónoma de México, Instituto de Investigaciones Económicas, 2008.

de la Cruz, J. P., Marti, M., Conejo, R., Morales-Bueno, R., and Fernandez, T. An expert system for identifying steels and cast irons. *Engineering Applications of Artificial Intelligence* 7 (4): 455–459, 1994.

Gavarito, J. Metalografía-curso de materiales. Tech. rep., Escuela Colombiana de Ingeniería, 2011.

Haralick, R. M., Shanmugam, K., and Dinstein, I. Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics* SMC-3 (6): 610–621, Nov., 1973.

Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. Extreme learning machine: Theory and applications. *Neurocomputing* 70 (1-3): 489–501, 2006.

Kalpakjian, S. and Schmid, S. *Manufactura, ingeniería y tecnología*. Pearson Educación, 2002.

Laws, K. I. Textured image segmentation. Tech. rep., DTIC Document, 1980.

Medeiros, F. N., Ramalho, G. L., Bento, M. P., and Medeiros, L. C. On the evaluation of texture and color features for nondestructive corrosion detection. *EURASIP Journal on Advances in Signal Processing* 2010 (1): 1, 2010.

Pathak, B. and Barooah, D. Texture analysis based on the gray-level co-occurrence matrix considering possible orientations. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering* 2 (9): 4206–4212, 2013.

Sandoval, A. E. L., Mendoza, C., Martínez, L. Á. R. C., Rivas, E. A., Araiza, J. M. R. A., Carlos, J., and Ortega, P. Sistema de autenticación facial mediante la implementación del algoritmo pca modificado en sistemas embebidos con arquitectura arm. *Personal de la Revista*, 2015.

Sebastian, V., Unnikrishnan, A., Balakrishnan, K., et al. Gray level co-occurrence matrices: Generalisation and some new features. *arXiv preprint arXiv:1205.4831*, 2012.

Shapiro, L. and Stockman, G. *Computer Vision*. Prentice Hall, 2001.

Ulyanov, P., Usachov, D., Fedorov, A., Bondarenko, A., Senkovskiy, B., Vyvenko, O., Pushko, S., Balizh, K., Maltcev, A., Borygina, K., Dobrotvorskii, A., and Adamchuk, V. Microscopy of carbon steels: Combined {AFM} and {EBSD} study. *Applied Surface Science* vol. 267, pp. 216 – 218, 2013. 11th International Conference on Atomically Controlled Surfaces, Interfaces and Nanostructures.

Graph-Based Semi-Supervised Learning for Semantic Role Diffusion

M. G. Carneiro[1], L. Zhao[2], J. L. G. Rosa[2]

[1] Universidade Federal de Uberlândia, Brazil
mgcarneiro@ufu.br
[2] Universidade de São Paulo, Brazil
zhao@usp.br, joaoluis@icmc.usp.br

Abstract. Semi-supervised learning (SSL) has recently attracted a considerable amount of research in machine learning. Among many categories of SSL techniques, graph-based methods are characterized by their ability to identify classes of arbitrary distributions. In this article we investigate the application of these methods for Semantic Role Labeling (SRL) which is the task of automatically identifying and classifying arguments with roles that indicate semantic relationship between an event and its participants. Such roles have great potential to improve a wide range of tasks, such as information extraction and machine translation. Experiments on a Brazilian Portuguese corpus named PropBank-br, which was built with text from Brazilian newspapers, were performed varying the number of labeled points, graph construction and label diffusion methods. The results show that the combination between symmetric k-nearest neighbors graph and local and global consistency method is a promising choice to semantic role diffusion on PropBank-br.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

Keywords: Graph-based SSL, Label Diffusion, PropBank-br, Semantic Role Labeling, Semi-supervised learning

1. INTRODUCTION

Semi-supervised learning (SSL) considers the general problem of learning from labeled and unlabeled data. It has turned out to be a new topic of machine learning research that has recently attracted a considerable amount of research [Chapelle et al. 2006; Zhu 2008]. While unlabeled data is far easier to obtain, the labeling process is often expensive, time consuming and requires the efforts of human annotators, who must often be quite skilled. Among many categories of SSL techniques which include generative models and low-density separation models, graph-based methods have as main advantage their ability to identify classes of arbitrary distributions [Silva and Zhao 2012]. See [Zhu et al. 2003; Zhou et al. 2004; Jebara et al. 2009; Ozaki et al. 2011; de Sousa et al. 2013] for more details about graph-based SSL.

In this article, we investigate graph-based SSL methods for Semantic Role Labeling (SRL) which is the task of automatically identifying and classifying the arguments of a predicate with roles. Such roles indicate meaningful relations among the arguments, as who did what to whom, where, when and how [Palmer et al. 2010]. Motivated by its potential to improve applications in a wide range of Natural Language Processing (NLP) tasks, such as information extraction, question answering, plagiarism detection, and so on, SRL has received much attention in the last years. In addition, many lexical resources, such as PropBank and FrameNet, have been built to allow the development of efficient semantic role labelers. Under the PropBank annotation framework each predicate is associated with

Authors thank the financial support given by the São Paulo State Research Foundation - FAPESP (grants numbers 2012/07926-3, 2011/50151-0 and 2013/07375-0). Authors also acknowledge support from CAPES and CNPq.

a set of core roles (named A0, A1, A2, and so on) which interpretation is specific to that predicate, and a set of adjunct roles (e.g., location, manner or time) which interpretation is common across predicates. Following we present an example about the SRL task through the sentences 1, 2 and 3. In 2, the arguments are identified, and the argument classification is showed in 3. The former aims to identify groups of words in a sentence that represent semantic arguments, and the latter aims to assign specific labels to the identified groups.

1. *Seymour Cray can do it again.*[1]
2. [*Seymour Cray*$_{arg}$] [*can*$_{arg}$] *do* [*it*$_{arg}$] [*again*$_{arg}$].
3. [*Seymour Cray*$_{A0}$] [*can*$_{mod}$] *do* [*it*$_{A1}$] [*again*$_{tmp}$].

To be specific, this work focus on the argument classification task using the PropBank-br [Duran and AluÃŋsio 2012] which is a Brazilian Portuguese corpus built with text from Brazilian newspapers that follows the PropBank style. An important motivation of this choice is related to the scarcity of annotated data in PropBank-br which size is about one seventh of the original PropBank [Fonseca and Rosa 2013]. This represents a difficulty scenario for machine learning where graph-based SSL methods can be evaluated through very arbitrary and unbalanced distributions. As the core roles are verb-dependant and in order to have a bigger number of unlabeled points, we perform experiments using three of the most frequent verbs in the PropBank-br, "dar" (to give), "fazer" (to do) and "dizer" (to say), which are evaluated using many graph construction methods and label diffusion strategies, and considering different number of labeled points. For graph construction, two widely used graph construction methods: symmetric k-nearest neighbors (SkNN) and mutual k-nearest neighbors (MkNN) are considered and they are also combined with the minimum spanning tree (MST) forming other two methods (SkNN+MST and MkNN+MST). For label diffusion, the gaussian field and harmonic function (GFHF)[Zhu et al. 2003] and local and global consistency (LGC) [Zhou et al. 2004] are analyzed.

The remainder of the article is organized as follows. Sect. 2 briefly presents some related works; Sect. 3 describes the framework adopted in this work. Computer simulations are presented in Sect. 4; and Sect. 5 concludes the article.

2. RELATED WORK

The objective of this work is simple. It aims to investigate graph-based SSL methods for SRL task in order to explore particular characteristics of the PropBank.br such as scarcity of labeled data and unbalanced class distributions.

Most part of the techniques developed for SRL lies on the supervised category [Gildea and Jurafsky 2002; Pradhan et al. 2008] where large (and expensive) amount of human annotated data are used to train classifiers. Although PropBank-br annotated data is scarse, some works explored it in the supervised context [Alva-Manchego and Rosa 2012; Fonseca and Rosa 2013; Hartmann et al. 2016; Carneiro et al. 2016].

SRL literature contains few works about unsupervised and semi-supervised learning. Unsupervised works such as presented in [Lang and Lapata 2011] refer to the task as semantic role induction and the objective is to cluster argument instances of each verb. SSL works include the investigation of semi-supervised and "semi-unsupervised" approaches. For instance, bootstrapping approaches such as self-training and co-training methods are proposed in [He and Gildea 2006] while a semi-unsupervised approach which employs a small number of labeled data to build an informed prior distribution over an unsupervised method is presented in [Titov and Klementiev 2012]. In addition, approaches

[1] `http://verbs.colorado.edu/propbank/framesets-english/do-v.html`

that increase the manually annotated instances with unlabeled instances which roles were infered through projection [Fürstenau and Lapata 2012] as well as the reduction of lexical features sparsity by exploiting word representation [Deschacht and Moens 2009] have been proposed.

3. GRAPH-BASED SEMI-SUPERVISED LEARNING

Problem definition. Given a set of arguments $\mathbf{X} = \{a_1, \ldots, a_l, a_{l+1}, \ldots, a_n\}$ and a set of semantic roles $\mathcal{L} = 1, \ldots, c$, the first l arguments are labeled $\{y_1, \ldots, y_l\} \in \mathcal{L}$ and the remaining arguments $(u = n - l)$ are unlabeled. Tipically, $l \ll u$, i.e., the great majority of arguments does not possess labels. The goal is to predict the semantic roles of the unlabeled arguments.

General framework. Following we present the general framework used in our investigation:

(1) Build an undirected graph $\mathcal{G}$ from $\mathbf{X}$;
(2) Generate a weighted matrix $\mathbf{W}$ from a similarity measure $\mathcal{K}$ and $\mathcal{G}$;
(3) Compute the graph Laplacian matrix $\mathbf{L}$ from $\mathbf{W}$;
(4) Label the unlabeled points from the output matrix $\mathbf{F}$ obtained by using a diffusion strategy.

3.1 Graph Construction

Consider an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each node $v_i \in V$ represents an argument $a_i \in \mathbf{X}$. Let $\mathbf{S}$ be a distance matrix in which $\mathbf{S}_{ij} = \delta(a_i, a_j)$ and $k\text{NN}_i$ be the set of k nearest neighbors of a_i, the adjacency matrix $\mathbf{A}$ of a kNN-graph is obtained as follows:

$$\mathbf{A}_{ij} = \begin{cases} 1, & \text{if } a_j \in k\text{NN}_i \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

As the kNN-graph may not be symetric, two strategies are commonly used to symmetrize it: symmetric kNN and mutual kNN. The symmetric kNN (SkNN) is obtained as follows:

$$\mathbf{A} = max(\mathbf{A}, \mathbf{A}^T) \ , \tag{2}$$

and the mutual kNN (MkNN) is obtained as follows:

$$\mathbf{A} = min(\mathbf{A}, \mathbf{A}^T) \ . \tag{3}$$

We also investigate the combination of both graph construction methods to the minimum spanning tree graph (MST) as a way to rigorously ensure the graph-based SSL assumption that each unlabeled point is on a connected subgraph in which there exists at least one labeled point. Let $\mathcal{G}$ denote a fully connected graph whose weights of the edges are given by a distance matrix $\mathbf{S}$. The minimum spanning tree (MST) of $\mathcal{G}$ is the subset of edges $\mathcal{E}' \subset \mathcal{E}$ which connect all the nodes in $\mathcal{V}$ minimizing the following quantity: $\sum_{i,j} \mathbf{S}_{ij}$.

3.2 Weighted Matrix Generation and Graph Laplacian

Despite the possibility to generate a fully connected weighted matrix, the graph construction step is important to promote sparsification in $\mathbf{W}$ which considerably reduces the computational complexity and usually improves the performance. Thus, a symmetric weighted matrix $W \subset \mathbb{R}^{n \times n}$ is given by:

$$\mathbf{W}_{ij} = \mathbf{A}_{ij} \mathcal{K}(a_i, a_j), \tag{4}$$

where $\mathcal{K}(a_i, a_j)$ denotes a similarity function. The Gaussian or RBF kernel has been used in this article:

$$\mathcal{K}(a_i, a_j) = exp\left(- \frac{\delta(a_i, a_j)}{2\sigma^2} \right), \tag{5}$$

where $\delta(\cdot)$ is a vector-based distance function such as the l_2-norm $\|a_i - a_j\|^2$, and σ is the kernel bandwidth parameter.

From the weighted matrix $\mathbf{W}$, the normalized Laplacian $\mathbf{L}$ is obtained:

$$\mathbf{L} = \mathbf{I}_n - \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}, \tag{6}$$

where $\mathbf{I}_n$ is the identity matrix and $\mathbf{D}$ is the diagonal matrix which contains the vertices degree. As the literature suggest the normalized Laplacian may lead better results in comparison with the combinatorial Laplacian ($\mathbf{\Delta} = \mathbf{D} - \mathbf{W}$), we used $\mathbf{L}$ in the formulation of the label diffusion algorithms which are presented in next sub-section.

3.3 Label Diffusion

The label diffusion process is conducted by the following graph-based techniques: gaussian field and harmonic function (GFHF) [Zhu et al. 2003] and local and global consistency (LGC) [Zhou et al. 2004]. Given a graph Laplacian $\mathbf{L}$ and let $\mathbf{Y} \in \mathbb{B}^{n \times c}$ be a label matrix in which $\mathbf{Y}_{ij} = 1$ if and only if a_i has label $y_i = j$, GFHF and LGC generate the output matrix $\mathbf{F}$ by label diffusion as follows:

Gaussian Field and Harmonic Function GFHF obtains the output matrix $\mathbf{F}$ as follows:

$$\mathbf{F}_{\mathcal{U}} = -\mathbf{L}_{\mathcal{U}\mathcal{U}}^{-1}\mathbf{L}_{\mathcal{U}\mathcal{L}}\mathbf{Y}_{\mathcal{L}} \ . \tag{7}$$

Local and Global Consistency LGC obtains the output matrix $\mathbf{F}$ as follows:

$$\mathbf{F}(t+1) = \alpha\mathbf{L}\mathbf{F}(t) + (1-\alpha)\mathbf{Y} \ , \tag{8}$$

where α defines the relative amount of the information from its neighbors and its initial label.

4. COMPUTER SIMULATIONS

This section provides experimental results using the general framework described in the previous section. The objective is to evaluate the performance of the label diffusion techniques on the SRL task. In the study, each simulation was performed in a transductive setting using different numbers of labeled and unlabeled points. The number of labeled points l is dependant of the number of classes c in the data set, i.e., $l = \eta c$ with $\eta \in \{1, 2, 3\}$. For each labeled set size l tested, we perform 30 trials. In each trial we randomly sample labeled data from the entire data set ensuring there is at least one labeled point per class, and use the rest of instances as unlabeled data. The error rate averaged over the trials is used to evaluate the quality of the semantic role diffusion process.

4.1 Datasets

In this article we present results for SRL task using the PropBank.br. For the experiment, we select all sentences related to the predicate "to give", "to do" and "to say", which are among the most frequent verbs in the corpus, and extract the attributes of each argument by using a set of features from the literature[2] [Gildea and Jurafsky 2002; Alva-Manchego and Rosa 2012; Pradhan et al. 2008]. As a pre-processing step, argument classes smaller than ten instances were excluded. Table I presents a brief description of the data sets obtained, named here "PBbr-give", "PBbr-do" and "PBbr-say", in terms of number of instances/arguments and classes. In the simulations, each argument was mapped as a vertex into an underlying network. As a data preparation, each instance attribute vector was normalized to have a magnitude of one and the euclidean distance was used in all simulations as

[2]The features used were: FirstForm+FirstPostag, FirstLemma, Head, HeadLemma, TopSequence, PostagSequence, PredLemma+PhraseType, LastForm+LastPostag, PredLemma+Path, FirstPostag, LeftHead, RightHead, VoicePosition, LeftHeadPostag, RightPhrase, and PredLemma.

the distance measurement. In order to avoid the dimensionality curse problem, we run principal component analysis (PCA) to reduce the dimensionality of the data to 100 features.

Table I: Brief description of the PropBank-br data sets.

	PBbr-give	PBbr-do	PBbr-say
#Instances / #Classes	148 / 3	397 / 8	506 / 5

4.2 Parameters and Baseline

The following parameters are employed in the computer simulations presented here: the value of k in the kNN-graph is optimized over the set $k \in \{1, 2, \ldots, 60\}$; the kernel bandwidth σ in the Gaussian Kernel $\mathcal{K}$ is defined as suggested in [Jebara et al. 2009] by $\sigma = \bar{d}_k/3$ where $\bar{d}_k$ is the average distance between each sample and its k-th nearest neighbor; and the parameter α in LGC is chosen at range $\{0.001, 0.01, 0.05, 0.1, 0.2, 0.5, 0.8, 1.0\}$. As in related works in literature, 1NN classifier is used for comparison purposes as a baseline.

4.3 Results and Discussion

This subsection provides results obtained in the simulations which are organized in three major experiments according to the number of labeled points $\eta \in \{1, 2, 3\}$. Note that $\eta = 1$ is the most relevant experiment to the addressed problem as it suggests approximately one sentence per verb is labeled.

The results of the three experiments are presented in Table II. In the first one ($\eta = 1$), which employs one labeled item per class, one can see: (i) LGC-SkNN presents the best performance on PBbr-give and PBbr-do data sets while GFHF-SkNN performs very well on PBbr-say; (ii) the baseline performance is very far from both GFHF and LGC when they are combined with SkNN and SkNN+MST graph construction methods. In the second and third one ($\eta = \{2, 3\}$), one can observe: (i) LGC-* (the four graph construction methods have very similar performance) presents the better performance on PBbr-give and LGC-SkNN on PBbr-do, and *-SkNN+MST performs very well on PBbr-say; (ii) MkNN graph construction has the worst performance on all data sets; (iii) by increasing the number of labeled points ($\eta > 1$), the baseline has its performance considerably improved on the data sets, but it is also outperformed. In order to analyze statistically the results obtained in the simulations, the Wilcoxon Signed Ranks test is adopted with $\alpha = 0.1$. The tests reveal LGC-SkNN and LGC-SkNN+MST provide the better results when comparing each two methods over all data sets.

Now we move on to analyze the general performance of both label diffusion algorithms GFHF and LGC in function of the graph construction methods and the variation of their parameter k. As LGC has also the parameter α, we set α according to the best performance obtained by the technique. Again we consider our division of the computer simulations in three experiments according to the number of labeled points. Figs. 1, 2 and 3 show the average error rates of the techniques for $\eta = 1$ (one labeled point per class), $\eta = 2$ and $\eta = 3$, respectively. By analyzing the figures, we can see:

GFHF × LGC: Despite GFHF does not require parameter selection (σ is defined heuristically), its predictive performance is usually worse than LGC in the PropBank-br data sets considered here.

SkNN × MkNN: Independently of the number of labeled points l and the label diffusion algorithm used, MkNN present the worst general performance on PBbr-give as well as on PBbr-say data sets. SkNN graph construction method is also better than MkNN on PBbr-do data set, with exception in the case when there is only one labeled point per class ($\eta = 1$).

MST × not MST: For $\eta > 1$, the minimum spanning tree (MST) graph improves the general performance of both SkNN and MkNN on PBbr-give and PBbr-say data sets.

Table II: Comparative results in terms of average error rates and standard deviations (over thirty runs) using $\eta = 1$ (one labeled point per class), $\eta = 2$ and $\eta = 3$. Best result for each data set is in bold face.

$\eta = 1$		PBbr-give (3)	PBbr-do (8)	PBbr-say (5)
	1NN	45.31 (11.37)	63.52 (9.80)	35.41 (14.97)
	GFHF-SkNN	41.26 (11.20)	50.21 (4.50)	**29.41 (19.84)**
	GFHF-SkNN+MST	41.54 (9.96)	50.21 (4.50)	30.95 (15.03)
	GFHF-MkNN	43.72 (9.05)	50.75 (4.52)	34.55 (7.93)
	GFHF-MkNN+MST	43.84 (8.70)	50.34 (4.53)	34.42 (7.16)
	LGC-SkNN	**37.65 (11.20)**	**47.02 (2.90)**	30.31 (19.36)
	LGC-SkNN+MST	**37.65 (11.20)**	47.78 (3.79)	30.82 (16.26)
	LGC-MkNN	39.61 (9.69)	48.93 (3.03)	34.47 (7.96)
	LGC-MkNN+MST	39.72 (10.38)	47.13 (3.30)	33.59 (4.01)
$\eta = 2$		PBbr-give (6)	PBbr-do (16)	PBbr-say (10)
	1NN	32.98 (8.96)	47.23 (3.77)	22.65 (13.15)
	GFHF-SkNN	31.78 (10.15)	44.28 (4.84)	23.47 (10.71)
	GFHF-SkNN+MST	31.73 (9.58)	45.74 (4.09)	**21.46 (12.36)**
	GFHF-MkNN	32.91 (9.21)	46.02 (4.28)	26.16 (7.77)
	GFHF-MkNN+MST	32.14 (9.92)	45.82 (4.12)	23.56 (12.10)
	LGC-SkNN	30.96 (9.43)	**43.57 (4.18)**	22.91 (10.83)
	LGC-SkNN+MST	30.96 (9.43)	44.82 (4.73)	21.70 (14.49)
	LGC-MkNN	30.96 (9.52)	46.63 (3.24)	26.04 (7.81)
	LGC-MkNN+MST	**30.89 (9.36)**	44.98 (5.67)	22.00 (12.67)
$\eta = 3$	Alg.	PBbr-give (9)	PBbr-do (24)	PBbr-say (15)
	1NN	33.76 (9.51)	43.22 (6.26)	15.48 (6.57)
	GFHF-SkNN	29.50 (8.64)	42.48 (5.07)	17.05 (8.69)
	GFHF-SkNN+MST	29.50 (8.64)	43.13 (4.28)	15.80 (8.26)
	GFHF-MkNN	30.36 (8.71)	43.44 (4.24)	20.98 (7.79)
	GFHF-MkNN+MST	29.86 (8.73)	43.34 (4.28)	15.84 (8.37)
	LGC-SkNN	**28.94 (8.36)**	**41.38 (5.07)**	16.71 (8.69)
	LGC-SkNN+MST	**28.94 (8.36)**	44.03 (5.09)	**15.12 (7.35)**
	LGC-MkNN	29.18 (8.74)	45.29 (4.87)	19.81 (8.17)
	LGC-MkNN+MST	29.02 (8.91)	43.31 (5.48)	16.00 (7.92)

PBbr-do: The best average error rates obtained to PBbr-do usually have small values of k. We believe this is because the data set is very unbalanced with 2 of 8 classes pursuing more than 68% of the data points. In this sense, LGC have some trouble to decrease its average error rate as k increases while GFHF not. Such result deserves more investigation.

5. CONCLUSION

In this article we investigated the application of graph-based SSL methods for semantic role labeling on a Brazilian Portuguese corpus named PropBank-br. Four graph construction methods and two label diffusion functions were evaluated in computer simulations with different number of labeled points. The results show LGC-SkNN as the best combination. The analysis also reveal MST can improve the general performance of the techniques as the number of labeled points increase. Forthcoming works include comparisons with other SSL methods and the inclusion of more verbs in the experiments.

REFERENCES

Alva-Manchego, F. E. and Rosa, J. L. G. Semantic role labeling for brazilian portuguese: A benchmark. In *Advances in Artificial Intelligence – IBERAMIA 2012*. Springer, pp. 481–490, 2012.

Carneiro, M. G., Zhao, L., Cheng, R., and Jin, Y. Network structural optimization based on swarm intelligence for highlevel classification. In *IEEE International Joint Conference on Neural Networks*. pp. 3737–3744, 2016.

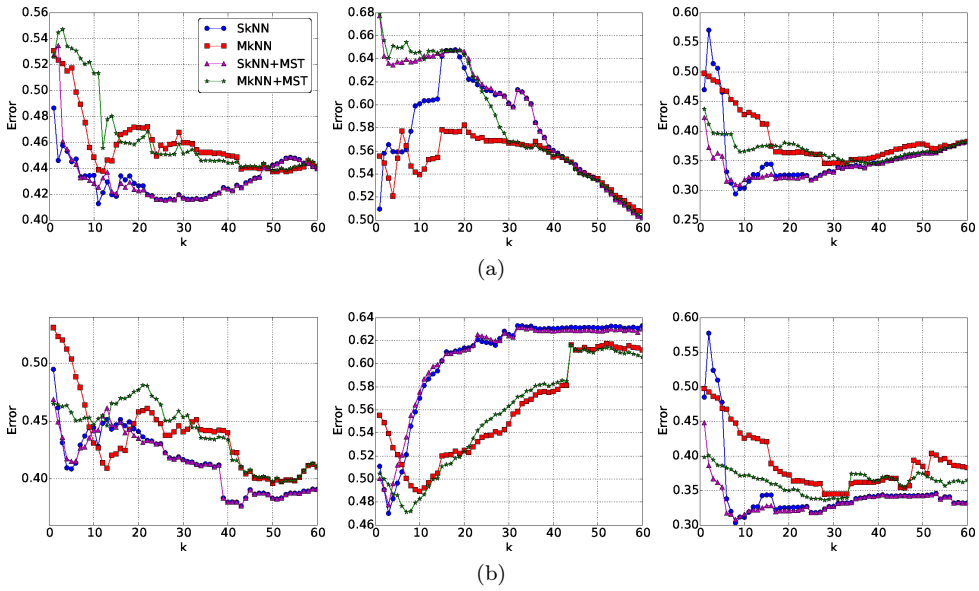


Fig. 1: Average error rates of (a) GFHF and (b) LGC on PBbr-give (left), PBbr-do (middle) and PBbr-say (right) data sets with one labeled point per class.

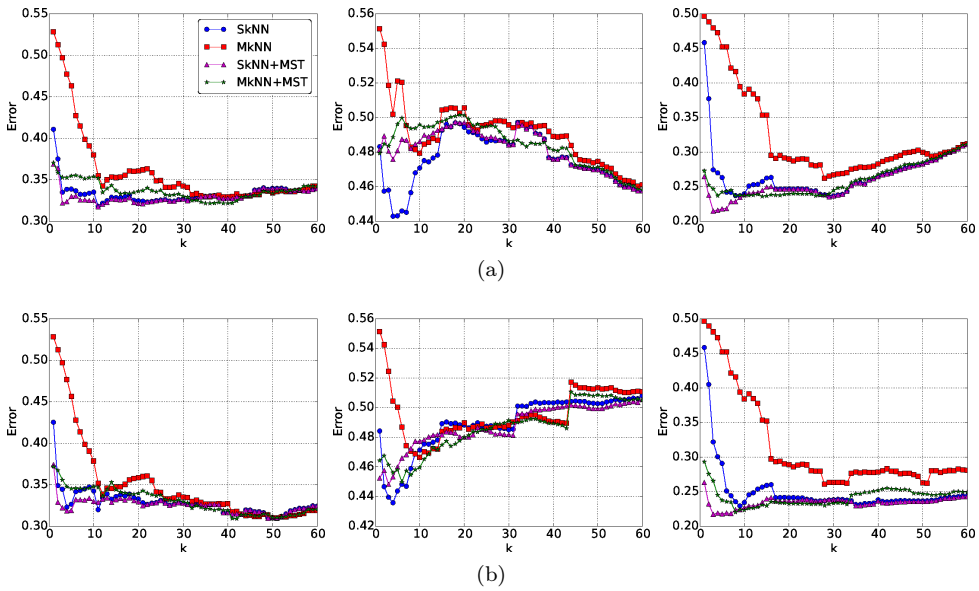


Fig. 2: Average error rates of (a) GFHF and (b) LGC on PBbr-give (left), PBbr-do (middle) and PBbr-say (right) data sets with the number of labeled points given by 6, 16 and 10, respectively.

Chapelle, O., Schölkopf, B., and Zien, A. *Semi-supervised learning*. MIT Press, 2006.

de Sousa, C. A. R., Rezende, S. O., and Batista, G. E. Influence of graph construction on semi-supervised learning. In *Machine Learning and Knowledge Discovery in Databases*. Springer, pp. 160–175, 2013.

Deschacht, K. and Moens, M.-F. Semi-supervised semantic role labeling using the latent words language model. In *ACL Conference on Empirical Methods in Natural Language Processing*. pp. 21–29, 2009.

Duran, M. S. and AluÃısio, S. M. Propbank-br: a brazilian treebank annotated with semantic role labels. In *International Conference on Language Resources and Evaluation*. pp. 1862–1867, 2012.

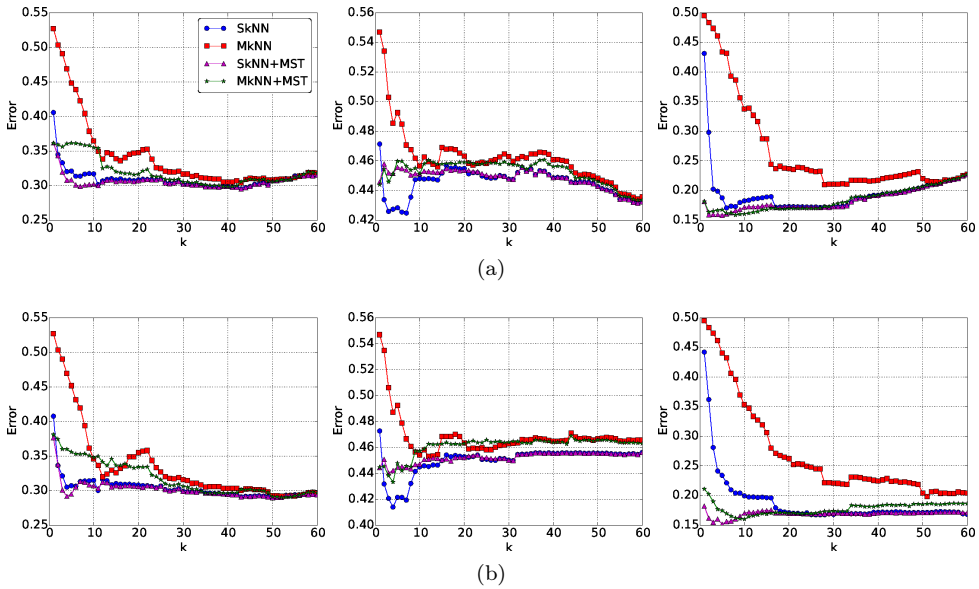


Fig. 3: Average error rates of (a) GFHF and (b) LGC on PBbr-give (left), PBbr-do (middle) and PBbr-say (right) data sets with the number of labeled points given by 9, 24 and 15, respectively.

Fonseca, E. R. and Rosa, J. L. G. A two-step convolutional neural network approach for semantic role labeling. In *IEEE International Joint Conference on Neural Networks*. pp. 2955–2961, 2013.

Fürstenau, H. and Lapata, M. Semi-supervised semantic role labeling via structural alignment. *Computational Linguistics* 38 (1): 135–171, 2012.

Gildea, D. and Jurafsky, D. Automatic labeling of semantic roles. *Computational Linguistics* 28 (3): 245–288, 2002.

Hartmann, N. S., Duran, M. S., and Aluísio, S. M. Automatic semantic role labeling on non-revised syntactic trees of journalistic texts. In *Computational Processing of the Portuguese Language*. pp. 202–212, 2016.

He, S. and Gildea, D. Self-training and co-training for semantic role labeling: Primary report. Tech. rep., 2006.

Jebara, T., Wang, J., and Chang, S.-F. Graph construction and b-matching for semi-supervised learning. In *International Conference on Machine Learning*. pp. 441–448, 2009.

Lang, J. and Lapata, M. Unsupervised semantic role induction via split-merge clustering. In *Annual Meeting of the Association for Computational Linguistics*. Vol. 1. pp. 1117–1126, 2011.

Ozaki, K., Shimbo, M., Komachi, M., and Matsumoto, Y. Using the mutual k-nearest neighbor graphs for semi-supervised classification of natural language data. In *ACL Conference on Computational Natural Language Learning*. pp. 154–162, 2011.

Palmer, M., Gildea, D., and Xue, N. *Semantic Role Labeling*. Morgan & Claypool Publishers, 2010.

Pradhan, S. S., Ward, W., and Martin, J. H. Towards robust semantic role labeling. *Computational Linguistics* 34 (2): 289–310, 2008.

Silva, T. C. and Zhao, L. Network-based stochastic semisupervised learning. *IEEE Transactions on Neural Networks and Learning Systems* 23 (3): 451–466, 2012.

Titov, I. and Klementiev, A. Semi-supervised semantic role labeling: Approaching from an unsupervised perspective. In *International Conference on Computational Linguistics*. pp. 2635–2652, 2012.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. Learning with local and global consistency. *Advances in Neural Information Processing Systems*, 2004.

Zhu, X. Semi-supervised learning literature survey. Tech. rep., 2008.

Zhu, X., Ghahramani, Z., and Lafferty, J. Semi-supervised learning using gaussian fields and harmonic functions. In *International Conference on Machine Learning*. pp. 912–919, 2003.

Detecção de Anomalia Aplicada a Pontos de Medição de Vazão em Plantas de Produção de Gás Natural

Hadriel Toledo Lima, Flavia Cristina Bernardini

Programa de Pós-Graduação em Engenharia
de Produção e Sistemas Computacionais
Universidade Federal Fluminense
hadriellima@gmail.com, fcbernardini@gmail.com

Abstract. Após o estabelecimento do Regulamento Técnico de Medição pela agência reguladora do setor petrolífero Brasileiro, ANP, garantir o valor reportado por Pontos de Medição em plantas de produção de Gás Natural passou a ter importância legal, além da operacional. Para facilitar o monitoramento de problemas nos Pontos de Medição, este trabalho propõe um modelo utilizando Redes Bayesianas Dinâmicas para Detecção de Anomalias. É um modelo espaço-temporal, pois é construído com base nas relações temporais e espaciais entre os pontos de medições da planta de produção. O artigo apresenta ainda uma avaliação das alternativas existentes para construção deste modelo, e uma comparação dos resultados em cada uma das alternativas.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: Anomaly Detection, Natural Gas Measurement Points, Dynamic Bayes Net

1. INTRODUÇÃO

A ANP — Agência Nacional do Petróleo, Gás Natural e Biocombustíveis — é o órgão regulador das atividades que integram a indústria do petróleo, gás natural e biocombustíveis no Brasil. Em 2000 lançou portaria em conjunto com o INMETRO — Instituto Nacional de Metrologia, Qualidade e Tecnologia — que regulamentou o processo de medição de petróleo e gás natural [ANP and INMETRO 2000]. Este regulamento, nomeado Regulamento Técnico de Medição — RTM — estabeleceu requisitos administrativos, técnicas, metrológicas e operacionais para contabilizar os volumes produzidos e movimentados. Este controle é importante para o correto recolhimento e distribuição das participações governamentais a estados e municípios. Desde o lançamento o regulamento passou por atualizações [ANP and INMETRO 2010] [ANP and INMETRO 2013], mas manteve o objetivo principal da busca por resultados acurados e completos na medição da produção. Após o RTM, detectar anomalias na medição de petróleo e gás natural passou a ter importância legal, além da operacional.

Em uma unidade de produção a medição do petróleo e gás natural produzido é feito por pontos de medição espalhados pela planta de processo da unidade. A distribuição desses pontos é projetado para garantir que todo o fluido produzido e movimentado seja medido. A rede de dutos que compõe a malha de escoamento do fluido pela planta de processo pode ser representada por um grafo dirigido, como o mostrado na Figura 1. Nesse grafo é representada uma planta de gás natural, que foi utilizada para os experimentos realizados neste trabalho. As arestas representam os dutos por onde o fluido escoa e os nós representam pontos de interesse para a medição na planta. Esses pontos podem representar a entrada ou saída de fluido da planta, ou bifurcação ou junção do fluido. Em algumas das arestas,

representados por traço duplo na Figura 1, estão presentes os pontos de medição. O fluido que entra na planta é produzido nos poços e escoa para: a) Exportação: saída para outra planta de processo ou gasoduto; b) Gás Lift: gás utilizado no método de elevação artificial Gás Lift; e c) Consumo: Gás consumido na planta de produção, por exemplo, em geradores de energia.

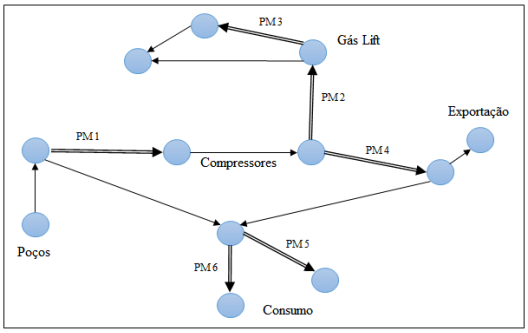


Fig. 1. Representação de malha de escoamento de gás natural

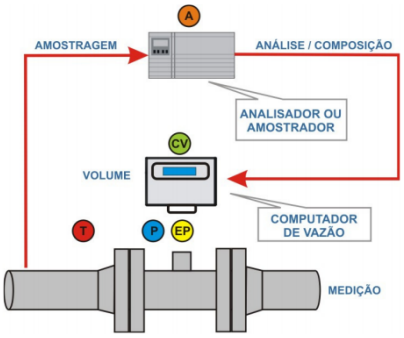


Fig. 2. Esquema de instalação de um ponto de medição

Um ponto de medição é composto por um conjunto de instrumentos como o mostrado na Figura 2. O Elemento Primário, representado por "EP" no figura, está localizado no duto de escoamento e tem participação direta na medição da vazão. Quando é do tipo Placa de Orifício ou V-cone, por exemplo, este instrumento gera o diferencial de pressão para o cálculo da vazão. Representado por "A", os Analisadores ou Amostradores são instrumentos que vão influenciar no resultado final da medição, tais como Analisadores BSW — porcentagem de água e sedimentos em relação ao volume total do fluido produzido, do inglês *Basic Sediments and Water* — e Densidade. Os símbolos "P" e "T" representam os instrumentos responsáveis por medir a pressão e temperatura, respectivamente. O Computador de Vazão, "CV", realiza o cálculo da vazão e volume produzido a partir dos valores fornecidos por todos os instrumentos que compõe o sistema. Os problemas mais comuns, no que tange à variação do valor medido, que podem ocorrer são: a) o valor medido permanece igual a zero; b) o valor medido "congela", ou seja, permanece fixo com o valor da última medição; e c) desvio no valor medido. Os dois primeiros problemas descritos são de fácil identificação, porém o terceiro é mais difícil de ser detectado, principalmente quando a variação é pequena.

Uma ferramenta computacional capaz de identificar pontos de medição em falha e alertar os responsáveis pelo tratamento é útil para evitar o descumprimento do regulamento. No entanto, há grandes desafios para implementação desta ferramenta: a) há um espaçamento temporal do fluxo entre os pontos de medição, bem como uma dependência temporal entre os diversos pontos de medição; b) variação do estoque no duto; e c) imprecisão dos instrumentos.

Este artigo tem como objetivo propor um modelo para identificação de falhas nos pontos de medição de gás natural nas plantas de produção e comparar alternativas na construção do modelo. Foi escolhido restringir o escopo do trabalho a pontos de medição de gás natural pois tais plantas apresentam um número maior de medidores, o que dificulta o monitoramento, e, normalmente, ocorrem mais falhas.

2. DETECÇÃO DE ANOMALIA E REDES BAYESIANAS

Para reconhecer padrões de funcionamento normal e problemático em uma rede de medição, podem ser utilizados algoritmos de aprendizado de máquina [Russell et al. 2003]. Na literatura, este tipo de problema é classificado como Detecção de Anomalias. Engloba problemas de encontrar padrões em dados que não estão de acordo com o comportamento esperado pois, em muitos domínios, anomalias em dados podem traduzir anomalias reais [Chandola et al. 2009].

Em domínios que o histórico de dados disponível seja tal que cada exemplo de treinamento seja rotulado como anomalia ou funcionamento correto, algoritmos clássicos de aprendizado supervisionado para problemas binários podem ser utilizados. No entanto, no problema abordado neste trabalho, os dados são coletados a cada minuto e a informação de anomalia é dada somente para períodos completos de 24 horas, fato que inviabiliza esta abordagem. É interessante que o modelo detecte anomalias em períodos bem menores que 24 horas, sendo este mais um ponto que reforça a inviabilidade do aprendizado supervisionado.

É difícil determinar nesse problema uma região que represente o comportamento normal dos pontos de medição. Um dos pontos de dificuldade é que há espaçamento temporal na passagem do fluido pelos pontos de medição. Em [Dereszynski and Dietterich 2011] e [Wang et al. 2008] os autores abordaram problemas com essa mesma característica. O primeiro mostra um método para detectar falhas em dados de sensores de temperatura que monitoram o clima. O segundo busca detectar falhas em sensores de gases tóxicos em uma mina de carvão. Os dois têm em comum a utilização de Redes Bayesianas Dinâmicas na construção do modelo, porém os dados dos dois trabalhos apresentam menor variação que os dados de vazão de gás, que apresentam variações consideráveis a cada minuto.

Uma dificuldade comum em problemas de detecção de anomalia é que há muito poucos exemplos de anomalias, fato que também ocorre neste domínio estudado. Assim, outra alternativa para a construção do modelo seriam algoritmos de aprendizado de uma classe. A dificuldade de modelar o problema nesta abordagem é devido a interdependência estatística entre todas as variáveis. Os dados de cada ponto de medição tanto seriam utilizados para predizer anomalias em outros medidores quanto neles mesmo. Assim, Redes Bayesianas (RBs) foi a alternativa que se mostrou mais indicada para o problema em questão.

RBs (ou Redes de Crença Bayesianas, Redes Causais ou Redes Probabilísticas) são utilizadas para representação do conhecimento incerto [Guo and Hsu 2002]. Russel e Norvig [Russell et al. 1995] definem RBs como um grafo no qual:

—Os nós são compostos por variáveis aleatórias;
—Os arcos são direcionados tal que o significado intuitivo de um arco a partir do nó X para o nó Y é que X tem uma influência direta sobre Y;
—Cada nó tem uma Tabela de Probabilidade Condicional (TPC) que quantifica os efeitos que os nós pais têm no nó em questão. Os pais de um nó X são todos os nós que possuem uma aresta dirigida para X;
—Os grafos têm ciclos não dirigidos, porém não têm ciclos dirigidos, e por esse motivo trata-se de um grafo direcionado acíclico — DAG (*Directed Acyclic Graph*).

A modelagem de séries temporais é possível por uma extensão de RB conhecida por Redes Bayesianas Dinâmicas (RBDs), no qual os nós representam as variáveis em intervalos de tempos determinados [Trifonova et al. 2015]. O termo dinâmico significa que ela é utilizada em sistemas dinâmicos, não sendo necessário que a rede mude a cada instante de tempo [Murphy 2002].

Segundo [Guo and Hsu 2002], uma RB pode ser considerada uma base de conhecimento probabilístico representada pela topologia da rede e a CPT de cada nó. A principal função de uma base de conhecimento é utilizá-la para computar resposta sobre o domínio, ou seja fazer inferência. Já a inferência em uma RB, denominada Inferência Bayesiana, é o processo de determinar informações com base nas probabilidades condicionais de qualquer hipótese dado os dados disponíveis, aplicando a regra de Bayes [Alves et al. 2015]. Os algoritmos de Inferência Bayesiana são divididos em dois grupos [Guo and Hsu 2002]: (i) **Inferência Exata**, que consiste no cálculo dos valores de probabilidades exatos da distribuição a posteriori completa [Alves et al. 2015] — em geral, apresentam tempo de execução exponencial à largura induzida do grafo, e por isso em muitos casos a inferência exata não é viável [Russell et al. 1995] —; e (ii) **Inferência Aproximada**, Utilizada nos casos em que a

inferência Exata não é viável — nesse caso, a exatidão é sacrificada e substituída por uma descrição aproximada da distribuição a posteriori. Neste trabalho, é utilizada a inferência aproximada.

3. DESENVOLVIMENTO DO MODELO

A implementação foi feita utilizando a linguagem R [R Core Team 2016] e o pacote bnlearn [Scutari 2010] que possui implementados alguns algoritmos para aprendizado de estrutura, estimativa de parâmetros e inferência em Redes Bayesianas. O código-fonte está disponível em `http://github.com/hadriellima/Anomaly-Detection-Bayesian-Network`.

A planta de produção utilizada nos testes foi apresentada na Figura 1. Ela possui 6 pontos de medição: PM1, PM2, PM3, PM4, PM5 e PM6. Foram coletados os dados de volume corrigido a cada k minutos destes pontos medição. As variações deste valor k são detalhadas na Seção 4. O conjunto de treinamento contém os dados de novembro a dezembro, e para o conjunto de testes, os dados de janeiro a abril do ano seguinte. Antes de estabelecer tais conjuntos, foram removidos dados espúrios e problemas de comunicação. Ainda, os dados foram normalizados para a o intervalo $[0, 1]$. As indicações de falha dos pontos de medição não estão atreladas apenas ao dia em que ocorreram, e não ao horário. Houve falha em 11 dos 180 dias que compreendem os período de dados.

Da mesma forma que [Dereszynski and Dietterich 2011], o modelo foi desenvolvido combinando dois componentes principais: a) o componente espacial, que representa a relação entre os pontos de medição em um instante de tempo; e b) o componente temporal, que representa a transição de um instante de tempo a outro.

Foi utilizado o aprendizado de estrutura de Redes Bayesianas para aprender o conjunto de relações espaciais entre os pontos de medição, e construção do componente espacial. Tal abordagem permite que o modelo se adapte a cada planta onde ele for implantado, sem a necessidade de conhecimento prévio destas relações. Foi utilizado o algoritmo de subida de encosta (*Hill-climbing*) para encontrar a melhor estrutura espacial com a métrica de pontuação BGe (*Bayesian Metric for Gaussian*). Para utilização de BGe foi assumido que o conjunto de dados corresponde a uma distribuição gaussiana multivariada. A estrutura da rede obtida é mostrada na Figura 3. Cada nó da rede, rotulado por X_i, representa o ponto de medição i dado $i \in \{PM1, PM2, PM3, PM4, PM5, PM6\}$.

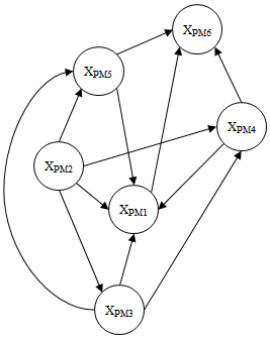


Fig. 3. Componente espacial da Rede Bayesiana construída pelo algoritmo de subida de encosta com métrica BGe

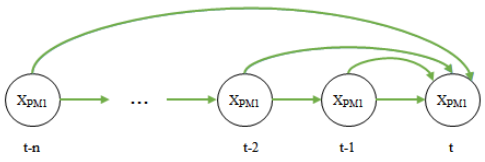


Fig. 4. Exemplo de Componente Temporal Adicionado ao Ponto de Medição PM1 com n variáveis de atraso

O componente temporal foi construído relacionando cada ponto de medição com as observações dos instantes de tempo anteriores. Estes nós também são chamados de variável de atraso (ou *lag*) neste trabalho. Na Figura 4 é mostrado o relacionamento do ponto de medição PM1 com as variáveis de atraso até o tempo n. As relações temporais nesta figura e nas demais estão desenhadas em verde, enquanto que as relações espaciais em preto. Essa relação temporal é conhecida como relação de

Markov [Dereszynski and Dietterich 2011], e neste trabalho fizemos testes até a relação de Markov de segunda ordem, ou seja dois instantes de tempo de atraso. A adição dessas variáveis é responsável por transformar uma Rede Bayesiana Estática em Dinâmica (RBD).

Sob a hipótese de um Modelo Linear Gaussiano, uma variável normalmente distribuída $X \sim N(\mu_x, \sigma_x^2)$ condicionada a uma outra variável, denominada variável pai, normalmente distribuída, $Y \sim N(\mu_y, \sigma_y^2)$, se for assumido que ambas são univariadas, então a função de densidade pode ser descrita por $P(X|Y) \sim N(\mu_x + w_1 y, \sigma_x^2)$, onde w_1 é um peso escalar multiplicado por uma entrada y, dada pela distribuição de Y [Dereszynski and Dietterich 2011]. O próximo passo na construção do modelo é estimar os valores dos parâmetros $(\mu_i, \sigma_i^2 e w_i)$ utilizando a abordagem MLE — do inglês, *Maximum Likelihood Estimates*. Esta tarefa se resume a um problema de regressão linear múltipla [Russell et al. 2003].

Após a construção do modelo, com os componentes espacial e temporal, e a estimativa de parâmetros, é possível prever o valor medido por um ponto de medição com base nos nós pai na Rede Bayesiana. É esperado que este modelo apresente o diagnóstico se o ponto de medição está funcionando corretamente ou quebrado. Com este objetivo, foi incorporado ao modelo um nó para representar a observação atual do ponto de medição O_i e uma variável para representar o estado deste ponto S_i, onde i representa o ponto de medição. Os nós S_i são discretos e podem apresentar os valores funcionando ou quebrado. Os parâmetros para estes nós foram atribuídos manualmente. Para S foi utilizado a razão entre o número de dias com falhas, 11, e o período de dados levantado, 180 dias. Foi atribuído ao valor *quebrado* $0,1$ de probabilidade de falha para deixar o modelo um pouco mais propenso a alarmar uma falha. O estado *funcionando* foi atribuído o complemento deste valor. Para O_i, foi utilizada a ideia de que quando um ponto de medição está *funcionando*, o valor previsto x_i deve ser muito próximo ao valor observado, e quando o ponto de medição está *quebrado* uma variação maior. Na Figura 5 é apresentado o modelo completo proposto. As relações espaciais estão em preto, as temporais em verde e as relações para a tomada de decisão em azul.

$$P(S_i = funcionando) = 0,90$$
$$P(S_i = quebrado) = 0,10$$
$$P(O_i|S_i = funcionando, X_i = x_i) \sim N(x_i, 0, 1)$$
$$P(O_i|S_i = quebrado, X_i = x_i) \sim N(0, 0001x_i, 10000)$$

A inferência foi feita em dois passos. No primeiro os nós S e O são eliminados do modelo, e é feita a predição do valor de um ponto de medição com base nos nós relacionados a este na estrutura espacial e do componente temporal. De posse do valor previsto, este é incluído na configuração do nó O correspondente ao ponto de medição, para o dignóstico de falha, com base no valor observado deste ponto de medição.

4. EXPERIMENTOS E RESULTADOS

Como mencionado na Seção 3 o conjunto de dados deste estudo compreende os meses de novembro a dezembro para treinamento da rede bayesiana, e janeiro a abril para os testes. Um dos valores reportados de vazão para os pontos de medição é um acumulador perpétuo que incrementa o volume passado pelo ponto de medição a cada segundo. Com este valor é possível extrair intervalos de volume acumulado variados para os pontos de medição. Foi chamado de k o intervalo de volume acumulado, em minutos, e foram feitos testes com 3 valores diferentes: $k = 10, k = 20, k = 60$.

Outra questão importante é quantas variáveis de atraso devem ser observadas. Por questões de performance foi feito teste com até duas variáveis de atraso. Este teste consiste em comparar o valor previsto pelo modelo com o valor observado utilizando a métrica de erro médio quadrático , MSE — do inglês, *Mean Squared Error*. Na Tabela I são apresentados os resultados destes testes. O Modelo CE é composto apenas pelo componente espacial; o Modelo CET1, pelo componente espacial e temporal

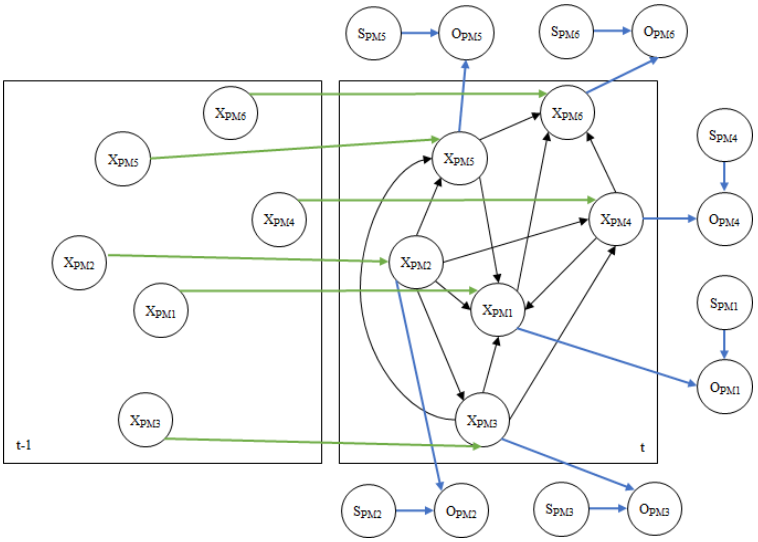


Fig. 5. Modelo proposto completo, com os componente Espacial, Temporal com uma variável de atraso e variáveis de observação e decisão

Table I. Comparativo da Predição entre Modelos utilizando métrica MSE

Ponto	k = 10			k = 20			k = 60		
	Modelo CE	Modelo CET1	Modelo CET2	Modelo CE	Modelo CET1	Modelo CET2	Modelo CE	Modelo CET1	Modelo CET2
PM1	0,003	0,001	0,001	0,004	0,002	0,002	0,010	0,003	0,003
PM2	0,055	0,001	0,001	0,061	0,002	0,002	0,061	0,003	0,003
PM3	0,039	0,001	0,001	0,027	0,001	0,001	0,034	0,005	0,006
PM4	0,142	0,012	0,010	0,041	0,007	0,006	0,032	0,006	0,006
PM5	0,081	0,010	0,042	0,060	0,011	0,055	0,067	0,030	0,028
PM6	0,394	0,050	0,010	0,498	0,072	0,010	0,222	0,043	0,038
Média	0,119	0,012	0,011	0,115	0,016	0,013	0,071	0,015	0,014

com 1 variável de atraso; e o Modelo CET2 pelo componente espacial e temporal com 2 variáveis de atraso. Este estudo comparativo mostra ganho significativo na precisão ao utilizar o componente temporal junto ao espacial. Também houve ganho com a adição da segunda variável de atraso, porém em menor escala. Quanto aos valores de k, percebe-se que a melhor precisão no menor valor, $k = 10$.

Testes estatísticos: Foi utilizado o teste de Friedman para verificar se as combinação do intervalo de tempo k e os parâmetros da Rede Bayesiana Dinâmica apresenta diferença estatística significativa [Demsar 2006]. Foi considerado então que a primeira variável aleatória v_1 é relativa aos resultados obtidos para $K = 10$ e Modelo CE; v_2 é relativa aos resultados obtidos para $K = 10$ e Modelo CET1; v_3 é relativa aos resultados obtidos para $K = 10$ e Modelo CET2; v_4 é relativa aos resultados obtidos para $K = 20$ e Modelo CE; v_5 é relativa aos resultados obtidos para $K = 20$ e Modelo CET1; v_6, para $K = 20$ e Modelo CET2; v_7, para $K = 30$ e Modelo CE; v_8, para $K = 30$ e Modelo CET1; e por fim, v_9, para $K = 60$ e Modelo CET2. Segundo o teste, há diferença significativa nos resultados obtidos (foi rejeitada a hipótese nula). Daí, foi realizado o pós-teste de Nemenyi. Na Figura 6 é exibido o resultado do teste, verificando-se que a significância estatística está entre dois grupos de resultados.

Também foi realizado o teste de Friedman para verificar se há diferença estatística nos resultados quando varia-se o valor de k. A primeira variável aleatória v_1 é equivalente a $k = 10$; v_2 é equivalente

a $k = 20$; e v_3 é equivalente a $k = 60$. Os valores observados foram: v_{1_1} — MSE para ponto PM1 e Modelo CE; v_{1_2} — MSE para ponto PM1 e Modelo CET1; v_{1_3} — MSE para ponto PM1 e Modelo CET2; v_{1_4} — MSE para ponto PM2 e Modelo CE; e assim sucessivamente, até $v_{1_{18}}$ — MSE para ponto PM6 e Modelo CET2. O teste não rejeitou a hipótese nula, significando que os resultados são estatisticamente semelhantes.

Por fim, foi realizado o teste de Friedman para verificar se há diferença estatística nos resultados quando se varia o modelo escolhido (CE, CET1 ou CET2). A primeira variável aleatória v_1 é equivalente ao modelo CE; v_2 é equivalente ao modelo CET1; e v_3 é equivalente ao modelo CET2. Os valores observados foram: v_{1_1} — MSE para ponto PM1 e $k = 10$; v_{1_2} — MSE para ponto PM1 e $k = 20$; v_{1_3} — MSE para ponto PM1 e $k = 60$; v_{1_4} — MSE para ponto PM2 e $k = 10$; e assim sucessivamente, até $v_{1_{18}}$ — MSE para ponto PM6 e $k = 60$. O teste rejeitou a hipótese nula, ou seja, há diferença significativa nos resultados obtidos. Daí, foi realizado o pós-teste de Nemenyi. Na Figura 7 é exibido o resultado do teste, verificando-se que a significância estatística está entre dois grupos de resultados.

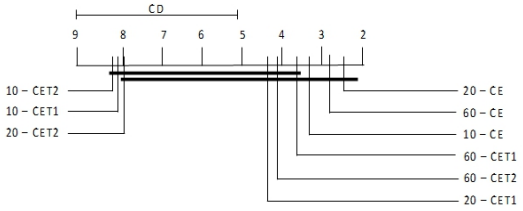


Fig. 6. Representação gráfica do pos-teste considerando cada combinação de cenário de experimento como uma variável aleatória

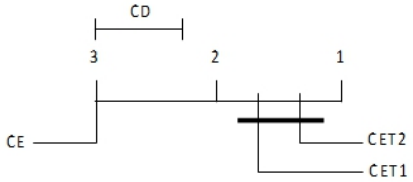


Fig. 7. Representação gráfica do pos-teste considerando cada modelo de rede bayesiana como uma variável aleatória

Análise de um dos modelos: Devidos aos resultados obtidos, observa-se na Figura 6 que o melhor modelo obtido foi considerando o modelo CET2 e $k = 10$. Assim, o modelo construído foi escolhido para avaliar o diagnóstico de falha. Essa parte do estudo consiste verificar a eficácia do modelo em detectar falhas nos pontos de medição. Foram testados todos os exemplos do conjunto de teste um por vez. Quando a probabilidade de falha de um exemplo fosse maior que 50%, o teste do exemplo seguinte utilizou apenas o componente espacial, pois um valor com potencial de estar incorreto como variável de atraso prejudicaria a previsão do valor seguinte.

Em alguns momentos o modelo apresenta falsos positivos em apenas um evento da série. É perceptível nos dados a variação naquele ponto de medição, porém nem sempre isto é uma falha. Não é comum que um problema se resolva em tempo inferior ao utilizado no teste (10 minutos) e, por isso, o modelo foi adaptado para reportar falha apenas quando o segundo evento de falha consecutivo for diagnosticado. Um resumo do comportamento do modelo é apresentado na Tabela II. Como as falhas reportadas estão associadas apenas ao dia em que ocorreram, a análise de aderência do modelo foi montada em dias. As linhas 3 e 4 da tabela apresentam uma matriz de confusão para o ponto de medição PM1; as linhas 5 e 6, uma matriz de confusão para o ponto de medição PM2; as linhas 7 e 8, uma matriz de confusão para o ponto de medição PM3; as linhas 9 e 10, uma matriz de confusão para o ponto de medição PM4; as linhas 11 e 12, uma matriz de confusão para o ponto de medição PM5; e por fim as linhas 13 e 14, uma matriz de confusão para o ponto de medição PM6. Apenas uma falha, ocorrida em PM5, não foi detectada pelo modelo. Esta falha ocorreu junto com a falha em PM6 que foi diagnosticada.

5. CONCLUSÃO

Neste artigo foi proposta a utilização de Redes Bayesianas Dinâmicas para construir um modelo espaçotemporal para Detecção de Anomalia em uma planta de produção de Gás Natural. É importante

Table II. Resultado da Detecção de Anomalia Feita Pelo Modelo Espaçotemporal Proposto

		Diagnóstico feito pelo modelo	
		Funcionando	**Quebrado**
PM1	**Func.**	180	0
	Queb.	0	0
PM2	**Func.**	180	0
	Queb.	0	0
PM3	**Func.**	180	0
	Queb.	0	0

		Diagnóstico feito pelo modelo	
		Funcionando	**Quebrado**
PM4	**Func.**	175	3
	Queb.	0	2
PM5	**Func.**	169	6
	Queb.	1	4
PM6	**Func.**	142	32
	Queb.	0	6

observar que os dados utilizados são reais. Nos testes realizados o modelo mostrou eficácia conseguindo detectar a maior parte das falhas presentes no conjunto de teste, deixando de detectar apenas uma. Houveram falsos positivos que ainda devem ser mais estudados, porém um alarme falso é menos crítico que a falta de um alarme para uma anomalia real. Assim sendo, a modelagem proposta apresentou resultados promissores para o problema em questão. Como trabalhos futuros, pretende-se aprofundar a exploração dos diversos parâmetros das redes bayesianas, considerar outros intervalos de tempo de atraso, e considerar a adaptação do modelo no tempo, já que há uma variação no comportamento dos dados em problemas de detecção de anomalia de medidores.

REFERENCES

Alves, J., Ferreira, J., Lobo, J., and Dias, J. Brief survey on computational solutions for bayesian inference. In *Workshop on Unconventional computing for Bayesian inference (UCBI), IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

ANP and INMETRO. Portaria conjunta anp e inmetro, nº. 1, 2000. dispõe sobre o regulamento técnico de medição de petróleo e gás natural. Tech. Rep. 1, Diário Oficial da União da República Federativa do Brasi, Brasília, DF, 2000.

ANP and INMETRO. Portaria conjunta anp e inmetro, nº. 1, 2010. dispõe sobre o regulamento técnico de medição de petróleo e gás natural. Tech. Rep. 1, Diário Oficial da União da República Federativa do Brasi, Brasília, DF, 2010.

ANP and INMETRO. Portaria conjunta anp e inmetro, nº. 1, 2013. dispõe sobre o regulamento técnico de medição de petróleo e gás natural. Tech. Rep. 1, Diário Oficial da União da República Federativa do Brasi, Brasília, DF, 2013.

Chandola, V., Banerjee, A., and Kumar, V. Anomaly Detection: A Survey. *ACM Computing Surveys*, 2009.

Demsar, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning Research* vol. 7, 2006.

Dereszynski, E. W. and Dietterich, T. G. Spatiotemporal models for data-anomaly detection in dynamic environmental monitoring campaigns. *ACM Transactions on Sensor Networks (TOSN)* 8 (1): 3, 2011.

Guo, H. and Hsu, W. A survey of algorithms for real-time bayesian network inference. In *AAAI/KDD/UAI02 Joint Workshop on Real-Time Decision Support and Diagnosis Systems*. Edmonton, Canada, 2002.

Murphy, K. P. *Dynamic bayesian networks: representation, inference and learning*. Ph.D. thesis, University of California, Berkeley, 2002.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.

Russell, S., Norvig, P., and Intelligence, A. A modern approach. *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs* vol. 25, 1995.

Russell, S. J., Norvig, P., Canny, J. F., Malik, J. M., and Edwards, D. D. *Artificial intelligence: a modern approach*. Vol. 2. Prentice hall Upper Saddle River, 2003.

Scutari, M. Learning bayesian networks with the bnlearn R package. *Journal of Statistical Software* 35 (3): 1–22, 2010.

Trifonova, N., Kenny, A., Maxwell, D., Duplisea, D., Fernandes, J., and Tucker, A. Spatio-temporal bayesian network models with latent variables for revealing trophic dynamics and functional networks in fisheries ecology. *Ecological Informatics* vol. 30, pp. 142–158, 2015.

Wang, X. R., Lizier, J. T., Obst, O., Prokopenko, M., and Wang, P. Spatiotemporal anomaly detection in gas monitoring sensor networks. In *Wireless Sensor Networks*. Springer, pp. 90–105, 2008.

Detecção *online* de *outliers* em agrupamento de fluxos contínuos de dados

M. A. Pereira[1], E. R. Faria[2], M. C. Naldi[1]

[1] Universidade Federal de Viçosa, Brasil
{mariana.a.alves, murilocn}@ufv.br
[2] Universidade Federal de Uberlândia, Brasil
elaine@ufu.br

Abstract. *Developments in hardware and software allowed the expansion of the generation of continuous and unlimited data, called data streams. In this scenario, appropriate adapted algorithms are essential due to memory management restrictions, inability to store all stream objects, return of a model analysis whenever the user requests, definition of critical input parameters, among others. Thus, many clustering algorithms to data streams emerged in recent decades, for example CluStream, one of the most widely cited algorithms in the data clustering process. However, many clustering algorithms in data streams are not able to detect outliers along the stream, or if they do, they require many critical input parameters. Inserting outliers tends to depreciate the model, leading to mistakes in representing properly the majority of the data flow to the user. This paper presents a proposal for detecting outliers objects for CluStream algorithm, namely CluStreamOD, which identifies such objects and stores them in an temporary memory. Periodically, clusters are sought in the objects of the temporary memory, when valid, they return to the model. The experiments show the proposal's effectiveness in detecting outliers objects at the time of arrival (online phase) and its ability to detect and return valid objects from memory to the model when they become relevant from new related objects.*

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: fluxos contínuos de dados, agrupamento de dados, detecção de *outliers*, *CluStream*

1. INTRODUÇÃO

Fluxos Contínuos de Dados (FCDs) são sequências ordenadas de objetos ($\chi = \mathbf{o_1}, \mathbf{o_2}, \mathbf{o_3}, ...$), onde $\mathbf{o_i}$ é um vetor que armazena um objeto de d dimensões, cujo tempo de chegada é t, analisados em ordem crescente do índice i, potencialmente ilimitados e que podem evoluir ao longo do tempo [Silva et al. 2013]. Problemas que envolvem FCDs tem sido extensivamente pesquisados por estarem envolvidos em um grande número de aplicações relevantes, como por exemplo, detecção de intrusos em redes de computadores, detecção de fraudes em cartões de crédito, predição e planejamento do consumo de energia elétrica, monitoramento em tempo real das atividade de um usuário, entre outras [Aggarwal 2003], [Aggarwal et al. 2003], [Guha et al. 2003].

O agrupamento de dados é uma das tarefas de grande importância para os FCDs, sendo que vários algoritmos foram desenvolvidos para esta tarefa [Aggarwal et al. 2003], [Cao et al. 2006], [Chen and Tu 2007], [Guha et al. 2003]. Os algoritmos de agrupamento de dados precisam considerar os seguintes requisitos: a capacidade da descoberta de grupos de formatos diferentes, mudanças nos grupos existentes ou surgimento de novos, habilidade para detectar *outliers*, flexibilidade para retornar ao usuário os grupos no momento solicitado, gerenciamento da memória, definição dos parâmetros de entrada, entre outros.

Ainda que muitos algoritmos tenham sido desenvolvidos para a tarefa de agrupamento em FCDs, a detecção de *outliers* ainda representa uma importante questão a ser tratada. A natureza dinâmica dos FCDs faz com que os papéis de *outliers* e grupos mudem frequentemente, e consequentemente novos grupos surgem, enquanto que os velhos desaparecem. Esse processo se torna mais complexo quando há presença de *outliers* [Cao et al. 2006]

A proposta discutida neste artigo tem por objetivo realizar uma melhoria no algoritmo *CluStream* [Aggarwal et al. 2003] quanto à detecção de *outliers*. Para isso, os objetos considerados potenciais *outliers* são detectados e inseridos em uma memória auxiliar para análise futura. Periodicamente, grupos válidos são procurados nessa memória para serem inseridos no modelo. Desta forma são realizadas apenas inclusões seguras no modelo, seja no momento da chegada do objeto ou após o processo de validação da memória auxiliar, evitando sua depreciação devido à inserção de *outliers*. Todo o processo é realizado buscando utilizar uma quantidade mínima de parâmetros de entrada, quando comparado aos demais algoritmos de agrupamento para FCDs que realizam detecção de *outliers*, devido a dificuldade de delimitar tais parâmetros.

O restante do artigo está organizado do seguinte modo: a Seção 2 discute os trabalhos relacionados. A Seção 3 descreve a estratégia proposta. A Seção 4 apresenta os experimentos e a discussão dos mesmos. A Seção 5 relata as conclusões e trabalhos futuros.

2. TRABALHOS RELACIONADOS

Em geral, os algoritmos para agrupamento de FCDs têm como estrutura básica duas componentes: i) *online* (ou abstração de dados), que sumariza os dados usando estruturas apropriadas, sem a necessidade de armazenar todos os dados do fluxo, e ii) *offline* (macro-agrupamento) que utiliza o sumário dos dados em conjunto com outras entradas do usuário para prover uma compreensão dos grupos quando requisitado [Silva et al. 2013]. Uma das estruturas comumente utilizadas para sumarizar os dados é chamada de micro-grupo (ou CF-Vector) (ver definição 3.1).

O *CluStream* [Aggarwal et al. 2003] é um dos algoritmos precursores na área de agrupamento em FCDs. Sua componente *online* mantém uma quantidade fixa de micro-grupos na memória e para cada novo objeto que chega ao fluxo é calculada a distância deste ao micro-grupo mais próximo do modelo. Caso sua distância esteja dentro do limite máximo desse micro-grupo, o objeto é inserido nele. Caso contrário, um novo micro-grupo é criado para absorver o novo objeto. Se a quantidade limite de micro-grupos estiver em seu máximo, a criação de um novo micro-grupo resulta na necessidade da exclusão do mais antigo ou, caso não seja possível, na união dos dois mais próximos. Para a criação dos micro-grupos iniciais uma versão adaptada do algoritmo k-médias é aplicada sobre os primeiros objetos do fluxo. Na fase *offline* um algoritmo de agrupamento, como por exemplo, o k-médias é executado sobre os micro-grupos do modelo para encontrar os grupos. O único tratamento de *outliers* do algoritmo *CluStream* ocorre no momento da exclusão de um grupo antigo, que por se tratar de um *outlier* não absorveu novos objetos do fluxo e portanto, se tornou obsoleto.

Outros algoritmos foram também propostos usando a estrutura de micro-grupos e com rotinas para tratamento de *outliers*. O *DenStream* [Cao et al. 2006] é um algoritmo baseado em densidade que trata *outliers*. Para isso, a componente *online* utiliza dois *buffers* denominados potenciais micro-grupos *(p-micro-grupos)* e *outliers* micro-grupos *(o-micro-grupos)*. Quando um novo objeto não consegue ser alocado no *p-micro-grupos*, ele é armazenado em *o-micro-grupos* até que seja promovido ou removido, que de fato o caracteriza como um *outlier*. Periodicamente uma análise sobre os *buffers* é realizada, buscando micro-grupos que devem ser promovidos a *p-micro-grupos* ou removidos dos *buffers*. Essa análise depende de parâmetros de entrada fornecidos pelo usuário, os quais são difíceis de serem identificados, especialmente em bases de dados reais.

D-Stream [Chen and Tu 2007] é um algoritmo de agrupamento baseado na densidade da grade. Possui a estrutura básica dos algoritmos de agrupamento, de forma que na componente *online* cada

objeto de entrada é mapeado em uma grade correspondente. Na componente *offline* são computadas as densidades de cada grade e as mesmas são agrupadas em relação a esses valores. Uma função de desvanecimento é utilizada para diminuir a densidade das grades com o tempo, se ela estiver abaixo de limiar definido pelo usuário e nenhum objeto foi adicionado desde a última verificação da densidade da grade, a mesma é descartada. O tratamento de *outliers* nesse algoritmo é realizado periodicamente por meio da remoção das grades que permaneceram marcadas como esporádicas, ou seja, possuem poucos objetos ou se tornaram muito antigas em relação ao *timestamp* atual, desde a última verificação. Caso essa verificação siga os parâmetros definidos pelos autores, o processo de análise das mesmas pode tornar-se computacionalmente custoso.

Alguns exemplos de algoritmos específicos para detecção de *outliers* em FCDs propostos na literatura são: *iLOF* [Salehi et al. 2015], *CORM* [Elahi et al. 2008], *Abstract-C* [Yang et al. 2009], *STORM* [Angiulli and Fassetti 2010], *MCOD* [Kontaki et al. 2011]. Em geral, esses empregam um critério para determinar se um objeto é um *outlier* baseado na distância [Aggarwal 2015]. Considerando que os objetos pertencem a um mesmo espaço, se existem menos que *minPts* objetos na vizinhança do objeto, o mesmo é denominado *outlier*. Entretanto, esses algoritmos não são empregados em conjunto com um algoritmo de agrupamento em FCDs, conforme é realizado na proposta em estudo.

3. ALGORITMO PROPOSTO: *CLUSTREAMOD*

Esse artigo apresenta o algoritmo *CluStreamOD* (*CluStream with Outlier Detection*), uma melhoria do algoritmo *CluStream* quanto a detecção de *outliers*. O *CluStreamOD* é capaz de detectar objetos *outliers*, na componente *online*, por meio da inclusão de uma memória auxiliar e uma etapa de validação interna aplicada sobre a mesma.

Os algoritmos 1 e 2 apresentam a fase *online* do *CluStreamOD* e o processo para análise, validação e remoção de elementos da memória temporária, respectivamente. Na fase *online* (algoritmo 1), assim como o *CluStream*, o *CluStreamOD* cria os q primeiros micro-grupos usando os M_{init} elementos dos FCDs. Para cada novo objeto dos FCDs o seu micro-grupo mais próximo é encontrado para absorvê-lo. Caso não seja possível, esse objeto vai para a memória temporária. Aqui reside a principal modificação em relação ao *CluStream*, o qual decide criar um novo micro-grupo quando nenhum dos existentes pode absorver o novo objeto. A estratégia usada pelo *CluStreamOD* é adicionar esse novo elemento a uma memória temporária (M_{aux}) para análise futura.

Definition 3.1. Dado que m seja um micro-grupo o mesmo possui as principais componentes: número de objetos ($m.N$), soma linear dos vetores de objetos $LS = \sum_{i=1}^{N} \mathbf{o_i}$ ($m.LS$), soma quadrática dos vetores de objetos $SS = \sum_{i=1}^{N} \mathbf{o_i}^2$ ($m.SS$), soma linear dos marcadores de tempo dos objetos $LST = \sum_{i=1}^{N} \mathbf{o_i}.\mathbf{t}$ ($m.LST$) e soma quadrática dos marcadores de tempo dos objetos $SST = \sum_{i=1}^{N} \mathbf{o_i}.\mathbf{t}^2$ ($m.SST$). Os componentes LS e SS são d dimensionais. Usando essas componentes é possível calcular o limite máximo de um micro-grupo e o seu centróide [Aggarwal et al. 2003].

O algoritmo 2 representa a análise de M_{aux}, a cada tp unidades de tempo, visando eliminar os objetos *outliers* e trazer para o modelo novos micro-grupos válidos, considerando que exista no mínimo min_{aux} objetos em M_{aux}. Neste trabalho o valor de min_{aux} foi definido empiricamente como 10. A análise feita sobre a M_{aux} passa por três etapas. A primeira é dada pelo agrupamento (G_{aux}) dos objetos de M_{aux} por meio do algoritmo k-médias++, com quantidade de grupos igual a k_{aux}. Para cada grupo G_l em G_{aux}, verifica-se a cardinalidade do grupo ($|G_l|$) e se essa for maior que *minPts*, a próxima etapa é executada. A segunda etapa consiste em aplicar a técnica de detecção local de *outliers* (LOF) [Breunig et al. 2000], para cada objeto o_y de G_l. A técnica apresenta como resultado uma pontuação (*score*) para cada objeto, determinando o seu grau de ser um *outlier*. A separação dos objetos em *inliers* e *outliers* é feita por meio de um valor de corte (*cuteLOF*) definido pelo usuário. Todos os objetos com *scores* menores ou iguais a *cuteLOF* são considerados *inliers* e inseridos em um potencial

Algorithm 1 Fase *online* algoritmo *CluStreamOD*

Entrada: q, M_{init}
$M \leftarrow k\text{-}médias++(q, M_{init})$ // gera os q primeiros micro-grupos
// sendo t o *timestamp* do objeto $\mathbf{o_t}$ do fluxo χ
para cada $\mathbf{o_t} \in \chi$ **faça**
    $j \leftarrow \arg\min_{m_j \in M}(dist(\mathbf{o_t}, m_j))$ // encontra o micro-grupo mais próximo
    se $dist(\mathbf{o_t}, m_j) < limiteMaximo(m_j)$ **então**
        adiciona($\mathbf{o_t}, m_j$)
    senão
        insere($\mathbf{o_t}, M_{aux}$) //insere objeto na memória temporária
    fim se
    $analise(M_{aux})$
fim para

Algorithm 2 Análise de M_{aux}

Entrada:$h, minPts, cuteLOF, tp$
se (t % $tp == 0$) e ($|M_{aux}| \geq min_{aux}$) **então**
    $k_{aux} \leftarrow \frac{|M_{aux}|}{minPts}$ // onde $|M_{aux}|$ é a cardinalidade de M_{aux}
    $G_{aux} \leftarrow k\text{-}médias++(k_{aux}, M_{aux})$ // obtém grupos sobre M_{aux}
    para cada $G_l \in G_{aux}$ **faça**
        $cont \leftarrow 0$
        se ($|G_l| > minPts$) **então**
            para cada $\mathbf{o_y} \in G_l$ **faça**
                se lof($\mathbf{o_y}$)$\leq cuteLOF$ **então**
                    $cont++$
                    adiciona($\mathbf{o_y}, m_{pot}$) // adiciona objetos em um potencial micro-grupo
                fim se
            fim para
        fim se
        se $cont > minPts$ **então**
            se (dispersao(m_{pot}) $\leq$ ($\max_{m_i \in M}$(dispersao(m_i)))) **então**
                $e \leftarrow \arg\min_{m_y \in M}(\frac{m_y.LST}{m_y.N})$
                se ($\frac{m_e.LST}{m_e.N} < (t-h)$) **então**
                    remove(m_e, M) //remove micro-grupo menos relevante
                senão
                    merge(M) // une dois micro-grupos mais próximos em M
                fim se
                adiciona(m_{pot}, M)
            fim se
        fim se
    fim para
fim se
// Verifique se está no momento de remover objetos em M_{aux}
se (t % $tp == 0$) **então**
    para ($\mathbf{o_y} \in M_{aux}$) **faça**
        se $\mathbf{o_y} < (t-h)$ **então**
            remove($\mathbf{o_y}, M_{aux}$)
        fim se
    fim para
fim se

micro-grupo (m_{pot}). Ao final dessa etapa, os grupos resultantes que contém apenas objetos *inliers* e possuem mais que *minPts* objetos são encaminhados para a última etapa.

Na etapa 3 é calculada a dispersão [Spinosa 2008] dos grupos resultantes da etapa 2, por meio da Equação 1. Para que m_{pot} seja válido e apto a ser inserido no modelo, é preciso que sua dispersão (dispersao(m_{pot})) seja menor ou igual a máxima dispersão dos micro-grupos do modelo (max(dispersao(M)). Caso verdadeiro, m_{pot} pode ser adicionado em M. Para adicionar um novo micro-grupo em M e manter a quantidade q fixa, é necessário que o micro-grupo menos relevante de M seja removido ou os dois micro-grupos mais próximos em M sejam unidos. Assim como no *CluStream*, um micro-grupo m_e só pode ser removido de M se o tempo de chegada médio de seus objetos ($\frac{m_e.LST}{m_e.N}$) for menor que (t - h). Ao final dessa verificação em M, m_{pot} é adicionado no modelo. Todos os objetos dos potenciais micro-grupos ou grupos que não satisfazem algumas das etapas de validação são mantidos em M_{aux} para um novo processo de validação interno ou até que se tornem menos relevantes e possam ser removidos.

A Equação 1 calcula a dispersão de um micro-grupo (m_i). A mesma é dada pela subtração entre a soma quadrada de todos os objetos pertencentes ao micro-grupo ($m_i.SS$) pelo tamanho do micro--grupo ($m_i.N$) e o quadrado da soma de todos os objetos pertencentes micro-grupo ($m_i.LS$) pelo tamanho do micro-grupo ($m_i.N$). Quanto maior o valor resultante, mais esparso é o micro-grupo.

$$dispersao(m_i) = \left(\frac{m_i.SS}{M_i.N} - \left(\frac{m_i.LS}{m_i.N} \right)^2 \right) \quad (1)$$

A remoção de objetos menos relevantes ou antigos para M ocorre na mesma periodicidade que a validação de M_{aux} e de forma semelhante à remoção do micro-grupo mais antigo de M. Esse processo é importante para evitar que objetos muito antigos que não representam mais o modelo M, sob o horizonte h, sejam em algum momento incluídos pelo processo de validação de M_{aux} em M, depreciando sua qualidade. A componente *offline* do *CluStreamOD* aplica o algoritmo k-médias++ sobre os micro-grupos do modelo M.

4. EXPERIMENTAÇÃO

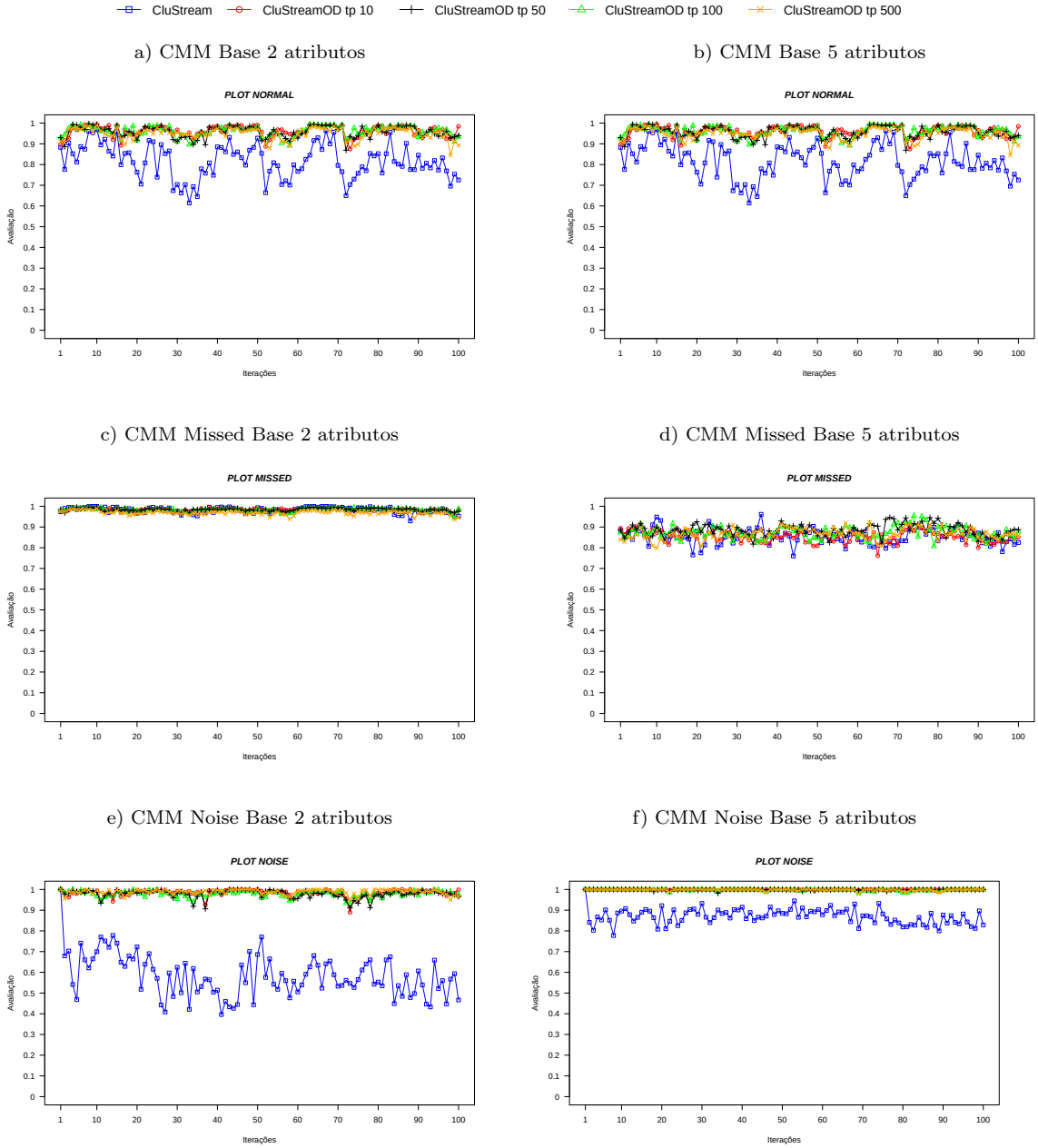
Foram comparados os algoritmos *CluStream* e *CluStreamOD* com a finalidade de medir o desempenho dos mesmos na detecção de *outliers* em FCDs. Ambos os algoritmos foram desenvolvidos em Java e comparados utilizando o arcabouço *MOA (Massive Online Analysis)*[Bifet and Kirkby 2009]. Duas bases de dados sintéticas com 2 e 5 atributos foram utilizadas, geradas à partir do próprio gerador MOA, que possibilita configurar o formato da evolução dos grupos ao longo do tempo. Cada base de dados possui 100.000 objetos, sendo que os primeiros 1.000 objetos, usados para a criação dos micro-grupos iniciais, estão livres de *outliers*. Cada base possui 5 grupos, sendo que novos grupos podem surgir ou desaparecer (mínimo de 2 e máximo de 8 grupos).

O algoritmo utilizado na fase *offline* do *CluStream* e *CluStreamOD* é o k-médias++. Nesse algoritmo o primeiro centróide é escolhido de forma aleatória e os demais são escolhidos a partir das probabilidades proporcionais aos objetos restantes. Essa inicialização é responsável pelo desempenho superior desse algoritmo em relação k-médias [Arthur and Vassilvitskii 2007].

Os experimentos foram conduzidos em um PC com Intel Core i5-4430 CPU 3.00GHz com 4GB de memória com Sistema Operacional Ubuntu 14.04 LTS. Foram utilizados os parâmetros de entrada padrão do *CluStream* implementados no *MOA* para ambos os algoritmos. Os parâmetros específicos do *CluStreamOD*, determinados de forma empírica, apresentam a seguinte configuração: *minPts: 3*, *tp: 10, 50, 100 e 500* e *cuteLOF: 1,5* (valor baseado em [Breunig et al. 2000]).

De maneira a garantir maior confiabilidade aos resultados, todos os experimentos foram executados 10 vezes para cada *tp* selecionado em cada base de dados. A métrica de avaliação externa selecionada foi a *CMM (Cluster Mapping Measure)* [Kremer et al. 2011], formada pelas componentes *Missed*,

Fig. 1: Análise da medida CMM e seus subconjuntos para as bases de 2 e 5 atributos



Misplaced e *Noise*, que indica de forma efetiva diferentes tipos de erros que devem ser considerados importantes no cenário de FCDs . Os valores médios de *CMM* e suas componentes *Missed* e *Noise* são apresentados na Figura 1.

Uma análise sobre a Figura 1 permite inferir que o *CluStreamOD*, para todas as medidas *CMM* e diferentes valores de *tp*, foi superior ou se manteve semelhante ao *CluStream*. Analisando a componente CMM *Noise* (figuras 1e e 1f) é possível ver que o *CluStreamOD* obtém desempenho superior ao *CluStream*, independente do valor de *tp* usado, ou seja, a estratégia para detecção de *outliers* proposta representa uma melhora no desempenho do algoritmo. Além disso, é possível perceber que a proposta para detecção de *outliers* não influenciou negativamente as demais etapas do algoritmo, o que pode ser confirmado analisando a componente CMM *Missed* (figuras 1c e 1d), na qual *CluStream*

e *CluStreamOD* obtém desempenho semelhante, e a medida geral CMM na qual *CluStreamOD* obtém melhor desempenho. Ainda que não apresentado por questões de espaço, a componente CMM *Misplaced* apresenta resultados semelhantes a CMM *Missed*.

Uma das características importantes a ser apresentada pelos algoritmos de agrupamento em FCDs é a capacidade de diferenciar *outliers* do surgimento de novos grupos, ou mudanças nos grupos existentes. A fim de fazer essa análise é importante observar se os exemplos *inliers*, em especial aqueles que representam mudanças nas características dos FCDs, não foram incorretamente classificados como *outliers*. O comportamento do *CluStreamOD* em relação a essa característica pode ser avaliado observando as tabelas Ia e Ib, que apresentam as proporções médias e desvios padrões, em parênteses, das 10 execuções realizadas. Ambas as tabelas mostram as proporções de inserção e remoção de objetos *inliers* e *outliers* em M_{aux} (memória temporária).

Table I: Média e desvio padrão das proporções de *inliers* e *outliers* em relação a M_{aux}

(a) Base 2 atributos

tp	*Inserção em* M_{aux}		*Remoção em* M_{aux}	
	Inliers	*Outliers*	*Inliers*	*Outliers*
10	0.33 (0.04)	0.94 (0.00)	0.94 (0.05)	0.07 (0.01)
50	0.38 (0.05)	0.95 (0.00)	0.93 (0.05)	0.07 (0.01)
100	0.41 (0.06)	0.95 (0.01)	0.92 (0.05)	0.07 (0.01)
500	0.57 (0.09)	0.97 (0.01)	0.89 (0.04)	0.07 (0.01)

(b) Base 5 atributos

tp	*Inserção em* M_{aux}		*Remoção em* M_{aux}	
	Inliers	*Outliers*	*Inliers*	*Outliers*
10	0.63 (0.10)	1.00 (0.00)	0.88 (0.06)	0.00 (0.00)
50	0.61 (0.07)	1.00 (0.00)	0.91 (0.03)	0.00 (0.00)
100	0.64 (0.08)	1.00 (0.00)	0.90 (0.04)	0.00 (0.00)
500	0.82 (0.08)	1.00 (0.00)	0.87 (0.03)	0.00 (0.00)

Analisando as Tabelas Ia e Ib, é possível constatar que, em média, mais de 94% dos *outliers* são identificados e inseridos na memória auxiliar. Em alguns casos, como Tabela Ib, todos os *outliers* são identificados, o que mostra que o método proposto consegue corretamente identificar os objetos *outliers*. Dessa proporção, no máximo 7% desses objetos conseguem retornar ao modelo pela aplicação do processo de validação sobre a memória temporária. Tal comportamento assegura que o processo de validação da memória auxiliar apresentado nesse artigo é apropriado para eliminar os *outliers*.

Considerando os objetos *inliers*, é possível constatar que, em média, até 82% desses objetos foram inseridos na memória temporária à medida que a quantidade de atributos e o valor de *tp* aumenta. É esperado que os objetos que representam mudanças com relação aos FCDs sejam primeiramente inseridos na memória temporária e futuramente retornem ao modelo quando novos exemplos similares a eles cheguem nos FCDs, possibilitando a criação de novos micro-grupos. É importante observar que em média, mais de 87% dos *inliers* retornam ao modelo pela aplicação do processo de validação na memória auxiliar. Esse comportamento assegura que o processo de validação da memória auxiliar apresentado nesse artigo é viável para manter os objetos *outliers* na memória auxiliar e promover os *inliers* ao modelo, mantendo-o atualizado e confiável ao longo do tempo.

5. CONCLUSÃO E TRABALHOS FUTUROS

Por meio da proposta e experimentos apresentados neste trabalho, podemos concluir que o algoritmo *CluStreamOD* é capaz de detectar os *outliers* do fluxo e manter o modelo constantemente atualizado e confiável para análise.

O uso da memória auxiliar para armazenamento dos objetos considerados *outliers* pelo algoritmo apresentou bom desempenho e tem auxílio significativo no processo geral de detecção de *outliers* proposto neste artigo. Porém, somente o uso dessa estrutura não é capaz de garantir a eficácia do mesmo. Sendo assim, fez-se necessário o uso das estratégias de validação aplicadas sobre a mesma com a finalidade de validar os objetos *inliers*, considerados em um dado momento *outliers*, para serem inseridos no modelo. Essa promoção dos objetos ao modelo garante que o mesmo se mantenha atualizado e confiável, proporcionando ao usuário uma análise de confiança sobre o comportamento do fluxo ao longo do tempo.

Tais resultados são relevantes para a problemática discutida nesse artigo e instiga ao desenvolvimento de novos trabalhos futuros sobre a proposta em estudo como: verificar o comportamento do algoritmo *CluStreamOD* em conjuntos de dados reais com o intuito de garantir maior confiabilidade ao mesmo; verificar alternativas para diminuição dos parâmetros de entrada eliminando os não-críticos; comparar o algoritmo *CluStreamOD* a algoritmos que possuem o processo de detecção de *outliers* em conjunto com o processo de agrupamento, com a finalidade de analisar o impacto causado pela quantidade de parâmetros de entrada sobre a acurácia dos algoritmos.

AGRADECIMENTOS

Esse trabalho foi subsidiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG).

REFERENCES

Aggarwal, C. C. A framework for diagnosing changes in evolving data streams. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*. SIGMOD '03. ACM, New York, NY, USA, pp. 575–586, 2003.

Aggarwal, C. C. Outlier analysis. In *Data Mining*. Springer, Springer, New York, USA, pp. 237–263, 2015.

Aggarwal, C. C., Han, J., Wang, J., and Yu, P. S. A framework for clustering evolving data streams. In *Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29*. VLDB '03. VLDB Endowment, Berlin, Germany, pp. 81–92, 2003.

Angiulli, F. and Fassetti, F. Distance-based outlier queries in data streams: the novel task and algorithms. *Data Mining and Knowledge Discovery* 20 (2): 290–324, 2010.

Arthur, D. and Vassilvitskii, S. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, pp. 1027–1035, 2007.

Bifet, A. and Kirkby, R. Data stream mining a practical approach, 2009.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. Lof: identifying density-based local outliers. In *ACM sigmod record*. ACM, ACM, New York, NY, USA, 2000.

Cao, F., Ester, M., Qian, W., and Zhou, A. Density-based clustering over an evolving data stream with noise. In *In 2006 SIAM Conference on Data Mining*. SIAM, Bethesda, Maryland, USA, pp. 328–339, 2006.

Chen, Y. and Tu, L. Density-based clustering for real-time stream data. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, ACM, San Jose, CA, USA, pp. 133–142, 2007.

Elahi, M., Li, K., Nisar, W., Lv, X., and Wang, H. Efficient clustering-based outlier detection algorithm for dynamic data stream. In *Fuzzy Systems and Knowledge Discovery, 2008. FSKD'08. Fifth International Conference on*. Vol. 5. IEEE, IEEE, Jinan, Shandong, China,, pp. 298–304, 2008.

Guha, S., Meyerson, A., Mishra, N., Motwani, R., and O'Callaghan, L. Clustering data streams: Theory and practice. *IEEE Trans. on Knowl. and Data Eng.* 15 (3): 515–528, Mar., 2003.

Kontaki, M., Gounaris, A., Papadopoulos, A. N., Tsichlas, K., and Manolopoulos, Y. Continuous monitoring of distance-based outliers over data streams. In *2011 IEEE 27th International Conference on Data Engineering*. IEEE, IEEE, Hannover, pp. 135–146, 2011.

Kremer, H., Kranen, P., Jansen, T., Seidl, T., Bifet, A., Holmes, G., and Pfahringer, B. An effective evaluation measure for clustering on evolving data streams. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, ACM, San Diego, CA, USA, pp. 868–876, 2011.

Salehi, M., Leckie, C., Bezdek, J. C., and Vaithianathan, T. Local outlier detection for data streams in sensor networks: Revisiting the utility problem invited paper. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on*. IEEE, IEEE, Singapore, pp. 1–6, 2015.

Silva, J. A., Faria, E. R., Barros, R. C., Hruschka, E. R., Carvalho, A. C. P. L. F. d., and Gama, J. a. Data stream clustering: A survey. *ACM Comput. Surv.* 46 (1): 13:1–13:31, July, 2013.

Spinosa, E. J. *Detecção de novidade com aplicação a fluxos contínuos de dados*. Ph.D. thesis, Universidade de São Paulo, 2008.

Yang, D., Rundensteiner, E. A., and Ward, M. O. Neighbor-based pattern detection for windows over streaming data. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*. EDBT '09. ACM, New York, NY, USA, pp. 529–540, 2009.

Automatic Ontology Generation for the Power Industry The Term Extraction Step

Alexandra Moreira[1], **Jugurta Lisboa Filho**[1], **Alcione de Paiva Oliveira**[1]

[1]Departamento de Informática
Universidade federal de Viçosa
Campus – 36570-900 –Viçosa – MG – Brazil

Abstract. *The development of an ontology is a long and laborious process that involves many hours of manual labor. The effort applied in this process can be alleviated if automated tools are employed. Particularly, the use of tools derived from computational linguistics may be helpful in this task. However, there is a shortage of these tools above all for the Portuguese language. This article describes an automatic tool for extracting noun phrases that can be adopted as terms for a certain domain. Also the tool creates a two levels hierarchy of terms. This hierarchy can be used as an first step for creating the ontology. The tool described in this article was used to select terms to be used in an ontology for the power sector domain. It showed better performance when compared to other tools developed for the Brazilian Portuguese.*

Resumo. *O desenvolvimento de uma ontologia é um processo longo e trabalhoso, envolvendo muitas horas de trabalho especializado. O esforço aplicado no processo pode ser aliviado se forem empregadas ferramentas automatizadas. Particularmente, poderia ser de grande utilidade a utilização de ferramentas oriundas de linguística computacional. No entanto, existe uma escassez dessas ferramentas,principalmente para o idioma Português. Este artigo descreve uma ferramenta automática para extrair sintagmas nominais que podem ser adotados como termos de um determinado domínio. A ferramenta, também foi usada para criar uma hierarquia dois níveis de termos. Esta hierarquia pode ser utilizada como um primeiro passo para criar o ontologia. A ferramenta descrita foi usada para selecionar termos a serem utilizados em uma ontologia para o domínio do sector de elétrico e mostrou um desempenho superior quando comparada com outras ferramentas de extração de sintagmas nominais para o idioma Português Brasileiro.*

1. Introduction

Ontologies have become an essential resource for systems related to the area of information processing. They establish a common meaning to the terms of a domain and are used for different tasks, such as enabling communication between computer systems and people, information extraction and natural language processing in general [Moreira et al. 2004]. However, the process of creating an ontology is complex, involving many hours of manual work and requires prior knowledge of the domain [Sanchez and Moreno 2004]. Any tool that might help tackle the process and increase its automation is welcome.

Particularly, most of the time spent at the ontology creation is devoted to the initial phase, which involves the selection of the ontology concepts. Aiming at minimizing this problem, some tools are being developed, using technology derived from the natural language processing (NLP) and data mining area (see section 2). However, the lack of such resources target for the Portuguese language still hinders the automation of the process of creating ontologies in Portuguese.

This paper describes an automated tool, named EχTerm, for extracting noun phrases that can be adopted as terms for a certain domain. Also the tool creates a two levels hierarchy of terms. This hierarchy can be used as an embryo for creating the ontology. It showed better performance when compared to other tools with similar goal that have been developed for the Brazilian Portuguese. The tool is currently being used in the process of creating an ontology for the electricity sector.

It is important to emphasize the difference between noun phrases (NPs) and terms. NPs are syntactic structures whose semantic component basically indicate that they refer to entities in a discourse. On the other hand, terms are words or noun phrase that have a specific meaning in a particular language in a particular area. That is, they are used to define a concept in a specific domain and they have the syntactic and semantic aspects better defined. Having said that, a noun phrase is a candidate domain term that must pass the scrutiny of the expert.

This paper is organized as follows: the next section presents the related research previously developed that are related to this research; Section 3 describes the extraction process adopted; Section 4 presents the results obtained; and Section 5 presents the final remarks.

2. Related Work

Term and concepts extraction from a textual database is a very active area of research and there are several projects being developed. [Lopes et al. 2009] are developing a software whose ultimate goal is the automatic generation of ontologies. The software must be applied to a previously syntactically annotated corpus by the software PALAVRAS [Bick 2000]. At the moment, the software extracts the terms of corpus annotated that belongs to the noun phrases category. From this point, it does some processing in order to eliminate candidates terms that are not relevant to the treated area. Our proposal differs from that presented by Lopes and her colleagues in the sense that our approach is more bottom-up. That is, rather than starting from noun phrases, we start from annotated lexemes with their syntactic classes or Part-Of-Speech (POS). From this point the algorithm aggregates the lexemes to the point that compose a relevant terminology unit for the domain.

Carvalheira [Carvalheira 2007] does not create a term extractor, but he proposes a semi automatic method for creating ontologies, which is our goal as well. The method uses linguistic and statistical resources to extract concepts and relations candidates to compose the ontology. However, the method does not eliminate the participation of a expert to determine which terms actually should be incorporated in the ontology to be built. In our case, we aim to minimize the involvement of people, setting automatically, the concepts and relationships between concepts in a two-level hierarchy. Another distinguishing feature is that the work [Carvalheira 2007] extracts terms using the Brown

corpus containing American English texts. Our work focuses on Brazilian Portuguese, where there is a shortage of linguistic resources and a small number of proposals.

Teline [Teline et al. 2003] developed the term extractor called Exporter (Evaluation of Terminology Authomatic Extraction Methods for Portuguese Texts). The system was used in the BLOC-Eco project [Zavaglia et al. 2007], whose goal was to create a knowledge base with the ontological information about ecology terms in Brazilian Portuguese. The system is based on POS annotation and syntactic sequence patterns for unigrams, bigrams and trigrams. For example, one of the patterns for trigram would be `<Noun Preposition Adjective>`. Our study differs from the Exporter system as it does not have a limit on the size of the composed terms and for not having a fixed number of syntactic class sequence patterns.

Batista [Batista 2011] used natural language processing tools to extract definitions contained in a text. He also had difficulties related with finding corpus in Brazilian Portuguese as well as tools available for data extraction. In addition, he used the Mac-morpho *corpus* and NLTK framework, similar choices that were adopted in the present study. In this case, the work was more focused on the extraction of settings than in NPs in general, and this is the main distinction in relation to this work.

Macken et al. [Macken et al. 2013] have created a bilingual terminology extraction system, called TExSIS, that uses a chunk based alignment method for the generation of candidate terms. The technique proposed requires multilingual *corpus* to perform the alignment of text segments. This fact distinguishes this research from the one proposed in this paper. Furthermore, the technique has not been tested in Portuguese.

Maia and Souza [Maia and Souza 2010] developed the software tool, named Ogma, to extract noun phrases from texts written in Portuguese language. The aim of the authors was to check use the noun phrases as indexers for classifying documents. The Ogma tool is an extractor based on rules and is available on the Web[1]. The current research results will be compared with the results obtained with the Ogma tool.

3. Applied Process

The term extraction process takes as its starting point a corpus annotated with lexemes according to their syntactic classes. To carry out the annotation we used the annotator Unigran Tagger from the NLTK package (Natural Language Toolkit)[2] trained with the Mac-Morpho corpus [Aluísio et al. 2003]. After this step the resulting corpus undergoes an annotation adjustment step to reduce mistakes in the annotation process. For this purpose, the adjusting module uses a word database extracted from Probank.BR [Duran and Aluísio 2011], and a Brazilian proper names and Locations list.

The next step is the most important and it is the heart of the system. It is the one executed by the module that performs the junction of the lexemes in order to create compound words or NPs. The joining is performed by a set of rules which basically combine separate nominal that may be united by prepositions and co-occurring adjectives. The rules adopted to detect NPs are as follows:

$$sn \rightarrow N(N \mid ADJ \mid < SPREP >)^*$$

[1]http://www.luizmaia.com.br/ogma/

[2]http://www.nltk.org/

$$SPREP \rightarrow PREP < SN >$$

Thus, the set of annotated terms

```
('nível', 'N'), ('dos', 'PREP'), ('reservatórios', 'N'),
('das', 'PREP'), ('usinas', 'N'), ('hidrelétricas', 'N')
```

becomes

```
('nível dos reservatórios das usinas hidrelétricas', 'N').
```

The last part of the process is to create a two-level hierarchy for the purpose of grouping the terms and facilitating the creation of an ontology. The terms have been grouped into classes named by the lemma form of the first lexeme of the term. The leftmost nominal in a noun phrase usually is the head of the phrase in Portuguese. In order to obtain the lemma form we used the lemmatizer available in NILC website (Inter-Institutional Center for Computational Linguistics USP)[3]. The steps are summarized in Fig.1 and make up the system known as EχTerm.

An excerpt from the final file issued by the process can be seen in Fig.2. It is the result of applying the system to an annual report of a company for generating and transmitting electrical energy. As can be seen, the system was able to extract noun phrases from the company's annual report, that even for a non-specialist, appear to be related to the domain. For example, the Fig. 3 shows a segment of a concept hierarchy for power sector market that could accommodate the NPs related to the market extracted by the system. The hierarchy was prepared manually, but the ultimate aim of this research is to build the concept hierarchy automatically .

4. Results

Ideally, to assess whether the proposed process is a viable alternative for term extraction, it would be necessary to compare their performance against other tools when applied to a pair of "corpus $\times$ terms" established in advance. However, as already mentioned, there is a lack of both tools and corpus of tests for the Brazilian Portuguese. In the scientific literature of the area one can find some tools for extracting terms from textual basis, however, the tools are not available for use. One such tool is the EχATOLP, [Lopes et al. 2009]. The tool is not yet available for download, but it was possible compare against it using the results obtained from the application of the tool over a previously used *corpus*. In this case, the glossary of terms of the Brazilian national electric energy agency-ANEEL[4] was used as a test *corpus*. This *corpus* was used only for comparison purposes between the tools and not with the aim of generating a set of terms for the elaboration of the ontology. The terms extracted by the two tools were compared with a total of 1403 terms selected by an expert independently. The results can be seen in Table1.

In Table1 one can see that the number of terms extracted by the EχTerm system is 65% greater than the number of terms extracted by the EχATOLP software. This can be

[3]http://www.nilc.icmc.usp.br/nilc/index.php

[4]http://www.aneel.gov.br/biblioteca/glossario.cfm

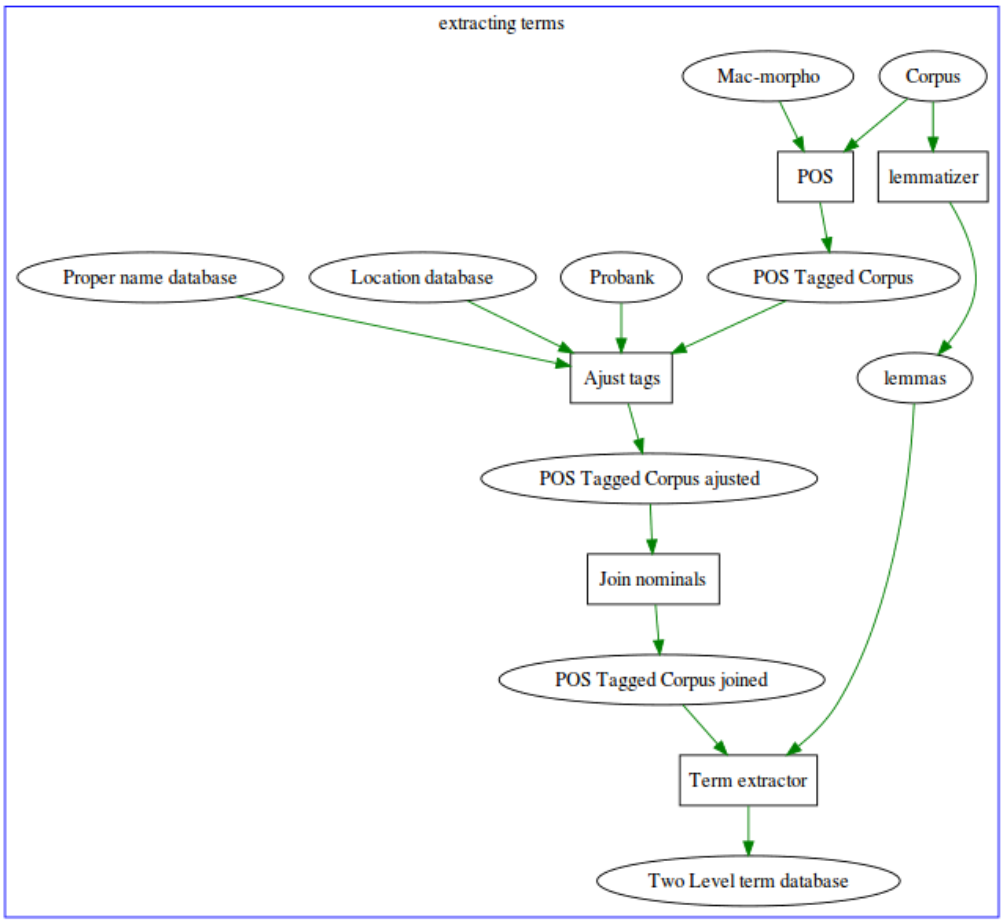


Figure 1. Steps in the term extraction process of the (EχTerm system). The system operates in pipeline fashion, where each module receives as input the output of the previous stage.

```
mercado      Mercado de Capitais
             Mercado de Energia
             mercado de capitais
             mercado de energia
             mercado do Grupo
             mercados
meta         meta de treinamento
             metas de crescimento
             metas de indicadores
             metas
metodologia  metodologia de cálculo para definição valores
             metodologia de cálculo das tarifas de fornecimento
             metas
```

Figure 2. A small segment of the initial grouping of extracted terms issued by the EχTerm system.

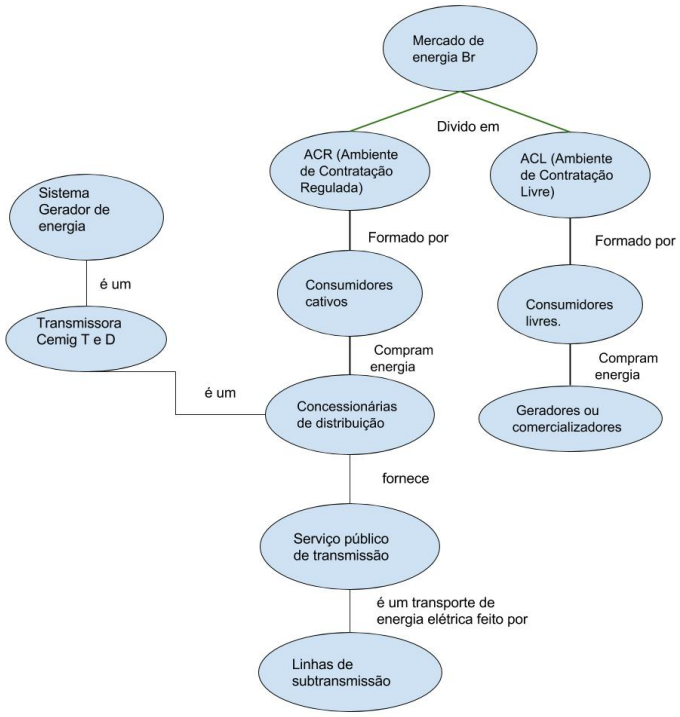


Figure 3. Segment of the relations between the concepts of electricity market domain. This concept hierarchy can be used to accommodate the NPs extracted by the system.

Tool	N. extracted terms	match w/ human	Precision P	recall R	$F1 = \frac{2PR}{P+R}$
EχATOLP	2688	252	9,38%	17,96%	0,1232
EχTerm	4114	345	8,39%	24,59%	0,1251

Table 1. Results of the application of the system on the ANEEL glossary. The performance measures were low for both systems, probably due to the fact that the *corpus* was restricted to the ANEEL glossary.

attributed to the difference in approach between the two systems. The approach to take as a basis the individual nominal and gradually build the compound terms has a smaller granularity than the approach that uses noun phrases, adopted by EχATOLP. This tends to increase the system recall. The disadvantage of this approach is the lower precision rate. The F-measure index ($F1$), which produces a balanced measure between precision and recall points that the two approaches obtained virtually identical results. The precision of the two systems was low, but it is necessary to point out that the elaboration of the terms made by the human was made using several documents, i.e. was not restricted to terms occurring in the ANEEL glossary. Furthermore, one must remember that the expert selected terms and not NPs, which is what is emitted by the systems. This semantic distinction is largely responsible for the low score.

Despite serving as a point of comparison the above analysis does not portray the true potential of the tool as an extractor of NPs. The lack of access to the EχATOLP tool forced a comparison using an unsuitable *corpus*. In order to carry out a test to show

the true potential of the tool it was made a comparison with another tool, the Ogma tool [Maia and Souza 2010], using a more appropriate corpus. To verify the precision and recall the results were compared with a list of terms extracted from the corpus manually by an expert. Is worth mentioning that we compared the output of the tools, that issue a list of noun phrases, with the list of terms prepared by the expert. The expert has prepared a list of 142 terms.

Tool	N. extracted terms	match w/ human	Precision P	recall R	$F1 = \frac{2PR}{P+R}$
Ogma	211	19	9,0%	13,38%	10,76
EχTerm	132	49	37,12%	34,5%	35,76

Table 2. Comparison of results with Ogma tool. Although they use similar techniques to show the proposed system superior in all performance measures. The comparison was made with a list 142 of terms produced by a specialist from the same text used by automatic extractors.

In Table2 one can see that the proposed tool has a much higher performance than Ogma tool. It was higher in both precision and recall and the F-measure strongly reflects this superiority. We believe the reason for this is that the rules of Ogma tool produce NPs that include some elements that do not belong to a term. For example, the tool issued the noun phrase “os dados financeiros” (the financial data) and, in this case, the more appropriate term candidate would be “dados financeiros” (financial data). Furthermore, the tool produces some spurious noun phrases, such as “às suas” (to their) and does not join some nominal to form a compound term, such as “empresa” and “controladora” to form “empresa controladora” (controlling company).

5. Conclusions

Satisfactory results for the execution of tools as described in this work depends largely on the performance of more basic tools on which they rely upon. These more basic tools would be the lemmatizers, POS taggers, Stemmers and, parsers. Despite the efforts of some Brazilian research groups to provide these tools, there is still a lack of basic tools for the Brazilian Portuguese and there is room for improvement in this area. The results obtained in this study suggest that a bottom-up approach for extracting terms can increase the recall without a great loss of precision. However, more tests and improvements are needed in order to reduce the terms extraction process dependency of a human expert.

It was hard to find tools available to perform a comparison. Only one was found to download and the results showed that the tool proposed in this paper has an notably superior performance. In the case of the second tool used for comparing, the tests were conducted indirectly through a previously obtained output. It was done this way by the lack of access to the tool. Due to an inappropriate corpus, both tools have a low performance, but the proposed tool showed better recall rate.

The next step is to place a statistical filter trained with a larger *corpus* to filter out spurious terms and improve the accuracy of the tool. It is intended to use the terms extracted by the tool to feed a tool for automatic creation of ontologies that is already under development.

Acknowledgments.

This research is supported in part by the funding agencies CEMIG, FAPEMIG, CNPq, and CAPES.

References

Aluísio, S., Pelizzoni, J., Marchi, A. R., de Oliveira, L., Manenti, R., and Marquiafável, V. (2003). An account of the challenge of tagging a reference corpus for brazilian portuguese. In *Computational Processing of the Portuguese Language*, pages 110–117. Springer.

Batista, A. H. (2011). Extração automática de definições: um estudo de caso em textos legislativos. Master's thesis, Universidade Católica de Brasília.

Bick, E. (2000). *The Parsing System "Palavras": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Aarhus Universitetsforlag.

Carvalheira, L. C. d. C. (2007). *Método semi-automático de construção de ontologias parciais de domínio com base em textos.* PhD thesis, Universidade de São Paulo.

Duran, M. S. and Aluísio, S. M. (2011). Propbank-br: a brazilian portuguese corpus annotated with semantic role labels. In *8th Brazilian symposium in information and human language technology*, pages 164–168.

Lopes, L., Fernandes, P., Vieira, R., and Fedrizzi, G. (2009). Eχatolp–an automatic tool for term extraction from portuguese language corpora. In *Proceedings of the 4th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics (LTC), Faculty of Mathematics and Computer Science of Adam Mickiewicz University*, pages 427–431.

Macken, L., Lefever, E., and Hoste, V. (2013). Texsis: Bilingual terminology extraction from parallel corpora using chunk-based alignment. *Terminology*, 19(1):1–30.

Maia, L. C. G. and Souza, R. R. (2010). Uso de sintagmas nominais na classificação automática de documentos eletrônicos. *Perspectivas em Ciência da Informação*, 15(1):154–172.

Moreira, A., Alvarenga, L., and Oliveira, A. P. (2004). O nível do conhecimento e os instrumentos de representação: tesauros e ontologias. *DataGramaZero-Revista de Ciência da Informação*, 5(6).

Sanchez, D. and Moreno, A. (2004). Creating ontologies from web documents. *Recent Advances in Artificial Intelligence Research and Development. IOS Press*, 113:11–18.

Teline, M. F., Almeida, G., and Aluisio, S. M. (2003). Extração manual e automática de terminologia: Comparando abordagens e critérios. In *16th Brazilian Symposium on Computer Graphics and Image Processing-SIBGRAPI*.

Zavaglia, C., Aluísio, S., Nunes, M. G. V., and Oliveira, L. (2007). Estrutura ontológica e unidades lexicais: uma aplicação computacional no domínio da ecologia. In *Proceedings of the 5th Workshop in Information and Human Language Technology*, pages 1575–84.

Identifying Locomotives' Position in Large Freight Trains: An investigation with Machine Learning and Fuel Consumption

Helder Arruda[1], Gustavo Pessin[1], Orlando Ohashi[2], Jair Ferreira[1], Cleidson de Souza[1,3]

[1] Vale Institute of Technology, Brazil
{helder.arruda, jair.ferreira}@pq.itv.org, gustavo.pessin@itv.org
[2] Amazon Federal Rural University, Brazil
orlando.ohashi@ufra.edu.br
[3] Federal University of Pará, Brazil
cleidson.desouza@acm.org

Abstract. Accurate identification of locomotives' position in large freight trains is important due to maintenance and management aspects. Current solutions employ RFIDs, image cameras or GPS, while the first two are expensive, the third is not an off-the-shelf hardware for all locomotives. In this paper we investigate a data driven solution to automatically identify locomotives' position in large freight trains. We take into account off-the-shelf hardware alone (that gather instant fuel consumption) seeking for a less expensive solution. We evaluate different machine learning approaches and algorithms and different inputs attributes, achieving significant results.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning; I.5.1 [**Pattern Recognition**]: Neural Nets; J.7 [**Computers in Other Systems**]: Industrial Control

Keywords: railway, locomotive, fuel consumption, machine learning, random forest, multilayer perceptron

1. INTRODUCTION

Long freight trains can use several locomotives. These locomotives might be distributed in different positions along the composition. Accurate identification of a locomotive position in large freight trains is important due to maintenance and management aspects. For instance, the erosion will be different on the wheels of the locomotives depending on their position. Solutions employing RFIDs or image cameras [Martin 2008; Zhijun et al. 2008] are expensive and are not off-the-shelf hardware in the locomotives. GPS is another potential solution, although, it is also a not off-the-shelf hardware for all locomotives. In this paper we investigate a data driven solution to automatically identify locomotives' position in large freight trains. We take into account off-the-shelf hardware alone, that gather instant fuel consumption, seeking for a less expensive solution. The gathered instant fuel consumption is a time series of fuel consumption in a given part of the railroad. We evaluate different machine learning approaches (multi and binary-class classifiers) and two machine learning algorithms: random forest and multi-layer neural nets. Furthermore, we take into account that the time series can be transformed in attributes using different spatial resolutions; hence, we evaluate attributes taking into account 3 different spatial resolutions for the machine learning algorithms.

2. MATERIALS AND METHODS

The train which we refer is composed by 4 locomotives and 330 train wagons (Fig. 1). This kind of formation is a default adopted by Vale S.A. [Matos et al. 2015]. The positions of the locomotives, in order, are: Leader, Commanded, Remote B and Remote C. We employ a dataset from large freight trains from Vale S.A., which has the largest trains in operation in the world with 330 wagons, 3.3 kilometers long, and usually 4 to 6 locomotives. The Carajás Ridge presents the largest iron ore deposit in the world [Piló et al. 2015]. From the mine to the harbor, the Carajás railway has 897 km

and crosses two of the largest Brazilian states, Pará and Maranhão, connecting Carajás ridge to Ponta da Madeira harbor.

We obtained diesel consumption from 3 Vales' locomotives from May 2014 to December 2015. This data was integrated with information from the Vale operational system which describes the locomotive position in the train. We have 3 locomotives with a total of 63 trips. A locomotive could be a Leader, Commanded, RemoteB or RemoteC, from these trips we have, respectively 5, 11, 16 and 31 trips.

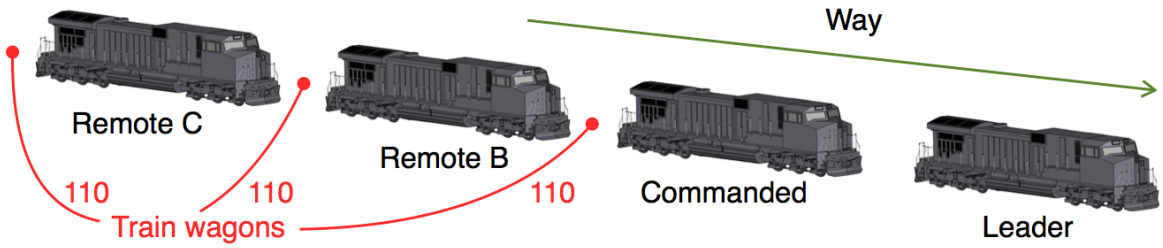


Fig. 1. The train formation composed by 4 locomotives and 330 train wagons. The locomotives' positions from the beginning to the end of the train are: Leader, Commanded, Remote B and Remote C. There are 110 train wagons between Commanded and Remote B, other 110 train wagons between Remote B and Remote C, and 110 more train wagons afer Remote C.

We employed a subset of 36 km of each trip to perform the classification because we had access only to a subset of the trips. Noteworthy to point out that, for all trips, the same 36 km were employed. The data coming from the monitored locomotives are in plain text format and require preprocessing before being imported into a database. The R programming language [Lantz 2013] was used to deal with raw data and to populate a PostgreSQL database [Vohra 2016]. Once the data was stored in the database, SQL queries embedded in Python scripts helped the extraction of feature vectors to be used in machine learning algorithms [Garreta and Moncecchi 2013; Richert and Coelho 2013]. A feature vector with the mean, standard deviation, trend, sum, and median of fuel total consumption, was used instead of raw data according to the best results obtained using this kind of approach in previous work [Furquim et al. 2014]. The algorithms used are part of Weka software [Hall et al. 2009]. As the chosen stretch is 36 km long, three spatial resolutions were used: 4 stretches of 9 km, 6 stretches of 6 km and 12 stretches of 3 km. The first comparison involved all the classes (Fig. 2). The second comparison used each of the classes against the rest (Fig. 3), which is a binary classification [Lorena and de Carvalho 2010].

Two machine learning methods were used in the comparison: random forest (RF) classifier and a bagging of multilayer perceptron (MLP) neural networks. Both methods were executed using cross validation with leave-one-out technique, due to the small amount of data [Cawley and Talbot 2003].

3. RESULTS AND DISCUSSION

The approach using binary classification showed better results for all the spatial resolutions. For RemoteB locomotives, RF was better for the 3 km spatial resolution (median), and the MLP was better in 6 and 9 km (Figure 3(c)). In the case of RemoteC locomotives against all, both the spatial resolutions of 6 and 9 km produced 95.24% accuracy for the MLP, for each one of the 30 executions. The RF presented the larger accuracy of 96.83%, also in 6 and 9 km. The best result of RemoteC was in 3 km resolution with 98.41% accuracy for the MLP (Figure 3(d)). Although the use of binary classifiers showed better results than the multi-classifier, we did not find a pattern related to the spatial resolutions.

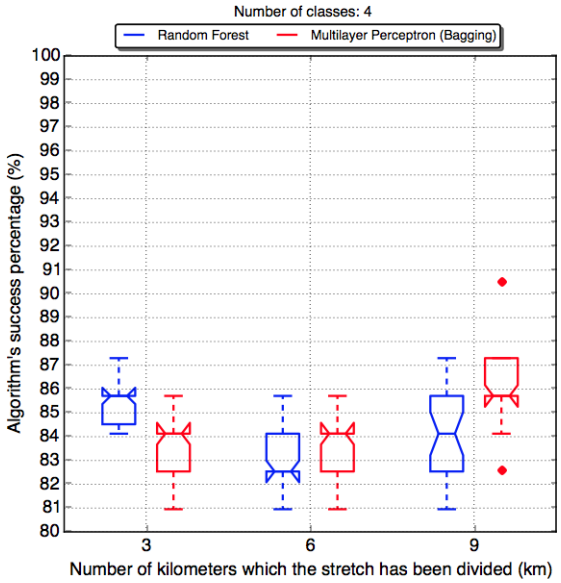


Fig. 2. The boxplots shows the comparison among 4 classes of locomotives (Leader, Commanded, RemoteB and RemoteC). There are 2 boxplots for each spatial resolution (3 km, 6 km and 9 km), as shown in the x axis. The blue ones indicate the Random Forest algorithm and the red ones show the Multilayer Perceptron Bagging. In the y axis are the percentual of success.

4. ACKNOWLEDGMENTS

The authors would like to thank CNPq (process numbers 440880/2013-0 and 310468/2014-0).

REFERENCES

Cawley, G. C. and Talbot, N. L. C. Efficient leave-one-out cross-validation of kernel Fisher discriminant classifiers. *Pattern Recognition* 36 (11): 2585–2592, 2003.

Furquim, G., Neto, F., Pessin, G., Ueyama, J., Albuquerque, J. P. D., Clara, M., Mendiondo, E. M., Souza, V. C. B. D., Souza, P. D., Dimitrova, D., and Braun, T. Combining Wireless Sensor Networks and Machine Learning for Flash Flood Nowcasting. In *Proceedings - 2014 IEEE 28th International Conference on Advanced Information Networking and Applications Workshops, IEEE WAINA 2014*. IEEE, Victoria, Canada, pp. 67–72, 2014.

Garreta, R. and Moncecchi, G. *Learning scikit-learn: Machine Learning in Python*. Packt Publishing, Birmingham, UK, 2013.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. The WEKA Data Mining Software: An Update. *SIGKDD Explorations Newsletter* 11 (1): 10–18, 2009.

Lantz, B. *Machine Learning with R*. Packt Publishing, Birmingham, UK, 2013.

Lorena, A. C. and de Carvalho, A. C. P. L. F. Building binary-tree-based multiclass classifiers using separability measures. *Neurocomputing* 73 (16-18): 2837–2845, 2010.

Martin, S. E. Overview of the Operation for the Proposed Locomotive Worker Identification with Cameras. In *2008 IEEE International Conference on Technologies for Homeland Security, HST'08*. IEEE, Boston, pp. 351–352, 2008.

Matos, J. C. L., Branco, V. H. L., Macêdo, A. N., and Oliveira, D. R. C. Structural assessment of a RC Bridge over Sororó river along the Carajás railway. *IBRACON Structures and Materials Journal* 8 (2): 140–163, 2015.

Piló, L. B., Auler, A. S., and Martins, F. Carajás National Forest: Iron Ore Plateaus and Caves in Southeastern Amazon. In *Landscapes and Landforms of Brazil*. Springer, Brasilia, pp. 273–283, 2015.

Richert, W. and Coelho, L. P. *Building Machine Learning Systems with Python*. Packt Publishing, Birmingham, UK, 2013.

Vohra, D. Using PostgreSQL Database. In *Kubernetes Microservices with Docker*. Apress, Berkeley, pp. 115–139, 2016.

Zhijun, S., Zhongpan, Q., and Shaozi, L. The Application of UHF RFID Technology in Mine Locomotive Positioning System. In *Proceedings of 2008 IEEE International Symposium on IT in Medicine and Education*. IEEE, Xiamen, China, pp. 1079–1082, 2008.

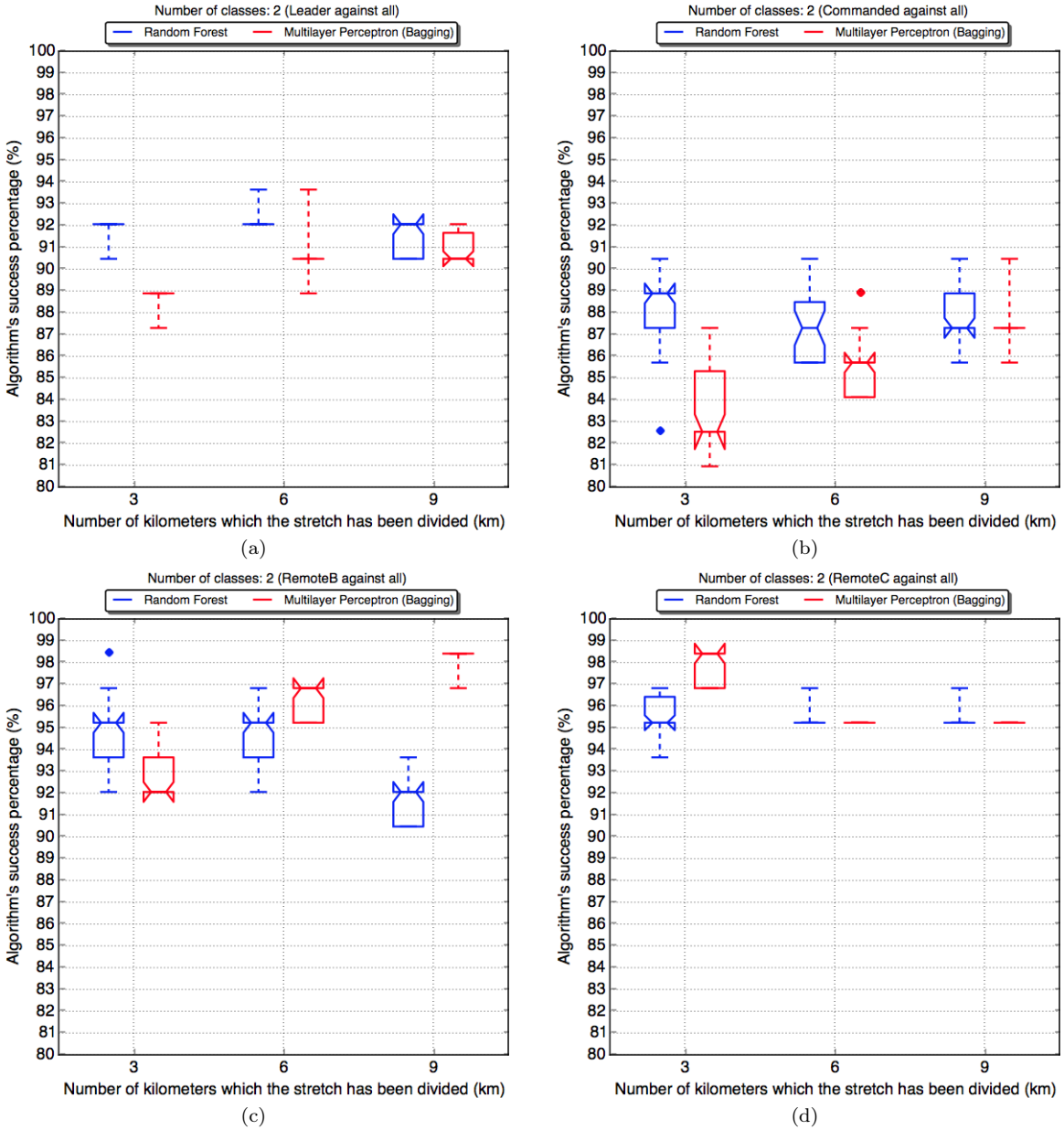


Fig. 3. Each one of the locomotive's positions against the others, featuring comparisons among 2 classes. Figure 3(a) shows Leader (5 trips) against all (58 trips). Figure 3(b) shows Commanded (11 trips) against all (52 trips). Figure 3(c) shows RemoteB (16 trips) against all (47 trips). Figure 3(d) shows RemoteC (31 trips) against all (32 trips).

Mineração de Opiniões: Um Classificador Ternário ou Dois Binários?

C. A. Fernandes Filho[1,2], J. Carvalho[1,3], A. Plastino[1]

[1] Universidade Federal Fluminense, Niterói, RJ, Brasil
{caugusto,joncarv,plastino}@ic.uff.br
[2] Instituto Federal do Rio de Janeiro, Rio de Janeiro, RJ, Brasil
[3] Instituto Federal Fluminense, Itaperuna, RJ, Brasil

Abstract. Mineração de opiniões é a área que avalia automaticamente opiniões expressas em textos publicados em blogs, redes sociais, microblogs etc. O Twitter é um microblog que permite a publicação de mensagens curtas, chamadas *tweets*, e tem sido foco da aplicação de mineração de opiniões em muitos trabalhos. Com o objetivo de classificar a opinião expressa em *tweets* quanto à sua polaridade, ou seja, se são opiniões positivas ou negativas, alguns trabalhos realizam a classificação em apenas uma etapa, enquanto outros utilizam uma estratégia que consiste em duas etapas distintas. Quando a classificação é realizada em apenas uma etapa, utiliza-se uma abordagem de classificação ternária, em que um classificador ternário é empregado para identificar se o *tweet* contém opinião positiva, negativa ou neutra. Por outro lado, a classificação em duas etapas utiliza uma abordagem de classificação duobinária, em que dois classificadores binários são utilizados. O primeiro classificador binário é utilizado para identificar se o *tweet* contém ou não opiniões, ou seja, se o *tweet* é subjetivo ou objetivo (neutro), e o segundo classificador determina a polaridade (positiva ou negativa) do *tweet* classificado como subjetivo na etapa anterior. Apesar de ambas as abordagens serem amplamente adotadas em muitos estudos, não existe um consenso sobre a abordagem mais apropriada na mineração de opiniões em *tweets* ou se existe alguma diferença entre o desempenho das abordagens. Nesse contexto, este trabalho apresenta a comparação entre a abordagem de classificação ternária e duobinária na mineração de opiniões em *tweets*. Nos experimentos computacionais realizados, a abordagem de classificação ternária se mostrou mais adequada para a maioria das situações testadas.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: análise de sentimentos, identificação de polaridade, mineração de opiniões, Twitter

1. INTRODUÇÃO

Com o desenvolvimento das tecnologias da informação e o surgimento das mídias sociais, a Web atingiu uma nova dimensão. Atualmente, os usuários da Web são responsáveis pela criação de grande parte de seu conteúdo, conhecido como *user-generated content*. Esse tipo de conteúdo pode ser encontrado em blogs, wikis, sites de opiniões e, nos últimos anos, em redes sociais e microblogs, como o Twitter[1]. O Twitter é um microblog que suporta a postagem de mensagens de até 140 caracteres, chamados de *tweets*, em que usuários expressam opiniões sobre qualquer assunto diariamente.

Devido à grande popularidade do Twitter, diversas empresas e instituições monitoram o conteúdo publicado pelos usuários dessa rede com o objetivo de identificar opiniões a respeito de pessoas, produtos ou eventos em geral. Entretanto, buscar opiniões de forma manual e identificá-las como sendo favoráveis a um determinado tópico de interesse, como um produto ou evento, pode se tornar uma

[1]http://www.twitter.com

O desenvolvimento deste trabalho contou com o apoio financeiro da CAPES e do CNPq.

tarefa impraticável devido à grande quantidade de mensagens publicadas diariamente. Nesse contexto, a mineração de opiniões é a área que avalia automaticamente as opiniões e sentimentos, expressos em formato textual, a respeito de um determinado tópico de interesse. A mineração de opiniões tem como objetivo identificar documentos que contêm opiniões e classificá-las quanto às suas polaridades, ou seja, se são opiniões positivas ou negativas [Tsytsarau and Palpanas 2012].

O Twitter tem sido o foco da aplicação da mineração de opiniões em muitos trabalhos [Agarwal et al. 2011; Barbosa and Feng 2010; Bravo-Marquez et al. 2013; Jiang et al. 2011; Kouloumpis et al. 2011; Pak and Paroubek 2010]. Enquanto alguns trabalhos determinam a polaridade das opiniões contidas em *tweets* em apenas uma etapa de classificação [Agarwal et al. 2011; Kouloumpis et al. 2011; Pak and Paroubek 2010], outros adotam duas etapas distintas [Barbosa and Feng 2010; Bravo-Marquez et al. 2013; Jiang et al. 2011]. Quando a classificação é realizada em apenas uma etapa, os *tweets* são classificados como positivos, negativos ou neutros, de acordo com o tipo de opinião que eles contêm. Nesse caso, como a finalidade da classificação é discernir entre três possíveis classes, a estratégia é denominada abordagem de classificação ternária. Em contrapartida, quando a polaridade das opiniões contidas em *tweets* é determinada em duas etapas distintas, primeiro, a partir de um conjunto de *tweets*, aqueles que contêm opiniões são identificados e classificados como subjetivos. Na etapa seguinte, os *tweets* subjetivos são classificados quanto às suas polaridades, ou seja, positivos ou negativos. Neste trabalho, essa estratégia de classificação é denominada abordagem de classificação duobinária, uma vez que, em cada uma das duas etapas, a finalidade é discernir entre duas possíveis classes (subjetiva ou objetiva e, em seguida, positiva ou negativa). É importante destacar que, neste trabalho, as classes objetiva e neutra referem-se a *tweets* que não contêm qualquer opinião.

Apesar de ambas as abordagens serem amplamente adotadas em muitos trabalhos, não existe um consenso sobre a abordagem que deve ser adotada na mineração de opiniões em *tweets* ou se existe alguma diferença em relação ao desempenho das abordagens. Nesse contexto, este trabalho apresenta uma comparação entre as abordagens de classificação ternária e duobinária na mineração de opiniões em *tweets*. A fim de comparar as abordagens, diferentes algoritmos de aprendizado de máquina e métodos de pré-processamento textual distintos foram utilizados nos experimentos computacionais realizados neste trabalho. Mais especificamente, inicialmente, os algoritmos e métodos de pré-processamento mais adequados para cada abordagem são identificados e, em seguida, eles são utilizados na comparação entre as duas abordagens.

O restante deste trabalho está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados, com foco na comparação entre as abordagens investigadas. Na Seção 3, as abordagens de classificação ternária e duobinária são definidas e, na Seção 4, os resultados dos experimentos computacionais são apresentados e as abordagens são comparadas. Por fim, na Seção 5, a conclusão e direções para trabalhos futuros são apresentados.

2. TRABALHOS RELACIONADOS

Na mineração de opiniões, muitos trabalhos focam apenas na classificação da polaridade das opiniões, ignorando a etapa da classificação da subjetividade, i.e., a etapa de identificar documentos que contêm ou não opiniões. No entanto, outros trabalhos realizam tanto a classificação da subjetividade quanto a da polaridade. Nesse caso, a classificação pode ser realizada em apenas uma etapa, a partir de um classificador ternário, ou em duas etapas distintas, com base em dois classificadores binários, um para a classificação da subjetividade e outro para a classificação da polaridade.

Entre os trabalhos que realizam a comparação das abordagens ternária e duobinária para mineração de opiniões estão [Esuli and Sebastiani 2006] [Wang et al. 2015] e [Wilson et al. 2009]. O trabalho precursor, apresentado em [Esuli and Sebastiani 2006], avalia três estratégias distintas para a classificação da subjetividade e da polaridade de um conjunto de termos (palavras e expressões).

Em [Esuli and Sebastiani 2006], a primeira estratégia avaliada, denominada E1, é baseada na

abordagem duobinária, em que, primeiro, os termos são classificados como subjetivos ou objetivos e, em seguida, as polaridades dos termos classificados como subjetivos são determinadas. A segunda estratégia, E2, também baseada na abordagem duobinária, discerne entre termos subjetivos positivos, subjetivos negativos e objetivos (ou neutros). Nesse caso, enquanto o primeiro classificador binário classifica os termos como "positivo" ou como seu complemento, "não positivo", o segundo, de forma semelhante, classifica os termos como "negativo" ou como "não negativo". Os termos que são classificados como "positivo" pelo primeiro classificador e como "não negativo" pelo segundo, são, ao final, considerados positivos. Os termos classificados como "não positivo" pelo primeiro classificador e como "negativo" pelo segundo, são considerados negativos. Os termos objetivos são aqueles classificados tanto como "não positivo" e "não negativo" ou como "positivo" e "negativo". Por fim, a terceira estratégia, E3, é baseada na abordagem ternária, em que os termos são classificados como neutros, positivos ou negativos, em apenas uma etapa. Dentre os 120 experimentos realizados, o melhor resultado foi obtido pela estratégia E2, seguida da estratégia E3. Os resultados menos significativos foram obtidos pela estratégia E1.

Em [Wilson et al. 2009], uma estratégia para identificar a polaridade contextual de expressões é proposta, considerando que uma palavra dentro de determinado contexto pode ter uma polaridade diferente de sua polaridade padrão. Nos experimentos realizados, o objetivo é identificar se determinada expressão possui ou não polaridade, i.e., se uma expressão é neutra ou subjetiva, e, caso seja subjetiva, sua polaridade contextual é determinada. Com esse propósito, é utilizada a abordagem de classificação duobinária. Contudo, um dos experimentos reportados tem como objetivo investigar o desempenho obtido pela abordagem de classificação ternária e os resultados de ambas as abordagens são comparados. Os resultados da comparação mostraram que a abordagem de classificação duobinária apresentou um F-measure superior para a classe neutra, enquanto que a abordagem ternária obteve F-measure superior para as classes de polaridade (positiva e negativa).

No contexto da classificação de opiniões contidas em mensagens publicadas em microblogs, o trabalho apresentado em [Wang et al. 2015] realiza a comparação entre as abordagens ternária e duobinária para uma base de dados com 1.584 mensagens publicadas na plataforma Weibo, na língua chinesa. Os resultados dessa comparação demonstram que, para este tipo de mensagem, a abordagem duobinária apresenta um melhor desempenho.

Apesar de a comparação entre as abordagens ternária e duobinária ter sido realizada em alguns trabalhos, em geral, essa comparação foi realizada apenas para bases de dados contendo palavras e expressões. Dessa forma, este trabalho apresenta uma comparação entre as abordagens ternária e duobinária no contexto da mineração de opiniões em *tweets*, considerando que ambas as abordagens são amplamente utilizadas na literatura sem uma prévia avaliação do desempenho de cada abordagem.

3. ABORDAGENS INVESTIGADAS

3.1 Abordagem de Classificação Ternária

Na abordagem de classificação ternária, a opinião contida em *tweets* é classificada em apenas uma etapa de classificação [Agarwal et al. 2011; Kouloumpis et al. 2011; Pak and Paroubek 2010]. Nesta abordagem, um modelo de classificação é construído a partir de um classificador ternário, capaz de classificar *tweets* em uma dentre três possíveis classes: neutra, positiva e negativa.

Na comparação das abordagens realizada neste trabalho, na Seção 4, o desempenho de cada abordagem é medido em termos de acurácia. Nesse sentido, na abordagem de classificação ternária, a acurácia da classificação consiste na razão entre a quantidade de *tweets* classificados corretamente em apenas uma etapa, i.e., a soma dos valores a, e e i da matriz de confusão apresentada na Tabela I, e a quantidade total de *tweets* na base.

Tabela I. Matriz de confusão da abordagem de classificação ternária.

		Classe predita		
		neutro	positivo	negativo
Classe real	neutro	a	b	c
	positivo	d	e	f
	negativo	g	h	i

3.2 Abordagem de Classificação Duobinária

A abordagem de classificação duobinária investigada neste trabalho consiste em classificar a opinião contida em *tweets* como neutra, positiva ou negativa, em duas etapas distintas de classificação [Barbosa and Feng 2010; Bravo-Marquez et al. 2013; Jiang et al. 2011].

A primeira etapa de classificação, chamada de classificação da subjetividade, tem como objetivo identificar *tweets* que contêm fatos (objetivos) e *tweets* que contêm opiniões (subjetivos). Com este propósito, nesta etapa, um modelo de classificação é construído a partir de um classificador binário, capaz de classificar *tweets* em uma dentre duas possíveis classes: subjetiva e objetiva.

Na segunda etapa, ou seja, na etapa de classificação da polaridade, os *tweets* classificados como subjetivos na etapa anterior são avaliados quanto às suas polaridades, i.e., se contêm opiniões positivas ou negativas. Dessa forma, assim como na etapa anterior, um modelo de classificação é construído a partir de um segundo classificador binário, capaz de classificar a opinião contida em *tweets* em positiva ou negativa. Devido ao fato de que essa abordagem utiliza dois classificadores binários distintos, em duas etapas de classificação, ela é denominada abordagem de classificação duobinária.

A fim de tornar os resultados das abordagens comparáveis, a avaliação da abordagem duobinária é realizada considerando o desempenho obtido pelos dois classificadores, ao invés de avaliar o desempenho de cada classificador individualmente. Mais especificamente, com base nas matrizes de confusão apresentadas na Tabela II, dada uma base de dados, a acurácia da classificação consiste na razão entre a quantidade de *tweets* classificados corretamente, i.e., a soma dos valores a, e e h das matrizes, e a quantidade total de *tweets* na base. É importante destacar que a quantidade de *tweets* subjetivos classificados corretamente na primeira etapa (valor d da matriz de subjetividade) não é considerada no cálculo final da acurácia, uma vez que esses *tweets* são a entrada para o classificador de polaridade.

Tabela II. Matrizes de confusão da primeira e segunda etapas da abordagem de classificação duobinária.

		Classe predita	
		objetivo	subjetivo
Classe real	objetivo	a	b
	subjetivo	c	d

	Classe predita	
	positivo	negativo
positivo	e	f
negativo	g	h

4. COMPARAÇÃO DAS ABORDAGENS

Nesta seção, é apresentada a comparação entre as abordagens de classificação ternária e duobinária, na mineração de opiniões em *tweets*.

A fim de comparar as abordagens, inicialmente, diferentes algoritmos de aprendizado de máquina e métodos distintos de pré-processamento textual serão avaliados para cada abordagem. Em seguida,

as melhores técnicas, compostas pelos dois melhores algoritmos e os dois melhores métodos de pré-processamento de cada abordagem serão utilizadas na comparação entre as duas abordagens.

4.1 Bases de Dados

Para avaliar as abordagens investigadas, foram utilizadas quatro bases de dados contendo *tweets* previamente rotulados: Sentiment140 [Go et al. 2009], Sanders[2], SentiStrength [Thelwall et al. 2012] e SemEval [Rosenthal et al. 2014]. Em cada base, os *tweets* encontram-se rotulados como "neutro", "positivo" ou "negativo". Para os experimentos realizados com a abordagem de classificação duobinária, os *tweets* rotulados como "positivo" e "negativo" são considerados como subjetivos na primeira etapa de classificação e os rotulados como "neutro" são considerados como objetivos. As características das bases são apresentadas na Tabela III.

Tabela III. Características das bases de dados.

Base de Dados	#neutros	#positivos	#negativos	Total
Sentiment140	139	182	177	498
Sanders	2.503	570	654	3.727
SentiStrength	1.953	1.340	949	4.242
SemEval	5.847	5.103	1.815	12.765

4.2 Metodologia de Avaliação

Cada abordagem foi avaliada utilizando-se seis conhecidos algoritmos para a tarefa de classificação: *Support Vector Machine* (SVM), *Naive Bayes* (NB), *Multinomial Naive Bayes* (MNB), *Random Forest* (RF), *Decision Tree* (DT) e *Maximum Entropy* (ME). Todos os experimentos reportados foram realizados com implementações desses algoritmos disponíveis na ferramenta Weka[3] [Hall et al. 2009].

Como pré-processamento textual, primeiro, todos os *tweets* são tokenizados, ou seja, os *tweets* são decompostos nas estruturas textuais que os compõem, denominadas *tokens*. Em seguida, outros métodos de pré-processamento são aplicados e, com o objetivo de avaliar a combinação mais apropriada de métodos de pré-processamento para cada abordagem, eles foram organizados e representados em quatro camadas distintas, de modo que cada nova camada incorpora os métodos contidos nas camadas anteriores, além de um método adicional. As camadas são descritas a seguir.

- **Camada de pré-processamento 1 (PP1)**: A camada mais básica, PP1, consiste nos métodos de conversão de todos os caracteres alfabéticos para minúsculos e de remoção de *stopwords*[4].
- **Camada de pré-processamento 2 (PP2)**: Na camada PP2, além dos métodos da camada PP1, os *tokens* específicos do Twitter, URLs, emoticons e pontuações são substituídos por tags especiais. Mais especificamente, menções a usuários (*tokens* seguidos pelo caractere especial @), *hashtags* (*tokens* seguidos pelo caractere especial #), URLs, emoticons e sinais de pontuação são substituídos pelas tags USERNAME, HASHTAG, URL, EMOTICON e PUNCTUATION, respectivamente.
- **Camada de pré-processamento 3 (PP3)**: A camada PP3 incorpora os métodos da camada PP2, além da aplicação de *stemming*. O *stemming* consiste em uma técnica para reduzir um termo ao seu radical (por exemplo, o radical das palavras "campo" e "campestre" é "camp"). Nessa camada, foi utilizado o algoritmo de *stemming Snowball*, disponível na Weka [Hall et al. 2009].
- **Camada de pré-processamento 4 (PP4)**: A última camada, PP4, assim como a camada PP3, incorpora os métodos da camada PP2 e consiste na aplicação de *stemming*. No entanto, o algoritmo de *stemming* utilizado na camada PP4 é o *Lovins*, também disponível na Weka [Hall et al. 2009].

[2]Disponível em http://www.sananalytics.com/lab/index.php.
[3]Disponível em http://www.cs.waikato.ac.nz/ml/weka/.
[4]Neste trabalho, foi utilizada a lista de *stopwords* disponível em http://snowball.tartarus.org/algorithms/english/stop.txt.

Após a realização do pré-processamento textual, cada *tweet* é representado apropriadamente, em forma de atributos. Neste trabalho, unigramas e bigramas foram extraídos dos *tweets* como atributos para representá-los na base de dados.

Os resultados dos experimentos são reportados em termos de acurácia, obtida por meio de validação cruzada com 10 iterações, ou seja, *k-fold cross validation*, com $k = 10$. A acurácia consiste na razão entre a soma da quantidade de *tweets* classificados corretamente em cada iteração e a quantidade total de *tweets* na base, considerando as especificidades das abordagens, conforme descrito na Seção 3.

4.3 Experimento I: Classificação Ternária

O primeiro experimento consiste na avaliação da abordagem de classificação ternária. A Tabela IV apresenta os resultados obtidos pela abordagem ternária para cada base de dados. Cada uma das quatro subtabelas representam cada uma das quatro camadas de pré-processamento. As colunas representam os algoritmos utilizados na avaliação. Os resultados destacados em negrito apontam os algoritmos com melhor desempenho por camada de pré-processamento e os melhores resultados por base de dados encontram-se sublinhados. Além disso, a linha #vitórias apresenta o total de vitórias de cada algoritmo, i.e., o número de vezes em que um algoritmo apresentou a maior acurácia por camada de pré-processamento.

Tabela IV. Acurácias (%) obtidas pela abordagem de classificação ternária.

Dataset	Camada PP1						Camada PP2					
	SVM	NB	MNB	RF	DT	ME	SVM	NB	MNB	RF	DT	ME
Sentiment140	**71,69**	67,87	70,28	59,84	62,05	69,68	**71,69**	67,27	69,28	61,65	59,84	68,88
Sanders	**78,48**	59,27	74,00	72,82	70,41	62,89	**76,42**	60,40	73,09	72,10	68,42	68,34
SentiStrength	**58,32**	54,24	53,56	50,78	51,18	43,12	**58,46**	54,53	53,32	50,26	52,24	42,55
SemEval	**67,87**	54,41	60,86	60,29	61,82	64,65	**67,67**	56,87	61,11	59,96	62,68	65,02
#vitórias	**4**	0	0	0	0	0	**4**	0	0	0	0	0
ranking médio	**1**	4,5	2,75	4,75	4	4	**1**	4,5	2,75	4,5	4,25	4
	Camada PP3						Camada PP4					
	SVM	NB	MNB	RF	DT	ME	SVM	NB	MNB	RF	DT	ME
Sentiment140	**72,49**	68,47	68,47	60,24	64,26	69,08	**73,09**	69,48	68,27	59,24	63,86	70,28
Sanders	**76,58**	60,16	73,44	72,77	69,87	67,99	**77,17**	60,18	73,84	72,12	70,59	68,82
SentiStrength	**58,49**	55,70	53,96	52,50	53,23	43,92	**58,09**	55,21	53,63	51,18	52,29	44,77
SemEval	**68,83**	57,42	61,23	60,75	63,99	66,12	**68,54**	57,05	60,59	61,61	64,27	65,98
#vitórias	**4**	0	0	0	0	0	**4**	0	0	0	0	0
ranking médio	**1**	4,375	3,125	4,75	4	3,75	**1**	4,25	3,5	4,5	4	3,75

Para a escolha dos dois melhores algoritmos, além do número de vitórias, foi utilizado um ranking, denominado, neste trabalho, de ranking médio. Para obter o ranking médio dos algoritmos, para cada base de dados e camada de pré-processamento textual, foram atribuídas pontuações de 1 a 6 para cada algoritmo, em ordem da maior acurácia (pontuação 1) para a menor (pontuação 6). Dessa forma, menores valores significam melhores algoritmos e maiores valores, piores algoritmos. Para cada camada de pré-processamento, a linha ranking médio apresenta a média das pontuações entre as quatro bases de dados para cada algoritmo. Por fim, para obter o ranking médio geral dos algoritmos, calcula-se, para cada algoritmo, a média dos valores de ranking médio nas quatro camadas de pré-processamento. Dessa forma, os valores de ranking médio geral dos algoritmos SVM, NB, MNB, RF, DT e ME são, respectivamente, **1**, 4,406, **3,031**, 4,625, 4,063 e 3,875. É possível observar que os dois melhores algoritmos, de acordo com o ranking médio geral, são SVM e MNB, com 1 e 3,031, respectivamente.

O ranking médio geral das camadas de pré-processamento foi obtido de forma semelhante ao ranking médio dos algoritmos. Nesse caso, foram atribuídas pontuações de 1 a 4 para cada camada, em ordem da maior acurácia (pontuação 1) para a menor (pontuação 4). Assim, os valores de ranking médio geral das camadas de pré-processamento PP1, PP2, PP3 e PP4 são, respectivamente, 2,896, 3,187, **1,875** e **2,042**. É possível observar que o melhor desempenho geral foi obtido pela utilização dos métodos de pré-processamento incorporados na camada PP3, com 1,875, seguido da camada PP4, com 2,042.

4.4 Experimento II: Classificação Duobinária

Esse experimento consiste na avaliação da abordagem de classificação duobinária. A Tabela V apresenta os resultados obtidos pela abordagem duobinária para cada base de dados. É possível observar que, assim como os resultados obtidos pela abordagem de classificação ternária, o algoritmo SVM apresentou desempenho superior aos demais, considerando todas as bases de dados e camadas de pré-processamento.

Tabela V. Acurácias (%) obtidas pela abordagem de classificação duobinária.

Dataset	Camada PP1						Camada PP2					
	SVM	NB	MNB	RF	DT	ME	SVM	NB	MNB	RF	DT	ME
Sentiment140	**70,68**	65,66	69,28	48,59	62,85	64,26	**72,29**	66,47	68,27	57,83	62,65	60,84
Sanders	**77,19**	60,96	73,84	73,81	69,44	65,39	**75,91**	61,66	73,17	72,31	68,58	65,07
SentiStrength	**57,00**	53,42	52,62	50,87	50,54	46,25	**57,43**	53,72	51,96	51,74	51,51	46,72
SemEval	**65,96**	53,07	60,24	58,75	59,73	63,25	**66,08**	56,40	60,73	58,64	62,02	63,50
#vitórias	**4**	0	0	0	0	0	**4**	0	0	0	0	0
ranking médio	**1**	4,25	2,5	4,5	4,5	4,25	**1**	4,25	3	4,25	4	4,5
	Camada PP3						Camada PP4					
	SVM	NB	MNB	RF	DT	ME	SVM	NB	MNB	RF	DT	ME
Sentiment140	**72,69**	66,87	66,67	52,81	59,04	63,65	**74,10**	67,67	67,67	56,43	59,24	65,66
Sanders	**75,88**	61,23	73,01	72,44	67,94	66,03	**75,69**	61,47	73,33	72,10	68,77	65,25
SentiStrength	**57,00**	54,64	52,40	51,13	51,82	50,87	**56,65**	54,10	51,89	50,21	52,19	50,66
SemEval	**66,98**	56,58	60,69	58,61	60,85	64,88	**66,96**	56,45	60,33	59,08	61,05	64,76
#vitórias	**4**	0	0	0	0	0	**4**	0	0	0	0	0
ranking médio	**1**	4	3	4,75	4	4,25	**1**	4,125	3,125	5	3,75	4

Para esta abordagem, a definição das melhores técnicas também é baseada na utilização de um ranking, como descrito no experimento anterior. Nesse contexto, para esta abordagem, os valores de ranking médio geral dos algoritmos SVM, NB, MNB, RF, DT e ME são, respectivamente, **1**, 4,156, **2,906**, 4,625, 4,063 e 4,25. Dessa forma, é possível observar que os algoritmos SVM e MNB obtiveram melhor desempenho, com ranking médio geral igual a 1 e 2,906, respectivamente.

Em relação aos métodos de pré-processamento, os valores de ranking médio geral das camadas de pré-processamento PP1, PP2, PP3 e PP4 são, respectivamente, 2,771, 2,458, **2,396** e **2,375**. É possível notar que o melhor desempenho geral foi obtido pela camada PP4, com 2,375, seguido pela camada PP3, com 2,396.

4.5 Um Classificador Ternário ou Dois Binários?

Esta subseção tem como objetivo responder à questão de pesquisa investigada neste trabalho: no contexto da mineração de opiniões em *tweets*, o melhor desempenho preditivo é obtido a partir da utilização de um classificador ternário, em apenas uma etapa de classificação, ou de dois classificadores binários, em duas etapas distintas?

A fim de comparar as abordagens investigadas, foram utilizadas as melhores técnicas identificadas nos experimentos reportados nas Subseções 4.3 e 4.4, ou seja, os dois melhores algoritmos e as duas melhores camadas de pré-processamento para as duas abordagens. Dessa forma, tanto para a abordagem de classificação ternária quanto para a abordagem duobinária, os melhores algoritmos foram o SVM e o MNB e as melhores camadas de pré-processamento foram a PP3 e a PP4. A Tabela VI apresenta a comparação entre as abordagens.

Em relação aos melhores resultados obtidos para cada base de dados (resultados sublinhados), é possível observar que a abordagem de classificação ternária apresentou desempenho superior em três das quatro bases de dados utilizadas (Sanders, SentiStrength e SemEval). A abordagem de classificação duobinária obteve melhor acurácia apenas para a base Sentiment140, com 74,10%.

De modo geral, considerando todas as técnicas avaliadas, ou seja, os dois melhores algoritmos e as duas melhores camadas de pré-processamento, ao comparar as abordagens, é possível observar que a

Tabela VI. Comparação entre as abordagens de classificação ternária e duobinária.

	Camada PP3				Camada PP4			
	SVM		MNB		SVM		MNB	
	ternária	duobinária	ternária	duobinária	ternária	duobinária	ternária	duobinária
Sentiment140	72,49	**72,69**	**68,47**	66,67	73,09	**74,10**	**68,27**	67,67
Sanders	**76,58**	75,88	**73,44**	73,01	**77,17**	75,69	**73,84**	73,33
SentiStrength	**58,49**	57,00	**53,96**	52,40	**58,09**	56,65	**53,63**	51,89
SemEval	**68,83**	66,98	**61,23**	60,69	**68,54**	66,96	**60,59**	60,33
#vitórias	**3**	1	**4**	0	**3**	1	**4**	0

abordagem de classificação ternária apresenta melhores resultados em 87,5% dos casos (14 vitórias), contra apenas 12,5% (2 vitórias) da abordagem duobinária.

5. CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho apresentou uma comparação entre as abordagens de classificação ternária e duobinária para mineração de opiniões em *tweets*. Considerando as quatro bases de dados, os dois melhores algoritmos de aprendizado de máquina e as duas melhores camadas de pré-processamento, os resultados demonstraram que a abordagem de classificação ternária se mostrou mais apropriada, apresentando melhores resultados em 87,5% das situações testadas.

Como trabalhos futuros, para que os resultados se tornem mais conclusivos, novos experimentos estão sendo planejados para um número maior de bases de dados de *tweets*. Além disso, os resultados deverão ser avaliados em relação a outras medidas de desempenho, além da acurácia.

REFERÊNCIAS

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., and Passonneau, R. Sentiment analysis of Twitter data. In *Proc. of the Workshop on Languages in Social Media*. ACL, pp. 30–38, 2011.

Barbosa, L. and Feng, J. Robust sentiment detection on Twitter from biased and noisy data. In *Proc. of the 23rd International Conference on Computational Linguistics: Posters*. ACL, pp. 36–44, 2010.

Bravo-Marquez, F., Mendoza, M., and Poblete, B. Combining strengths, emotions and polarities for boosting Twitter sentiment analysis. In *Proc. of the 2nd International Workshop on Issues of Sentiment Discovery and Opinion Mining*. ACM, pp. 2:1–2:9, 2013.

Esuli, A. and Sebastiani, F. Determining term subjectivity and term orientation for opinion mining. In *Proc. of the 11th Conference of the European Chapter of the Association for Computational Linguistics*. pp. 193–200, 2006.

Go, A., Bhayani, R., and Huang, L. Twitter sentiment classification using distant supervision. Tech. Rep. S224N, Stanford, 2009.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. The weka data mining software: an update. *SIGKDD Explorations Newsletter* 11 (1): 10–18, 2009.

Jiang, L., Yu, M., Zhou, M., Liu, X., and Zhao, T. Target-dependent Twitter sentiment classification. In *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. ACL, pp. 151–160, 2011.

Kouloumpis, E., Wilson, T., and Moore, J. Twitter sentiment analysis: The good the bad and the OMG! In *Proc. of the 5th International AAAI Conference on Web and Social Media*. pp. 538–541, 2011.

Pak, A. and Paroubek, P. Twitter as a corpus for sentiment analysis and opinion mining. In *Proc. of the 7th International Conference on Language Resources and Evaluation*. Vol. 10. pp. 1320–1326, 2010.

Rosenthal, S., Ritter, A., Nakov, P., and Stoyanov, V. Semeval-2014 task 9: Sentiment analysis in Twitter. In *Proc. of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. ACL, pp. 73–80, 2014.

Thelwall, M., Buckley, K., and Paltoglou, G. Sentiment strength detection for the social Web. *Journal of the American Society for Information Science and Technology* 63 (1): 163–173, 2012.

Tsytsarau, M. and Palpanas, T. Survey on mining subjective data on the Web. *Data Mining and Knowledge Discovery* 24 (3): 478–514, 2012.

Wang, X., Zhang, C., and Wu, M. Sentiment classification analysis of chinese microblog network. In *Proc. of the 6th Workshop on Complex Networks*. Springer, pp. 123–129, 2015.

Wilson, T., Wiebe, J., and Hoffmann, P. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational Linguistics* 35 (3): 399–433, 2009.

Uma estratégia de geração de dados artificiais para classificadores de larga margem aplicada em bases de dados desbalanceadas

Marcelo Ladeira Marques, Saulo Moraes Villela, Carlos Cristiano Hasenclever Borges

Universidade Federal de Juiz de Fora, Brasil
marcelo.ladeira@ice.ufjf.br, {saulo.moraes, carlos.cristiano}@ufjf.edu.br

Abstract. O presente trabalho tem como proposta o desenvolvimento de um método capaz de melhorar os resultados obtidos por classificadores de larga margem quando aplicados em bases desbalanceadas. O algoritmo apresentado realiza um procedimento iterativo baseado em classificadores de margem, visando gerar amostras sintéticas com o intuito de reduzir o nível de desbalanceamento. No processo são utilizados vetores suporte como referência para a geração das novas instâncias, o que tende a posicioná-las em regiões mais representativas. Introduz-se também uma estratégia de extrapolação do raio da classe minoritária, objetivando o reposicionamento do hiperplano separador, de forma a melhorar a capacidade de predição. Após a explicação do método são apresentados experimentos comparativos entre a técnica proposta e o *Synthetic Minority Over-sampling TEchnique* (SMOTE), os quais forneceram fortes evidências da aplicabilidade da abordagem proposta.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: imbalanced learning, large margin classifiers, oversampling, synthetic sample generation

1. INTRODUÇÃO

Determinadas particularidades das bases de dados sujeitas a aprendizado supervisionado podem ocasionar uma redução drástica no desempenho dos classificadores. Dentre estas particularidades, destaca-se o desbalanceamento entre as classes, o qual caracteriza-se pela concentração de amostras em um grupo reduzido de classes, enquanto as demais possuem poucos exemplos. No caso específico da classificação binária, a classe com mais amostras é denominada majoritária e a com menos amostras minoritária.

Nestes problemas é de fundamental importância a classificação correta das amostras associadas a classe minoritária. Esta padrão é necessário devido a natureza da classe minoritária, a qual, geralmente, caracteriza-se como a classe de principal interesse. Em decorrência desta particularidade, a utilização de algoritmos tradicionais usualmente não apresenta bons resultados, por não considerar a distribuição dos dados, o que pode levar à construção de classificadores onde as amostras tendem a ser inferidas como sendo da classe majoritária [Pourhabib et al. 2015].

Diversos são os problemas reais que apresentam como principal característica o desbalanceamento da base, a saber: detecção de fraudes em transações de cartão de crédito [Chan and Stolfo 1998], diagnósticos médicos [Sun et al. 2007], categorização de textos [Dumais et al. 1998], identificação de vazamento de óleo através de satélite [Kubat et al. 1998] e previsão de colisões entre aeronaves [Castro and Braga 2011].

Outra aplicação de grande importância destas técnicas é a solução de problemas multiclasse através da adaptação de métodos de classificação binários, pois uma das estratégias comumente utilizadas nestes casos é o treinamento de cada classe em relação às demais, o que acaba ocasionando um desbalanceamento nos dados.

De forma geral, pode-se identificar dois tipos de abordagem para a solução do problema de classificação em bases desbalanceadas [Castro and Braga 2011]. A primeira tem como foco a realização de modificações no próprio classificador, tornando-o compatível com a distribuição dos dados, onde podem-se citar: algoritmos de *ensemble* [Wang and Yao 2009] e estratégias de modificação na função de custo do classificador [Tao et al. 2007]. A segunda efetua um pré-processamento dos dados, balanceando a base antes de repassá-la para um algoritmo de classificação, sendo os modelos mais comuns o *oversampling* [Liu et al. 2007] e o *undersampling* [Yen and Lee 2009].

Apesar de diversos trabalhos já terem sido desenvolvidos na área, o estudo da classificação em bases desbalanceadas continua sendo um problema de grande interesse, visto que ainda não é consenso o real efeito do desbalanceamento na qualidade da predição. Desta forma justifica-se a busca por novas técnicas capazes de atingir melhores resultados.

2. CONCEITOS PRELIMINARES

2.1 *Support Vector Machine* – SVM

A abordagem proposta não se limita a utilização de um classificador específico, entretanto, se faz necessária uma introdução à técnica utilizada ao longo dos experimentos. Com esta finalidade, a seguir será apresentada uma breve descrição da formulação matemática das Máquinas de Vetores Suporte (*Support Vector Machines*) [Vapnik 1995] para a solução do problema binário de classificação.

Assumindo $Z = \left\{z_i = \left(x_i, y_i\right) : i \in \{1, \ldots, m\}\right\}$ onde $x_i \in \mathbb{R}^d$ são as amostras de treinamento com d dimensões e $y_i \in \{-1, +1\}$ são os possíveis rótulos das classes. Adotando também $Z^+ = \left\{\left(x_i, y_i\right) \in Z : y_i = +1\right\}$ e $Z^- = \left\{\left(x_i, y_i\right) \in Z : y_i = -1\right\}$. O problema de classificação binária consiste em identificar um hiperplano, dado pelo seu vetor normal $w \in \mathbb{R}^d$ e pela constante $b \in \mathbb{R}$, de tal forma que o hiperplano separe os conjuntos Z^+ e Z^-. Com base nesta formulação também é possível introduzir o conceito de margem, que consiste em definir uma distância mínima $\gamma \geq 0$ entre o hiperplano separador e os conjuntos Z^+ e Z^-. Assim, o problema resume-se a definir (w, b) tal que:

$$y_i \left(w \cdot x_i + b\right) \geq \gamma, \quad \forall \left(x_i, y_i\right) \in Z.$$

O SVM classifica as amostras através de um processo de otimização que identifica o hiperplano com a máxima margem entre as duas classes presentes no conjunto de treinamento, podendo este problema de otimização ser formulado como:

$$\begin{array}{ll} \min\limits_{w,b,\xi} & \frac{1}{2} w^T w + C \sum_i \xi_i \\ \text{sujeito a} & y_i \left(w^T \phi\left(x_i\right) + b\right) \geq 1 - \xi_i, \ \xi_i \geq 0, \ \forall i \in \{1, \ldots, m\} \end{array}$$

onde $\phi\left(.\right)$ é uma função de mapeamento para um espaço de características, ξ_i são as variáveis de folga utilizadas para evitar *overfitting* e C é uma constante de penalização utilizada no caso margem flexível (*soft margin*), sendo os vetores suporte as amostras mais próximas do hiperplano separador.

É válido ressaltar que o SVM binário também pode ser utilizado para solucionar problemas multiclasse, sendo nestes casos utilizados métodos de adaptação, como o treinamento um-contra-todos (*one-against-all*) ou um-contra-um (*one-against-one*) [Hsu and Lin 2002].

2.2 *Synthetic Minority Over-sampling TEchnique* – SMOTE

Dentre os algoritmos de *oversampling*, o SMOTE [Chawla et al. 2002] destaca-se por sua ampla utilização e pelos bons resultados obtidos. Esta técnica consiste na geração de exemplos artificiais da classe minoritária através da interpolação das amostras pré-existentes, sendo aplicada, em linhas gerais, da seguinte forma: para cada amostra pré-existente é escolhido aleatoriamente um dos K vizinhos mais próximos, os quais são determinados por distância euclidiana. Em seguida, para cada dimensão da instância, é gerado um valor aleatório entre a amostra de origem e o vizinho escolhido, sendo estes valores utilizados na construção do novo exemplo artificial. O pseudocódigo apresentado no Algoritmo 1 ilustra o processo de geração destas instâncias. Dependendo do número de amostras a serem geradas, para cada amostra de origem, podem ser utilizados diversos vizinhos.

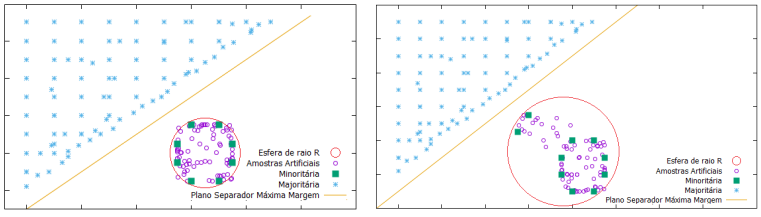
Algoritmo 1 Geração de amostra artificial (SMOTE)

Entrada: amostra: A; vizinho escolhido: V;
Saída: amostra artificial: G;
início
  para i ***de*** 1 ***até*** d **faça**
    $dif \leftarrow V_i - A_i$;
    $gap \leftarrow \text{rand}[0, 1]$;
    $G_i \leftarrow A_i + gap \times dif$;
  fim para
fim

3. ABORDAGEM PROPOSTA

Uma das alternativas de interesse na área de aprendizado em bases desbalanceadas têm como proposta a geração de dados artificiais, sendo o SMOTE um dos principais representante desta classe de soluções. Apesar dos bons resultados obtidos pelo algoritmo, existem situações onde as amostras geradas são pouco representativas no treinamento do classificador, o que implica em modificações irrelevantes na posição do hiperplano obtido, em comparação ao gerado através da base original. Nestes casos, o processo de balanceamento mostra-se ineficaz, pois a classe majoritária continua sendo priorizada, resultando assim na construção de um classificador sem nenhuma aplicação prática.

As amostras artificias geradas pelo SMOTE encontram-se sempre entre os segmentos de reta que unem cada uma das características pertencentes as duas instâncias que lhe deram origem. Esta abordagem garante uma certa robustez, porém promove a geração de uma quantidade reduzida de exemplos nas áreas próximas ao hiperplano de decisão, mostrando-se ainda mais raros os casos onde as novas amostras conseguem extrapolar o raio R da classe minoritária, ou seja, o raio que contém a menor hiperesfera capaz de circunscrever todas as instâncias da classe minoritária. A Figura 1a ilustra este comportamento.

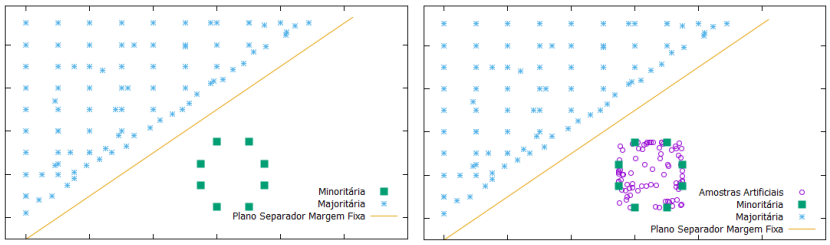


(a) Exemplo SMOTE. (b) Exemplo SMOTE base dispersa.

Fig. 1: Exemplos duas dimensões SMOTE

As características citadas anteriormente limitam a eficácia do SMOTE em algumas situações, como em casos onde se deseja utilizar classificadores baseados em margem rígida (*hard margin*). Isso ocorre pois os vetores suporte raramente são modificados, considerando que as novas instâncias dificilmente são gerados fora do raio da classe minoritária. Como a construção do classificador é baseada nos vetores suporte, os hiperplanos gerados com e sem as amostras artificiais permanecem similares, conforme pode ser observado no exemplo mostrado na Figura 2.

Outro aspecto importante é que, quando uma suavização da margem é permitida, o SMOTE é capaz de evitar que a classe minoritária seja tratada como ruído, garantindo assim uma melhora significativa dos resultados. Entretanto, sua utilização dificilmente possibilita uma movimentação do hiperplano em direção à classe majoritária, o que seria desejável na tentativa de priorizar a classe de maior interesse, comportamento este ilustrado na Figura 3. Apesar deste problema poder ser amenizado em situações simples, através de uma melhor escolha do parâmetro de flexibilização C, o mesmo não ocorre em bases onde a separação entre as classes não é bem definida, sendo então de vital importância a utilização de métodos capazes de reposicionar o hiperplano de forma automática.

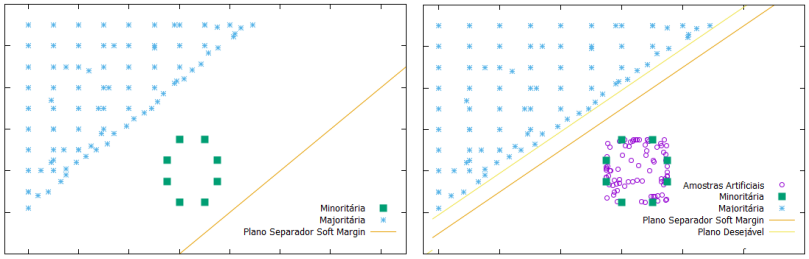


(a) Base Original (b) Base Original + SMOTE

Fig. 2: Margem rígida: a utilização do SMOTE não altera o plano separador.

Novamente, baseando-se na utilização de máquinas vetores suporte, um outro exemplo onde amostras pouco representativas podem ser geradas ocorre quando existe uma alta concentração de instâncias da classe minoritária em regiões distantes do hiperplano separador, pois, nestes casos, as novas amostras serão, em sua maioria, geradas em regiões irrelevantes para o processo de definição do hiperplano. A Figura 1b ilustra este comportamento em uma base de duas dimensões.

Alguns métodos com foco na geração de amostras mais significativas já foram propostos na literatura, com duas abordagens de especial interesse para o presente trabalho: as técnicas baseadas no aumento do raio da classe minoritária [Koto 2014] e as que priorizam a geração de amostras artificiais nas áreas próximas ao hiperplano separador [Han et al. 2005].



(a) Base Original (b) Base Original + SMOTE

Fig. 3: Margem flexível: a utilização do SMOTE apenas permite mover o hiperplano até a posição alcançada pela margem rígida, não sendo possível uma maior aproximação do plano desejável.

Este trabalho tem como objetivo obter um método capaz de integrar os benefícios encontrados nas duas abordagens citadas anteriormente. Para tal, a proposta apresentada é baseada nos seguintes princípios: (I) geração de dados artificiais; (II) extrapolação do raio da classe minoritária; e (III) utilização de vetores suporte como base para geração das amostras.

Em relação ao processo de extrapolação, propõe-se a introdução de um novo parâmetro de entrada, denotado por α. Esta modificação tem como objetivo redefinir os limites impostos pelo SMOTE na geração dos dados artificiais, consistindo basicamente na alteração da equação de cálculo da variável *gap* contida no Algoritmo 1, a qual passa a ser reescrita da seguinte forma:

$$\text{gap} \leftarrow \text{rand}\left[(0-\alpha),(1+\alpha)\right]$$

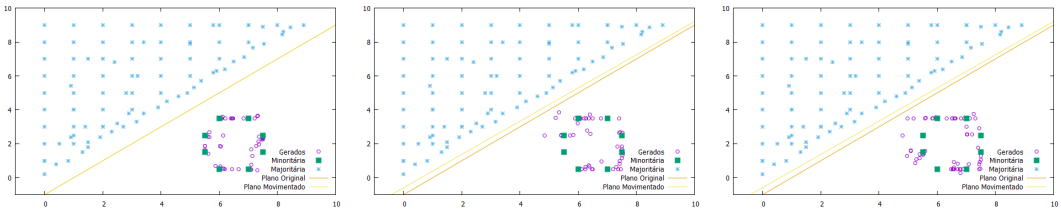
Com esta alteração, as novas amostras, não mais encontram-se delimitadas pelos exemplos que lhe deram origem, podendo também extrapolar estes limites. Sendo assim, o parâmetro α possibilita um aumento no raio da classe minoritária, consequentemente favorecendo a movimentação do hiperplano separador na direção da classe majoritária, o que tende a melhorar a qualidade do classificador gerado, pois oferece uma priorização da classe de maior interesse. Entretanto, deve-se ressaltar a importância de se adotar um valor para α de forma a não permitir que a extrapolação descaracterize a distribuição dos dados, pois isto resultaria na construção de uma base sem a devida representatividade da classe minoritária.

Como forma de calibrar o parâmetro α foi proposta uma abordagem baseada na margem fornecida pelo classificador, sendo necessária, portanto, uma execução do processo de otimização antes de dar início à geração das amostras. Uma descrição deste método é apresentada no Algoritmo 2.

Algoritmo 2 Validação de amostra sintética baseada em margem

Entrada: amostra: A; vizinho selecionado: V; amostra gerada: G; parâmetro de descarte: γ;
Saída: amostra artificial *aceita* ou *descartada*
início
  se *G não extrapolar os limites definidos por A e V* **então**
    retorna *amostra aceita*
  senão
    $dist_1 \leftarrow$ distância euclidiana entre A e G;
    $dist_2 \leftarrow$ distância euclidiana entre V e G;
    $dist \leftarrow \min\{dist_1, dist_2\}$;
    se $dist < \gamma$ **então**
      retorna *aceita*
    senão
      retorna *descartada*
    fim se
  fim se
fim

O Algoritmo 2 é executado para cada amostra gerada, com isto espera-se reduzir o risco da utilização da extrapolação, pois a margem funciona como um limite que é automaticamente adaptado para cada problema, dado que, quanto maior a margem, mais seguro é aumentar o raio da classe minoritária, sem que esta se misture com a outra classe. O número de amostras descartadas durante o processo pode ser utilizado no ajuste de α, levando-se em conta que caso muitas instâncias estejam sendo descartadas, uma redução no valor de α é aconselhável. A Figura 4 contém um exemplo de extrapolação. Conforme esperado, o aumento do raio da classe minoritária foi capaz de movimentar o plano em direção à classe majoritária, entretanto um aspecto pode ser aprimorado: instâncias pouco representativas continuam sendo obtidas quando as amostras utilizadas como referência para a geração do dado sintético estão localizadas em regiões afastadas do hiperplano separador (vide Figura 1b).



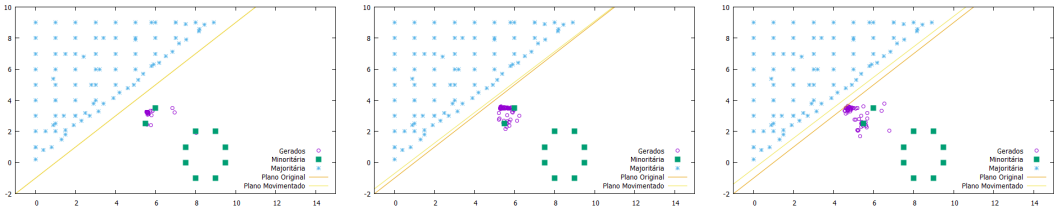
(a) $\alpha = 10$ (b) $\alpha = 50$ (c) $\alpha = 100$

Fig. 4: Gráficos de diferentes extrapolações.

Buscando uma melhoria no processo de geração dos dados sintéticos, uma nova modificação foi proposta no método. Com esta alteração, o algoritmo passa a utilizar somente os vetores suportes como referência para a geração das amostras artificiais. Sua descrição é apresentada no Algoritmo 3.

Algoritmo 3 Método proposto

início
  enquanto *base estiver desbalanceada* **faça**
    executar classificador para obtenção dos vetores suporte e do valor da margem;
    calcular lista dos K vizinhos mais próximos de cada vetor suporte (sendo vetor suporte ou não);
    para i ***de*** *1* ***até*** *total de vetores suportes* **faça**
      selecionar o i-ésimo vetor suporte (SV_i);
      selecionar aleatoriamente um dos K vizinhos de SV_i;
      executar **Algoritmo 1 com extrapolação** utilizando as amostras selecionadas;
      executar **Algoritmo 2** para validar a amostra gerada;
      se *aceita* **então**
        inserir amostra na base;
    fim para
    //`caso o Algoritmo esteja descartando amostras em excesso, é necessário reduzir valor de` α
  fim enquanto
fim



(a) $\alpha = 10$ (b) $\alpha = 50$ (c) $\alpha = 100$

Fig. 5: Diferentes extrapolações utilizando vetores suporte.

A partir da utilização dos vetores suporte na geração das novas instâncias foi possível obter amostras mais representativas, movimentando o plano em direção à classe majoritária. É importante destacar que a base original não foi descaracterizada, dado que as novas instâncias encontram-se sempre em suas redondezas e utilizam a margem como medida de referência para que as duas classes envolvidas no problema não acabem sendo misturadas. Outro aspecto a ser ressaltado é a concentração das amostras geradas, que ocorre devido ao reduzido número de vetores suporte, fato que seria minorado em casos com um número maior de dimensões.

4. EXPERIMENTOS E RESULTADOS

Para avaliar a performance do método proposto foram utilizadas 4 bases binárias desbalanceadas e não linearmente separáveis retiradas do *KEEL-dataset repository* [Sánchez et al. 2011]: Ecoli2 (E), Abalone9-18 (A), Yeast5 (Y) e 7-winequality-red-8_vs_6-7 (W). Em todos os experimentos as bases foram separadas de forma estratificada na proporção de 2/3 para treino e 1/3 para teste.

As métricas adotadas na avaliação dos resultados foram as medidas de *precision* (P), *recall* (R) e *F-measure* (F), sendo o *recall* a mais significante delas, pois indica a ocorrência de falso negativo, considerado o caso mais grave de erro, uma vez que representa classificar uma instância da classe minoritária como sendo da majoritária. Entretanto, também é importante alcançar um certo balanceamento entre as medidas de *precision* e *recall*, pois um classificador que tende a prever muitos falsos positivos não possui aplicação prática. Com esta finalidade foi introduzida a medida F, a qual é construída levando-se em conta as duas outras métricas. Considerando para os cálculos o número de ocorrências de verdadeiros positivos (VP), falsos positivos (FP) e falsos negativos (FN), estas medidas podem ser definidas como:

$$P = \frac{VP}{VP + FP} \qquad R = \frac{VP}{VP + FN} \qquad F = 2 * \frac{P * R}{P + R}$$

Durante os teste foram utilizadas três versões de cada base: uma contendo apenas os dados originais, uma balanceada através do SMOTE e outra balanceada pelo método proposto. O classificador adotado foi o SVM com margem flexível e *kernel* linear, sendo escolhido o algoritmo de Otimização Mínima Sequencial (*Sequential Minimal Optimization* – SMO) [Platt 1999] como forma de solução do processo de otimização. Também foram realizados experimentos utilizando o SVM com margem rígida, entretanto, estes apresentaram resultados inconclusivos, não sendo portanto incluídos.

Uma breve revisão da literatura indicou um possível motivo para o mal funcionamento da margem rígida. Como todas as bases são não linearmente separáveis, é necessária a utilização de *kernel* para que o SVM passa convergir. Entretanto, no modelo atual, a geração das instâncias artificiais é feita no espaço de entrada, enquanto o hiperplano é construído no espaço de características. Esta distorção provavelmente afeta a qualidade dos resultados. Pretende-se aprimorar o modelo em um futuro próximo, gerando os dados sintéticos diretamente no espaço de características, conforme proposto em [Mathew et al. 2015].

A geração das amostras pelo SMOTE foi realizada em uma única execução, enquanto a técnica proposta funciona de forma iterativa, buscando conseguir um reposicionamento gradativo do hiperplano construído. Em ambos os casos, o ponto de parada foi a obtenção do balanceamento mais igualitário possível. Para a definição do parâmetro de flexibilização foram utilizados valores decrescente de C múltiplos de 10, variando entre 1000 e 0,0001. Para o SMOTE, este processo foi realizado uma única vez, sendo o valor de C reduzido até o SVM convergir. No caso do método proposto, uma nova flexibilização era realizada sempre que a geração das novas instâncias fazia com que o SVM parasse de convergir. Por fim, os valores adotados para as variáveis K e α foram 5 e 10, respectivamente. A tabela I contém os resultados obtidos, além da taxa de balanceamento (Tx.) das bases.

Table I: Resultados dos algoritmos.

Base	Tx.	SVM			SMOTE			Proposto		
		P	R	F	P	R	F	P	R	F
E	5,46	0,7857	0,6111	0,6875	0,6667	0,7778	0,7180	0,6400	0,8889	0,7442
A	16,40	0,7273	0,5714	0,6400	0,3750	0,8571	0,5217	0,3333	0,9286	0,4906
Y	32,73	0,4000	0,1333	0,2000	0,3947	1,0000	0,5660	0,3409	1,0000	0,5085
W	46,50	0,0000	0,0000	0,0000	0,0581	0,8333	0,1086	0,0547	0,6667	0,1011

Uma análise dos experimentos realizados permite a observação dos seguintes comportamentos: (I) em todas os casos o balanceamento das classes levou a um aumento significativo no valor do *recall*, fornecendo fortes evidências sobre a eficácia dos métodos, considerando que este resultado indica uma redução no número de falsos negativos; (II) o aumento do valor do *recall* pode ser observado de forma mais acentuada quando o método proposto é empregado, indicando a validade da abordagem; (III) ambos os métodos de balanceamento apresentaram estabilidade no valor de *F-measure* quando comparados ao SVM original, o que significa que o *precision* não foi exageradamente reduzido na busca de melhores valores de *recall*.

5. CONCLUSÃO

O presente trabalho propõe um método para geração de dados, tendo como objetivo popular mais densamente as zonas de fronteira entre as classes. Através das analises teóricas e do estudo empírico foi possível obter fortes evidências sobre a validade da abordagem. O estudo indicou uma melhora nos resultados quando as instâncias são geradas de forma gradativa, pois assim o hiperplano vai sendo redirecionado de forma mais eficiente. Entretanto a realização de diversos processos de otimização resulta em um custo computacional muito alto. Visando solucionar este contratempo, e considerando a independência da abordagem proposta com relação ao classificador utilizado, uma opção de grande potencial seria a utilização de métodos *online* [Leite and Neto 2008], [Gentile 2001]. Com esta modificação seria possível atingir resultados similares ou superiores, porém a um custo muito inferior. Também é de grande interesse a parametrização automática de α, que pode ser obtida através de métodos evolucionistas ou de um estudo mais aprofundado de sua relação com o valor da margem.

REFERENCES

Castro, C. L. and Braga, A. P. Aprendizado supervisionado com conjuntos de dados desbalanceados. *RCA*, 2011.

Chan, P. K. and Stolfo, S. J. Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection. In *KDD*, 1998.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. Smote: synthetic minority over-sampling technique. *JAIR*, 2002.

Dumais, S., Platt, J., Heckerman, D., and Sahami, M. Inductive learning algorithms and representations for text categorization. In *P 7th ICIKM*, 1998.

Gentile, C. A new approximate maximal margin classification algorithm. *JMLR*, 2001.

Han, H., Wang, W., and Mao, B. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *ICIC*, 2005.

Hsu, C. and Lin, C. A comparison of methods for multiclass support vector machines. *TNN*, 2002.

Koto, F. Smote-out, smote-cosine, and selected-smote: An enhancement strategy to handle imbalance in data level. In *ICACSIS*, 2014.

Kubat, M., Holte, R. C., and Matwin, S. Machine learning for the detection of oil spills in satellite radar images. *ML*, 1998.

Leite, S. C. and Neto, R. F. Incremental margin algorithm for large margin classifiers. *Neurocomputing*, 2008.

Liu, A., Ghosh, J., and Martin, C. E. Generative oversampling for mining imbalanced datasets. In *DMIN*, 2007.

Mathew, J., Luo, M., Pang, C. K., and Chan, H. L. Kernel-based smote for svm classification of imbalanced datasets. In *IECON*, 2015.

Platt, J. C. 12 fast training of support vector machines using sequential minimal optimization. *AKM*, 1999.

Pourhabib, A., Mallick, B. K., and Ding, Y. Absent data generating classifier for imbalanced class. *JMLR*, 2015.

Sánchez, R. L., Alcalá, F. J., Fernández, H. A., Luengo, M. J., Derrac, R. J., García, L. S., and Herrera, T. F. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *JMLSC*, 2011.

Sun, Y., Kamel, M. S., Wong, A. K. C., and Wang, Y. Cost-sensitive boosting for classification of imbalanced data. *PR*, 2007.

Tao, X., Ji, H., and Xie, Y. A modified psvm and its application to unbalanced data classification. In *ICNC*, 2007.

Vapnik, V. The nature of statistical learning theory, 1995.

Wang, S. and Yao, X. Diversity analysis on imbalanced data sets by using ensemble models. In *CIDM*, 2009.

Yen, S. and Lee, Y. Cluster-based under-sampling approaches for imbalanced data distributions. *ESA*, 2009.

CFI Blocking: Uma estratégia eficaz para blocagem em pareamento probabilístico de registros

R.G Pereira[1], W.Meira Jr.[1], A.A.G Junior[1]

Universidade Federal de Minas Gerais, Brazil
ramonbhb@ufmg.br, meira@dcc.ufmg.br, augustoguerra@ufmg.br

Abstract.

O CFI Blocking é um algoritmo para otimizar a blocagem no contexto de pareamento probabilístico de registros. A blocagem é responsável por pré selecionar e agrupar registros com maior probabilidade de pertencerem a mesma entidade no mundo real. As estratégias de blocagem atuais são definidas pelo conhecimento prévio do pesquisador. CFI Blocking se baseia na mineração de padrões frequentes fechados e no conhecimento intrínseco das instâncias dos atributos, em particular, foram utilizadas propriedades de conjuntos fechados para enumeração automatizada dos blocos. A avaliação experimental foi realizada utilizando dados sintéticos e reais, e permitiu concluir que CFI Blocking apresenta melhor desempenho que outras abordagens existentes.

Categories and Subject Descriptors: 10003183 [**Deduplication**]: ; 10003351 [**Data Mining**]: ; 10003219 [**Information Integration**]:]

Keywords: Blocagem, Pareamento Probabilístico, Mineração de Dados, Resolução de Entidade

1. INTRODUÇÃO

Durante as últimas décadas observamos a construção e o contínuo crescimento das mais variadas bases de dados, em particular aquelas contendo dados sobre pessoas, sendo comum que haja registros referentes a uma pessoa em várias bases ou até na mesma base. Uma demanda cada vez mais comum e desafiadora é a resolução de entidades [Fellegi and Sunter 1969] , ou seja, identificar múltiplas ocorrências de uma entidade na mesma base (deduplicação) ou não (pareamento). Estas tarefas podem ser realizadas de forma determinística, onde um par de registros é considerado equivalente, se todos os seus atributos são exatamente iguais, ou probabilística, que permite atribuir a um par de registros uma probabilidade deste ser verdadeiro através de comparações entre os seus atributos. À primeira vista, o problema é quadrático, uma vez que potencialmente temos que comparar todos os pares de registros a serem deduplicados ou pareados, o que torna a tarefa inviável para grandes bases de dados.

Uma estratégia para reduzir a complexidade desta tarefa é a blocagem, que consiste na criação de blocos de registros que possuam maior probabilidade de representarem a mesma entidade no mundo real, ou seja, que possuam atributos em comum entre si. Os blocos são conjuntos de registros que devem ser todos comparados par a par. Caso sejam similares o suficiente, é inferido que pertencem à mesma entidade e denotam pares verdadeiros. Este artigo propõe uma estratégia de blocagem que minimiza o número de pares a serem comparados baseado na teoria dos conjuntos fechados.

Antes de descrevermos a estratégia de blocagem padrão, introduzimos o conceito de blocagem ótima, que é aquela que resulta na verificação apenas de pares verdadeiros. Considerando que os registros

a serem pareados ou deduplicados são vértices de um grafo, isso seria equivalente a determinar uma floresta geradora, onde cada árvore geradora seria uma entidade. Assim, considerando n registros referentes a m entidades, seriam necessárias apenas $n - m + 1$ comparações. Não foi possível localizar na literatura nenhum algoritmo que gere uma blocagem ótima, o que é explicado pela irregularidade do problema e o conceito subjetivo de similaridade entre registros.

A blocagem padrão [Jaro 1989] consiste em escolher alguns atributos e determinar os blocos a partir de instâncias destes atributos que sejam comuns aos registros que compõem um bloco. O problema neste caso é que nem sempre as instâncias dos atributos são discriminativas o suficiente, podendo gerar blocos muito grandes e, portanto, computacionalmente ineficientes, pelo grande número de pares falsos. Neste caso, é interessante buscar uma estratégia de blocagem que se aproximasse da blocagem ótima, o que é alcançado neste trabalho pela determinação de blocos a partir do CFI Blocking.

O CFI Blocking, do inglês *Closed Frequent Itemset*, é um algoritmo proposto por este trabalho para a enumeração de blocos em um pareamento de registros através de conjuntos frequentes fechados. Um conjunto fechado [Zaki and Hsiao 2002], neste contexto, é um conjunto de registros que satisfaz um predicado conjuntivo de instâncias de atributos. Este predicado tem a propriedade de ser o fechamento em relação a todos os predicados associados ao conjunto de registros mencionado, sendo então a descrição mais precisa e restritiva destes registros. Cada conjunto fechado define um bloco, que verifica, através da comparação de pares de registros, quais se referem a uma mesma entidade.

Em suma, o presente trabalho apresenta as seguintes contribuições: (i) a criação de uma nova estratégia de blocagem baseada em conjuntos fechados; (ii) a integração da nova estratégia ao algoritmo de pareamento probabilístico de registros; (iii) validação utilizando dados reais do Sistema Único de Saúde do Brasil.

2. TRABALHOS RELACIONADOS

Nesta seção descrevemos brevemente estratégias de blocagem populares, metodologias de avaliação dessas estratégias e estratégias de blocagem baseadas em aprendizado de máquina e mineração de dados.

O método mais popular de blocagem é a blocagem padrão, do inglês *standard blocking* (SB), que é uma técnica tradicional que agrupa registros em blocos que são idênticos a uma dada chave de blocagem separadas a nível de atributo [Jaro 1989].

Em termos de metodologias de avaliação, o trabalho de [Coeli and Camargo Jr. 2002] teve como objetivo comparar a eficiência de diferentes estratégias de blocagem e estudar a eficácia da utilização de estratégias combinadas e estratégias únicas para encontrar as melhores para o pareamento em dados de saúde pública do Brasil.

No contexto da aplicação de técnicas de aprendizado de máquina e mineração de dados ao problema de blocagem, podemos destacar os seguintes trabalhos. [de Carvalho et al. 2012] apresenta um algoritmo de programação genética para deduplicação de registros. O trabalho de [McNeill et al. 2012] utiliza uma técnica de classificação para tentar prever os melhores blocos a serem avaliados, o que é feito de forma incremental, a cada novo atributo considerado. [Kenig and Gal 2013] apresenta o primeiro algoritmo para blocagem com o uso de mineração de padrões frequentes, o MFI Blocking. Este algoritmo determina blocos baseado em conjuntos frequentes maximais, que são os maiores conjuntos que satisfazem um limiar de frequência pré-definido chamado suporte. A estratégia também realiza uma análise de qualidade dos blocos para estimar a possibilidade de existência de pares duplicados dentro de blocos distintos.

No contexto de revisões sistemáticas e análise dos métodos de blocagem existentes para o pareamento de registros. [Papadakis et al. 2016] apresenta uma revisão comparativa e classificativa dos métodos de blocagem existentes na literatura. Este trabalho classifica o MFI Blocking como um método pró

ativo de baixa escalabilidade, O CFI Blocking é semelhante a ele mas o uso de conjuntos fechados permite obter blocos mais eficientes aumentando o poder de escalabilidade, como será mostrado.

3. BLOCAGEM

Antes de falar da blocagem, é importante relembrar os passos de um pareamento de registros [Fellegi and Sunter 1969], este processo é composto de 5 etapas: limpeza e padronização dos dados, análise dos dados, blocagem, comparação e classificação. Como mencionado, a blocagem é a etapa que torna o pareamento de grandes bases de dados possível.

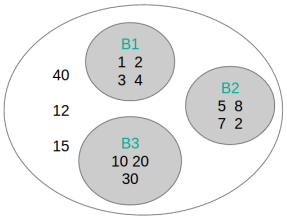


Fig. 1. Exemplos de blocos em um conjunto de registros

A Figura 1 exibe um exemplo de blocagem. Dada uma base dados A onde A = { 1, 2, 3, 4, 5, 7, 8, 10, 12, 15, 20, 30, 40 }, um exemplo de blocos pode ser B_1 ={1,2,3,4}, B_2 = {2,5,7,8} e B_3 = {10,20,30} e os registros { 12,15,40 }, neste exemplo, não se encaixam em nenhum bloco por não atenderem aos critérios estabelecidos de blocagem.

Atualmente, os métodos de blocagem utilizam o conhecimento prévio do analista, ou seja, o analista infere como os blocos devem ser enumerados através dos atributos da base de dados, o que pode resultar em falhas como, por exemplo, utilização de atributos com grande número de valores ausentes e portanto poucos blocos, ou ainda atributos com baixa qualidade, o que inviabiliza a separação correta dos registros nos grupos.

Os métodos pró-ativos, como o MFI Blocking e o CFI Blocking, se distinguem pela capacidade de enumerar os blocos de forma automatizada através da mineração da base de dados. Enquanto a blocagem padrão enumera os blocos a partir de atributos definidos pelo pesquisador, o *CFI Blocking* realiza a enumeração dos padrões frequentes fechados, determina os melhores padrões e gera os blocos para a etapa de comparação.

4. CFI BLOCKING

O *CFI Blocking* é dividido em 3 passos principais: mapeamento do banco de dados, extração dos conjuntos frequentes fechados e criação dos blocos de comparação. Além disso, deve ser considerada a etapa de comparação, que faz parte do processo final do pareamento probabilístico de registros e é utilizada neste artigo para avaliação do método proposto. Para a utilização do CFI Blocking é necessário definir um limiar de frequência denominado suporte mínimo, isto é, o número mínimo de registros que contenham determinadas instâncias de atributos para este conjunto ser considerado frequente. As estratégias CFI Blocking e MFI Blocking são similares e discutimos ambas conjuntamente nas próximas seções, para maior clareza.

Para ilustrar os passos de execução do CFI Blocking, a Tabela I é um exemplo de entrada para o algoritmo e se propõe a auxiliar no entendimento do funcionamento do mesmo, que está descrito nas seções 4.1, 4.2, 4.3. Um conjunto fechado extraído dos dados da Tabela I, {cpf=157890 & nome da mae=Sarah J Santos}, contém os registros A e B nos quais essas instâncias ocorrem. Para o pareamento de registros, isto quer dizer que os registros A e B possuem pelo menos o valor dos atributos

Table I. Tabela de Exemplo: Estado Inicial.

ID	NOME	CIDADE	ESTADO	CEP	NOME DA MÃE	DT NASC	CPF
A	Mary Jane Berg	BH	MG	12180	Sarah J Santos	18-11-1950	157890
B	Mary J Berg	BH	MG		Sarah J Santos	18-11-1950	157890
C	Maryah Jani Berg		MG	12180	Sarah J Santos		

cpf e nome da mãe em comum e por isso devem ser comparados. Como o pareamento probabilístico visa encontrar registros que se refiram à mesma entidade, é intuitivo que, quanto maior a cardinalidade do predicado de instâncias de atributos que define um conjunto fechado, mais atributos em comum existem entre os registros que contém estas instâncias e, consequentemente, maior é a probabilidade desses registros se referirem à mesma entidade.

4.1 Mapeamento do Banco de Dados

A primeira parte do *CFI Blocking* tem como objetivo assinalar um número inteiro distinto para cada valor de instância de atributo, permitindo inclusive empregar diferentes critérios de similaridade por atributo ou por instância de atributo sem que as etapas subsequentes do algoritmo sejam afetadas. Caso um mesmo valor de instância se repita em atributos distintos, valores distintos são assinalados para cada uma dessas instâncias.

ITEM	VALOR
1	Mary Jane Berg
2	Mary J Berg
3	Maryah Jani Berg
4	BH
5	MG
6	12180
7	Sarah J Santos
8	18-11-1950
9	157890

(a)

ID DA TRANSAÇÃO	ITENS						
A	1	NULO	5	6	7	8	9
B	2	4	5	NULO	7	8	9
C	3	4	5	6	7	NULO	NULO

(b)

Fig. 2. Tabela de Exemplo: Mapeamento.

A Figura 2 mostra o resultado do mapeamento aplicado na Tabela I. É possível visualizar na Figura 2(a) o conjunto de itens únicos, que é formado por $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$. A Figura 2(b), ilustra a substituição das instâncias dos atributos pelos valores assinalados aos mesmos. Por exemplo, para o atributo *cidade*, o valor BH, para os três registros, se transforma no valor 4.

4.2 Extração dos Conjuntos Frequentes Fechados (e Maximais)

Os conjuntos fechados (CFI Blocking) e maximais (MFI Blocking) foram extraídos com o uso dos algoritmos Eclat e Charm [Zaki and Hsiao 2002], respectivamente. A Figura 3 mostra os conjuntos frequentes extraídos com suporte 2, frequência mínima para se formar um par, da tabela mapeada na Figura 2. Apesar de serem padrões frequentes semelhantes, é possível observar como as suas características impactam de forma diferenciada a blocagem. Por exemplo, o conjunto {5 & 7} é um conjunto frequente fechado, mas não é um conjunto frequente maximal pois existe um subconjunto, {4 & 5 & 7}, frequente, que também é maximal. Na prática, não é possível, utilizando apenas os conjuntos maximais, fazer as comparações dos registros A,B e C com apenas um bloco, sendo necessário utilizar três blocos para realizar essa comparação, o que pode aumentar o número de comparações, tornando a abordagem baseada em maximais menos eficiente.

PADRÕES FREQUENTES			
Padrão	Transações	Padrão	Transações
4	B,C	8 & 5	A,B
5	A,B,C	8 & 7	A,B
6	A,C	9 & 5	A,B
7	A,B,C	9 & 7	A,B
8	A,B	5 & 7	A,B,C
9	A,B	4 & 5 & 7	B,C
4 & 5	B,C	6 & 5 & 7	A,C
4 & 7	B,C	8 & 9 & 5	A,B
6 & 5	A,C	8 &9 & 7	A,B
6 & 7	A,C	8 & 9 & 5 & 7	A,B
8 & 9	A,B	8 & 5 & 7	A,B
9 & 5 & 7	A,B		

(a)

PADRÕES FECHADOS	
Padrão	Transações
4 & 5 & 7	B,C
6 & 5 & 7	A,C
8 & 9 & 5 &7	A,B
5 & 7	A,B,C

(b)

PADRÕES MAXIMAIS	
Padrão	Transações
4 & 5 & 7	B,C
6 & 5 &7	A,C
8 & 9 & 5 &7	A,B

(c)

Fig. 3. Tabela de Exemplo: Extração de Conjuntos frequentes de instâncias de atributos.

4.3 Enumeração dos Blocos para Comparação

A segunda etapa do *CFI Blocking* seleciona os melhores blocos a serem avaliados. Cada conjunto fechado encontrado é considerado um possível bloco para o pareamento de registros e para selecionar quais blocos devem ser avaliados é realizado um ordenamento dos blocos considerando a maior cardinalidade em número de instâncias de atributos no conjunto e a menor cardinalidade em número de registros, nesta prioridade. A seleção é baseada em um parâmetro implementado como *limiar*, ou limite superior. Este parâmetro define o tamanho máximo, em número de registros, permitido para um bloco. Portanto, dado um bloco X, este bloco só será avaliado se o tamanho de X for menor que *limiar* e se X não for subconjunto, em termos de instâncias de atributos, de algum bloco avaliado anteriormente.

4.4 Análise de Complexidade

O mapeamento de dados tem complexidade temporal $O(n \times m)$, uma vez que para cada atributo (m) devem ser percorrido todos os registros da base de dados (n) de entrada. A complexidade é $O(n \times m)$ em espaço se todos os valores forem distintos, pois todos devem ser armazenados. Para encontrar os conjuntos frequentes fechados utilizando o algoritmo Eclat, a complexidade é de $O(n * 2^i)$, no pior caso, uma vez que podem existir 2^i conjuntos frequentes e a interseção de dois conjuntos de registros é $O(n)$. Já a complexidade de espaço é $O(2^i/i)$ [Zaki and Meira Jr 2014].

5. AVALIAÇÃO

Para avaliar o desempenho do CFI Blocking foram executados testes utilizando dados sintéticos e dados reais. A base de dados sintética foi gerada através do FEBRL [Christen 2008] contendo 5000 registros em distribuição Poisson, com 1000 registros duplicados e com alteração randômica dos valores das instâncias dos seus atributos. Foi possível perceber que o CFI Blocking foi superior ao MFI Blocking e também a blocagem padrão em precisão e revocação, obtendo até 97% de revocação enquanto os outros alcançaram 91% e 64%, respectivamente.

A base de dados reais é uma amostra referente à região metropolitana de Belo Horizonte, selecionada por referência de localidade, com aproximadamente 1.700.000 registros do Sistema Único de Saúde (SUS) do Brasil no período 2000-2010. Foram selecionados os atributos de identificação do paciente: primeiro nome, nome do meio, último nome, nome da mãe, sobrenome da mãe, data de nascimento, sexo, município de residência, cpf, cns(cartão nacional de saúde) e cep.

O *CFI Blocking* foi avaliado com o arcabouço de precisão e revocação utilizado nos trabalhos

anteriores conforme [Gu et al. 2003]. Dados p_v, o número de pares verdadeiros, p_t, o número de pares comparados totais, p_f, o número de pares falsos e p_{v_t} o número de pares verdadeiros totais, definimos a precisão p por $p = \frac{p_v}{p_t}$, ou seja a porcentagem de acertos de um pareamento/bloco. A revocação r é definida por $r = \frac{p_v}{p_{v_t}}$, que é a porcentagem de acerto nos pares verdadeiros encontrados.

Para a execução e análise do *CFI Blocking*, foram estabelecidos os valores de suporte mínimo com variação de 2 até 5 e o limiar foi variado do suporte escolhido até 20. Todos os testes foram realizados em computador com processador com 20 núcleos e 180Gb de memória RAM.

A estratégia de blocagem padrão utilizada para avaliação foi um combinado de estratégias definido por: {cns}, {primeiro nome, último nome, ano de nascimento, sexo}, {primeiro nome da mãe, sobrenome da mãe, data de nascimento}, {município de nascimento, data, sexo}, {primeiro nome, último nome, primeiro nome da mãe, último nome da mãe, sexo} que foram avaliadas no trabalho de [Coeli and Camargo Jr. 2002] e utilizadas nos trabalhos de [de Queiroz 2007], [Pereira et al. 2015].

Devido à dificuldade de se determinar todos os pares verdadeiros em bases de dados reais, estimamos este conjunto com base em todos os pares verdadeiros encontrados em todos os experimentos, mesmo aqueles que geram grande número de pares falsos. Embora seja uma aproximação, pudemos notar um comportamento assintótico em termos de novos pares para limiares maiores, o que nos permite utilizar esse conjunto de pares verdadeiros como próximo do real. Este conjunto é denominado Conjunto Estimado de Pares Verdadeiros (CEPV).

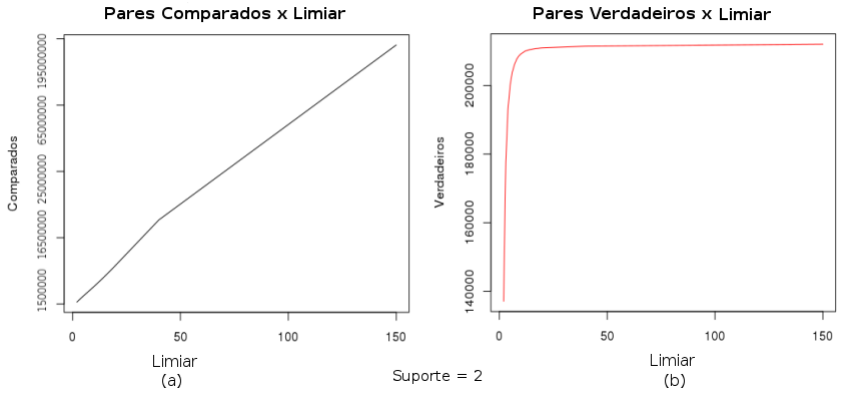


Fig. 4. Pares x limiar.

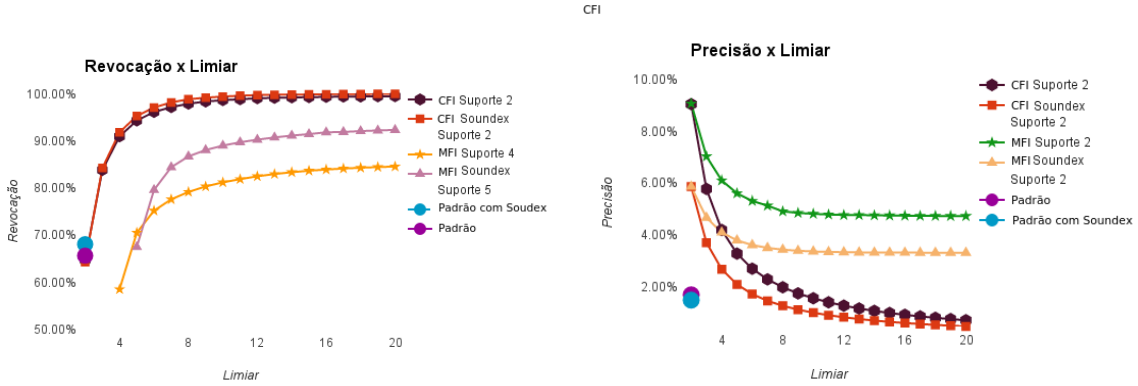
Assim, foi possível perceber conforme o gráfico da Figura 4(b), que para $limiar >= 15$, o crescimento do número de pares verdadeiros tornou-se marginal em relação ao crescimento do número de pares comparados e assumimos este valor como o valor total de pares verdadeiros existentes.

Além disso, os algoritmos também foram avaliados com o mapeamento dos dados utilizando sistema de codificação de nomes: *soundex* em sua versão brasileira implementada no trabalho de [dos Santos 2008] e utilizada pra codificação dos nomes similares. Na avaliação com *soundex*, os atributos que caracterizam nomes foram mapeados tal que nomes com probabilidade de serem o mesmo foram detectados através da fonética e foram mapeados como uma mesma instância de atributo, por exemplo, WAGNER e VAGNER foram considerados como iguais para a enumeração dos conjuntos fechados. Como impacto, é possível perceber um número menor de blocos e consequentemente um maior número de registros presente em cada bloco com a utilização do soundex.

6. RESULTADOS

Foram realizados 280 testes para as avaliações propostas. O foco dos resultados demonstrados neste trabalho está nos melhores resultados obtidos. O primeiro fenômeno interessante está capacidade de revocação dos pares verdadeiros. Para os conjuntos maximais, ao se utilizar um valor baixo de suporte, a revocação apresenta um comportamento assintótico. Isto ocorre pois conjuntos maximais com n transações restringem a formação de conjuntos maximais com suporte maior que n, ou seja, ao aumentar o limiar, o número de pares gerados adicionais é baixo, uma vez que os conjuntos já foram limitados pelo próprio suporte.

O *CFI Blocking* foi superior à blocagem padrão em termos de revocação, tendo encontrado, em seu melhor resultado, 100% do CEPV enquanto a blocagem padrão encontrou 68.47%. Esses resultados podem ser vistos nos gráficos da Figura 5(a), que representa o resultado três configurações analisadas: o CFI Blocking, os conjuntos maximais e a blocagem padrão, com os melhores resultados encontrados para cada uma. Foi possível perceber que, quanto maior o limiar, maior é a revocação alcançada por uma estratégia.



(a) (b)

Fig. 5. Precisão e Revocação x Limiar

Os resultados com melhor precisão são apresentados no gráfico da Figura 5(b). Os melhores resultados foram encontrados utilizando o suporte 2 e para quase todos os experimentos realizados o *CFI Blocking* atinge uma precisão, maior que a precisão da blocagem padrão. O melhor caso em termos de precisão (9.06%) é o MFI Blocking, utilizando os conjuntos maximais com suporte 2, acompanhado de perto pelo CFI Blocking com (9.04%), enquanto que o melhor caso da blocagem padrão alcança 1.74%. Nesta mesma figura é possível observar uma queda na precisão à medida que o limiar aumenta, isto ocorre pela característica do algoritmo no qual grandes valores de limiar permitem a avaliação de grandes blocos, o que consequentemente gera um elevado número de pares falsos, diminuindo a precisão.

Foi possível perceber que testes realizados com o mapeamento dos dados através do *soundex* são menos precisos se comparados com o mapeamento puro, ou seja, sem a utilização do *soundex*. Isto ocorre porque o agrupamento de nomes por proximidade aumenta o número de elementos em um mesmo bloco que não necessariamente são as mesmas pessoas. Em contrapartida, os testes executados com essa estratégia possuem maior revocação, isto é explicado pelo aumento do espaço de busca de pares.

O terceiro fator importante de análise é o tempo gasto pelo algoritmo para sua execução. A Tabela II contém os resultados dos melhores testes em ordem crescente do tempo de execução. Os resultados indicados nesta tabela mostram que o *CFI Blocking* foi superior em precisão e revocação

com um mesmo custo, em tempo, que a blocagem padrão. Os testes com conjuntos maximais foram compatíveis, ou seja, tiveram resultados com precisão e revocação próximos ao *CFI blocking*.

Table II. Algoritimo Suporte - Limiar x Tempo(minutos)

Conjunto	**Blocagem**	**Comparação**	**Total**	**Precisão**	**Revocação**
MaxCSoundex 5-6	0:01:29	0:02:13	0:03:42	2.57%	79.58%
MaxSSoundex 4-4	0:02:54	0:00:51	0:03:45	5.52%	58.48%
CFI CSoundex 5-6	0:01:59	0:02:05	0:04:04	2.49%	80.86%
MaxCSoundex 3-3	0:02:36	0:01:33	0:04:09	4.98%	76.86%
Padrão	0:00:45	0:03:19	0:04:04	1.70%	65.34%
Padrão CSoundex	0:00:40	0:04:08	0:04:48	1.49%	68.05%

7. CONCLUSÃO

A enumeração de blocos através do *CFI Blocking* se mostrou eficaz e eficiente sendo superior a blocagem padrão e seu desempenho superou conjuntos maximais. Os fatores determinantes de escolha entre os algoritmos foram o tempo de execução e o espaço de busca pois o desempenho, em termos de precisão e revocação, para os conjuntos fechados e maximais é semelhante. Os melhores valores para o CFI Blocking foram encontrados com suporte 3 e limiar até 5 para a base de dados real utilizada. Como trabalhos futuros sugere-se estudos para permitir execução de forma paralela e reduzir o tempo gasto na enumeração dos conjuntos fechados, além de estratégias de ordenação dos blocos que aumentem a precisão do processo de blocagem.

REFERENCES

Christen, P. Febrl: A freely available record linkage system with a graphical user interface. In *Proceedings of the Second Australasian Workshop on Health Data and Knowledge Management - Volume 80*. HDKM '08. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 17–25, 2008.

Coeli, C. M. and Camargo Jr., K. R. Avaliação de diferentes estratégias de blocagem. Vol. 5. Rev. Bras. Epidemiol., 2002.

de Carvalho, M. G., Laender, A. H. F., Gonçalves, M. A., and da Silva, A. S. A genetic programming approach to record deduplication. *IEEE Trans. Knowl. Data Eng.* 24 (3): 399–412, 2012.

de Queiroz, O. V. Relacionamento probabilístico de registros na integracao de sistemas de informacao do sus: O caso da base nacional de dados em terapia renal substitutiva, 2007.

dos Santos, W. Algoritmo paralelo e eficiente para o problema de pareamento de dados, 2008.

Fellegi, I. and Sunter, A. A theory for record linkage, 1969.

Gu, L., Baxter, R., Vickers, D., and Rainsford, C. Record linkage: Current practice and future directions. Tech. rep., CSIRO Mathematical and Information Sciences, 2003.

Jaro, M. A. Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. Journal of the American Statistical Association 84:414–420, 1989.

Kenig, B. and Gal, A. Mfiblocks: An effective blocking algorithm for entity resolution. *Inf. Syst.* 38 (6): 908–926, Sept., 2013.

McNeill, B., Kardes, H., and Borthwick, A. Dynamic record blocking: Efficient linking of massive databases in mapreduce. In *10th International Workshop on Quality in Databases (QDB), in conjunction with VLDB 2012*, 2012.

Papadakis, G., Svirsky, J., Gal, A., and Palpanas, T. Comparative analysis of approximate blocking techniques for entity resolution. *PVLDB* 9 (9): 684–695, 2016.

Pereira, R. G., Leal, G. S., Dias, L. V., Acúrcio, F., Junior, A. A. G., Cherchiglia, M., and Gurgel, E. I. Unified database creation applied to public brazilian health information systems from the hospital, outpatients and mortality information systems. In *ISPOR 20th Annual International Meeting*, 2015.

Zaki, M. J. and Hsiao, C.-J. CHARM: An efficient algorithm for closed itemset mining. In *2nd SIAM International Conference on Data Mining*, 2002.

Zaki, M. J. and Meira Jr, W. *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press, New York, NY, USA, 2014.

Avaliação dos Ganhos de Combinação de Múltiplas Coleções de Documentos em Recuperação de Informação

Felipe de Almeida Costa[1] e Wagner Meira Júnior[1]

Universidade Federal de Minas Gerais, Brazil
{felipealco,meira}@dcc.ufmg.br

Abstract. Com o advento da Web, a produção de conteúdo quebra novos recordes todos os anos. O baixo custo do acesso a tecnologia permite a existência de muitos colaboradores nessa grande fábrica de conteúdo que é a Web. Qualquer um destes colaboradores pode produzir sua própria versão de um conhecimento. Por mais cuidadosos que sejam, é natural que cada um adicione um viés próprio na sua versão. Muitos são documentos semanticamente equivalentes, em alguns casos coleções inteiras de documentos. A diversidade das versões de documentos semanticamente equivalentes levanta a dúvida sobre qual versão escolher para a construção de um Sistema de Recuperação de Informação. Para contornar essa questão sugerimos a utilizar todas versões através da combinação de resultados. A coleção de documentos que escolhemos foi a Bíblia Sagrada, pois além da facilidade da obtenção dos dados e sólida estruturação, apresenta a propriedade de simetria entre versões. A avaliação dos resultados foi feita usando revocação (*recall*) dos resultados relevantes em rankings de tamanho um, cinco e dez. Resultados comprovam que a combinação entre versões de coleções tem melhor revocação quando comparada às versões indexadas individualmente.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords: recuperação da informação, combinação de scoring, similaridade entre coleções de documentos

1. INTRODUÇÃO

No contexto de Recuperação de Informação, os itens indexados são considerados documentos. No caso de uma busca por imagens, cada imagem é um documento. Em nosso trabalho definimos como versão um documento com valor semântico. Quando existem dois documentos de semanticamente equivalentes dizemos que estes documentos são múltiplas versões de uma representação semântica. No exemplo de busca de imagens poderíamos dizer que essas múltiplas versões seriam imagens iguais de tamanhos distintos ou ainda imagens iguais de diferente tons de coloração.

Os cenários de documentos de múltiplas versões são diversos. Na Web muitas são as páginas que se repetem semanticamente. Por exemplo, uma notícia sobre um acontecimento que relatada por diferentes portais de notícias. Cada jornalista tem seu próprio vocabulário e seu estilo de escrita, criando assim diferentes versões de um mesmo acontecimento que neste trabalho consideramos ser múltiplas versões de um documento, no caso a notícia em si. Múltiplas versões de documentos ocorrem também com frequência em traduções, pois durante o processo de tradução o tradutor precisa adaptar expressões e contextos para facilitar o entendimento do leitor. Em geral, versões de documentos estão presentes onde existem diferentes perspectivas de um mesmo conhecimento.

A diversidade de versões implica na diversidade de Sistemas de Recuperação de Informação, de forma que para cada versão tem-se um indexador diferente ou uma estratégia de busca diferente, ou ainda ambos. O resultado da busca pode variar dependendo da versão indexada. Portanto, a escolha da versão altera o desempenho do buscador, caracterizando um problema de decisão arquitetural.

A arquitetura tradicional de um Sistema de Recuperação de Informação (SRI) é composta por uma Consulta (C), que processada torna-se uma Formulação (F) [Van Rijsbergen 1986]. Essa formulação é submetida a um Esquema (E) que por sua vez retorna uma Lista dos Documentos (Ld) em ordem de relevância para a consulta dada. Essa relevância é definida com um score calculado para cada documento.

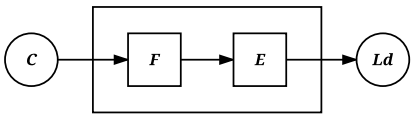


Fig. 1: Processo de Recuperação de Informação.

O que diferencia os SRIs é o método para Formulação da Consulta e a abordagem do Esquema. A diversidade nos métodos e nas abordagens torna possível a melhora dos SRIs utilizando técnicas de combinação de resultados. Hsu e Taksa definiram três tipos de SRIs com arquitetura combinada: (i) múltiplas formulações num único esquema (MFSS), (ii) única formulação em múltiplos esquemas (SFMS) e (iii) múltiplas formulações em múltiplos esquemas (MFMS) [Frank Hsu and Taksa 2005].

Este trabalho analisa uma abordagem para a arquitetura de única formulação em múltiplos esquemas (SMFS), onde cada esquema indexa uma versão diferente do documento. O objetivo dessa abordagem é verificar se a combinação de bases melhora o desempenho do SRI preservando o ganho de informação de cada versão. A coleção de documentos que escolhemos neste trabalho foi a Bíblia Sagrada, pois além da facilidade da obtenção dos dados e sólida estruturação, apresenta também a propriedade de simetria entre versões.

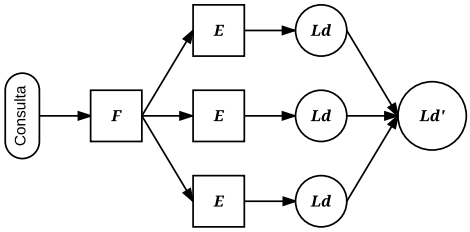


Fig. 2: Arquitetura de única formulação em múltiplos esquemas.

Zhang et al. (2002) formularam uma combinação de dois esquemas com diferentes modelos (SMART e Okapi) e verificaram uma melhora de até 17% comparados aos desempenhos individuais. Li et al. (2011) verificaram o desempenho de 7 diferentes modelos e as combinações possíveis de tamanho até 4 utilizando combinação de rank e combinação de score. Com destaque para a combinação de score que obteve melhores resultados.

2. A BÍBLIA SAGRADA

A Bíblia Sagrada é uma coleção de livros e cartas com relatos relevantes ao cristianismo. Independente do aspecto religioso, a Bíblia é um livro de grande importância na cultura ocidental. Está entre os livros mais vendidos de todos os tempos. Disponível em mais de 500 línguas, a Bíblia é também o livro mais traduzido. Destaca-se também por ter sido o primeiro livro impresso da história.

Ao longo da história, a estrutura Bíblia foi organizada diversas vezes. A versão disponível hoje possui dois testamentos chamados *Velho Testamento* e *Novo Testamento*. Cada testamento é composto por uma série de livros divididos em capítulos e subsequentemente em versículos.

3. LUCENE JAVA FRAMEWORK

Lucene é um framework de busca e indexação de documentos escrito em Java. Lucene combina duas técnicas de Recuperação de Informação para calcular o score dos documentos. A primeira é o Vector Space Model (VSM) de Recuperação da Informação, técnica dominante em Recuperação de Informação, que representa a consulta e os documentos como vetores para então definir uma similaridade [Zhang and Ma 2002]. A outra técnica é o Standard Boolean Model, baseado em Teoria de Conjuntos e Álgebra Booleana [Lucene 3.6.1 API] .

4. METODOLOGIA

Para verificar o desempenho de SRI de esquemas de múltiplas versões utilizamos cinco versões do Novo Testamento da tradução em inglês: American Standard Version (ASV), King James Version (KJV), Weymouth (WNT), World English Bible (WEB) e Young's Literal Translation (YLT). O Novo Testamento possui 7957 versículos, em nosso SRI representados como documentos. Assim, a lista de documentos é uma lista dos versículos mais relevantes.

Toda a análise foi feita baseada nos resultados de consultas. Para simplificar a interpretação dos resultados definimos uma consulta sendo um versículo, dessa forma é razoável considerar o esquema eficiente se o versículo alvo aparace em primeiro no ranking. Então, uma vez que um versículo é selecionando aleatoriamente, tem-se cinco consultas distintas, uma para cada versão. Essas consultas submetidas a cinco esquemas, um para cada versão indexada.

Para análise selecionamos aleatoriamente 20 versículos de livros distintos. Considerando 5 consultas para cada versículo tem-se ao todo 100 consultas distintas. Submetidas a 5 esquemas, um para cada versão, obtém-se 500 rankings. O procedimento detalhado é mostrado no Pseudocódigo 1.

Algorithm 1 Psedudocódigo 1.

```
1: lucene(c, i) ← retorna a lista de documentos da consulta c no índice
2: versiculos ← lista dos versículos aleatórios
3: versoes ← lista das versões
4: for v in versiculos do
5:     for i in versoes do
6:         for j in versoes do
7:             lucene(i.getVersiculo(v), j)
8:         end for
9:     end for
10: end for
```

4.1 Correlação entre versões

A correlação entre as versões foi obtida utilizando o score do documento alvo em cada consulta. Para cada versão obtivemos um vetor de tuplas onde cada posição representa o par: (versiculoAlvo,score). A matriz de correlação obtida é a da Tabela 1.

	ASV	KJV	WNT	WEB	YLT
ASV	1.000	0.350	-0.290	0.240	-0.071
KJV	0.350	1.000	-0.300	-0.062	0.172
WNT	-0.290	-0.300	1.000	-0.255	-0.304
WEB	0.240	-0.062	-0.255	1.000	-0.179
YLT	-0.071	0.172	-0.304	-0.179	1.000

Table I: Matriz de Correlação entre as versões.

4.2 Combinação dos Resultados

Embora existam diversos métodos para combinação de score e ranking na literatura, optamos por usar a média simples, pois o nosso foco é no cenário de múltiplas versões, independente da abordagem do esquema e do método usado na combinação.

Com a combinações das 5 versões obtém-se 31 rankings, isto é, todas as $2^5 - 1$ possibilidades.

4.3 Exemplo de combinação

Considerando o versículo Romans2_18 da versão Weymouth uma consulta C.

> "and know the supreme will, and can test things that differ–being a man who receives instruction from the Law–" Romans 2:18

Seja E_v o esquema que utiliza a versão v. Para simplificar a visualização dos resultados dos esquemas utilizamos a tabela II, que na primeira coluna indica o esquema utilizado, na segunda o versículo com maior score no resultado, na terceira o score do maior versículo, na quarta o score do versículo alvo e na última coluna a posição do versículo alvo no ranking.

Esquema	Top 1 Documento	Score Top 1 Documento	Score Documento Alvo	Posição no Ranking
E_{ASV}	John4_29	0.23473	0.15016	8
E_{KJV}	1 Corinthians2_14	0.22181	0.11968	22
E_{WNT}	**Romans2_18**	**5.16255**	**5.16255**	**1**
E_{WEB}	Galatians5_3	0.33823	0.16893	2
E_{YLT}	John7_51	0.23941	0.16893	3

Table II: Resultados de Romans2_18.

O único esquema que obteve o versículo alvo na primeira posição foi o esquema que utilizou a versão Weymouth, como esperado, uma vez que a consulta é idêntica ao versículo alvo. A combinação dos resultados dois a dois é apresentada na tabela III.

Combinação 2-2	Top 1 Documento
E_{ASV} e E_{KJV}	Mark7_15
E_{ASV} e E_{WNT}	**Romans2_18**
E_{ASV} e E_{WEB}	Romans7_1
E_{ASV} e E_{YLT}	John7_51
E_{KJV} e E_{WNT}	**Romans2_18**
E_{KJV} e E_{WEB}	Mark7_15
E_{KJV} e E_{YLT}	John7_51
E_{WNT} e E_{WEB}	**Romans2_18**
E_{WNT} e E_{YLT}	**Romans2_18**
E_{WEB} e E_{YLT}	**Romans2_18**

Table III: Combinação 2-2 dos Resultados de Romans2_18.

O destaque na tabela III é para a combinação E_{WEB} e E_{YLT} que separadamente erraram no sentido de que não priorizaram o versículo alvo na primeira posição, mas quando combinados acertaram.

Para as combinações 3-3 e 4-4 (Tabela IV) percebemos que, à medida que novas versões são adicionadas, os acertos tornam-se mais frequentes. Como esperado, a combinação envolvendo todas as versões do modelo também acerta o versículo alvo.

Combinação 3-3	Top 1 Documento
E_{ASV}, E_{KJV} e E_{WNT}	**Romans2_18**
E_{ASV}, E_{KJV} e E_{WEB}	Mark7_15
E_{ASV}, E_{KJV} e E_{YLT}	John7_51
E_{ASV}, E_{WNT} e E_{WEB}	**Romans2_18**
E_{ASV}, E_{WEB} e E_{YLT}	**Romans2_18**
E_{ASV}, E_{WNT} e E_{YLT}	**Romans2_18**
E_{KJV}, E_{WNT} e E_{WEB}	**Romans2_18**
E_{KJV}, E_{WNT} e E_{YLT}	**Romans2_18**
E_{KJV}, E_{WEB} e E_{YLT}	**Romans2_18**
E_{WNT}, E_{WEB} e E_{YLT}	**Romans2_18**

Combinação 4-4	Top 1 Documento
E_{ASV}, E_{KJV}, E_{WNT} e E_{WEB}	**Romans2_18**
E_{ASV}, E_{KJV}, E_{WNT} e E_{YLT}	**Romans2_18**
E_{ASV}, E_{KJV}, E_{WEB} e E_{YLT}	Mark7_15
E_{ASV}, E_{WNT}, E_{WEB} e E_{YLT}	**Romans2_18**
E_{KJV}, E_{WNT}, E_{WEB} e E_{YLT}	**Romans2_18**

Table IV: Combinações 3-3 e 4-4 dos Resultados de Romans2_18.

5. AVALIAÇÃO

O objetivo da avaliação é comparar o desempenho dos SRIs tradicionais (com uma única versão) com as combinações de tamanho 2 até 5. O desempenho do SRI é medido através da fórmula de revocação tradicional:

$$\text{revocação} = \frac{|\text{documentos relevantes}| \cup |\text{documentos recuperados}|}{|\text{documentos relevantes}|}$$

Em nossa avaliação consideramos apenas o versículo alvo como documento relevante, simplificando a fórmula da revocação para:

$$\text{revocação} = \begin{cases} 1 & \text{versículo alvo} \in \text{documentos recuperados} \\ 0 & \text{caso contrário} \end{cases}$$

Como mencionado anteriormente, cada esquema avaliou 100 consultas distintas. Na tabela V cada linha representa o resultado de revocação de um esquema para 1, 5 e 10 documentos recuperados. Comparativamente percebemos que, para o caso em que o documento aparece na primeira posição (top-1), o esquema da versão World English Bible tem o melhor desempenho enquanto a versão Weymouth o pior.

Esquema	top-1	top-5	top-10
E_{ASV}	92	93	94
E_{KJV}	92	94	95
E_{WNT}	80	91	91
E_{WEB}	93	96	97
E_{YLT}	88	94	94

Table V: Revocação dos esquemas inicias.

Todas as combinações 2-2 apresentam melhora quantitativa na revocação de resultados top-5, independente da correlação entre as coleções (Subseção 4.1). Para a revocação top-1 há uma redução para a combinação E_{KJV} e E_{YLT}. Ambos resultados, de aumento e redução, mostram-se independentes da correlação entre os documentos, o que é um indício de que os ganhos associados à combinação de múltiplas coleções não são facilmente estimáveis.

Na Tabela VII as combinações 3-3 alcançam resultados próximos à perfeição, e isso se repete nas combinações 4-4 e na combinação de todas as versões. De forma que percebemos haver um limite superior na quantidade de versões que aumenta o ganho de informação no SRI.

Combinação 2-2	Correlação	$max(E_1, E_2)$ top-5	top-1	top-5	top-10
E_{ASV} e E_{KJV}	0.35	93	95	97	97
E_{ASV} e E_{WNT}	-0.29	93	92	98	98
E_{ASV} e E_{WEB}	0.24	96	94	97	97
E_{ASV} e E_{YLT}	-0.071	94	93	98	98
E_{KJV} e E_{WNT}	-0.3	94	93	97	97
E_{KJV} e E_{WEB}	-0.062	96	97	99	100
E_{KJV} e E_{YLT}	0.172	94	89	98	98
E_{WNT} e E_{WEB}	-0.255	96	92	99	99
E_{WNT} e E_{YLT}	-0.304	94	91	99	99
E_{WEB} e E_{YLT}	-0.179	96	96	99	99

Table VI: Revocação dos esquemas combinados 2-2.

Combinação 3-3	top-1	top-5	top-10
E_{ASV}, E_{KJV} e E_{WNT}	95	99	99
E_{ASV}, E_{KJV} e E_{WEB}	96	99	99
E_{ASV}, E_{KJV} e E_{YLT}	94	98	98
E_{ASV}, E_{WNT} e E_{WEB}	93	99	99
E_{ASV}, E_{WNT} e E_{YLT}	94	99	99
E_{ASV}, E_{WEB} e E_{YLT}	95	99	99
E_{KJV}, E_{WNT} e E_{WEB}	96	100	100
E_{KJV}, E_{WNT} e E_{YLT}	91	98	99
E_{KJV}, E_{WEB} e E_{YLT}	96	99	99
E_{WNT}, E_{WEB} e E_{YLT}	94	100	100

Combinação 4-4	top-1	top-5	top-10
E_{ASV}, E_{KJV}, E_{WNT} e E_{WEB}	96	100	100
E_{ASV}, E_{KJV}, E_{WNT} e E_{YLT}	94	99	99
E_{ASV}, E_{KJV}, E_{WEB} e E_{YLT}	95	99	99
E_{ASV}, E_{WNT}, E_{WEB} e E_{YLT}	95	100	100
E_{KJV}, E_{WNT}, E_{WEB} e E_{YLT}	95	100	100

Table VII: Revocação dos esquemas combinados 3-3 e 4-4.

6. CONCLUSÃO E TRABALHOS FUTUROS

Como mostrado experimentalmente, a combinação de evidências implica na melhoria do resultado. Podemos com este estudo concluir que combinação de versões de coleções de documentos melhora o desempenho de SRIs.

Esse resultado abre caminho para outras discussões. A maior delas é descobrir quais fatores implicam no desempenho do SRI, para o qual sugerimos a análise de técnicas de similaridade entre documentos aplicada à coleções de documentos. Outra questão em aberto é o limite no ganho de informação pela adição de novas versões, pois se antes o problema de decisão arquitetural era decidir qual versão usar, agora ele se torna decidir quantas versões usar. A Bíblia ainda abre mais possibilidades para combinação de resultados. Saracevie e Kantor (1988) comprovaram que múltiplas formulações de consultas tem melhor desempenho que uma única formulação. Uma possibilidade seria utilizar vários versículos como uma única consulta, uma vez que existe uma correspondência semântica entre versículos distintos numa mesma versão.

REFERENCES

Similarity, (Lucene 3.6.1 API). Disponível em `https://lucene.apache.org/core/3_6_1/api/core/org/apache/lucene/search/similarity.html`, Acesso em 10-Agosto-2016.

Frank Hsu, D. and Taksa, I. Comparing rank and score combination methods for data fusion in information retrieval. *Information Retrieval* 8 (3): 449–480, 2005.

Li, Y., Shi, N., and Hsu, D. F. Fusion analysis of information retrieval models on biomedical collections. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*. pp. 1–8, 2011.

Saracevic, T. and Kantor, P. A study of information seeking and retrieving. iii. searchers, searches, and overlap. *Journal of the American Society for Information Science* 39 (3): 197–216, 1988.

Van Rijsbergen, C. J. A new theoretical framework for information retrieval. *SIGIR Forum* 21 (1-2): 23–29, Sept., 1986.

Zhang, M. and Ma, S. Efficient information retrieval based on a combination of vector space and probabilistic models. In *Systems, Man and Cybernetics, 2002 IEEE International Conference on*. Vol. 2. pp. 466–470, 2002.

Previsão de horários dos ônibus do sistema de transporte público coletivo da cidade de Campina Grande

M. A. Maciel[1], N. Andrade[1], L. B. Marinho[1], H. B. Carlos[2]

[1] Universidade Federal de Campina Grande, Brasil
matheusmaciel@copin.ufcg.edu.br, {nazareno, lbmarinho}@dsc.ufcg.edu.br
[2] Superintendência de Trânsito e Transportes Públicos, Brasil
helder@ciomcg.com.br

Abstract.

A previsibilidade dos serviços de transporte público é um aspecto central para a melhoria da experiência de seus usuários. Contudo, por funcionar dentro de um ambiente estocástico, essa previsibilidade é tipicamente prejudicada. Neste trabalho investigamos a possibilidade de tornar um sistema de transporte público mais previsível através do uso das informações históricas em um contexto onde não há disponível tecnologia de localização tempo real dos veículos ou informação atualizada sobre a operação do serviço. Embora GPS e outras tecnologias de *Automatic vehicle location* (AVL) em tempo real existam, muitos municípios brasileiros não as têm disponíveis. Considerando essa situação, utilizamos dados do sistema de ônibus da cidade de Campina Grande para avaliar o desempenho de quatro algoritmos de regressão na tarefa de prever no início do dia como os horários programados para os ônibus serão cumpridos. Os resultados apontam que embora a falta de informação em tempo real prejudique a capacidade preditiva dos algoritmos em determinadas situações, utilizá-los torna possível a previsão dos horários de saída reais dos ônibus com erro mediano de 28 segundos, e a previsão dos horários de fim de viagem com erro de mediano de -167 segundos.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data Mining*; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: Bus Arrival Time, Intelligent Transportation Systems, Machine Learning, Prediction

1. INTRODUÇÃO

A pontualidade dos ônibus é um fator importante tanto para nível de satisfação do usuário quanto para a entidade gestora do serviço [Eboli and Mazzulla 2011]. Ainda assim, pode ser complicado garantir um nível de pontualidade aceitável devido a aleatoriedade do sistema em que o serviço está inserido [Washington et al. 2010; Bazzan and Klügl 2013]. Por esse motivo, o planejamento e a tomada de decisão tem um papel crucial tanto na gestão do sistema quanto na utilização dele. Nesse momento *Intelligent transportation systems* (ITS) usam TI para gerar informação útil para o cidadão e para o gestor. Alguns exemplos de ITS bastante utilizados são sistemas de navegação encontrados em carros, lombadas eletrônicas e localização automática de veículos usando sensores GPS.

No contexto do transporte público, especificamente no cenário dos ônibus, um dos problemas mais conhecidos é o do cumprimento dos horários estabelecidos. Por ser um problema que em muitos casos é causado por fatores que não podem ser controlados, nem sempre é possível evitar que ele aconteça. Vários estudos aplicaram aprendizado de máquina e estatística com o objetivo de prever a ocorrência desses eventos [Baptista et al. 2012; Chen et al. 2009; Padmanaban et al. 2010; Gong et al. 2013]. Tal previsão permite que os interessados tenham essa informação com antecedência e possam decidir como contornar a situação.

No caso de Campina Grande, os ônibus não possuem nenhum sistema de localização automática. A execução das viagens é coletada em pontos de fiscalização no inicio e no final da viagem. Além disso, toda a informação coletada durante o dia só é disponibilizada no dia seguinte. Por isso, só é possível saber o horário em que o ônibus passou em um desses pontos até o dia anterior. Como todas as abordagens de previsão consolidadas utilizam basicamente dados de localização em tempo real, não existem evidências de que os métodos mais usados na previsão de horário dos ônibus conseguem resultados relevantes em cidades como Campina Grande.

O objetivo desse trabalho é analisar a qualidade das previsões do horário de inicio e fim de todas viagens de um determinado dia usando quatro métodos de predução comprovadamente eficientes em diversos domínios: *k-Nearest Neighbors* (k-NN), *Artificial Neural Networks* (ANN), *Support Vector Machines* (SVM) e *Random Forests* (RF). As previsões foram baseadas unicamente em dados históricos. Os algoritmos foram avaliados com base nos seguintes fatores: quantidade de dias usados no treino, tamanho do histórico de uma viagem, dia da semana e mês do ano.

Foi verificado que tanto para o inicio quanto para o fim da viagem a mediana dos dos erros foi de aproximadamente 28 e -167 segundos respectivamente. As análises indicam que os fatores apresentados influenciam o erro mediano, tendo a hora do dia e o mês do ano como os fatores que geram maior variação.

2. TRABALHOS RELACIONADOS

A previsão de variáveis relacionadas a sistemas de transporte é um problema bastante estudado em diferentes perspectivas e com diferentes abordagens. A utilização de localização em tempo real é um fator dominante em todas as abordagens, entretanto, algumas delas também usam dados históricos como entrada para os modelos de previsão.

Liu et al. [Liu et al. 2012] avaliam a utilização do algoritmo *k-Nearest Neighbors* (k-NN) na previsão dos horários de chegada e saída, tempo parado e atraso em cada uma das paradas do ônibus número 16 da cidade de Pequim, na China. Para tal tarefa, foi necessário uma etapa de pré-processamento dos dados de localização em tempo real, para resolver o problema de dados ausentes. Eles ainda comparam a abordagem em questão com a utilização de *Artificial Neural Networks*(ANN) no mesmo contexto.

Jeong e Rillet [Jeong and Rilett 2004] usam dados históricos mais informações em tempo real para prever o horário de chegada na parada dos ônibus da rota 60 da cidade de Houston no Estados Unidos. O objetivo dos autores foi a comparação da solução proposta usando *Artificial Neural Networks* com soluções usadas na época como Regressão Linear Múltipla. Foram utilizados dados históricos da execução da rota 60 como base da previsão e as informações de localização em tempo real como ajuste.

Vanajakshi e Rilett [Vanajakshi and Rilett 2004] comparam a utilização de *Artificial Neural Networks* e *Support Vector Machines* (SVM) utilizando dados históricos para a previsão da velocidade do ônibus. Os autores mostraram que SVM supera ANN quando a quantidade de dados usados no treino é menor e que o aumento dessa quantidade faz com que ANN obtenha melhores resultados.

Wang et al. [Wang et al. 2009] avaliou o desempenho da utilização de *Support Vector Regression*, com *Radial Basis Function* (RBF) como kernel do modelo, na previsão da duração do deslocamento entre paradas. Os autores utilizaram o horário de saída da primeira parada da viagem como base da previsão e adicionaram informações físicas e históricas de cada trecho entre os pares de paradas da rota estudada. A única informação em tempo real necessária é o horário de passagem do ônibus pela primeira parada do percurso.

Bejan et al. [Bejan et al. 2010] analisou a execução dos ônibus da cidade de Cambridge, no Reino Unido, que passam por uma via chamada Histon Road. Os autores verificaram que fatores como

dia da semana e hora do dia podem influenciar a operação do serviço. Eles ainda mostram que essa variação pode aumentar se o período do ano em questão não for de férias na escolas.

3. PROBLEMA

Uma consideração central em nosso trabalho é a indisponibilidade de informação em tempo real ou de informação histórica detalhada sobre o posicionamento dos ônibus na cidade. Embora existam tecnologias para tanto – a exemplo do GPS – a administração de várias cidades brasileiras não possui acesso a uma infraestrutura implantada com GPS ou aos dados de controle da frota usada em suas cidades. Em maio de 2016, João Pessoa, Natal e Campina Grande ainda figuram como exemplos de municípios com essa limitação na infraestrutura de monitoramento. Além disso, na ocasião da disponibilidade dessa informação, os modelos baseados na informação histórica e menos detalhada que utilizamos neste estudo podem ser usados em complemento à informação de localização dos veículos em tempo real.

Neste contexto, a informação que consideramos disponível a partir do sistema de transporte público consiste apenas dos horários de saída e chegada executados pelos ônibus até a meia noite do dia anterior ao momento da previsão, e da informação de quantos passageiros foram transportados em cada viagem. Essas informações tipicamente estão disponíveis nas administrações públicas por necessitarem de um esforço de implantação menor, dependerem de tecnologias desenvolvidas relativamente simples, e por serem centrais à distribuição da renda do sistema entre as empresas concessionárias do serviço de transporte público. Embora em alguns contextos a informação do horário de início e fim das viagens de cada ônibus possa estar disponível em tempo real, consideramos aqui um cenário de pior caso e baseado na realidade de Campina Grande - PB no momento da condução deste trabalho.

Especificamente, as tarefas de previsão que estudamos consistem de, dados (i) o histórico de viagens executadas até a meia noite do dia anterior, (ii) informações sobre o dia atual tais como mês, dia da semana e clima, e (iii) uma viagem programada para o dia atual, prever (a) o horário de início da viagem tal qual ela será realizada e (b) o horário de seu fim. De posse dessas duas informações é possível interpolar previsões para saber em que momento o ônibus realizando a viagem estará em cada ponto da rota. Por fim, a medida de erro que consideramos que o modelo de previsão deve minimizar é a raiz quadrada do erro quadrático médio (RMSE).

4. DADOS E ALGORITMOS

Os dados utilizados no estudo foram obtidos em uma parceria com a Superintendência de Trânsito e Transportes Públicos de Campina Grande (STTP-CG). Esses dados consistem de dois tipos de informação sobre o serviço de ônibus da cidade: o quadro de horários determinados pela STTP-CG e os horários executados pelo ônibus em cada viagem realizada. O primeiro especifica o horários de inicio e fim que uma determinada viagem deve cumprir. Esse valores são determinados com base na demanda dos locais que a viagem atende como também no conhecimento prévio dos responsáveis pela gerência e planejamento do serviço.

Como Campina Grande não possui informação sobre a localização dos ônibus em tempo real, a única maneira de obter informação sobre as viagens é a partir dos registros de início e fim de viagem realizados pelos motoristas ou fiscais. Essas informações são coletadas através do validador usado para o pagamento da tarifa. Sempre que uma viagem é iniciada ou finalizada o motorista, o cobrador ou o fiscal tem a obrigação de registrar o fato no validador. Com essa prática é possível obter horário de inicio e fim de cada viagem, além de informações como motorista, quantidade de passageiros, quantidade de passageiros que pagaram meia passagem e quantidade de passageiros gratuitos.

4.1 Pareamento entre programação e viagens realizadas

Nos dados de horários de viagens realizadas não há uma marcação que especifique qual viagem do quadro de horários corresponde a uma viagem feita por um ônibus. Por isso, um primeiro passo em nossa análise é parear as viagens executadas aos horários programados. Foi utilizado para isso o mesmo algoritmo de pareamento usado pela STTP-CG. O algoritmo define que uma viagem programada deve ser pareada à viagem realizada com menor diferença de tempo entre seus horários de início, atendendo a condição de que tal diferença não pode exceder valor do *headway*. O *headway* de uma viagem $\mathbf{x}$ é a diferença do horário de início de $\mathbf{x}$ e de sua viagem seguinte. Também é necessário salientar que uma viagem programada só pode ser pareada com uma realizada e vice-versa.

4.2 Viagens excessivamente longas

A coleta dos horários de inicio e fim dependem do registro por parte do motorista, do cobrador ou do fiscal no validador existente no veículo. Contudo, existem situações em que as viagens não são finalizadas criando viagens com duração anormal. Essas viagens "enormes" na verdade são várias viagens que não foram finalizadas e iniciadas corretamente. Esse comportamento gera um falso atraso na execução do horário final da viagem e por isso elas foram desconsideradas no experimento. Sendo assim, todas as viagens com duração maior que 2,5 vezes sua duração programada foram removidas do conjunto de dados. Esse fator foi definido com ajuda do órgão gestor do serviço em Campina Grande.

4.3 Atributos

Além dos horários realizados, utilizamos os seguintes atributos das viagens como preditores nos modelos.

4.3.1 *Categóricos.* Estão nos dados viagens de 46 rotas que podem ser agrupadas em 17 linhas. Seis empresas foram responsáveis pela execução dessas rotas utilizando um total de 469 veículos. Cada viagem também é classificada pelo dia da semana, dia normal, férias e se ela foi realizada a noite. Para a STTP-CG são considerados dias normais terça, quarta e quinta-feira. Definimos também que todas as viagens que ocorreram em Janeiro, Junho, Julho e Dezembro pertencem à categoria de realizadas no período de férias. Por último, se a viagem teve inicio após às 17:59 ela é categorizada como tendo acontecido à noite.

4.3.2 *Lotação.* A quantidade de pessoas dentro do ônibus durante a viagem pode influenciar seu cronograma tanto diretamente quanto indiretamente. O ônibus tende a ter mais tempo parado se necessitar embarcar e desembarcar um número elevado de passageiros. Além disso, a lotação pode ser um indicador da demanda de locomoção da cidade no momento da viagem, por isso, se a demanda é alta muitos veículos devem estar circulando o que pode gerar demoras no trânsito.

4.3.3 *Meteorológicos.* As condições meteorológicas no momento da viagem podem ter relação com as condições das vias, tempo de embarque e desembarque de passageiros e quantidade de passageiros. Todos esses fatores influenciam a execução do serviço por parte dos ônibus. Por esses motivos, três dos atributos de uma viagem referem as condições climáticas do dia em que a viagem ocorreu, são eles: precipitação total, temperatura média e umidade média. Esses estão disponíveis no banco de dados observacionais do Centro de Previsão de Tempo e Estudos Climáticos [CPTEC/INPE 2016].

4.3.4 *Temporais.* Parte dos atributos de uma viagem são cronológicos: dia do mês, mês do ano e o horário de início programado. Cada viagem também possui dados referentes as viagens respectivas em dias anteriores. Viagens respectivas são aquelas que foram pareadas com o mesmo horário programado. Sendo $v_{r,i,d}$ a viagem do dia d da rota r que foi pareada com a viagem programada i, $v_{r,i,d}$

possui horário de inicio/fim observado, quantidade de passageiros, atraso inicial e final, precipitação, temperatura e umidade das viagens $v_{r,i,(d-k)}$ onde $k \in 1, 2, 3$.

4.4 Análise descritiva da variável de resposta

Os modelos treinados usando os atributos citados acima tem como objetivo prever dois valores: a pontualidade referente ao início da viagem e a pontualidade do final da viagem. Ao observar a variação desses valores nos dados é possível notar que em mediana os ônibus começam as viagens com 1 minuto de atraso e terminam aproximadamente 3 minutos atrasados. Nas duas situações a execução dos ônibus está dentro do previsto na perspectiva da STTP, que define uma margem de 15 minutos para que uma viagem seja classificada como atrasada.

Mesmo que em mediana o serviço esteja funcionando corretamente existem casos que os horários executados pelos ônibus são distantes do programado como pode ser visto na Figura 1. Por exemplo, a viagem com o maior variação do horários inicial começou com um pouco mais de 5 horas de atraso, e a viagem com maior variação final terminou aproximadamente 6 horas após o horário previsto. Ainda existem casos em que as viagens começam ou terminam antes do horário previsto, chamadas de adiantadas. As viagens adiantadas são tão problemáticas quanto as atrasadas e também possuem uma margem de 15 minutos para serem punidas. A viagem mais adiantada teve inicio aproximadamente 5 horas antes do programado e a viagem que terminou com quase 5 horas a menos que o programado foi a viagem com maior adiantamento no horário final.

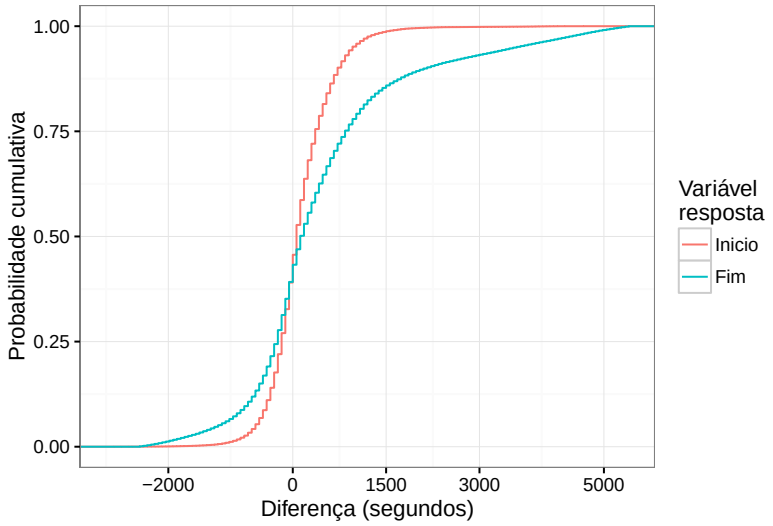


Fig. 1: Diferença entre horários programados e executados pelos ônibus.

5. RESULTADOS

Cada ensaio do experimento divide os dados em treino e teste. O conjunto de teste é o equivalente a um dia completo de viagens pareadas com os horários estabelecidos. Para o treino foram utilizados os 7, 15 ou 45 dias imediatamente predecessores do dia usado no teste. Além disso, dentre os atributos de cada viagem existe o histórico recente das viagens respectivas nos 1, 2 ou 3 dias anteriores. Usando este formato, foram escolhidos 12 dias aleatórios entre 01/01/2015 até 28/10/2015 e para cada um deles o experimento foi realizado. Em cada um destes dias todas as combinações dos demais fatores são exercitadas como parte do experimento.

Não existiu nenhuma restrição na definição dos dias selecionados como pode ser notado na Figura 2. O conjunto de dias escolhidos totaliza 15.269 viagens de 46 rotas agrupadas em 17 linhas executadas por 6 empresas do total de 394.295 viagens do conjunto de dados usado no experimento.

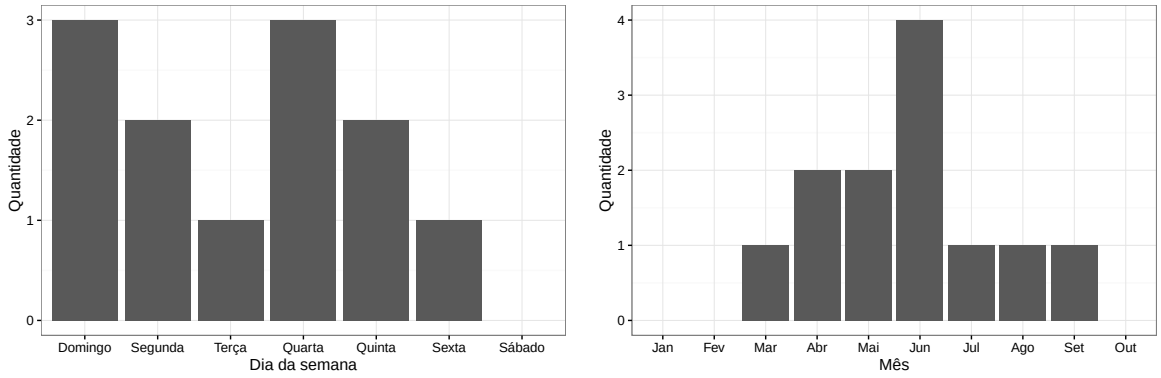


Fig. 2: Datas escolhidas para o conjunto de teste agrupadas por dia da semana e mês do ano.

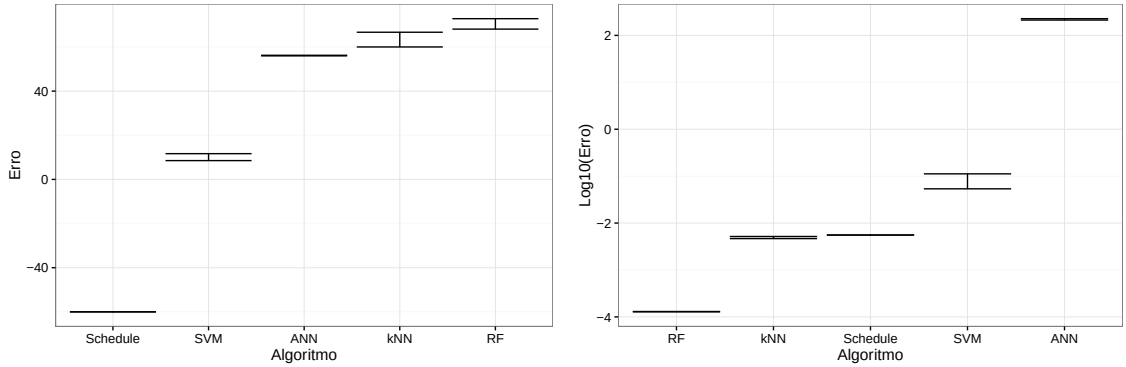
O valor previsto para cada viagem é a diferença de tempo, em segundos, entre o horário programado e o horário que irá acontecer. Por exemplo, para uma viagem programada para as 17h, um modelo pode ter como resposta -1.080s (= −18min), implicando em um horário previsto para essa viagem de $17h - 18min = 16h42$, 18 minutos antes do programado.

Neste trabalho exploramos o desempenho de quatro algoritmos de regressão: kNN, ANN, SVM e Random Forest, sendo os três primeiros bastante utilizados na literatura para a previsão de horário de ônibus [Liu et al. 2012; Jeong and Rilett 2004; Vanajakshi and Rilett 2004; Wang et al. 2009]. Na comparação também foi considerado como *baseline* um algoritmo que chamamos de *schedule* que prevê a diferença entre o horário programado e o observado sempre como 0. Este algoritmo equivale à ausência do uso de qualquer informação além da programação criada pela STTP.

A escolha dos parâmetros de aprendizado foi feita empiricamente usando validação cruzada e com o objetivo de minimizar o RMSE do conjunto de treino. Todo esse processo foi feito com o auxílio da função *train()* do pacote *caret* do R. O k escolhido para o kNN foi 9 tanto para o inicio quanto para o final da viagem. Para a rede neural foi avaliado o par de parâmetros *size* e *decay*, onde o primeiro representa a quantidade de neurônios na camada escondida e o segundo o fator de decadência dos pesos. Para o inicio da viagem os valores escolhidos foram 1 e 0, respectivamente. Para o final foram selecionados os valores 3 e 0.1. Três parâmetros são necessário para a SVM: função do *kernel*, sigma e C. Tanto para o inicio quanto para o final da viagem os valores escolhidos para esse parâmetros foram os mesmos. Função de base radial como função do *kernel*, aproximadamente 0,0067 para o *sigma* e 1 para p *C*. Por último, foi avaliado a quantidade de árvores e o *mtry* da floresta aleatória. A quantidade de árvores escolhida foi 10 em todas as situações. O *mtry* selecionado para o inicio da viagem foi 2 e para o final foi 11.

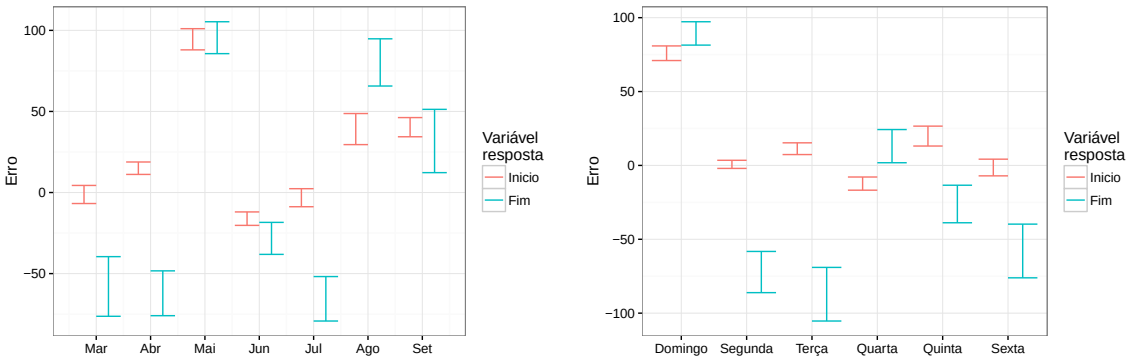
Os resultados mostram que os erros na previsão do horário de inicio da viagem foram menores do que na previsão do horário final. Os erros para o inicio da viagem ficaram entre -900 e 780 segundos com mediana de aproximadamente 28 segundos. A variação do erro para o final da viagem foi maior estando entre -38870 e 1649 segundos com mediana de quase -167 segundos.

Considerando os fatores variados em cada ensaio, percebe-se que a varição do tamanho do conjunto de treino e do tamanho do histórico recente não são tão influentes quanto o algoritmo utilizado. Nas Figuras 3a e 3b podemos ver que para as duas variáveis resposta o algoritmo que a mediana mais se aproxima do 0 é o SVM. O fato dele ter desempenho melhor que o *baseline* mostra que é possível melhorar a qualidade da informação existente.



(a) Inicio da viagem. (b) Final da viagem

Fig. 3: Intervalo de confiança da mediana do erro, agrupado por algoritmo.



(a) Mês do ano (b) Dia da semana

Fig. 4: Intervalos de confiança da mediana do erro.

O desempenho dos algoritmos também foi avaliado com relação a varição da quantidade de dias usados no conjunto de treino e no histórico recente de cada viagem. Os resultados mostram que de maneira geral esse dois fatores não proporcionam nenhuma diferença significativa nos erros. Entretanto, nota-se que utilizar 7 dias no conjunto de treino tende a ter melhores resultados do que usar 15 dias no treino, porém, não é possível afirmar que usar 7 dias é melhor do que usar 45 dias.

Por último, os erros foram avaliados com base em dois fatores temporais: mês do ano e dia da semana. A Figura 4a mostra que o mês dos ano em que a viagem aconteceu pode influenciar o desempenho do algoritmo em questão, nesse caso SVM. Por exemplo, o mês de Maio parece ser o mês mais imprevisível, porém, em Maio o algoritmo possui desempenho bastante similar tanto para o inicio quanto para o final da viagem. De maneira semelhante, a Figura 4b mostra que o dia da semana em que a viagem aconteceu interfere no desempenho do SVM. O domingo é o dia com o pior desempenho tanto no inicio quanto no final da viagem. Enquanto a segunda-feira e a sexta-feira parecem ser os melhores dias para prever o inicio da viagem, a quarta-feira parece ser o melhor dia para se prever o horário final da viagem.

Com base na análise apresentadas é possível afirmar que a necessidade de interpolar os horários das paradas internas da viagem com base no inicio e fim da viagem diminui a qualidade dos horários previstos, contudo, os resultados apontam que é possível usar dados históricos para tornar os valores interpolados mais acurados. De um ponto de vista mais geral, o experimento mostrou que é possível tornar o sistema de ônibus de Campina Grande mais previsível.

6. CONCLUSÃO E TRABALHOS FUTUROS

Neste artigo foi analisado o desempenho de 4 algoritmos, sendo três deles considerados estado da arte na previsão de horários de ônibus usando informação de localização em tempo real, quando aplicados no contexto de Campina Grande que não possui tal informação. Para suprir a falta de dados mais precisos e atualizados foram utilizados dados históricos de 3 tipos: categóricos, lotação e meteorológicos. Até onde sabemos esse é o primeiro estudo que trabalhou com dados que não possuem informação em tempo real para prever o horários dos ônibus.

A partir do experimento realizado foi possível concluir que é possível tornar o sistema de transporte público coletivo de Campina Grande mais previsível. Foi verificado que o algoritmo utilizado pode interferir na qualidade das previsões e que o mesmo não pode ser afirmado com relação a quantidade de dias usados no conjunto de treino e no histórico de viagens recentes. Além disso, foi mostrado que o desempenho dos algoritmos pode mudar com o decorrer dos meses e com o passar dos dias da semana.

Como trabalhos futuros pretendemos refazer as mesmas analises considerando as melhores combinações de atributos e parâmetros para cada uma das situações necessárias. Um ponto que pode ser atacado é a adição de outros algoritmos de regressão usados na previsão de horários de ônibus como *Kalman filter* e avaliar o desempenho de usar um ou mais algoritmos simultaneamente. Também pretendemos testar os modelos na prática para coletar e analisar a avaliação dos usuários.

7. ACKNOWLEDGEMENTS

Este trabalho foi financiado pelo projeto EU-BR BigSea (MCTI/RNP 3rd Coordinated Call).

REFERENCES

Baptista, A. T., Bouillet, E. P., and Pompey, P. Towards an uncertainty aware short-term travel time prediction using GPS bus data: Case study in Dublin. In *2012 15th International IEEE Conference on Intelligent Transportation Systems*. pp. 1620–1625, 2012.

Bazzan, A. L. and Klügl, F. Introduction to Intelligent Systems in Traffic and Transportation. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 7 (3): 1–137, 2013.

Bejan, A. I., Gibbens, R. J., Evans, D., Beresford, A. R., Bacon, J., and Friday, A. Statistical modelling and analysis of sparse bus probe data in urban areas. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. pp. 1256–1263, 2010.

Chen, P., p. Yan, X., and h. Li, X. Bus Travel Time Prediction Based on Relevance Vector Machine. In *2009 International Conference on Information Engineering and Computer Science*. pp. 1–4, 2009.

CPTEC/INPE. Portal de Tecnologia da Informação para Meteorologia. http://bancodedados.cptec.inpe.br/, 2016.

Eboli, L. and Mazzulla, G. A methodology for evaluating transit service quality based on subjective and objective measures from the passenger's point of view. *Transport Policy* 18 (1): 172–181, 2011.

Gong, J., Liu, M., and Zhang, S. Hybrid dynamic prediction model of bus arrival time based on weighted of historical and real-time GPS data. In *2013 25th Chinese Control and Decision Conference (CCDC)*. pp. 972–976, 2013.

Jeong, R. and Rilett, R. Bus arrival time prediction using artificial neural network model. In *Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on*. pp. 988–993, 2004.

Liu, T., Ma, J., Guan, W., Song, Y., and Niu, H. Bus Arrival Time Prediction Based on the k-Nearest Neighbor Method. In *Computational Sciences and Optimization (CSO), 2012 Fifth International Joint Conference on*. pp. 480–483, 2012.

Padmanaban, R. P. S., Divakar, K., Vanajakshi, L., and Subramanian, S. C. Development of a real-time bus arrival prediction system for Indian traffic conditions. *IET Intelligent Transport Systems* 4 (3): 189–200, 2010.

Vanajakshi, L. and Rilett, L. R. A comparison of the performance of artificial neural networks and support vector machines for the prediction of traffic speed. In *Intelligent Vehicles Symposium, 2004 IEEE*. pp. 194–199, 2004.

Wang, J.-n., Chen, X.-m., and Guo, S.-x. Bus travel time prediction model with v - support vector regression. In *2009 12th International IEEE Conference on Intelligent Transportation Systems*. pp. 1–6, 2009.

Washington, S. P., Karlaftis, M. G., and Mannering, F. *Statistical and econometric methods for transportation data analysis*. CRC press, 2010.

Multi-Armed Bandits to Recommend for Cold-Start User

Crícia Z. Felício[1,2], Klérisson V. R. Paixão[2], Celia A. Z. Barcelos[2], Philippe Preux[3]

[1] Federal Institute of Triângulo Mineiro, IFTM, Brazil
[2] Federal University of Uberlândia, UFU, Brazil
cricia@iftm.edu.br, {klerisson, celiazb}@ufu.br
[3] University of Lille & CRIStAL, France
philippe.preux@inria.fr

Abstract. To deal with cold-start user, recommender systems often rely on prediction models that exploit various sources of information. Such models are valuable to compensate the lack of ratings, but the abundance of them opens an important problem of model selection. So far, several methods have been proposed to deal with cold-start problem in recommender systems. However, facing a set of models, very little work exists on selecting the model to cope with a given cold-start user. To address this gap, this work in progress quantitatively investigates the implementation of multi-armed bandits for model selection during the cold-start phase. We present an encouraging preliminary experiment.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: cold-start problem, multi-armed bandits, recommender systems

1. INTRODUCTION

In a recommendation system, different models are commonly used to deal with different stages of a user experience. For example, a particular model works better in earlier stages when the recommender system does not know the user's tastes yet. However, in later stages, a different model should be more effective, and therefore one switches to the more effective model. Originally, switching methods [Burke 2002] were designed to handle cold-start problem. The idea is to switch from one model to an other once the system has enough data about the user, so he is not cold anymore.

While the concept of switching models [Billsus and Pazzani 2000] is not new for recommender systems (henceforth RS), the availability of several cold-start methods provides enriched resources to *model selection*. Applied to cold-start stage, a model selection method may be seen as a framework to alternate among prediction model in order to find a more suitable one. Few works have sought to empirically assess the efficacy of a model selection specifically within the cold-start stage. Based on this gap, the aim of this work in progress is to explore how a model selection can be useful to provide better recommendations. Our hypothesis is that recommendation model fails in part of its predictions, therefore a model selection that maximizes the recommendation gain might be more precise.

In this paper, we pose one research question and report preliminary results to identify the role of a feedback-oriented method for model selection. In particular, we investigate whether *bandit algorithms* are useful for this model selection task. These algorithms weigh models, so that the worst performing models end up with a very little weight. Therefore, the overall recommendation takes advantage of different models and might be better by selecting the best to use to make a particular recommendation, based on their past performance.

C. Z. Felício would like to thank the Federal Institute of Triângulo Mineiro for study leave granted. We also thank the Brazilian research agencies CAPES, CNPq and FAPEMIG for supporting this work.

2. BACKGROUND AND LITERATURE REVIEW

To understand our approach, called MAB-Rec, we introduce some RS formalism:

Let U be a set of users and I be a set of items. Each user $u \in U$ and each item $i \in I$ has a unique identifier. The user-item rating matrix is $R = [r_{u,i}]_{m \times n}$, where each entry $r_{u,i}$ is the rating given by user u on item i, and m is the number of users, and n is the number of items. The recommendation task is based on the predictions of the missing values of the user-item rating matrix. Then, prediction models are used to recommend those top-k ranked.

Multi-Armed Bandit (MAB) problem can be understood as a sequential decision problem where an algorithm continually chooses among a set of arms (in this paper, we assume the set of arms is finite). In each step t, an arm a is selected and pulled which leads to a reward $X_a(t)$. This reward is distributed according to a certain unknown law. Here, we consider that the goal is to learn, as fast as possible, through repeated arm pulls, the arm that returns the maximum expected reward.

In this work, we assume a set of prediction models as the arms from MAB problem. Then, our bandit algorithm is sequentially applied to choose a prediction model either the best performing one at the moment (exploitation), or an other arm to learn how it performs (exploration). We rely on the ϵ-Greedy algorithm to implement our model selection. ϵ-Greedy maintains the mean reward of each arm (prediction model) a, denoted by $\bar{X}_a$. Each time t that the arm a is played, the mean reward $\bar{X}_a$ is updated. The mean reward of the arm a at time t is represented by $\bar{X}_a(t)$.

We represent the probability of selecting arm a at time t as $\mathbb{P}_a(t)$. In each round t the ϵ-Greedy algorithm selects the arm with the highest mean reward with probability 1 - ϵ, and selects an arm uniformly at random with probability ϵ.

Related Work. This paper follows the line on applications of bandits in RS [Mary et al. 2015].

Li et al. [2010] reports on personalized recommendation of news articles as a contextual bandit problem. They propose LinUCB, an extension of the UCB algorithm. It selects the news based on mean and standard deviation. It also has a factor α to control the exploration / exploitation trade-off. Moreover, Caron and Bhagat [2013] incorporate social components into bandit algorithms to tackle the cold-start problem. They designed an improved bandit strategy to model the user's preference using multi-armed bandits. Several works model the recommendation problem using a MAB setting in which the items to be recommended are the arms [Bouneffouf et al. 2012; Girgin et al. 2012].In a different way, Lacerda et al. [2013; 2015] model users as arms to recommend daily-deals. They consider strategies for splitting users into exploration and exploitation.

In comparison, the goal of MAB-Rec is the selection of existent prediction models that might offer better recommendations for cold-start users. Our MAB setting is also different, whereas the arms are the prediction models.

3. MAB-REC APPROACH

MAB-Rec is made of 3 phases: (i) computation of the prediction models, (ii) sort prediction models, and (iii) recommendation. To define the set of prediction models, we applied three steps: Rating prediction, Preference clustering, and Consensus computation as described in [Felício et al. 2016]. We obtain $M = \{M_0 = (C_1, \hat{\theta}_1), \ldots, M_K = (C_K, \hat{\theta}_K)\}$, the set of prediction models where each M_s is composed of a cluster of users C_s and its consensual preference vector $\hat{\theta}_s$.

Table I shows an example of how a prediction model is built. In Table Ia we have a user-item rating matrix example with 2 users and 7 items. With BiasedMF algorithm[1] [Koren 2008] we obtain the predicted rating matrix, see Table Ib. Clustering the predicted rating matrix rows and considering that the two users is in the same cluster, we present the consensual preference vector $\hat{\theta}_1$ in Table Ic.

[1]The name BiasedMF comes from the LibRec library that we use in the experiments.

Table I. (a) Example of a user-item rating matrix. "-" means that the user has not rate the item. (b) Predicted rating matrix. (c) Consensual preference vector.

(a)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
u_1	5	2	4	-	5	1	-
u_2	4	-	5	-	5	-	1

(b)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
u_1	4.6	2.09	4.23	4.24	4.84	1.07	1.0
u_2	4.2	3.8	4.42	5.0	4.86	2.28	1.2

(c)

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
u_1	4.6	2.09	4.23	4.24	4.84	1.07	1.0
u_2	4.2	3.8	4.42	5.0	4.86	2.28	1.2
$\hat{\theta}_1$	4.4	2.94	4.32	4.62	4.85	1.67	1.1

After obtaining the prediction models, we sort the consensual preference vectors according to their ratings. So, for each $\hat{\theta}_s$ we have a $\hat{\theta}'_s$ that represents the consensual preference vector in a sorted order. The idea is to recommend the items with high ratings in each model first. We hypothesize that this strategy can contribute to learn users preference faster. For instance, the correspondent $\hat{\theta}'_1$ to $\hat{\theta}_1$ in Table Ic will have the sorted list of items equal to $\{i_5, i_4, i_1, i_3, i_2, i_6, i_7\}$.

Making Recommendations: at each time t a recommendation for a user u is made according to a bandit algorithm $\mathcal{B}$ following these steps:

(1) Select a prediction model M_s using the bandit algorithm $\mathcal{B}$;
(2) Select the next item i not recommended yet from $\hat{\theta}'_s$ (consensual preference vector of M_s sorted by ratings);
(3) Recommend item i to user u;
(4) Receive a rating $r_{u,i}$ as feedback from u;
(5) Return the reward;
(6) Update the prediction model statistics in $\mathcal{B}$.

We consider a binary reward where $X_{M_s} = 1$ if $r_{u,i} \geq r_{max} - \beta$, $\beta \in [1, 2]$, otherwise $X_{M_s} = 0$; r_{max} is the max rating in the dataset. Then, the reward is based on the proximity of user rating and r_{max}.

4. PRELIMINARY FINDINGS

4.1 Research Method

This section describes our methodology by outlining our research question, our dataset, and our analysis method. We structure our work around the following research question:

RQ: How effective are Multi-Armed Bandits to select initial recommendations for cold-start users?

To answer the above question, we mimic a cold-start scenario. This is done using the standard *leave-one-out cross-validation*, where the number of folds is equal to the number of instances in the dataset. Therefore, the selected prediction model is applied once for each instance, using all other instances as a training set, the remaining one being used as a single-user test set; then the performance are averaged over all users, each being used as a test set. Note that, to simulate a realistic cold-start scenario, we do not provide for train any preference, for instance, movie ratings from the test users and that is why we called this protocol **0-rating**.

The experiments were performed on a real-world dataset collected from Facebook users [Felício et al. 2015]. The dataset has 49,729 ratings from 498 users over 169 movies and with 40.9% of sparsity. Here, we took users that rated at least 20 items, because we want to evaluate MAB-Rec against *nDCG* metric for the firsts 20 items recommended.

We extended the LibRec [Guo et al.] implementation of BiasedMF to build the prediction models and incorporate the bandit algorithm in recommendation process. Experiments were executed with 10 latent factors and 100 iterations. The optimal number of consensual prediction models is 3.

4.2 Results

Table II presents the *nDCG* at rank size of 5, 10, 15, and 20 using ϵ-Greedy as the bandit algorithm. MAB-Rec achieves until 0.8620 for ndcg@5 with $\beta = 2$ and 0.8592 with $\beta = 1$. The difference is

quite small between the results using different ϵ values and between the two ways to compute reward. The explanation for this can be the dataset features where we have few items (169 movies) and a small number of prediction models (optimal values was got for 3 consensual prediction models, 3 clusters). Future work will investigate the MAB-Rec approach in others dataset and with others bandits algorithms.

Table II. nDCG results with ϵ-Greedy: (a) Binary Reward with $\beta = 1$; (b) Binary Reward with $\beta = 2$

(a)

ϵ	Rank size			
	@5	@10	@15	@20
0.1	0.8564	0.8474	0.8463	0.8449
0.2	**0.8592**	0.8495	0.8484	0.8479
0.3	0.8555	**0.8506**	0.8477	**0.8480**
0.4	0.8590	0.8504	0.8483	0.8479
0.5	0.8585	0.8504	**0.8495**	0.8476

(b)

ϵ	Rank size			
	@5	@10	@15	@20
0.1	0.8565	0.8481	0.8472	0.8468
0.2	0.8563	0.8481	0.8470	0.8467
0.3	0.8591	0.8496	0.8475	0.8473
0.4	0.8592	0.8501	0.8481	**0.8480**
0.5	**0.8620**	**0.8514**	**0.8490**	0.8476

5. FINAL REMARKS

We presented preliminary results on recommendation model selection through multi-armed bandit algorithm. Our proposed approach focus on the important problem of recommending for cold-start users. While prior works on model selection mainly aim at discovering when a user is not cold anymore, then switch to a new model, we are investigating new ways to foster RS when they have few or none information about the user. Our preliminary experimental results reached 86% of accuracy levels in terms of nDCG@5. We plan to look at different datasets, others bandits algorithms and different strategies to filter the set of prediction models.

REFERENCES

Billsus, D. and Pazzani, M. J. User modeling for adaptive news access. *User Modeling and User-Adapted Interaction* 10 (2): 147–180, 2000.

Bouneffouf, D., Bouzeghoub, A., and Gançarski, A. L. A contextual-bandit algorithm for mobile context-aware recommender system. In *Proc. IntĊonfón Neural Information Processing*. ICONIP. pp. 324–331, 2012.

Burke, R. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction* 12 (4): 331–370, 2002.

Caron, S. and Bhagat, S. Mixing bandits: A recipe for improved cold-start recommendations in a social network. In *Proc. Workshop on Social Network Mining and Analysis*. SNAKDD. ACM, pp. 11:1–11:9, 2013.

Felício, C. Z., Paixão, K. V. R., Alves, G., and de Amo, S. Social prefrec framework: leveraging recommender systems based on social information. In *Proc. Symposium on Knowledge Discovery, Mining and Learning*. KDMiLe. pp. 66–73, 2015.

Felício, C. Z., Paixão, K. V. R., Barcelos, C. A. Z., and Preux, P. Preference-like score to cope with cold-start user in recommender systems. In *Proc. IEEE Int. Conf. on. Tools with Artificial Intelligence*. ICTAI, 2016.

Girgin, S., Mary, J., Preux, P., and Nicol, O. Managing advertising campaigns – an approximate planning approach. *Frontiers in Computer Science* 6 (2): 209–229, Apr., 2012.

Guo, G., Zhang, J., Sun, Z., and Yorke-Smith, N. Librec: A java library for recommender systems. In *Proc. User Modeling, Adaptation, and Personalization*. UMAP.

Koren, Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proc. Int. Conf. on Knowledge Discovery and Data Mining*. ACM SIGKDD. Las Vegas, Nevada, USA, pp. 426–434, 2008.

Lacerda, A., Santos, R. L. T., Veloso, A., and Ziviani, N. Improving daily deals recommendation using explore-then-exploit strategies. *Information Retrieval Journal* 18 (2): 95–122, 2015.

Lacerda, A., Veloso, A., and Ziviani, N. Exploratory and interactive daily deals recommendation. In *Proc. ACM Conference on Recommender Systems*. RecSys. ACM, New York, NY, USA, pp. 439–442, 2013.

Li, L., Chu, W., Langford, J., and Schapire, R. E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the International World Wide Web Conferences*. ACM, pp. 661–670, 2010.

Mary, J., Gaudel, R., and Preux, P. Bandits and recommender systems. In *Machine Learning, Optimization, and Big Data*. Lecture Notes in Computer Science, vol. 9432. Springer International Publishing, pp. 325–336, 2015.

Contagem e Cognição Numérica: Experimentos com *Eye-Tracking*

D. A. Dal Fabbro, C. E. Thomaz

Centro Universitário da FEI, Brazil
ddfabbro@fei.edu.br cet@fei.edu.br

Abstract. This work investigates a cognitive approach to the mechanisms used by humans to perform enumeration of non-symbolic quantities by measuring the reaction time and eye-tracking fixations of human volunteers. Using the eye-tracking technology, it has been shown that the way we perceive a certain amount is strongly related to our reaction time and our visual perception varies for quantities up to five and higher than six. We believe that such detailed exploratory data analysis can be used for other data mining processes, disclosing the human perception of visual patterns.

Categories and Subject Descriptors: H.1.2 [**Models and Principles**]: User/Machine Systems—*Human information processing*; J.4 [**Social and Behavioral Sciences**]: ; H.2.8 [**Database Management**]: Database Applications—*Data mining*

Keywords: data mining, enumeration, eye-tracking

1. INTRODUÇÃO

Quando temos a nossa frente quarenta moedas de mesmo valor é evidente que fazemos uso da aritmética para concluirmos com precisão que realmente são quarenta moedas que estão ali. Esta aritmética pode ser feita pela soma unitária de cada moeda ou pela divisão de pequenos conjuntos cujas quantidades já são conhecidas para posteriormente somá-las. Qualquer que seja a técnica escolhida, não deixamos de recorrer ao uso da matemática.

Além do homem, outros seres vivos também são capazes de notar que existem quantidades diferentes entre dois conjuntos de objetos [Capaldi 1993]. Este foi o primeiro indício de que podemos fazer contagem sem conhecer aritmética.

No entanto, esta contagem feita por outros seres é meramente comparativa, ou seja, é uma forma de distinguir a quantidade entre dois conjuntos sem conhecer a quantidade exata de cada um. Para o indivíduo ser capaz de reconhecer o número presente em um conjunto é necessário que seja atribuído um símbolo ou nomenclatura àquela quantidade [Warren 1897; Campbell 1992]. Isto possibilitará que ela possa ser discriminada.

No seculo XX, um estudo sobre a percepção humana durante a interpretação de cenas complexas [Yarbus 1967] popularizou uma ferramenta com vasta aplicações em questões cognitivas, o *eye-tracking*. Este equipamento busca rastrear com precisão os movimentos oculares feitos pelo homem, facilitando novas descobertas e o entendimento da função que nossa visão exerce em atividades cognitivas, como uma contagem básica. Em 1984, foi publicado um estudo que utilizou este equipamento para estudar os padrões visuais gerados durante a contagem de quantidades unidimensionais (pontos) entre 19 e 23 quando dividida em grupos menores, comprovando que o olhar procura e conta por conjuntos de pontos, ao invés de individualmente [Van Oeffelen and Vos 1984].

Com o avanço da tecnologia, particularmente a eletrônica, o *eye-tracking* foi modernizado, gerando novos estudos que resgataram as questões levantadas por pesquisadores passados para experimental-

Este projeto foi parcialmente financiado pela CAPES e pelo CNPq (444964/2014-2).

mente entender a influência das quantidades quanto à frequência dos movimentos dos olhos [Watson et al. 2007], além de mostrar que o número de fixações oculares durante a contagem de quantidades a partir de cinco é altamente linear [Li et al. 2010]. Porém, até hoje não foi publicado um trabalho que investigasse os mecanismos de contagem de forma tão conclusiva utilizando experimentos de *eye-tracking*.

Este trabalho realiza o levantamento das informações do tempo de reação a partir da atividade visual que o individuo gera durante a contagem e um entendimento da percepção das quantidades com a forma em que esses estímulos são apresentados. Para coletar os dados das amostras, além de utilizar o tempo cronometrado para se fazer a contagem, também será analisado para onde o olhar é fixado, de forma que seja possível detectar padrões na maneira que olhamos durante a contagem, bem como sob a perspectiva da situação a que somos submetidos.

As demais seções podem ser assim resumidas. Na seção 2 são descritos os mecanismos de contagem existentes e um breve histórico sobre a evolução científica nesta área. Na seção 3 são detalhados os recursos utilizados durante o desenvolvimento do trabalho, bem como o procedimento experimental e os critérios de seleção dos dados. A seção 4 é uma extensa análise exploratória dos dados coletados em busca de relações entre os mecanismos estudados. Por fim, na última seção, o trabalho é concluído evidenciando o equipamento de *eye-tracking* como uma ferramenta de *data mining* em imagens que contém informações implícitas e as suas possíveis aplicações em domínios mais complexos.

2. MECANISMOS DE CONTAGEM

Os primeiros estudos sobre a forma que as pessoas fazem contagem sugeriram que possuímos dois principais métodos que são recorridos para situações distintas: a contagem comum e a contagem inferencial. No primeiro tipo, a discriminação é rápida, precisa e feita instantaneamente para pequenos grupos (perceptiva) ou pela aritmética de unidades para grupos maiores (progressiva). Já o segundo método é feito pela aproximação da quantidade baseada em padrões conhecidos e convenientes [Warren 1897].

Em 1949, foi publicado um estudo mostrando que até uma quantidade específica (geralmente quatro), um indivíduo com conhecimentos em algum sistema numérico consegue intuitivamente acusar a quantidade de objetos de determinado conjunto com precisão e sem necessariamente contá-los [Kaufman et al. 1949]. Esta técnica foi denominada pelo autor de *subitizing*[1], é inerente a todos nós e pode ser a explicação para a capacidade cognitiva de outros animais também saberem fazer comparações entre quantidades, mesmo que até um número limitado.

Através do *subitizing* é possível discriminar a quantidade de pequenos números quase instantaneamente, porém para quantidades superiores, este método é inconscientemente descartado e passamos a fazer a contagem de forma progressiva ou inferencial. Quando estes mecanismos de contagem são ativados, o tempo de reação para discriminação da quantidade é elevado consideravelmente, e isto pode ser notado facilmente na Figura 1 com dados levantados experimentalmente em trabalhos anteriores [Campbell 1992].

Uma suposta explicação para o *subitizing* ser tão rápido é que relacionamos os indivíduos a serem contados com a forma geométrica formada por eles [Akin and Chase 1978]. Por exemplo: quando temos a nossa frente três moedas, visualmente recordaremos de um triângulo. Já quando temos quatro moedas a nossa frente, na maioria dos casos irá assemelhar-se a um quadrado.

Em casos muito específicos, podemos recorrer ao *subitizing* para quantidades superiores a quatro. Um experimento comprovou que se a disposição dos indivíduos estiver organizada em padrões (homogênea), ainda somos capazes de contá-los de maneira muito rápida [Frick 1987], como por exemplo

[1]Etm. do latim *subitus*

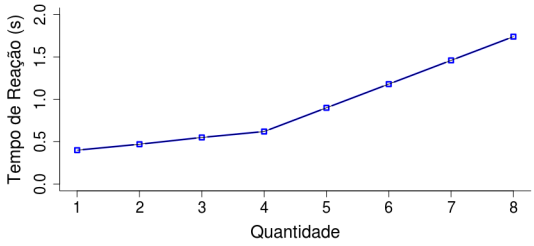


Fig. 1. Tempo de reação para contagem de quantidades inteiras.

o número seis representado pela face de um dado, ou até mesmo a quantidade nove disposta em um teclado numérico. Portanto a forma como apresentamos os objetos tem grande influência nos resultados.

3. MATERIAIS

3.1 Participantes

A amostragem utilizada consiste em 18 voluntários adultos (14 homens e 4 mulheres) com idades entre 20 e 50 anos. Todos os participantes possuem vínculos com a instituição de ensino onde foi realizado este trabalho e são alunos da graduação e pós-graduação de Engenharia, ou funcionários e professores. Nenhum dos participantes apresenta problemas de visão não corrigido e todos demonstram conhecimentos básicos em aritmética.

3.2 Procedimento Experimental

Este trabalho, além de medir o tempo de reação dos pontos de interesse durante a contagem, também faz uma comparação para diferentes maneiras de se organizar o mesmo número de moedas. Portanto, o experimento foi dividido em duas situações.

A primeira situação apresenta ao voluntário moedas com quantidades que variam entre um e quinze em momentos distintos. A distribuição dessas moedas é heterogênea, ou seja, não apresentam regra ou sentido em sua disposição. Além disso, a ordem que as distribuições são apresentadas deve ser aleatória para não induzir o individuo a nenhum valor.

A segunda situação apresenta as moedas de forma homogênea, ou seja, com disposições geométricas uniformes. Nesta situação, apenas as quantidades cinco, seis, sete, nove, doze e quinze são analisadas, uma vez que padrões não têm grande influência em quantidades inferiores a quatro e as demais quantidades não podem ser dispostas em padrões triviais.

Outro ponto a ser considerado é que, durante o experimento, não há distinção entre qual situação o individuo está submetido. Ou seja, as duas situações estão contidas na mesma sequência de experimentos.

O primeiro contato do voluntário com o experimento apresenta uma tela com as instruções para que não exista dúvida durante as medições. Antes de iniciar o experimento, uma prévia com uma quantidade aleatória é apresentada para ter certeza de que as regras expostas durante as instruções foram devidamente seguidas. As transições entre cada quantidade são constituídas por uma tela preta, de duração de 1 segundo, e têm como objetivo garantir o relaxamento do foco ocular.

Por fim, o experimento permite ao usuário determinar o número de moedas sem nenhuma restrição

de tempo, mas somente é possível realizar contagem visualmente, sem o uso das mãos ou outros artifícios.

3.3 Equipamento

Para avaliar os resultados cognitivos do experimento, bem como o tempo de reação, um equipamento de *eye-tracking* foi utilizado. Este equipamento é projetado pela Tobii, modelo TX300 e consiste em um monitor TFT de 23" com sensores infravermelho acoplados na parte inferior do monitor. A taxa de amostragem dos movimentos oculares foi de 300Hz com uma duração mínima de fixação de 60ms e um limiar de 0,5 ° de máxima dispersão. Estes parâmetros são considerados como padrões em pesquisas cognitivas com *eye-tracking* [Tobii 2012]. Durante o experimento um teclado padrão foi utilizado pelos voluntários para controlar a sequência de estímulos visuais. A calibração e o processamentos dos dados foram realizados pelo *software* da Tobii, instalado na plataforma Windows 7, processador Core i7 e 16Gb de RAM.

3.4 Processamento dos dados

Após a realização dos experimentos, os dados capturados durante a atividade ocular foram armazenados e tratados pelo *software* Tobii Studio, que permite a elaboração de mapas de calor com as principais áreas de interesse e exportação desses dados no formato de arquivo *".csv" (comma separated values)*. O raio de visão considerado no mapa de calor foi de 85 *pixels* para todos os mapas. O critério para geração do mapa de calor é relativo ao tempo total de análise da imagem e uma fixação só será considerada pelo equipamento caso o participante foque por mais tempo que a duração mínima de fixação, que é 60ms [Tobii 2012], caso contrario, ela será descartada.

Para iniciar a análise gráfica, 3 das 21 amostras foram descartadas pois para alguns testes o equipamento não foi capaz de rastrear o movimento ocular do voluntário, resultando em um tempo de reação igual a zero.

4. RESULTADOS

A taxa de acerto das quantidades informadas pelos voluntários é de 94,74% ($\pm$ 6.49%).

Apesar do experimento se basear em apenas 18 amostras, é possível detectar uma clara tendência nos resultados do tempo de reação para quantidades de 1 até 15 distribuídas de forma heterogênea, assim como comportamentos padronizados para as distribuições homogêneas.

A Figura 2 mostra um gráfico no formato *box-plot*, que possibilita evidenciar a mediana, os *outliers* e suas respectivas variâncias para cada quantidade. É possível notar que para quantidades até cinco a mediana do tempo de reação não possui grande diferença. Já a partir de seis moedas tanto uma mediana crescente, quanto uma variação maior entre as quantidades, podem ser observadas, comprovando assim o gráfico levantado por experimentos de outros autores.

Para as quantidades quatorze e quinze a variância das medições é notavelmente maior do que as anteriores. Uma possível explicação seria o mecanismo de aritmética utilizado por cada participante. Foi observado que para grandes quantidades, os voluntários contavam em grupos de dois ou em grupos de cinco e só é possível evidenciar a eficiência de cada um desses métodos quando testadas para grandes quantidades. Outra explicação encontrada é que para grandes quantidades, o voluntário poderia se confundir durante a contagem e ter de começar novamente.

O resultado cognitivo do experimento é analisado nas Figuras 3 e 4 no formato de mapa de calor, onde as regiões em vermelho são os pontos que os participantes focaram por mais tempo.

Como era de se esperar, as quantidades até cinco distribuídas de forma heterogênea possuem pontos de interesse concentradas na região central da imagem. Para quantidades acima desse valor, as regiões

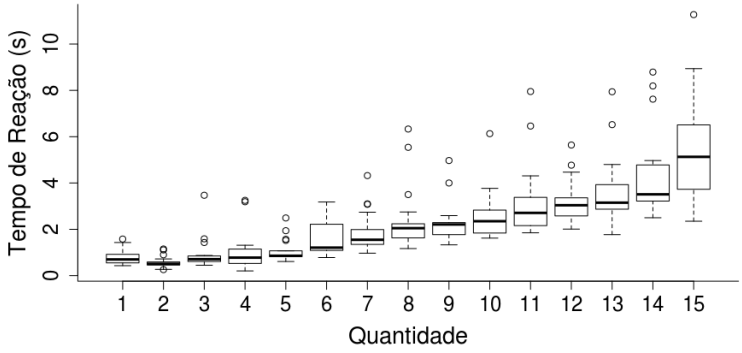


Fig. 2. Tempo de reação para contagem de distribuições heterogêneas.

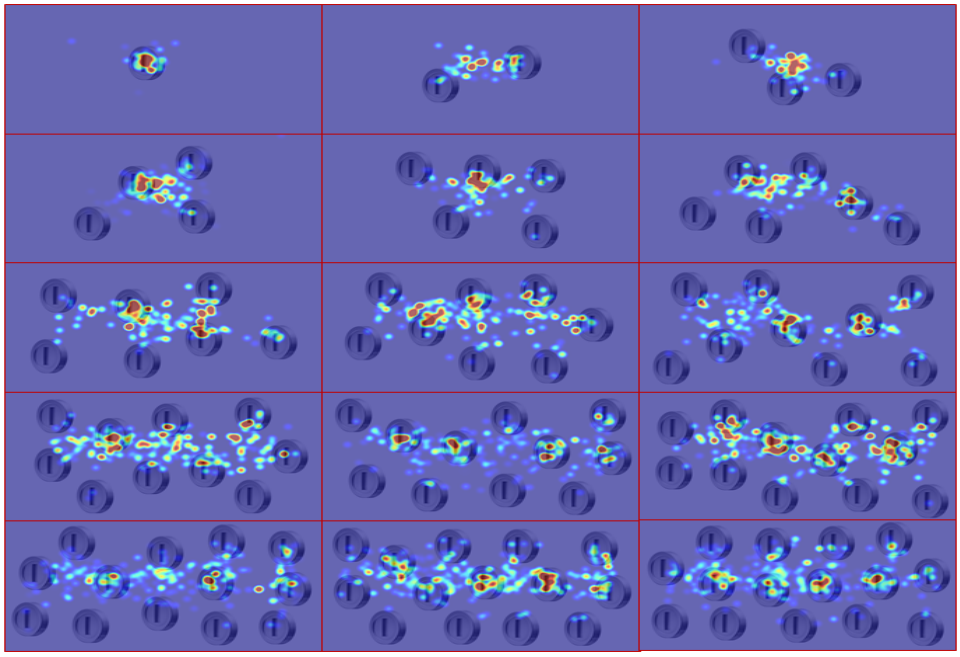


Fig. 3. Resultado cognitivo para contagem em distribuições heterogêneas.

de interesse começam a ficar dispersas. Isto significa que para estas quantidades é necessário que o olhar percorra a imagem para se ter consciência do número de moedas, o que pode revelar que um processo de aritmética é recorrido.

Para as moedas distribuídas de forma homogênea, a dispersão das regiões de interesse dos voluntários é menor, o que supostamente demonstra que a percepção sobre a mesma quantidade de moedas é diferente.

Por fim, para notar uma evidente diferença entre os dois tipos de distribuições, um gráfico no formato *box-plot* é analisado na Figura 5, comprovando a discrepância entre o tempo de reação para cada situação.

6 ·

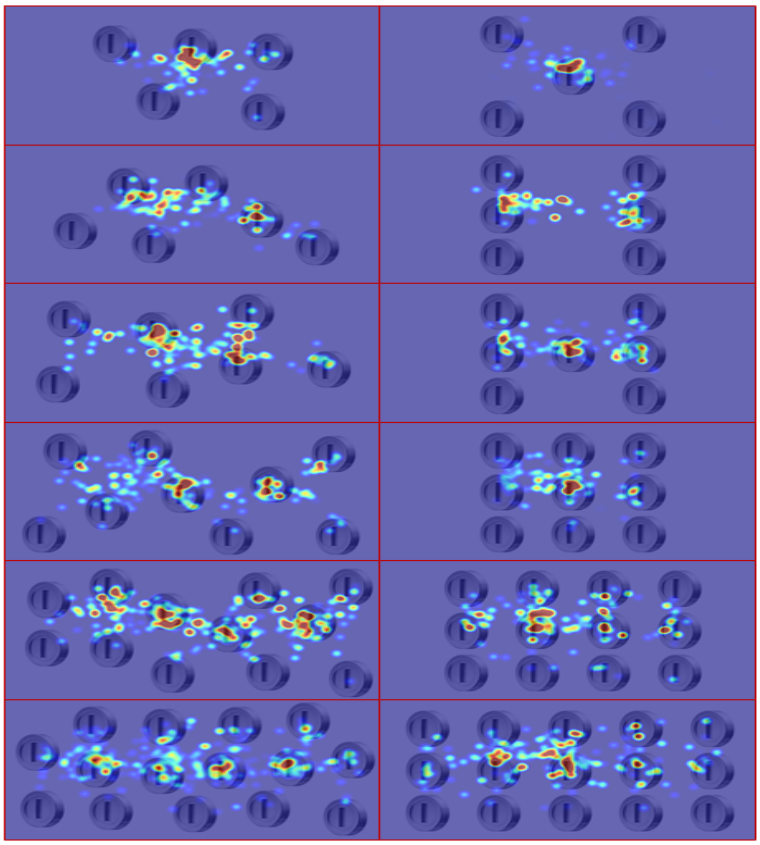


Fig. 4. Comparação entre o resultado cognitivo para contagem em distribuições heterogêneas (esquerda) e distribuições homogêneas (direita).

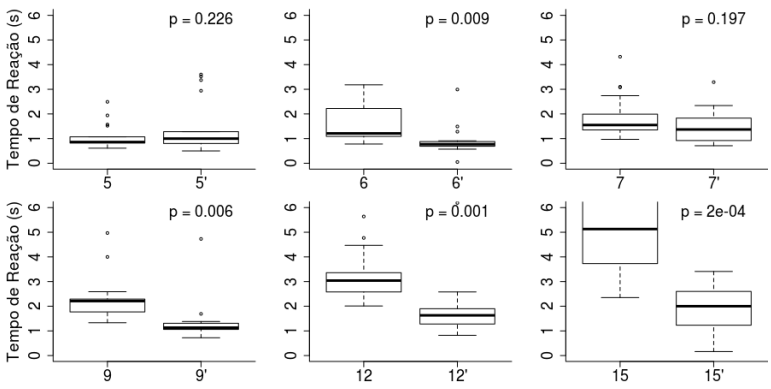


Fig. 5. Comparação de tempos de reação para distribuição heterogênea (x) e homogênea (x').

4.1 Relação com o número de fixações

Uma outra forma para analisar os dados é pelo número de fixações que os voluntários fazem durante a contagem, ou seja, quantas vezes precisam mudar o foco ocular. A Figura 6 mostra o *box-plot* do levantamento do número de fixações realizadas para cada quantidade com distribuição heterogênea.

Este gráfico apresenta comportamento similar ao gráfico para o tempo de reação, ilustrando uma forte relação entre as duas variáveis. Isto significa que o número de fixações cresceu proporcionalmente ao tempo de reação.

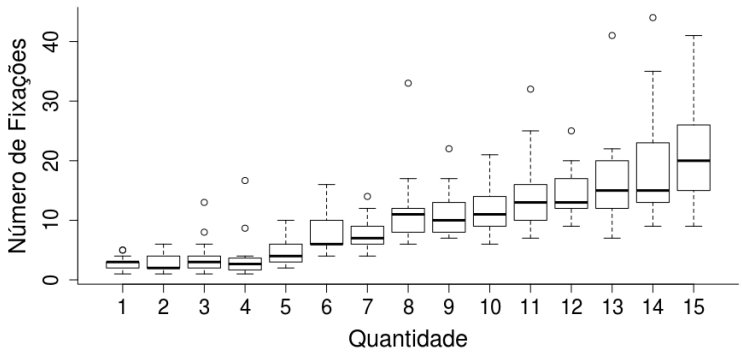


Fig. 6. Número de fixações para contagem de distribuições heterogêneas.

A Figura 7 mostra a comparação entre o número de fixações para distribuições heterogêneas e distribuições homogêneas para diferentes quantidades com aspectos muito similares aos analisados para os tempos de reação, reforçando ainda mais o resultado de que uma correlação entre o tempo de reação e o número de fixações existe.

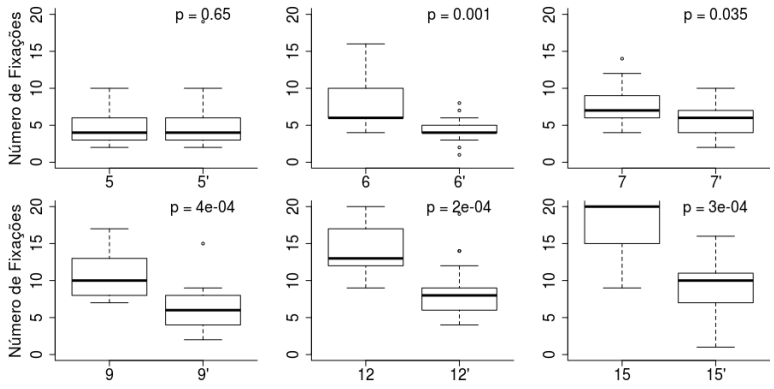


Fig. 7. Comparação de número de fixações para distribuição heterogênea (x) e homogênea (x').

Por fim, esta relação entre tempo de reação e número de fixações fica completamente clara quando as duas variáveis são analisadas em um mesmo gráfico. A Figura 8 mostra esta correlação para distribuições heterogêneas e homogêneas separadamente e nota-se que, independentemente do padrão da distribuição, a correlação é estatisticamente significativa.

5. DISCUSSÃO

Com este experimento, levantamos o gráfico que relaciona o tempo de reação de contagem para diferentes quantidades, comprovando os dados de trabalhos anteriores. Além disso, relacionamos estes tempos com a cognição numérica e visual para diferentes quantidades e, através dos resultados,

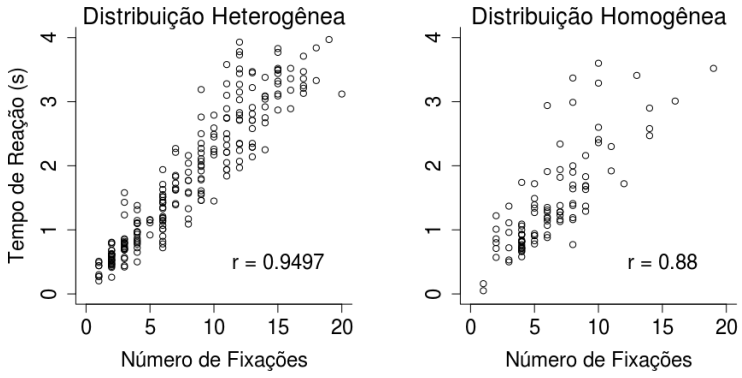


Fig. 8. Relação entre o número de fixações e o tempo de reação em distribuições heterogêneas e homogêneas.

mostramos que o tempo de reação dos voluntários está fortemente relacionado à nossa percepção de quantidades.

Esta percepção pode variar conforme estas quantidades são organizadas. A organização das quantidades em distribuições homogêneas nos possibilita relacioná-las com padrões já conhecidos ou até mesmo utilizar técnicas mais avançadas, como a multiplicação. Por conta disso, em alguns casos, o tempo para contagem de grandes quantidades pôde ser reduzido a tempos próximos aos observados para quantidades onde o *subtizing* é naturalmente recorrido, indicando que esta técnica é possivelmente um processo cognitivo inconsciente, que acontece baseado em experiências anteriores ou mesmo intuitivamente, o que consequentemente caracteriza uma forma de conhecimento implícito.

Acreditamos que este é o primeiro trabalho na literatura que aborda a questão do *subitizing* usando experimentos de *eye-tracking* de forma tão conclusiva e, que a questão da contagem estudada neste trabalho, mostra que o *eye-tracking* pode trazer uma nova abordagem para descoberta de conhecimento com aplicações em outras áreas da ciência, especialmente em problemas onde existe conhecimento implícito pela análise visual de um especialista sobre o domínio, como segmentação de imagens, recomendação de produtos, navegação de robôs, análise de imagens médicas, reconhecimento de faces, entre outros.

REFERÊNCIAS

Akin, O. and Chase, W. Quantification of three-dimensional structures. *Journal of Experimental Psychology: Human Perception and Performance* vol. 4, 1978.

Campbell, J. I. *The nature and origin of mathematical skills*. Vol. 91. Elsevier, 1992.

Capaldi, E. J. Animal number abilities: Implications for a hierarchical approach to instrumental learning. *The development of numerical competence: Animal and human models*, 1993.

Frick, R. W. The homogeneity effect in counting. *Perception & Psychophysics* vol. 41, 1987.

Kaufman, E. L., Lord, M. W., Reese, T. W., and Volkmann, J. The discrimination of visual number. *The American journal of psychology* vol. 62, 1949.

Li, X., Logan, G. D., and Zbrodoff, N. J. Where do we look when we count? the role of eye movements in enumeration. *Attention, Perception, & Psychophysics* 72 (2): 409–426, 2010.

Tobii, T. User manual - tobii studio. *Manual Version 3.2 Rev A*, 2012.

Van Oeffelen, M. P. and Vos, P. G. Enumeration of dots: An eye movement analysis. *Memory & Cognition* 12 (6): 607–612, 1984.

Warren, H. The reaction time of counting. *Psychological Review* vol. 4, 1897.

Watson, D. G., Maylor, E. A., and Bruce, L. A. The role of eye movements in subitizing and counting. *Journal of Experimental Psychology: Human Perception and Performance* 33 (6): 1389, 2007.

Yarbus, A. L. *Eye Movements and Vision*. Springer, 1967.

Uso de Redes Neurais Recorrentes para Localização de Agentes em Ambientes Internos

E. Carvalho[1,3], B. Ferreira[2,3], M. Ferreira[2,3], G. Pereira[4], J. Ueyama[4], G. Pessin[3]

[1] Instituto SENAI de Inovação em Tecnologias Minerais, Brasil
eduardo.isi@sesipa.org.br
[2] Universidade Federal do Pará, Brasil
[bruno.ferreira, mylena.ferreira]@pq.itv.org
[3] Instituto Tecnológico Vale, Brasil
gustavo.pessin@itv.org
[4] Universidade de São Paulo, Brasil
[geraldop,joueyama]@icmc.usp.br

Abstract. Sistemas de localização de agentes em ambientes internos ainda apresentam problemas em aberto. Propostas com conjuntos específicos de sensores tem produzido soluções com diversos graus de precisão. A utilização de métodos de aprendizado de máquina tem sido uma maneira de melhorar a acuracidade com esses conjuntos de sensores; sendo que a utilização de variações de redes neurais tem gerado resultados satisfatórios para os diferentes ambientes internos que se tem estudado. Este artigo apresenta a utilização de uma variação de rede neural, a Rede Neural Recorrente (RNN) para estimar a localização de agentes em ambientes internos; indicando a potencialidade da utilização da mesma para se construir um sistema de localização para ambientes internos com baixo custo financeiro e computacional e com alta acuracidade.

Categories and Subject Descriptors: I.2.6 [**Artificial Intelligence**]: Learning

Keywords: Redes Neurais Recorrentes, Localização *indoor*, *Long Short-Term Memory*

1. INTRODUÇÃO

Locais fechados, tais como prédios, e residências, até cavernas e minas subterrâneas possuem limitações de acesso em casos de acidentes, como: um prédio comercial em chamas, ou explosões e deslizamentos de terra em uma mina subterrânea. Como pessoas e equipamentos necessitam estar presentes em tais ambientes internos, e esses acidentes tem potencialidades de ocorrer, a localização desses agentes se torna importante para prover um nível mais alto de segurança para os mesmos.

A segurança de pessoas em ambientes internos é um assunto importante nos contextos indicados (prédios e minas subterrâneas). O departamento de incêndios dos EUA fez um levantamento do número de incêndios e mortes em prédios residenciais e em prédios comerciais no período entre 2003-2012. O número aproximado de incêndios em prédios residenciais foi de mais de 3,5 milhões com cerca de 25.000 mil óbitos e mais de 130.000 pessoas acidentadas. Já em áreas comerciais o número de incêndios chegou a mais de 970.000 casos, com 875 óbitos, 13.300 pessoas feridas [Department 2012]. Em mineração, um estudo feito no período entre 2006-2010 mostrou o número de fatalidades que aconteceu em minas a céu aberto e minas subterrâneas em cinco diferentes países (China, EUA, Índia, Austrália e África do Sul) em que mais de 4.000 mortes entretanto, para fatores de explosões e quedas de minérios mais de 2.500 de óbitos foram em minas subterrâneas e apenas 61 óbitos ocorreram por esses fatores em minas de céu aberto. Assim, se um sistema de localização estivesse presente, caso uma

pessoa se perca ou fique presa nesses ambientes, essas fatalidades poderiam ser minimizadas [Harris et al. 2014].

Técnicas de localização de pessoas em ambientes externos estão bem evoluídas com o GPS. Entretanto, em ambientes internos esse problema ainda não foi totalmente solucionado, apesar da grande quantidade de trabalhos na área. O motivo de não haver tal solução fica a cargo da complexidade das diversas edificações, haja vista que a propagação do sinal é um problema, pois paredes e demais objetos presentes nesses ambientes acabam por refletir ou atenuar os sinais propagados [Parameswaran et al. 2009]. Assim, trabalhos utilizam métodos de captação de potências de sinal para inferências das posições de pessoas ou objetos em ambientes internos. Esses sinais podem sofrer diversas análises, como a simples triangulação dos sinais e produzindo distâncias entre os dispositivos emissores e os receptores, entretanto tecnicas de aprendizado de máquinta tem sido muito utilizadas.. Trabalhos como os de [Jekabsons et al. 2011] [Marti et al. 2012] que tratam da utilização de Redes Neurais Artificiais, *K-Nearest Neighbors* (KNN) entre outros algoritmos para aprender padrões e classificar a posição de uma pessoa por exemplo.

Com o passar dos anos esses algoritmos sofreram constantes evoluções, assim como a quantidade de dados a serem processados, que cresceram, passando a cada vez mais exigir essas melhorias de código. Nesse sentido as Redes Neurais Artificiais cresceram em uma direção de *Deep Learning* que é a aplicação de conhecimento extraído de uma base de dados enorme e aplicado as entradas do sistema, assim como em RNA convencionais, entretanto com mecanismos de memórias para acelerar o processamento. Redes Neurais Recorrentes (RNN) e Redes Neurais Convolucionais (ConvNets) são frutos dessa evolução [Dalto 2015], [Sak et al. 2014].

Nesse artigo apresentamos uma avaliação de Redes Neurais Recorrentes para localizar pessoas em um ambiente interno. O protótipo tem como entrada uma coleta de potências de sinal e o algoritmo da RNN classifica a posição da pessoa de acordo com essa coleta. Devido o difícil acesso as áreas de cavernas ou minas subterrâneas, o sistema é testado em um ambiente mais controlável: o andar de prédio. O ambiente conta com dispositivos que emitem sinais Wi-Fi e a pessoa fica com um dispositivo que receba esse tipo de sinal, recebendo sinais de potência em *Received Signal Strength Indication* (RSSI)para criação de uma tabela de potências com os sinais coletados dos dispositivos e a RNN possa classificar as posições da pessoa nas coletas.

O restante desse artigo está organizado da seguinte forma: a Seção 2 apresenta trabalhos relacionados com o problema de localização. A Seção 3 apresenta os conceitos e alguns trabalhos sobre Redes Neurais Recorrentes (RNN). A Seção 4 mostra o ambiente de teste em que as coletas foram realizadas. A Seção 5 indica os resultados com RNN. Por fim, são apresentadas as considerações finais e os trabalhos futuros almejados.

2. TRABALHOS COM LOCALIZAÇÃO UTILIZANDO *MACHINE LEARNING*

Esta Seção apresenta alguns trabalhos relacionados, destacando a aplicação de redes de sensores sem fio na localização de agentes em ambientes internos.

O trabalho de [Yoo et al. 2014] teve como objetivo calcular como robôs poderiam se locomover dentro da estação espacial internacional para realizar operações dentro da mesma. No trabalho aparelhos de celular foram utilizados para emitir sinais de potências em Wi-Fi, como visto na Figura 1, e um algoritmo de regressão realizou a localização dos robôs, o algoritmo utilizado foi o *Gaussian Process* (GP). A Figura 1 está dividida em três partes, sendo a primeira parte (Figura 1(a)) coleta de sinais RSSI em um mês, a segunda parte (Figura 1(b)) indica a coleta em um mês diferente da primeira, e por fim a terceira parte (Figura 1(c)) tem um mapa das duas coletas sobrepostas na Estação. Com posse do mapa indicado em (c) o algoritmo baseado em filtros de partícula foi aplicado a fim de estimar as posições dos robôs na estação espacial, que chegaram a ter resultados com erros até 1.90 metros de indicação do local em que o robô se encontra.

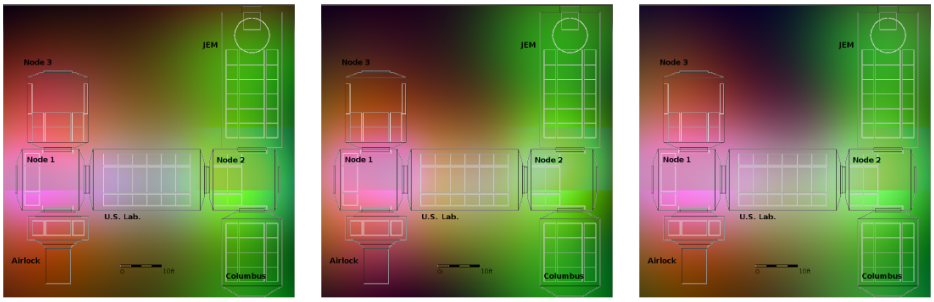


Fig. 1. Mapa de cores da *International Space Station* ISS, em que (a) Coletas de RSSI para o primeiro *dataset*, (b) Coletas de potência em RSSI para o segundo *dataset*, e (c) Junção das duas coletas em um mapa de cor [Yoo et al. 2014].

[Marti et al. 2012] realizam testes reais em ambientes com pouca visibilidade, e para tal utilizam algoritmos como KNN, RNA, juntamente com uma solução proposta pelos autores chamada de ARIEL, que é baseada no algoritmo do KNN com base de cálculo na distância euclidiana. O método ARIEL teve um desempenho melhor que os outros métodos já citados (KNN, distância mínima e redes neurais). Como a intenção do trabalho era auxiliar robôs em ambientes com pouca visibilidade o mesmo conta com uma placa que emite luz para locomoção, entretanto apenas informações coletadas por nós de rede Zigbee foram considerados, em vez de unir os dados da rede e os de luminosidade, mesmo assim, o sistema pôde permitir que um robô navegue em uma área com baixa visibilidade e chegar a pontos específicos de interesse. Em [Jekabsons et al. 2011] é utilizado o WiFi como técnica para localização em ambientes internos reais para determinar posições de pessoas. O trabalho indica que um aparelho móvel com WiFi utiliza o sinal de diversas estações-base espalhadas em um ambiente interno, sem a necessidade de um servidor *back-end*, se locomove e com auxílio dos algoritmos classificadores (KNN e WKNN) o trabalho conclui de que com esses diferentes dispositivos com a tecnologia Wi-Fi o erro calculado com os algoritmos chegam a ficarem entre 2,5 e 4,0 metros no ambiente testado.

De acordo com os trabalhos mostrados, vem sendo bastante utilizado métodos de aprendizado de máquina para localização de agentes em ambientes internos, nesse sentido a utilização de redes neurais recorrentes acabam por serem levadas em considerações, por se tratar de uma variação de redes neurais que são bastantes utilizadas em situações de localização.

3. REDES NEURAIS RECORRENTES

Esta seção mostra os conceitos básicos de uma rede neural recorrente e como se dá a aplicabilidade da mesma em duas situações: quando há recorrência de camadas escondidas ou recorrência de entradas.

Redes neurais recorrentes (RNN) são ditas recorrentes devido a sua característica de executar a mesma tarefa para cada um dos elementos de uma sequencia, com a saída sendo dependente da execução anterior. Em outras palavras, as RNNs tem um tipo de memória neural que captura informações calculadas até aquele momento da execução do problema [Schuster and Paliwal 1997].

A seguir na Figura 2 tem-se a estrutura básica de uma rede neural recorrente. Nesse modelo cada neurônio recorrente (r1, r2, r3, r4, r5) ocorre o desdobramento do neurônio em que a entrada de informações de tempos de execuções anteriores sejam consideradas no passo a ser realizado. E a utilização dessas informações passadas que caracteriza a chamada memória de uma RNN. Diferentemente de uma rede neural tradicional, que usa diferentes parâmetros em cada camada, a RNN utiliza os mesmos parâmetros para todos os passos, entretanto, apesar dos mesmos parâmetros, existem diferentes entradas para cada passo [Graves and Jaitly 2014].

Assim como em um cérebro humano, o computador aprende sequencias de informações, como em

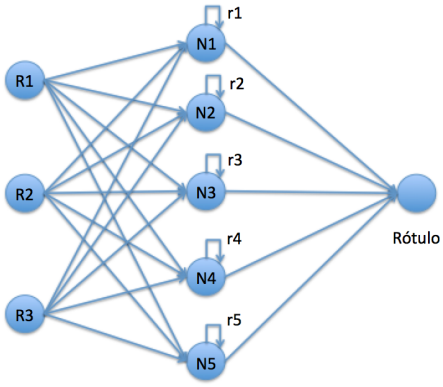


Fig. 2. Esquema de uma rede neural recorrente. Neste caso foi uma rede que foi utilizada no experimento, em que tem-se três entradas e cinco neurônios com recorrência, para apenas uma saída que significa o rótulo com a posição de um agente em um ambiente interno.

uma lista encadeada, em que uma informação puxa a próxima de maneira mais fácil [Sak et al. 2014]. Em geral quando os dados possuem sequencias as RNN são utilizadas, como um problema de classificação de posição de pessoas, se um grupo de informações dizem que uma pessoa encontra-se em uma sala e o tempo entre as coletas é pequeno, essa pessoa tem grande chance de ainda estar nessa mesma sala.

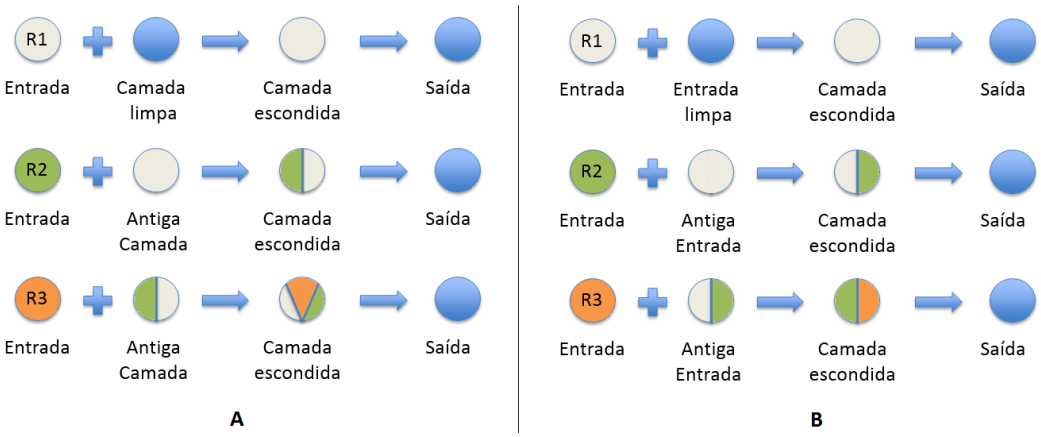


Fig. 3. Comportamento das camadas escondidas em uma RNN. (a) Recorrência de entradas. (b) Recorrência de camadas escondidas.

Redes neurais geralmente tem sua camada escondida baseada apenas na camada de entrada, entretanto as redes neurais recorrentes possuem dois tipos clássicos de recorrências: a recorrência por camada de entrada e a recorrência por camada escondida. A recorrência por camada de entrada indica que a camada escondida é baseada nas últimas camadas de entrada, como na Figura 3 (a). A recorrência por camada escondida indica que a próxima camada escondida será baseadas nas camadas escondidas anteriores [Pascanu et al. 2013] como na Figura 3 (b). Redes neurais recorrentes podem ser de diversos tipos de arquiteturas, tais como as bi-direcionais RNN, redes recorrentes totalmente conectadas, ou as *Long short term memory* (LSTM) que são as mais comuns dentre todas as arquiteturas. As LSTM possuem blocos de memória em vez de neurônios escondidos, em que um bloco pode ser formado por uma ou mais células de memória que liga as entradas com as células, e as células com as saídas da rede, para indicar o que a rede está aprendendo [Corrêa et al. 2008].

4. AMBIENTE DE TESTE

Na presente Seção é descrito o conjunto de passos empregado para realização do trabalho. A metodologia de coleta, o tratamento e análise dos resultados, bem como detalhamento do ambiente de teste escolhido. A seguir é apresentado em formatos de itens esses passos:

—Demarcação do ambiente e definição de regiões de um andar de prédio comercial;
—Preparação dos dispositivos e cenário de teste.
—Coleta e armazenamento dos RSSI para o cenário de teste.
—Preparação dos dados coletados no cenário de teste.
—Treinamento e validação, usando a base de dados coletada.
—Análise dos resultados obtidos.

Para tais itens o ambiente utilizado está presente na Figura 4, e nela é vista a demarcação do mesmo través das regiões de coleta. Cada uma dessas regiões de coleta (C1, C2, C3, C4) tem três pontos de interesse representados por pontos. Cada uma das coletas duraram cerca de 10 minutos, sendo aplicada a triplicata aleatória, ou seja, as coletas não foram realizadas de maneira sequencial, além disso foram realizadas por três vezes, o que resulta em cerca de 120 minutos de coleta de potências. Cada uma dessas regiões de coleta foram divididas em três pontos de interesse com aproximadamente 3 minutos de duração. Com a duração completa do teste houve cerca de mais de 12.000 potências de sinais dos três Roteadores presentes na Figura 4 (R1, R2, R3).

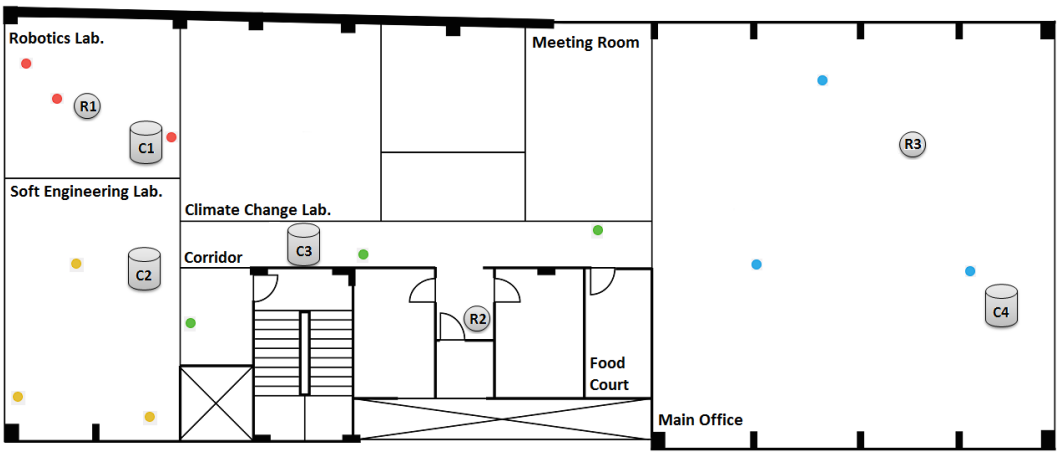


Fig. 4. Planta-baixa do ambiente testado. No próprio estão identificadas as posições dos roteadores (R1, R2, R3) e as regiões de coletas (C1, C2, C3, C4, C5), sendo que em cada uma dessas regiões houveram três pontos de coletas, representadas pelos pontos em cada região.

Com base na planta do experimento, os roteadores indicados na Figura 4 foram configurados como *access points* para sempre serem visíveis a pessoa que indica sua posição, a Figura 5 mostra essa arquitetura. Nela a pessoa sempre indica a sua posição buscando as informações da rede e criando uma lista de potências em RSSI para ser a predição do sistema. e assim ser visível a posição da pessoa a outras pessoas, e a ela mesma.

Uma vantagem dessa abordagem ocorre quando não existe um time de resgate disponível em uma situação de desabamento de teto em uma mina subterrânea, por exemplo. Nesse tipo de arquitetura a pessoa que sabe a sua posição poderia empregar mecanismos para sair de tal situação baseada no seu local e nas possíveis rotas próximas a ele, ou até mesmo, através das orientações de uma pessoa externa a situação.

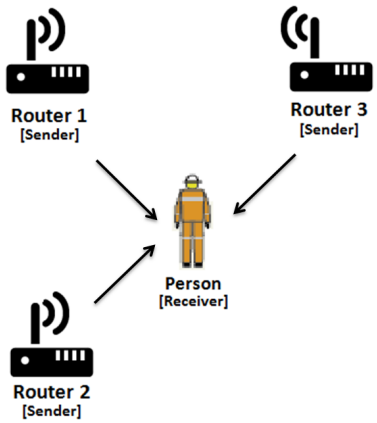


Fig. 5. Arquitetura de Localização em que a pessoa deve ter um dispositivo que recebe RSSI de roteadores e emprega aprendizagem de máquina para obter a sua localização.

Quanto aos parâmetros da RNN, o modelo foi executado dez vezes para cada uma das três diferentes quantidades de neurônios na camada de memória. Foi utilizado vinte mil épocas para cada execução. Primeiramente o código foi utilizado com apenas 2 neurônios e em seguida 5 neurônios e por fim 10 neurônios na camada de memória, ou camada escondida.

Com uma função de ativação linear foi-se calculado duas métricas, que foram: o erro quadrático médio e a acurácia da execução. O modelo de RNN foi considerado uma LSTM totalmente conectada entre suas camadas, com três neurônios de entrada, caracterizando os três roteadores e apenas um neurônio de saída, que é a sala em que a pessoa se encontra. A Figura 2 mostrou a rede recorrente com cinco neurônios de recorrência. Cada execução levou em conta todos os dados de coleta nos três pontos de cada sala em um único arquivo dividido em janelas de tempo. Cada uma dessas janelas possuiam um conjunto de dez leituras de cada roteador. A execução dos testes também utilizou esse único arquivo, dividindo essa base em duas partes: 70% para treino e 30% para valização.

5. RESULTADOS E DISCUSSÕES

Nessa seção serão mostrados os resultados das redes neurais recorrentes obtiveram em acurácia na localização de agentes em ambientes internos. Além da matriz de confusão para o melhor caso e o erro médio quadrático para esse caso.

O gráfico da Figura 6 indica o resultado da taxa de acerto contida na execução de dez vezes o algoritmo de RNN para diferentes quantidades de neurônios recorrentes, sendo escolhidos, dois, cinco e dez neurônios para a comparação.

Ainda sobre a Figura 6, a média de execução das mesmas foram bem diferentes. Em que com dois neurônios a média ficou em torno de 39% de taxa de acerto da posição de agentes em ambientes internos. Com cinco neurônios a situação melhora um pouco, entretanto houve uma variação muito grande entre os resultados, sendo que a média foi cerca de 59% de acerto. Com dez neurônios houve o melhor resultado dentro de todas as execuções do algoritmo com uma taxa de acerto de 94% e com uma média de 79% de acurácia em dez execuções da RNN.

Em seguida na Tabela 1 é verificada a matriz de confusão para o melhor caso das RNNs, em que com uma taxa de acerto de 94% com dez neurônios de recorrência, a matriz se apresenta com erros gerados em salas vizinhas, como visto na Figura 4, existem quatro pontos de coleta sendo que esses quatro pontos são divididos por três pontos distintos dento de uma sala. Os erros do algoritmo foram originados justamente por esses pontos próximos as diferentes salas testadas.

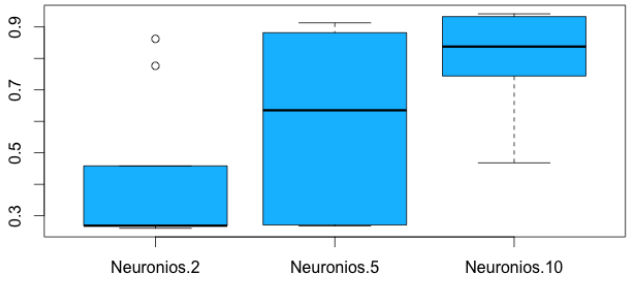


Fig. 6. Gráfico de Boxplot contendo o resultado de taxa de acerto do algoritmo de rede neural recorrente com 2 neurônios, com 5 neurônios e com 10 neurônios.

Table I. Matriz de confusão para o melhor caso de RNN em localização de agentes em ambientes internos.

	SalaRob.	Sala Eng.	Corredor	Sala ADM
Sala Rob.	406	51	0	0
Sala Eng.	54	309	9	0
Corredor	0	2	409	10
Sala ADM	0	0	0	409

Por fim a Figura 7 mostra o decaimento do erro na execução do melhor caso para as 20000 épocas de treinamento do modelo contendo 10 neurônios e com taxa de acerto de 94%. Sendo possível ver a convergência do resultado ocorrendo próximo das primeiras épocas. Para melhor visualização um corte foi realizado na época 10000, pois a mesma já havia atingido o resultado.

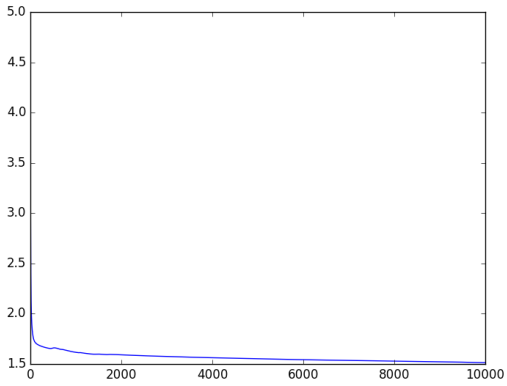


Fig. 7. Gráfico do erro quadrático médio para as 20000 ciclos de treinamento para o melhor caso com 94% de taxa de acerto.

6. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

A localização de agentes em ambientes internos ainda é um problema em aberto. Este trabalhos está inserido em um projeto de localização de pessoas em minas subterrâneas, buscando uma arquitetura de localização que seja de baixo custo e alta acurácia para os padrões de mineração. Nessa etapa

um algoritmo de redes neurais recorrentes foi desenvolvido para testar uma possível solução com um *testbed* realizado em um prédio comercial.

A utilização de aprendizado de máquinas tem sido uma área de pesquisa para localização, pois procura-se um método que consiga generalizar os ambientes internos com entradas de dados em RSSI, coletadas por roteadores satélites. Nesse sentido que a utilização dessa variação de rede neural foi escolhida, tendo em vista que a sintonia fina da mesma foi avaliada neste artigo. A utilização de redes neurais recorrentes para o problema de localização proposto atingiu uma taxa de acerto de 94% para o melhor caso, e a média de execuções com a maior quantidade de neurônios foi de 79%.

Para os próximos passos, (i) a comparação da utilização de redes neurais clássicas, redes neurais convolucionais e as redes neurais recorrentes são buscadas, (ii) teste de tolerância a falhas, (iii) teste de regressão, indicando a posição com erros em metros, (iv) além da coleta de dados no ambiente de mina subterrânea e a avaliação do comportamento dessas redes anteriormente citadas.

Assim, em termos de inovação industrial se tem a aplicação de desse problema em termos de resgate e salvamento de vidas humanas, bem como resgate de bens materiais em áreas inóspitas, além de vários campos a serem explorados na área científica, dado o fato de ainda possuírem campos abertos para essa pesquisa.

Agradecimentos

Os autores agradecem aos colegas Hanna Vitória (ITV), Helder Arruda (ITV), Gerson Serejo (ITV) e Bruno Faiçal (ICMC-USP) pelas conversas e inspirações. Além disso, os autores agradecem ao Prof. Dr. Cleidson R. B. Souza (ITV) e ao Dr. Joner Oliveira Alves (SENAI) pelos diversos auxílios no desenvolvimento desta pesquisa. Finalmente, os autores agradecem o apoio financeiro do Edital SENAI SESI de Inovação (CNI), da Vale S. A. e do CNPQ pela chamada 59/2013 MCTI/CT-Info/CNPq, processo 440880/2013-0.

REFERENCES

Corrêa, D. C., Salvadeo, D. H., Levada, A. L., Saito, J. H., and DA, N. Using lstm network in face classification problems, 2008.

Dalto, M. Deep neural networks for time series prediction with applications in ultra-short-term wind forecasting. *Industrial Technology (ICIT), 2015 IEEE International Conference on*, 2015.

Department, U. F. U.S. fire statistics, 2012.

Graves, A. and Jaitly, N. Towards end-to-end speech recognition with recurrent neural networks. In *ICML*. Vol. 14. pp. 1764–1772, 2014.

Harris, J., Kirsch, P., Shi, M., Li, J., Gagrani, A., Krishna, E. S., Tabish, A., Arora, D., Kothandaraman, K., Cliff, D., and Others. Comparative analysis of coal fatalities in Australia, South Africa, India, China and USA, 2006-2010. In *14th Coal Operator's Conference*, 2014.

Jekabsons, G., Kairish, V., and Zuravlyov, V. An analysis of wi-fi based indoor positioning accuracy. *Scientific Journal of Riga Technical University. Computer Sciences* 44 (1): 131–137, 2011.

Marti, J. V., Sales, J., Marin, R., and Jimenez-Ruiz, E. Localization of mobile sensors and actuators for intervention in low-visibility conditions: the zigbee fingerprinting approach. *International Journal of Distributed Sensor Networks* vol. 2012, 2012.

Parameswaran, A. T., Husain, M. I., Upadhyaya, S., et al. Is rssi a reliable parameter in sensor localization algorithms: An experimental study. In *Field Failure Data Analysis Workshop (F2DA09)*, 2009.

Pascanu, R., Gulcehre, C., Cho, K., and Bengio, Y. How to construct deep recurrent neural networks. *arXiv preprint arXiv:1312.6026*, 2013.

Sak, H., Senior, A., and Beaufays, F. Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *arXiv preprint arXiv:1402.1128*, feb, 2014.

Schuster, M. and Paliwal, K. K. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45 (11): 2673–2681, 1997.

Yoo, J., Kim, T., Provencher, C., and Fong, T. Wifi localization on the international space station. In *Intelligent Embedded Systems (IES), 2014 IEEE Symposium on*. pp. 21–26, 2014.

DataSex: um *dataset* para indução de modelos de classificação para conteúdo adulto

Gabriel S. Simões[1], Jônatas Wehrmann[1], Thomas S. Paula[1], Juarez Monteiro[1], Rodrigo C. Barros[1]

Pontifícia Universidade Católica do Rio Grande do Sul, Brasil - PUCRS
{gabriel.simoes.001, jonatas.wehrmann, thomas.paula, juarez.santos}@acad.pucrs.br
rodrigo.barros@pucrs.br

Abstract. O consumo indiscriminado de conteúdo adulto na Internet gera problemas comportamentais que, em alguns casos, pode desencadear patologias. A facilidade de acesso e o anonimato criam um ambiente propício ao consumo deste tipo de conteúdo. Neste cenário, classificadores podem ser treinados para identificar automaticamente conteúdo adulto, abrindo espaço para o monitoramento e o controle de acesso. Este trabalho descreve a criação do *DataSex*, um *dataset* composto por 286.920 imagens para indução de modelos de Aprendizagem de Máquina que possam efetivamente contribuir para este controle. Experimentos apontam que Redes Neurais Convolucionais podem atingir aproximadamente 95% de acurácia de teste quando treinadas com este conjunto de dados.

Categories and Subject Descriptors: I.2.6 [**Learning**]: Connectionism and neural nets, Induction

Keywords: machine learning, convolutional neural networks, computer vision, dataset

1. INTRODUÇÃO

O volume de conteúdo adulto disponível na Internet cresce constantemente e, em paralelo, cresce também a quantidade de problemas que o uso indiscriminado deste pode causar. Casos de vício em sexo são frequentes [Young 2008], no entanto, outras desordens como impotência e falta de apetite sexual começam a ser ralatadas[1][2]. Neste sentido, [Cooper 1998] chamou de *Triple A Engine* a relação dos 3 fatores que motivam o consumo de pornografia na Internet: i)*Accessibility*, em função da facilidade de acesso; ii)*Affordability*, referente ao baixo custo; iii) *Anonymity*, preservando a identidade do usuário.

Pequisas recentes em Aprendizagem de Máquina e Inteligência Artificial levaram ao surgimento de *Deep Learning* que, conforme [Bengio 2009], tem por objetivo descobrir *features* a partir de recursos de baixo nível com pouco esforço humano. Uma das estratégias mais difundidas de *Deep Learning* são as Redes Neurais Convolucionais (ConvNets)[LeCun and Bengio 1995] que, quando aplicadas em contextos de imagens, definem o novo estado-da-arte para tarefas como classificação e segmentação. Por estarem inseridas em um contexto de aprendizado supervisionado, a indução de modelos de ConvNets é significativamente dependente do conjunto de dados de treinamento.

Motivado pelos fatores apontados pelo *Triple A Engine* e pela capacidade de classificação de ConvNets, este trabalho descreve a criação do *DataSex*, um *dataset* que relaciona 286.920 imagens para indução de modelos por aprendizagem supervisonada, permitindo a identificação automática de conteúdo adulto. Experimentos apontam que *DataSex*, quando utilizado no treinamento de ConvNets, gera modelos que atingem aproximadamente 95% de acurácia em seu conjunto de teste, além de um

[1]https://goo.gl/RgyHOq
[2]http://goo.gl/tjuVVt

(a) animais. (b) invertebrados. (c) artrópodes.

Fig. 1: Exemplo de hierarquia de classes disponível em ImageNet.

volume significativo de classificações em função do tempo (100 imagens por segundo). Plotagens *t-distributed stochastic neighbor* (t-SNE), uma técnica para visualização de dados de alta dimensionalidade descrita em [Maaten and Hinton 2008], ilustram ganho de significado das *features* extraídas de modelos induzidos a partir do *DataSex*.

A seguir, a Seção 2 lista duas iniciativas relacionadas. A Seção 3 descreve a estrutura e o método para construção do *DataSex*. Resultados experimentais são apontados na Seção 4. Finalmente, discussão e iniciativas futuras são apresentadas na Seção 5.

2. TRABALHOS RELACIONADOS

Com base na literatura, diversas iniciativas compondo *datasets* de imagens para indução de modelos de aprendizagem de máquina são conhecidas. Algumas destas iniciativas tem por objetivo identificar conceitos em larga escala (i.e., ImageNet para mais de 2.000 conceitos), enquanto outras focam em conceitos específicos (i.e., MNIST para dígitos escritos à mão).

A seguir serão introduzidas duas iniciativas relacionadas: i) ImageNet, um *dataset* que aborda conceitos generalistas; ii) NPDI, um *dataset* binário que busca identificar conteúdos pornográficos e não-pornográficos.

2.1 *Dataset* ImageNet

O ImageNet [Deng et al. 2009] é um *dataset* constituído por aproximadamente 14 milhões de imagens rotuladas. A estruturação de conteúdo segue o WordNet [Kilgarriff and Fellbaum 2000], um banco de dados léxico para língua inglesa que classifica seus conceitos através de um conjunto de sinônimos (*synonym set* ou *synset*). Atualmente o ImageNet possui 21.841 *synsets*, sendo uma de suas principais características a organização de suas classes que compõe uma hierarquia semântica. A Figura 1 apresenta um exemplo de hierarquia entre os *synsets* contidos no ImageNet, onde o conceito ilustrado pela Figura 1c é uma especialização dos conceitos ilustrados pelas Figuras 1b e 1a.

Utilizando *crowdsourcing*[Brabham 2008], o conteúdo do ImageNet foi analisado de maneira a torná-lo um *dataset* livre de ruído. Um dos objetivos do *dataset* é ter diversidade de instâncias para os conceitos por ele cobertos. Desta maneira, cada *synset* é composto por imagens com diferentes objetos, posições e planos de fundo, dispondo cada um de uma média aproximada de 650 imagens, características suficientes para a criação de modelos de classificação com alto poder de generalização.

2.2 *Dataset* NPDI

O dataset NPDI [Caetano et al. 2014] é composto por imagens extraídas de aproximadamente 80 horas de vídeos pornográficos e não-pornográficos obtidos através da Internet. Neste dataset, a classe não pornográfica é sub-dividida em instâncias fáceis e difíceis. São considerados vídeos não-pornográficos fáceis aqueles que apresentam situações e/ou elementos genéricos do mundo real, tais como bicicletas, carros, aviões ou brinquedos em ações como correr, caminhar ou dirigir. Por outro lado, a sub-divisão não-pornográfico difícil relaciona situações que destacam a exposição corporal, como fotos em trajes de banho, lutas ou amamentação de bebês. A Figura 2 apresenta instâncias de exemplo onde é

possível observar diferentes níveis de exposição de pele proporcionadas por situações cotidianas que, não necessariamente, significam pornografia.

(a) não pornográfico fácil. (b) não pornográfico difícil. (c) pornográfico

Fig. 2: Exemplos de imagens do *dataset* NPDI.

Para a composição dos conjuntos, os vídeos foram pré-processados identificando todas as cenas, o que permitiu a extração dos *keyframes*[3]. O resultado deste pré-processamento são 802 conjuntos contendo entre 1 e aproximadamente 320 *frames*, conforme a quantidade de cenas de cada vídeo. Ao todo foram identificadas 16.727 cenas, onde os vídeos pornográficos apresentam em média 15.6 cenas, os não-pornográficos fáceis 33.8 cenas e os não-pornográficos difíceis 17.5 cenas, totalizando 16.727 imagens. O conteúdo do *dataset* NPDI apresenta diversidade étnica, já que participam das cenas asiáticos, negros, brancos e mestiços. Por outro lado, o *dataset* apresenta pontos controversos, como a ocorrência de vídeos com uma única cena, além da presença de desenhos e animações que dificultam a classificação.

3. *DATASEX*

Este trabalho apresenta *DataSex*, um *dataset* binário formado por duas classes, *adult* e *benign*, compostas por imagens extraídas de duas fontes distintas: i) uma seleção do ImageNet e ii) consultas direcionadas coletadas por *web crawling* [Kobayashi and Takeda 2000], resultando em um *dataset* significativamente maior (aproximadamente 17×) que o *dataset* NPDI, outra iniciativa descrita pela literatura. *DataSex* não faz distinção entre as instâncias de suas classes, além de não manter qualquer relação temporal entre elas. Até o presente momento, pelo melhor que se conhece dentre o estado-da-arte, nenhum outro *dataset* apresenta volume semelhante de instâncias, fazendo deste o maior conjunto rotulado de imagens pornográficas e não-pornográficas até então relatado. Uma versão resumida de *DataSex* contendo 28.692 imagens, aproximadamente 10% do total de instâncias da versão final, está disponível para *download* através do endereço `https://goo.gl/J6yWbH`.

3.1 Composição dos conjuntos

DataSex é dividido em sub-conjutos de treino, teste e validação, sendo estes compostos por imagens rotuladas conforme às classes *benign* e *adult*. *DataSex* ditingue-se de NPDI fundamentalmente por: i) tratar instâncias de maneira isolada; ii) restringir o conteúdo da classe positiva em imagens de forte apelo pornográfico; iii) pelo volume de imagens aproximadamente 17× maior (16.727 vs. 286.920). A distribuição de instâncias em *DataSex*, assim como seus respectivos volumes de dados, podem ser observados na Tabela I. A Figura 3 exibe amostras para ambas as classes. Para exibição neste trabalho, por motivos óbvios, imagens pertencentes à classe *adult* foram editadas.

Table I: Distribuição dos subconjuntos em *DataSex*.

Conjunto	*Benign*	*Adult*	*Benign* + *Adult*	Volume em Disco (≈)
Treino	95.388	95.388	190.776	18,0GB
Teste	32.048	32.048	64.096	5,9GB
Validação	16.024	16.024	32.048	3,0GB
Total	143.460	143.460	286.920	26,9GB

[3] *Keyframe* é o frame posicionado ao centro de uma dada cena.

(a) exemplos da classe *benign*.

(b) exemplos da classe *adult*.

Fig. 3: Amostra dos conjuntos *benign* e *adult* de *DataSex*.

3.2 Extração e composição de dados

A composição de dados do *DataSex* foi realizada por estratégias distintas. A classe *benign* foi criada a partir de um sub-conjunto de imagens extraídas do ImageNet, enquanto que a classe *adult* utilizou imagens obtidas através de consultas específicas, conforme descrito a seguir.

O conjunto *benign* foi formado por imagens para compor a classe negativa, partindo de uma seleção de imagens do ImageNet. Esta seleção priorizou 197 *synsets* que possuem ao todo 93.116 imagens relacionando temas que exibem pessoas em contextos genéricos. Desta maneira é possível contrapor pessoas em cenários unicamente pornográficos, o que permite treinar modelos de classificação suficientemente generalistas. Para equilibrar a quantidade de instâncias de ambas as classes, outras 50.344 imagens do ImageNet foram selecionadas aleatoriamente e incorporadas ao conjunto *benign*.

Por outro lado, a classe positiva foi composta com base em 243.968 imagens de diferentes *web sites* de conteúdo adulto. As imagens foram obtidas por um *web crawler* configurado para realizar consultas com termos específicos (e.g. nudez, sexo, pornografia). Dentre o total de imagens, 138.475 são estáticas enquanto que 105.493 são animações no formato GIF, das quais foram extraídos somente o primeiro *frame*. As imagens obtidas apresentam configurações de qualidade e dimensões diversas, além de diferentes formatos (GIF, JPG, PNG). Após a realização de pré-processamento, foram removidas as imagens duplicadas, corrompidas ou que não atingiram as dimensões mínimas (largura ou altura < 128). As 143.460 imagens resultantes do pré-processamento foram utilizadas para compor a classe positiva (*adult*).

3.3 Pré-processamento

Para garantir a acurácia e corretude dos modelos treinados a partir de *DataSex*, é fundamental que não existam instâncias duplicadas, especialmente repetições que envolvam treino/teste ou treino/validação. Desta maneira, duplicidades foram eliminadas através de uma análise automática de similaridade entre imagens que utilizou *features* extraídas de uma rede convolucional para comparar todo o conjunto.

Uma Rede Neural Convolucional (ConvNet) é uma estratégia de *Deep Learning* que combina três ideias para atingir um grau de variação de deslocamento, escala e distorção sendo elas: i) campos receptivos locais (conhecidos como filtros); ii) parâmetros compartilhados; iii) *pooling* espacial. A operação de convolução substitui a multiplicação de matrizes totalmente conectadas, presentes em redes neurais convencionais. Esta operação garante as duas primeiras ideias apontadas anteriormente, além de reduzir a quantidade de parâmetros. Filtros convolucionais, também conhecidos como *kernels*, são otimizados utilizando o algoritmo *backpropagation*[Rumelhart et al. 1988]. Este processo pode ser compreendido como a construção de um extrator de *features*. Quando isoladas, estas *features* podem ser aplicadas em diversas tarefas (e.g., agrupamento, recuperação de conteúdo ou aproximação por similaridade) [LeCun and Bengio 1995].

$$v_{ij}^{xy} = relu\left(b_{ij} + \sum_{m}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1} w_{ijm}^{pq} v_{(i-1)m}^{(x+p)(y+q)}\right) \quad (1)$$

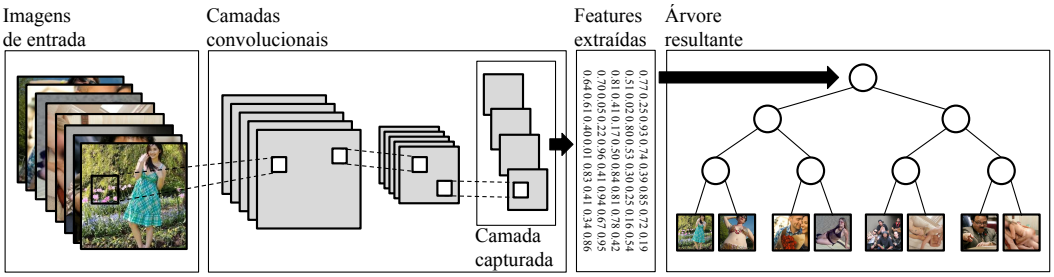


Fig. 4: Geração de árvore para remoção de duplicatas.

A Equação 1 define uma convolução onde (x, y) é a posição no j^{th} mapa de *features* da i^{th} camada; m indexa o conjunto de mapas de *features*, b_{ij} representa o valor do *bias* correspondente, w_{ijm}^{pq} representa os valores dos pesos na posição (p, q), e P_i e Q_i representam largura e altura do filtro, respectivamente. Nesta equação, a função de ativação ReLU [Krizhevsky et al. 2012] é aplicada como fonte de não-linearidade. Essencialmente, ReLU limita minimamente o valor de saída de convoluções a zero (i.e., $relu(v) = max(0, v)$).

O procedimento adotado, ilustrado pela Figura 4, extraiu 1024 *features* convolucionais de cada imagem a partir da última camada de convolução de uma rede convolucional com arquitetura *GoogLeNet* [Szegedy et al. 2015], treinada com o *dataset* ImageNet. Após extraídas, as *features* foram utilizadas para treinar uma estrutura de árvore do tipo KD-Tree [Bentley 1975], que apresenta uma complexidade de tempo para operação de busca por vizinhos mais próximos de $O(\log n)$. Finalmente, sendo I o conjunto de todas as imagens e $Q(i)$ uma operação de consulta que retorna *true* para $q > 1$, onde q representa a quantidade de réplicas da imagem i no conjunto I, após a execução de $Q(i) \forall i \in I$ foram excluídas todas as ocorrências de i para valores de $Q(i) = true$. Observou-se que este procedimento resultou na identificação e remoção de 31.448 duplicatas presentes no *DataSex*.

Mesmo após a execução do pré-processamento, é possível observar a ocorrência de ruído na classe *adult*. Entende-se por ruído instâncias rotuladas erroneamente. O ruído foi mensurado a partir da observação de 3 amostras de 95 imagens aleatórias (aproximadamente $0,01\%$) extraídas da classe *adult*. Após análise das 3 amostras foram contabilizados em média 10.5% de ruído com desvio padrão de $\pm 2\%$. Em função de tratar-se de um sub-conjunto de ImageNet, declaradamente um *dataset* livre de ruído, não foi contabilizado ruído para a classe *benign*. A Figura 5 apresenta uma amostra de instâncias erroneamente rotuladas extraídas dos conjuntos avaliados. Observa-se que, mesmo não-pornográficas, a amostra apresenta uma forte tendência à exposição corporal.

3.4 Aplicações

Paralelamente à massiva popularização do uso de dispositivos que permitem exibir e registrar imagens, surgem diversas possibilidade de aplicações para classificadores treinados a partir de *datasets* de conteúdo adulto. Iniciativas como restrição de acesso a conteúdos inadequados em redes sociais ou controle parental são breves exemplos.

Redes sociais recebem diariamente montantes significativos de conteúdo, sendo parte deste composto por imagens e vídeos, sabidamente dados não estruturados. Dado o volume, a análise manual do

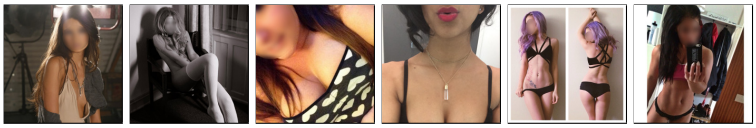


Fig. 5: Amostra de instâncias ruidosas da classe *adult*.

conteúdo torna-se impraticável. Por outro lado, um classificador automático pode realizar esta tarefa, promovendo o encaminhamento necessário aos casos onde conteúdo inapropriado seja identificado.

O consumo de séries, filmes e shows por *streaming* vem crescendo significativamente. Estes conteúdos normalmente apresentam classificação indicativa previamente apontada por entidades reguladoras (i.e., Cocind[4], MPAA[5]). No entanto, a classificação indicativa aponta uma análise holística da obra, permitindo muitas vezes que partes específicas apresentem conteúdo inapropriado para crianças, por exemplo. Desta maneira, a análise de vídeos em tempo de execução pode impedir automaticamente a visualização de segmentos específicos, conforme avaliação automática de cada *frame*.

4. EXPERIMENTOS

Para verificar a acurácia de modelos treinados com *DataSex*, assim como mensurar o tempo necessário para classificar conjuntos de imagens, foram executados experimentos que abordaram treinamento, validação e teste de uma arquitetura *GoogLeNet* para classificação de imagens em pornográficas e não-pornográficas. Metodologia e resultados serão descritos a seguir.

4.1 Metodologia experimental

O treinamento foi executado por 200 épocas, partindo de pesos aleatórios. Cada época corresponde a uma observação completa do conjunto de treinamento. Foi utilizado o otimizador Gradiente Descendente Estocástico, ajustado conforme mecanismo descrito em [Bottou 2012], utilizando os seguintes hiperparâmetros: $\alpha = 1 \times 10^{-3}$, α (γ)=4%, *momentum* $= 9 \times 10^{-1}$, *weight decay* $= 5 \times 10^{-4}$, sendo α a taxa de aprendizagem e α (γ) um fator para redução da taxa de aprendizagem aplicado a cada 2 épocas. Para inicialização dos pesos foi utilizado o método descrito em [He et al. 2015].

O treinamento foi realizado com as seguintes configurações de *hardware*: GPU NVIDIA M40 com 12GB de memória, executando sobre um CPU Intel Xeon E5-2603 com 128GB de memória principal, com o *dataset* armazenado em disco de estado sólido. Quanto ao *software*, foi utilizado o *framework Caffe*[6], sobre sistema operacional Ubuntu 14.04.

O período de treinamento consistiu em aproximadamente 6 dias. Analisando os resultados de validação obtidos para cada época, foi possível observar a formação de um topo de acurácia, o que indicou a convergência do modelo, sustentando o encerramento do treinamento.

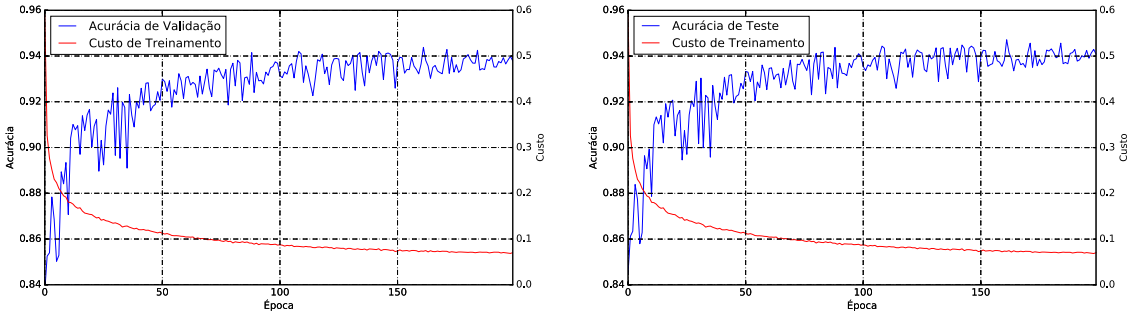
4.2 Resultados

Encerrado o treinamento, observou-se os melhores resultados de validação ocorrendo na 162º época. A partir desta observação, utilizou-se o melhor modelo para executar os procedimentos de avaliação sobre o conjunto de teste. A Tabela II ilustra um cenário bastante promissor, onde valores para acurácia e custo para ambos os conjuntos apresentaram nítido equilíbrio, diminuindo a hipótese de *overfitting*.

Table II: Resultados obtidos na 162º época.

Conjunto	Acurácia	Custo	Precisão	Revocação
Teste	0,9471	0,2430	0,9492	0,9559
Validação	0,9438	0,2551	0,9445	0,9539

[4]Coordenação de Classificação Indicativa
[5]Motion Picture Association of America's
[6]`http://caffe.berkeleyvision.org/`



(a) custo e acurácia de validação.

(b) custo e acurácia de teste.

Fig. 6: Evolução das acurácias e custos para validação e teste.

Após a avaliação dos resultados de teste para o modelo da 162º época, o procedimento de observação foi conduzido para os 199 modelos restantes. A Figura 6 exibe os valores de acurácia e custo para cada uma das 200 épocas treinadas, para ambos os conjuntos: validação e teste. Observando a figura é possível concluir que os resultados mantiveram o equilíbrio para todos os modelos testados, o que reforça a hipótese da não ocorrência de *overfitting*. A Figura 7 exibe uma amostra de erros de classificação do conjunto de teste para ambas as classes.

Ainda durante a execução dos teste foi possível analisar o tempo necessário para avaliar cada imagem. Foi observado que lotes de 300 imagens são classificadas em aproximadamente 3 segundos, resultando em 1×10^{-2} segundo por imagem, ou 100 imagens por segundo. Assumindo que 1 segundo de vídeo apresenta aproximadamente 24 imagens pode-se concluir que, executando em hardware compatível, é possível aplicar análise semelhante (quadro-a-quadro) sobre vídeos.

(a) erroneamente classificadas como *adult*.

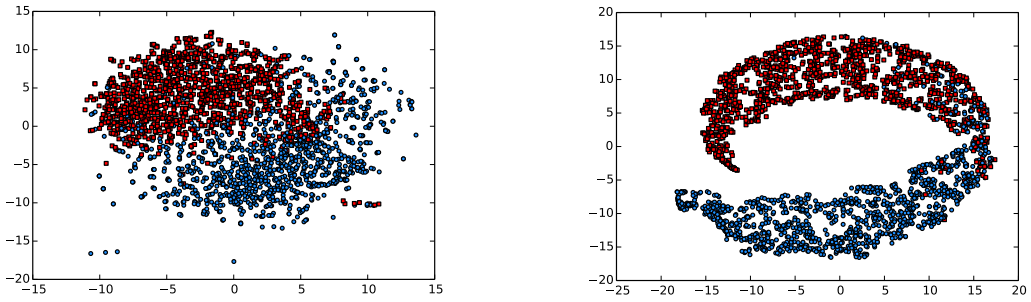
(b) erroneamente classificadas como *benign*.

Fig. 7: Amostra de erros de classificação extraídos do conjunto de teste.

A Figura 8 apresenta gráficos t-SNE para visualização das *features* de instâncias de validação do *DataSex* extraídas de uma ConvNet com arquitetura *GoogLeNet*. Observando os gráficos é possível perceber significativa distinção dos conjuntos quando as *features* são extraídas do modelo treinado com *DataSex*(Figura 8b) em comparação com a separação gerada com *features* extraídas do modelo treinado com ImageNet (Figura 8a), reforçando a eficácia do *DataSex* como fonte de dados para indução de modelos para detecção de conteúdo adulto.

5. DISCUSSÃO E TRABALHOS FUTUROS

Este trabalho descreveu *DataSex*, um *dataset* binário de imagens para classificação de conteúdo adulto. Com base no atual estado-da-arte, pode-se afirmar que esta iniciativa trata-se do maior conjunto rotulado para este propósito. Dada a acurácia de teste observada nos experimentos ($\approx 95\%$) e o volume de instâncias disponível, pode-se concluir que *DataSex* pode ser aplicado como fonte de dados para indução de modelos de Aprendizagem de Máquina para diferentes fins, tais como gerenciamento de conteúdo e controle parental, além de também poder ser utilizado como *benchmark* para avaliação de algoritmos de classificação.



(a) *features* de modelo treinado com ImageNet. (b) *features* de modelo treinado com *DataSex*.

Fig. 8: Representações t-SNE para instâncias de validação. Azuis = *benign*, vermelhos =*adult*.

Como trabalhos futuros, observa-se a necessidade de: i) incrementar o número de instâncias; ii) diminuir a quantidade de ruído; iii) rotular instâncias *pixel*-a-*pixel*, de maneira que *DataSex* possa ser aplicado para problemas de segmentação de imagens.

6. AGRADECIMENTOS

Esta iniciativa agradece a Motorola Mobility^{TM} e as agências brasileiras de fomento à pesquisa científica CAPES e CNPq, por custearem este trabalho. Finalmente, registra-se o agradecimento a NVIDIA Corporation^{TM} pela doação de um GPU Tesla K40 por vezes utilizado neste trabalho.

REFERENCES

Bengio, Y. Learning deep architectures for ai. *Foundations and trends in Machine Learning* 2 (1): 1–127, 2009.

Bentley, J. L. Multidimensional binary search trees used for associative searching. *Communications of the ACM* 18 (9): 509–517, 1975.

Bottou, L. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*. Springer, pp. 421–436, 2012.

Brabham, D. C. Crowdsourcing as a model for problem solving an introduction and cases. *Convergence: the international journal of research into new media technologies* 14 (1): 75–90, 2008.

Caetano, C., Avila, S., Guimaraes, S., and Araújo, A. d. A. Pornography detection using bossanova video descriptor. In *2014 22nd European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 1681–1685, 2014.

Cooper, A. Sexuality and the internet: Surfing into the new millennium. *CyberPsychology & Behavior* 1 (2): 187–193, 1998.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, pp. 248–255, 2009.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

Kilgarriff, A. and Fellbaum, C. Wordnet: An electronic lexical database, 2000.

Kobayashi, M. and Takeda, K. Information retrieval on the web. *ACM Comput. Surv.* 32 (2): 144–173, June, 2000.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. pp. 1097–1105, 2012.

LeCun, Y. and Bengio, Y. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks* 3361 (10): 1995, 1995.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of Machine Learning Research* 9 (Nov): 2579–2605, 2008.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. Neurocomputing: Foundations of research. MIT Press, Cambridge, MA, USA, Learning Representations by Back-propagating Errors, pp. 696–699, 1988.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 1–9, 2015.

Young, K. S. Internet sex addiction: Risk factors, stages of development, and treatment. *American Behavioral Scientist* 52 (1): 21–37, 2008.

Análise de Redes Sociais Profissionais por meio de Análise Formal de Conceitos

Paula. R. C. Silva, Wladmir. C. Brandão, Luis. E. Zárate

Pontifícia Universidade Católica de Minas Gerais, Brasil
paula.raissa@sga.pucminas.br, {wladmir,zarate}@pucminas.br

Abstract. Diante da recente proliferação das redes sociais, em especial daquelas focadas em negócios, as empresas passaram a considerar importante a análise do comportamento de profissionais, próprios ou potenciais, dentro dessas redes. Nesse cenário e, considerando o volume massivo de informação presentes nas redes sociais profissionais, torna-se fundamental a adoção de métodos e técnicas computacionais efetivas para a análise das relações presentes nos dados obtidos a partir dessas redes. Nos últimos anos uma nova técnica vêm se mostrando efetiva para análise de dados de redes sociais, por identificar estruturas conceituais dentro de conjuntos de dados. Trata-se da Análise Formal de Conceitos (AFC). No presente trabalho propomos uma abordagem baseada em AFC para analisar redes sociais profissionais a partir da obtenção de um tipo de regra de associação específico, conhecido como implicação própria, que permite visualizar de maneira direta correlações existentes entre variáveis consideradas na análise. Como estudo de caso, a abordagem será aplicada à rede social profissional *LinkedIn*.

Categories and Subject Descriptors: H.2.8 [**Database Applications**]: Data Mining

Keywords: formal concept analysis, impec algorithm, proper implications, social networks

1. INTRODUÇÃO

Num mundo cada vez mais interconectado, as redes sociais objetivam atender as necessidades de comunicação e informação de diferentes grupos de usuários [Russell 2013]. O *LinkedIn*[1] destaca-se como a maior e mais popular rede social profissional do mundo, com mais de 400 milhões de usuários distribuídos em mais de 200 países e territórios [LinkedIn 2016]. Essa massiva fonte de informação profissional tem sido explorada por gestores de recursos humanos para descobrir profissionais com competências adequadas para ocupar posições organizacionais específicas. No *LinkedIn*, há informações disponibilizadas pelo próprio usuário que caracterizam de forma direta ou indireta suas competências profissionais.

Com o objetivo de minerar redes sociais, uma teoria baseada nos reticulados conceituais, vem sendo utilizada com este fim [Dufour-Lussier et al. 2010], [Cuvelier and Aufaure 2011] e [Missaoui et al. 2013] . Análise Formal de Conceitos (AFC) é uma teoria de análise de dados, com fundamentação matemática, que identifica estruturas conceituais a partir de um conjunto de dados [Ganter et al. 2005]. Um dos produtos dessa técnica são as regras de implicação, extraídas a partir de um contexto formal. A técnica permite que, a partir da modelagem conceitual do problema, representado por meio de um contexto formal, seja possível, após a aplicação de algoritmos específicos, extrair regras de implicação.

Neste artigo, é proposta uma abordagem baseada na teoria da AFC para análise de redes sociais (ARS), objetivando descobrir padrões de comportamento profissional a partir de dados disponíveis na rede social *LinkedIn*, combinando implicações próprias e abordagens teóricas de modelos de seleção por competências. A abordagem proposta trata do problema de mineração de perfis profissionais fornecendo um modelo de domínio do problema adequado ao contexto. Para tanto, aplicou-se o

[1] https://www.linkedin.com/

algoritmo *Impec* [Taouil and Bastide 2001], cujo objetivo é extrair implicações próprias, e analisou-se o conjunto encontrado a partir de métricas de classificação de regras de implicação. O conjunto de regras é composto por implicações próprias, cujo lado esquerdo é mínimo e o direito possui apenas um atributo. Esse tipo de regra foi adotado por possibilitar visualizar, de maneira direta, o conjunto mínimo de requisitos que implicam em um objetivo (competência profissional).

Este trabalho está organizado da seguinte maneira: a Seção 2 apresenta uma breve revisão da teoria da análise formal de conceitos. Na Seção 3 os trabalhos relacionados são sucintamente descritos. Na Seção 4 é apresentada a metodologia baseada em AFC para ARS. Na Seção 5 são apresentados os experimentos e análises dos resultados. Na Seção 6 são apresentadas as considerações finais e trabalhos futuros.

2. ANÁLISE FORMAL DE CONCEITOS

Análise Formal de Conceitos (AFC) é uma teoria que identifica estruturas conceituais dentro de um conjunto de dados, relacionados a um domínio de problema específico[Ganter and Wille 2012]. AFC pode oferecer instrumentos teóricos e algoritmos adequados para a resolução de problemas específicos de mineração de dados [Valtchev et al. 2004]. Os principais conceitos associados à AFC são contexto formal e conceito formal, os quais estão descritos sucintamente a seguir:

2.1 Contexto Formal

Um contexto formal é uma estrutura utilizada para representar instâncias através de uma tabela de incidências. Essa tabela é composta por objetos (linhas), atributos (colunas) e suas respectivas incidências (relações binárias). Em [Ganter and Wille 2012] um contexto formal é definido pela notação (G, M, I), onde G é um conjunto de objetos, M é um conjunto de atributos e I é o conjunto de incidências, definidas como $I \subseteq G \times M$. Se um objeto $g \epsilon G$ e um atributo $m \epsilon M$ tem uma incidência I, essa representação $(g, m) \epsilon I$ ou gIm, pode ser lida como "um objeto g possui um atributo m".

2.2 Conceito Formal

De acordo com [Ganter et al. 2005] a partir do contexto formal (G, M, I) podem-se obter os conceitos formais, os quais são definidos como um par ordenado (A, B), em que $A \subseteq G$ é chamado de extensão e $B \subseteq M$ é chamado de intenção e ambos seguem as seguinte condições: $A \subseteq G, B \subseteq M, A' = B \quad e \quad B' = A$; $A' := \{m \in M | gIm \quad \forall g \in A\}$; $B' := \{g \in G | gIm \quad \forall m \in B\}$.

2.3 Implicações Próprias

A regra de implicação é um tipo específico de regra de associação, na qual é obtida a partir de contextos formais. Uma regra de implicação entre dois conjuntos Q e R recebe a notação $Q \rightarrow R$. É válida para um contexto, se o mesmo conjunto de objetos definido pelos atributos de Q também é descrito pelos atributos de R. Os conjuntos Q e R são denominados, respectivamente, premissa e conclusão.

Uma implicação própria é uma regra de implicação cuja premissa é mínima e a conclusão é composta por apenas um atributo. Uma premissa mínima é formada pela menor quantidade de atributos que levam a uma única conclusão. De uma maneira formal, implicações próprias são do tipo: $A \rightarrow b$ no qual $A \subseteq M$, $b \epsilon M$ e não existe implicação válida da forma $D \rightarrow b$ em que $D \subset A$ [Taouil and Bastide 2001]. Isso faz com que o conjunto de implicações gerado forneça mais novidades sobre os dados, além de ser mais legível para o usuário final. Em 2001, o algoritmo *Impec* foi proposto por Taouil e Bastide, cujo objetivo é encontrar o conjunto de implicações próprias a partir de contextos formais [Taouil and Bastide 2001]. Esse algoritmo é baseado em operações de fecho definidas sobre os conjuntos de conceitos formais e possibilita uma análise detalhada do conjunto de instâncias, a partir de cada uma das variáveis do contexto.

3. TRABALHOS RELACIONADOS

Diversos estudos tem aplicado a teoria de AFC para analisar comportamentos em redes sociais. Em [Neto et al. 2015], os autores apresentam uma abordagem para analisar uma base de dados, contendo *logs* de acesso à internet, utilizando o conjunto mínimo de regras de implicação e teorias de redes complexas para identificação de subestruturas que não são facilmente visualizadas na rede social. Em [Jota Resende et al. 2015], foi proposta uma abordagem baseada em AFC para construir modelos canônicos que representam os padrões de acesso ao Orkut e são compostos por um conjunto de regras mínimas. Esses se assemelham ao presente trabalho, por analisarem redes sociais através de regras de implicação obtidas pelos principais algoritmos da AFC.

Assim como o presente trabalho, em [Barysheva et al. 2015], os autores utilizaram o *LinkedIn* como fonte de dados e aplicaram AFC para análise da rede. Porém não é feito algum tipo de relação com o mercado de trabalho, pois o objetivo foi classificar os usuários a partir de suas interações na rede. Os autores não chegaram a mostrar regras de implicação, pois extraem informação a partir do reticulado conceitual.

Os artigos [Rodriguez et al. 2014] e [Xu et al. 2014] são alguns dos principais trabalhos sobre aplicação de mineração de dados, para identificar tendências de comportamento no mercado de trabalho. É importante destacá-los, por também terem o objetivo de modelar perfis profissionais e propor trajetórias para alcançar determinadas carreiras.

No geral, a teoria da AFC vem chamando atenção das comunidades de mineração de dados e redes sociais, por apresentar um formalismo para a representação, identificação de comportamento e extração de conhecimento, por meio da representação formal de um domínio de problema a partir de instâncias e atributos.

4. METODOLOGIA BASEADA EM AFC PARA ARS

Neste trabalho, o problema de análise e representação do perfil profissional em redes sociais, pode ser fundamentado por meio da construção de um modelo conceitual que sirva para relacionar a rede social com conceitos de competências profissionais e a transformação desse modelo em um contexto formal. Depois de coletar e processar os dados para um contexto formal, é possível extrair o conjunto de regras para serem analisadas. A Figura 1 ilustra as etapas da metodologia proposta para analisar a rede social *LinkedIn* por meio da AFC.

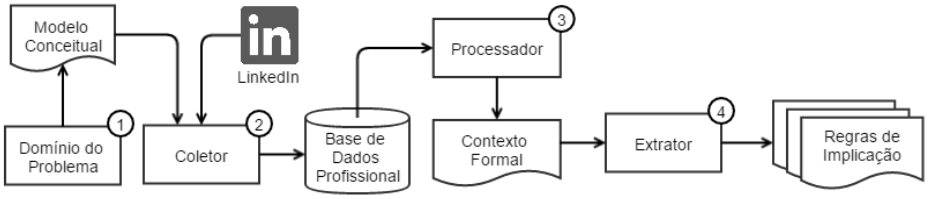


Fig. 1: Metodologia baseada em AFC para ARS

4.1 Domínio do problema

De acordo com a Figura 1, o primeiro passo (1) foi construir o modelo conceitual do domínio do problema a ser tratado. Com o apoio de especialistas, realizou-se um levantamento das variáveis que caracterizam uma pessoa enquanto profissional. Por ser mais utilizado no mercado de trabalho e devido à maior aceitação acadêmica, foi adotado o modelo de seleção por competências proposto por [Durand 1998] como base para modelar o problema. Nesse modelo, um profissional é caracterizado por três dimensões básicas: *Conhecimento*, *Habilidade* e *Atitudinal*. A dimensão *Conhecimento* está

ligada basicamente à formação acadêmica e cursos complementares. *Habilidade* é composta por dados sobre a experiência do profissional. A *Atitudinal* é composta por aspectos relacionados às interações dos usuários na rede social.

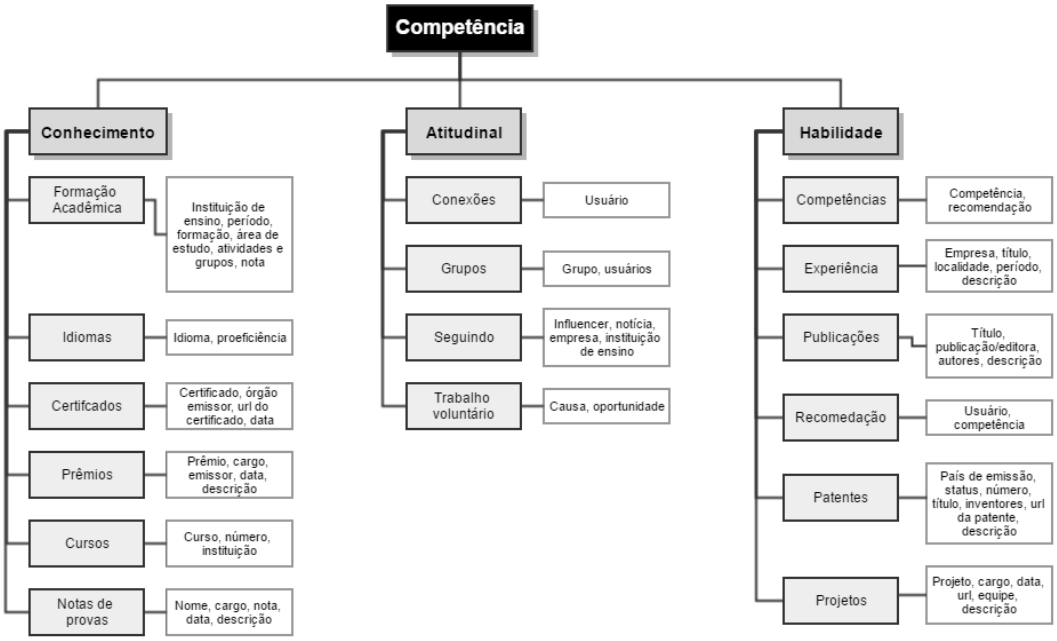


Fig. 2: Identificação profissional por competência

Na Figura 2 pode-se observar um modelo conceitual, já associado às informações disponíveis na rede *LinkedIn*, construído a partir da classificação das categorias informacionais, de acordo com a teoria de seleção por competências. É possível observar as dimensões, os aspectos e variáveis associadas a estes. No total foram identificadas 3 dimensões (competências profissionais), 16 aspectos e 61 variáveis vinculadas ao modelo. Devido a limitação de páginas, essas variáveis não foram descritas, mas o significado é intuitivo.

4.2 Coletor

A segunda etapa (2), da metodologia, representa o componente Coletor que é responsável por coletar dados do *LinkedIn*. O processo de coleta foi dividido em duas fases. Na primeira fase, foi utilizada a *People Search API*, disponibilizada pelo *LinkedIn*, para recuperar os perfis profissionais, considerando os campos predefinidos, como área de atuação e localização. Nesse caso, para o escopo do trabalho, foram selecionados dados de pessoas da área de computação da região metropolitana de Belo Horizonte, obtendo no total 2223 perfis.

Na segunda fase foi realizada uma coleta aberta, devido ao fato da API (Interface de Programação de Aplicativos) não disponibilizar o perfil completo do usuário com os dados necessários para o estudo. Essa coleta foi feita diretamente nas páginas públicas de cada usuário encontrado na primeira fase, para extrair os dados complementares.

4.3 Processador

O componente Processador (3) é o responsável pelo processamento dos dados extraídos na etapa anterior. Para este trabalho, foi considerado apenas a variável *competência* que compõe a dimensão

habilidade. Porém, em trabalhos futuros, serão consideradas as demais dimensões para analisar as relações entre elas. Foi identificado que os dados contidos no campo "Competências e Recomendações" do *LinkedIn*, compõem a variável selecionada. Para a construção do contexto formal, cada valor da variável *competência* foi considerado como um atributo, e cada usuário (profissional do *LinkedIn*) como um objeto. Na primeira versão do modelo, foram detectados 2000 atributos e 2206 objetos.

O algoritmo *Impec*, baseado em análise combinatória, é sensível à quantidade de atributos, objetos e à densidade do contexto formal. Este algoritmo não suporta a quantidade de atributos e objetos definidos na primeira versão do contexto formal, logo foi necessário adotar estratégias de redução do contexto formal. Como os atributos são nominais, a junção de variáveis altamente correlacionadas causaria uma considerável redução na quantidade de atributos. Nesse caso foram consideradas 17 competências globais, nas quais os 2000 atributos puderam ser reduzidos. A redução foi realizada em duas etapas: a primeira foi identificar sinônimos e padronizá-los; a segunda foi definir listas de competências específicas que poderiam compor cada competência global. Para isto, foi aplicado o algoritmo *Shift-End* Aproximado, para fazer o casamento de *strings* entre os atributos do contexto formal e os itens da lista de competências específicas. Assim os atributos com alta similaridade com itens da lista foram classificados de acordo com a competência global correspondente ao item. Ao final desse processo foi gerado o contexto formal *C1*, com 17 atributos (ver Tabela I para descrição dos atributos) e 2206 objetos.

Em AFC é possível executar um processo chamado clarificação de contextos formais, que pode ser realizado para objetos e atributos. A clarificação de objetos consiste em remover do contexto todos os objetos duplicados. Vale ressaltar que um objeto duplicado é àquele que é composto pelos mesmos atributos. A clarificação de atributos consiste em remover do contexto todos os atributos comuns à todos objetos, pois parte do princípio que esses atributos não agregam informação. Na segunda fase de redução do contexto formal, foi considerada a clarificação de objetos. Ao final desse processo foi gerado o contexto *C2*, com 146 objetos e 17 atributos.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q
Usuário 1	×				×	×	×		×	×		×	×				

Table I: Parte do contexto formal gerado a partir das habilidades globais dos usuários do *LinkedIn*: (a) Programação, (b) Gerência de projetos, (c) Suporte, (d) Segurança, (e) Business Intelligence, (f) Servidores, (g) Redes, (h) Aplicação móvel, (i) Engenharia de software, (j) Banco de dados, (k) Processamento gráfico, (l) Arquitetura de computadores, (m) Sistemas operacionais, (n) Aplicação de escritório, (o) Governança de TI, (p) Educação e pesquisa, (q) Gestão da informação

4.4 Extrator

A última fase da metodologia (4) corresponde à aplicação do algoritmo *Impec* para extrair as regras de implicações, a partir do contexto formal resultante da etapa anterior. Na tabela I, é mostrada uma parte do contexto formal construído a partir de dados dos usuários do *LinkedIn*.

O algoritmo *Impec*, proposto por [Taouil and Bastide 2001] e implementado por [Vimieiro and Vieira 2007], foi aplicado nos contextos *C1* e *C2* para extrair as implicações próprias.

5. EXPERIMENTOS E RESULTADOS

Nesta seção estão descritos os procedimentos adotados para execução dos experimentos e análise dos resultados obtidos através da metodologia proposta. Foram executados três experimentos com o objetivo de responder as seguintes questões: A clarificação de contextos interfere no conjunto final de implicações? Qual o impacto da escolha do suporte mínimo para selecionar implicações próprias relevantes? Como classificar as implicações próprias geradas em triviais e não-triviais?

5.1 Clarificação do Contexto

Como esperado, os contextos *C1* (não clarificado) e *C2* (clarificado) retornaram o mesmo conjunto de regras. No total foram obtidas 965 regras de implicação, das quais descartou-se 238 regras por terem suporte igual a 0 e foram consideradas as 727 regras com suporte maior que 0. Como o algoritmo *Impec* é baseado em combinações lógicas e não considera o suporte no momento de extração das implicações próprias, ele gera implicações sem suporte. Do ponto de vista lógico e de acordo com a ideia original do algoritmo, essas regras estão corretas e podem representar um comportamento que poderá ocorrer no futuro. Mas, de acordo com alguns estudos como [Webb 2006], regras com suporte nulo representam falsas descobertas e podem ser descartadas.

5.2 Definição do Suporte Mínimo

Diversas métricas são utilizadas na seleção de regras relevantes, como suporte e confiança. O suporte de um conjunto de implicações próprias é dado pela proporção de objetos do contexto formal que contêm um conjunto de atributos. Vale ressaltar que a confiança de uma regra de implicação é sempre igual a 100%, devido às implicações próprias serem derivadas de conceitos formais. Como mencionado na Seção 5.1, a quantidade de regras pode ser grande para serem analisadas, o que torna necessário determinar um limiar para selecionar a amostra relevante.

A Tabela II apresenta a distribuição da frequência absoluta do suporte das implicações. O suporte médio do conjunto de regras foi de 5,96% com um desvio padrão mínimo de 0,11%. O resultado mostra uma forte concentração de implicações na faixa 1% - 10%. Pode-se observar que ao utilizar essa métrica para selecionar implicações próprias relevantes, deve-se ter cautela ao definir o suporte mínimo para o corte das regras. Por exemplo, se o estudo considerasse um suporte mínimo de 70%, haveria um descarte de 99% das regras, tornando o procedimento pouco eficiente para a descoberta de informação relevante. Nesse caso, o ideal seria definir o suporte mínimo com base na média e no desvio padrão. Então ao considerar a média mais um desvio padrão (aproximadamente 15,87% da distribuição gaussiana, equivalente a 115 regras de implicação). Assim o espaço amostral foi reduzido para 152 implicações.

Intervalos de suporte (%)	0-1	1-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80-100
Nº de regras	176	437	68	14	11	8	9	3	0	1
Percentual	24,21%	60,11%	9,35%	1,93%	1,51%	1,10%	1,24%	0,41%	0,00%	0,14%

Table II: Distribuição de frequência da métrica suporte

5.3 Classificação de Regras em Triviais e Não-Triviais

Neste trabalho foi considerado que uma relação não-trivial ocorre quando, na visão de um especialista, as competências que compõem uma regra não são oriundas de áreas correlacionadas. Em contextos muito específicos, como o do presente trabalho, avaliar as medidas de interesse baseadas em estatística, não são suficientes para selecionar regras compostas por relações não-triviais, pois para determinar esse tipo de relação é necessário a interpretação das implicações, baseada em algum conhecimento prévio. Por exemplo, a regra *Aplicação móvel* → *Programação*, com suporte de 43,84%,é considerada trivial, pois ter conhecimento em *programação* é um requisito básico para desenvolver o conhecimento em *aplicação móvel*.

A criação de um oráculo, com o apoio de especialistas, é uma alternativa para a classificação das regras em triviais e não-triviais, podendo ocorrer um maior aproveitamento das regras em comparação com a seleção baseada em suporte. Essa classificação é baseada no conhecimento do especialista, que permite a análise das relações entre as habilidades e a identificação de informações relevantes. Como é inviável avaliar o conjunto completo de regras, foi necessário definir uma amostra de implicações

próprias para compor o oráculo. Neste trabalho a amostra foi composta pelas 152 implicações selecionadas através da definição do suporte. Tais implicações foram submetidas a um especialista que classificou as relações em triviais e não-triviais.

Na primeira versão do oráculo, foram identificadas algumas regras não-triviais como: Na regra {arquitetura de computadores, gestão de TI} → {business intelligence}, com suporte de 55,48%, nota-se que ao visualizar *business intelligence* como habilidade principal, as habilidades *arquitetura de computadores* e *gestão de TI* não foram consideradas como conhecimentos esperados para que uma pessoa ocupe uma função relacionada com inteligência empresarial. Outra relação interessante ocorre na regra {segurança, redes, processamento gráfico} → {sistemas operacionais}, com suporte de 7,53%, em que ter conhecimento em *processamento gráfico* não é necessariamente um pré-requisito básico para ter conhecimento em *sistemas operacionais*, o que esta regra pode representar é um conjunto de competências essenciais para uma função muito específica. Esses resultados demonstram que as implicações próprias não-triviais podem ser identificadas através do oráculo, porém ainda é necessário que mais profissionais participem da classificação para aprimorar a sua eficiência.

6. CONCLUSÃO E TRABALHOS FUTUROS

Neste artigo foi apresentado o uso da AFC para ARS, utilizando, como estudo de caso, a rede social *LinkedIn*. A AFC permite a exploração das relações entre os atributos selecionados para caracterizar o domínio do problema, permitindo que seja possível observar habilidades complementares e correlacionadas obtidas pelos profissionais. Quando é selecionada uma competência de interesse, as implicações próprias permitem visualizar facilmente a quantidade mínima de competências correlacionadas. Por exemplo, quando deseja-se um profissional para trabalhar com aplicações móveis, tem-se o mínimo de competências que estão relacionadas a essa função. Optou-se por um algoritmo clássico para extração de implicações próprias a partir do contexto. Como trabalhos futuros, pretende-se explorar outros algoritmos, em especial, aqueles capazes de obter o conjunto de implicações próprias a partir do reticulado conceitual ou do subconjunto de conceitos formais [Dias 2016]. Ademais, pretende-se identificar qual algoritmo apresenta melhor desempenho em função das características dos dados explorados neste trabalho.

Foi abordado apenas um aspecto definido no modelo conceitual da Seção 4.1. Porém, também é interessante relacionar aspectos diferentes. Por exemplo, ao adicionar cargos neste contexto formal, será possível extrair os requisitos mínimos para atingir uma determinada ascensão profissional.

O principal objetivo foi explorar a aplicação de implicações próprias para análise de redes sociais profissionais e como estudo de caso foram utilizados dados de usuários do *LinkedIn*. Para atingir o objetivo principal, na seção 5, foram abordados três objetivos específicos. O primeiro foi avaliar se a clarificação de contextos formais interfere na extração do conjunto de implicações próprias, permitindo concluir que utilizar contextos clarificados trazem ganho na eficiência computacional do algoritmo, sem ocorrer perdas no conjunto final de implicações próprias. O segundo objetivo foi avaliar o impacto da definição de um suporte mínimo para seleção de regras relevantes, observou-se que tal definição deve ser cautelosa, pois para o maior aproveitamento das regras, foi proposto que o suporte mínimo seja baseado na média e desvio padrão calculados a partir do suporte das regras. E, por último, foi apresentado uma estratégia para identificar implicações não-triviais, onde chegou-se a conclusão que, por ser baseado em conhecimento prévio sobre o assunto tratado, o oráculo pode ser utilizado para identificação de implicações próprias não-triviais. Vale ressaltar que, de um modo geral, a técnica encontrou padrões óbvios. Esse já era um resultado esperado, já que as regras são dependentes dos dados, sendo que neste caso foram geradas a partir de dados fatuais. Como demonstrado neste trabalho, padrões gerados a partir de bases fatuais não trazem muitas relações não-triviais. Assim, para extrair padrões não-triviais é necessário enriquecer a base com dados oriundos de diversas fontes.

Uma desvantagem em utilizar a AFC é o limite de variáveis que podem ser incluídas no contexto

formal. Devido à ordem de complexidade $O(|M||F|(|G||M| + |F||M|))$, que está diretamente relacionada à quantidade de atributos do contexto, os algoritmos atuais são pouco eficientes para processar e extrair o conjunto completo de regras. Uma das formas de reduzir a quantidade de variáveis é agrupá-las em categorias e utilizar tais categorias como atributos no contexto, porém essa solução pode causar perda de informação relevante para o estudo. Assim, em trabalhos futuros serão abordadas outras técnicas para redução de contextos formais e será implementado um novo algoritmo para gerar implicações próprias a partir do contexto formal. Também deseja-se aplicar implicações próprias para identificar trajetórias profissionais bem como fazer a comparação com resultados de pessoas de diferentes regiões do Brasil. Esses trabalhos já foram iniciados.

7. AGRADECIMENTOS

Os autores agradecem o suporte financeiro recebido da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ).

REFERENCES

Barysheva, A., Golubtsova, A., and Yavorskiy, R. Profiling less active users in online communities, 2015.

Cuvelier, E. and Aufaure, M.-A. A buzz and e-reputation monitoring tool for twitter based on galois lattices. In *Conceptual Structures for Discovering Knowledge*. Springer, Berlin Heidelberg, pp. 91–103, 2011.

Dias, S. M. *Redução de Reticulados Conceituais (Concept Lattice Reduction)*. Ph.D. thesis, Department of Computer Science of Federal University of Minas Gerais (UFMG), Belo Horizonte, Minas Gerais, Brazil, 2016. In Portuguese.

Dufour-Lussier, V., Lieber, J., Nauer, E., and Toussaint, Y. Text adaptation using formal concept analysis. In *Case-Based Reasoning. Research and Development*. Springer, pp. 96–110, 2010.

Durand, T. Forms of incompetence. In *Proceedings Fourth International Conference on Competence-Based Management. Oslo: Norwegian School of Management*, 1998.

Ganter, B., Stumme, G., and Wille, R. *Formal concept analysis: foundations and applications*. Vol. 3626. springer, 2005.

Ganter, B. and Wille, R. *Formal concept analysis: mathematical foundations*. Springer Science & Business Media, 2012.

Jota Resende, G., De Moraes, N. R., Dias, S. M., Marques Neto, H. T., and Zarate, L. E. Canonical computational models based on formal concept analysis for social network analysis and representation. In *Web Services (ICWS), 2015 IEEE International Conference on*. IEEE, pp. 717–720, 2015.

LinkedIn. Sobre nós - linkedin, 2016. Acessado em 30/03/2016.

Missaoui, R., Université, E., Lyon, L., Université, L., and Pascal, B. Social network analysis using formal concept analysis, 2013.

Neto, S. M., Song, M., Dias, S., et al. Minimal cover of implication rules to represent two mode networks. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. Vol. 1. IEEE, pp. 211–218, 2015.

Rodriguez, M., Helbing, D., Zagheni, E., et al. Migration of professionals to the us. In *Social Informatics*. Springer, pp. 531–543, 2014.

Russell, M. A. *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*. "O'Reilly Media, Inc.", 2013.

Taouil, R. and Bastide, Y. Computing proper implications. In *9th International Conference on Conceptual Structures: Broadening the Base-ICCS'2001*. pp. 13–p, 2001.

Valtchev, P., Missaoui, R., and Godin, R. Formal concept analysis for knowledge discovery and data mining: The new challenges. In *Concept lattices*. Springer Berlin Heidelberg, pp. 352–371, 2004.

Vimieiro, R. and Vieira, N. J. Uma análise de algoritmos para extração de regras de associação usando análise formal de conceitos. In *III Workshop em Algoritmos e Aplicações de Mineração de Dados da Universidade de Federal de Minas Gerais*, 2007.

Webb, G. I. Discovering significant rules. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 434–443, 2006.

Xu, Y., Li, Z., Gupta, A., Bugdayci, A., and Bhasin, A. Modeling professional similarity by mining professional career trajectories. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 1945–1954, 2014.

Fitted Q-Iteration Fatorado no controle de Redes de Regulação Gênica

Cyntia E. H. Nishida, Anna H. R. Costa

Escola Politécnica, Universidade de São Paulo, Brasil
{cyntiaeico, anna.reali}@usp.br

Abstract. A modelagem de uma rede de regulação gênica permite simular estratégias de controle para alterar o comportamento do sistema biológico no longo prazo. Quando o modelo da dinâmica do comportamento do sistema não está completamente disponível, pode-se derivar uma estratégia de controle diretamente de amostras temporais da expressão gênica. Os algoritmos de controle mapeiam a expressão gênica de uma amostra, dada pelos valores de expressão dos genes, em números decimais que representam um estado. Porém, isso prejudica a noção de similaridade entre estados, uma vez que a distância numérica será maior que a distância de Hamming entre dois estados. Esse ponto é especialmente importante quando existem poucas amostras disponíveis, pois métodos de aproximação podem associar valores a estados não visitados de acordo com valores dos estados vizinhos. Em vista disso, esse trabalho propõe aplicar diretamente o valor da expressão dos genes como representação de estado multivariável no algoritmo de aprendizado por reforço em lote Fitted Q-Iteration. Essa abordagem facilita a noção de vizinhança entre estados e permite identificar melhor o relacionamento entre os genes.

Categories and Subject Descriptors: G.3 [**PROBABILITY AND STATISTICS**]: Markov processes; J.3 [**LIFE AND MEDICAL SCIENCES**]: Biology and genetics

Keywords: Aprendizado por Reforço em Lote, Processo de Decisão de Markov, Redes de Regulação Gênica

1. INTRODUÇÃO

Rede de Regulação Gênica (GRN – *Gene Regulatory Network*) é um modelo que descreve as interações entre os genes no genoma, possibilitando compreender o comportamento do sistema biológico. Esse comportamento é capturado principalmente por um conjunto de estados que formam um ciclo que será percorrido indefinidamente [de Jong 2002]. No entanto, nem todo comportamento é desejável e, por isso, alguns ciclos podem estar associados a doenças. Por exemplo, no caso do melanoma, o gene WNT5A é um forte indicador da agressividade do tumor [Bittner et al. 2000], pois quando esse gene passa a produzir uma grande quantidade de proteínas, a capacidade do tumor entrar em metástase aumenta [Weeraratna et al. 2002]. Nesse caso o gene está em um ciclo que requer constantemente a sua ativação e, dessa forma, é importante encontrar um procedimento para manter o gene WNT5A desligado ou não expresso. Isso pode ser feito por meio de simulações na GRN, onde é possível verificar as consequências da aplicação de certas ações [Shmulevich and Dougherty 2010]. Tais simulações permitem encontrar quais manipulações devem ser executadas em um determinado momento para controlar a GRN [Shmulevich et al. 2002]. Com isso, essas respostas podem auxiliar a criação de novos remédios que consigam reproduzir essas manipulações e controlar o sistema biológico.

Uma das modelagens mais utilizadas para controlar a GRN é a rede booleana probabilística (PBN – *Probabilistic Boolean Networks*) [Shmulevich et al. 2002]. Além disso, o seu controle pode ser modelado como um Processo de Decisão de Markov (MDP – *Markov Decision Process*) [Puterman 2005], que é uma forma de modelar processos estocásticos para interferir periodicamente por meio

Agradecemos à Capes e ao CNPq (Processo nº: 311608/2014-0) pelo auxílio.

da execução de ações. Cada ação tem uma recompensa, que depende do estado atual e do estado para o qual o processo transitou devido à ação tomada. Entretanto a inferência da PBN possui um elevado custo computacional, evidenciado pela complexidade exponencial da sua construção e da quantidade de estados, dados pelo número de genes [Faryabi et al. 2007]. Logo, métodos de controle que consigam gerar políticas efetivas sem o conhecimento prévio do modelo são alternativas aplicáveis a grandes redes. Uma possibilidade é aplicar técnicas de aprendizado por reforço (RL – *Reinforcement Learning*), nas quais um sistema autônomo aprende o que fazer em cada situação de um ambiente desconhecido de forma a maximizar a recompensa esperada [Sutton and Barto 1998]. Dentre os métodos de RL, o RL em lote (BRL – *Batch Reinforcement Learning*) é especialmente vantajoso para trabalhar com GRN, pois nesse caso o agente pode utilizar um conjunto de observações previamente realizadas ou planejadas para atuar, ao invés de atuar e tomar decisões apenas em tempo real (de forma *online*). A primeira técnica de BRL aplicada no controle de GRN foi proposta em [Sirin et al. 2013]. Os autores propuseram empregar o algoritmo de BRL Fitted Q-Iteration usando a expressão gênica como característica da função de aproximação paramétrica, cujos parâmetros obtêm-se aplicando regressão por mínimos quadrados. No entanto, a função de aproximação não obteve bons resultados para todos os casos apresentados [Sirin 2013].

Em geral, utiliza-se a representação do estado como valor da expressão gênica em sua forma decimal. Essa abordagem facilita a manipulação dos dados, contudo isso prejudica a noção de similaridade entre estados, que é um ponto importante para o desempenho das funções de aproximação. O conceito de ter um estado composto por diversos fatores é uma das propostas do MDP Fatorado [Boutilier et al. 2000], que decompõe os estados para obter uma representação mais compacta. O método proposto nesse trabalho utiliza a expressão gênica como estado fatorado no algoritmo BRL denominado Fitted Q-Iteration (FQI) [Ernst et al. 2005]. Essa representação do estado aumenta a noção de similaridade e vizinhança entre estados; além disso, possibilita, de forma mais clara, explorar o relacionamento entre genes. Os resultados dos experimentos realizados apontaram que a abordagem fatorada consegue controlar de forma mais efetiva que o FQI tradicional e o método de [Sirin et al. 2013]. Também foi constatado o aumento da qualidade da política de controle conforme cresce o conjunto de treinamento usado no BRL.

O restante deste artigo está estruturado da seguinte forma: Trabalhos relacionados são descritos na seção 2. Na seção 3 é feita uma revisão sobre GRN. Já na seção 4, os conceitos de BRL são explicados. Em seguida, na seção 5, o método FQI Fatorado é proposto para controle de GRN. A seção 6 apresenta os experimentos realizados. E, por fim, a seção 7 descreve as conclusões deste trabalho.

2. TRABALHOS RELACIONADOS

A Média dos Tempos de Primeira Passagem (MFPT – *Mean First Passage Time*) foi a primeira técnica independente de modelo para controlar uma PBN. Nela, calcula-se a média dos tempos que leva para transitar entre estados desejados e indesejados e, depois, em escolher a ação que faça estados indesejados chegarem mais rápido em estados desejados. Por ser uma técnica gulosa, pode escolher ações com um menor valor a longo prazo. Já [Faryabi et al. 2007] propuseram aplicar uma técnica de aprendizado por reforço denominada Q-Learning com um simulador para emular as interações com o sistema biológico. No entanto, ele também pode ser utilizado diretamente nos dados de amostras temporais se eles contiverem a informação de qual ação foi tomada em cada instante.

O trabalho mais similar à nossa proposta talvez seja o proposto por [Sirin et al. 2013], no qual foi utilizado o algoritmo FQI com aproximação por mínimos quadrados [Busoniu et al. 2010] (LSFQI – *Least Squares Fitted Q-Iteration*) para gerar uma política a partir dos dados temporais. Cada gene da expressão gênica define uma característica do algoritmo de aproximação paramétrica, o qual obtém os parâmetros que minimizem o erro quadrático entre o valor alvo e o resultado da aproximação. O trabalho não utilizou representação fatorada e apresentou um teste com apenas um conjunto de dados; desta forma, não foi possível verificar se a técnica obtém bons resultados para qualquer GRN.

3. CONTROLE DE REDES DE REGULAÇÃO GÊNICA

Os genes no genoma formam uma rede de interações conhecida como GRN, onde cada gene pode ser ativado ou inibido por outros genes [El Samad et al. 2005]. Em outras palavras, o valor de expressão de cada gene é determinado por uma função denominada preditora, cuja saída depende do valor atual de todos os genes da rede. Um dos modelos mais utilizados para o controle de GRN é a PBN, o qual é um modelo discreto que descreve o relacionamento estocástico entre genes. Nesse caso, além das funções preditoras, cada gene terá uma probabilidade p de sofrer uma perturbação e ter o seu valor alterado. Essa perturbação pode ser entendida como uma influência externa, que modifica o seu comportamento. Por ser um modelo discreto, é necessário discretizar as amostras de expressão gênicas, que medem o quanto cada gene está ativo [Shmulevich and Dougherty 2010]. Este trabalho foca em PBNs binárias como em [Pal et al. 2006], mas seus resultados podem ser estendidos para qualquer intervalo de discretização, visto que o controle continuará sendo modelado como um MDP. Ademais, mesmo para organismos mais complexos, a binarização demonstra resultados consistentes, assim como requer menos dados para a inferência da modelagem [Shmulevich and Dougherty 2010].

A partir das amostras de expressão gênica, o algoritmo de inferência procura encontrar uma PBN que consiga reproduzir o que foi observado. Formalmente, uma PBN binária consiste de um conjunto $V = \{x_i\}_{i=1}^{n}, x_i \in \{0,1\}$ de n nós ou genes, e um conjunto de funções preditoras $\{f_i\}_{i=1}^{n}, f_i : \{0,1\}^n \rightarrow \{0,1\}$ que calcula o valor do gene i. O parâmetro p indica a probabilidade de um gene sofrer uma perturbação e ter o seu valor alterado. Cada expressão gênica denota um estado da PBN; desta forma, como existem n genes e cada um possui dois possíveis valores, existem 2^n estados possíveis.

Como o estado é a expressão gênica, ele é composto por um vetor com n elementos que varia de 00..0 até 11...1. Esse vetor pode ser mapeado em seu equivalente decimal, tomando valores do intervalo $[1, 2^n]$ [Pal et al. 2006]. O estado como número decimal simplifica a representação, assim como facilita a manipulação dos dados.

O controle de PBN por controle externo é baseado em aplicar uma ação para alterar, $a = 1$, ou manter, $a = 0$, o valor de um determinado gene [Shmulevich and Dougherty 2010]. Esse gene, cujo valor pode ser alterado, é denominado gene de controle, pois é utilizado para controlar a GRN. Como o seu novo valor é utilizado como entrada das funções preditoras, ele consegue influenciar a transição de estados. A melhor política informará quando o gene de controle deve ser alterado para aumentar a probabilidade de afastar a rede de ciclos indesejados [Sutton and Barto 1998]. Conforme foi dito anteriormente, o controle de GRN pode ser modelado como um MDP. Formalmente, um MDP é descrito pela quádrupla $\langle S, A, T, R\rangle$ [Puterman 2005], no qual S é o conjunto de possíveis estados; A é o conjunto de ações disponíveis para executar; T é a função de transição $T : (S, A, S) \rightarrow [0, 1]$ e R é a função de recompensa $R : (S, A, S) \rightarrow \mathbb{R}$. A função T especificada por $T\left(s^t, a^t, s^{t+1}\right)$ indica a probabilidade de transitar do estado atual s^t para o estado s^{t+1}, após aplicar a ação a no tempo t. O problema central do MDP é encontrar uma política de atuação $\pi : S \rightarrow A$, ou seja, especificar qual ação $\pi(s)$ deverá ser escolhida quando o processo estiver no estado s a fim de maximizar a recompensa esperada.

No caso do controle de GRN, em vez de uma função de recompensa é utilizada uma função de custo a ser minimizada. Assim, o custo esperado de seguir a política π:

$$E[C(\pi)] = \sum_{s}\sum_{s\prime} p(s)p(s\prime|s, \pi(s))custo(s, \pi(s), s\prime), \tag{1}$$

pode ser utilizado como critério de comparação entre soluções, com $p(.)$ indicando probabilidade. Outra métrica de qualidade é a taxa de transferência, que indica o quanto estados indesejados deixaram de ser visitados a longo prazo:

$$\Delta T = \frac{\sum_{s=estado\ indesejado} \omega(s) - \sum_{s=estado\ indesejado} \omega^c(s)}{\sum_{s=estado\ indesejado} \omega(s)}, \tag{2}$$

onde ω é a distribuição limite e ω^c é a distribuição limite após o controle. Apesar de ΔT medir diretamente a qualidade da solução em relação ao objetivo, ela precisa de informações completas do modelo MDP.

4. APRENDIZADO POR REFORÇO EM LOTE

A aprendizagem por reforço (RL) é definida como um paradigma de aprendizado tradicionalmente modelado como um MDP [Sutton and Barto 1998]. No RL, um sistema aprende, por meio de interações com o ambiente, quais ações o levarão a alcançar o objetivo desejado. Uma forma de resolver o MDP por meio de RL é com o algoritmo Q-Learning [Watkins and Dayan 1992]. Este algoritmo avalia a qualidade de cada par estado-ação definida na função Q, de acordo com o valor esperado de recompensas a longo prazo. Conforme o sistema atua no ambiente, o Q-Learning atualiza o valor da função $Q_{t+1}(s^t, a^t) \leftarrow r^t + \gamma \max_a Q_t(s^{t+1}, a)$, de forma incremental com as recompensas recebidas a cada passo. γ é um fator de desconto ($0 < \gamma < 1$) que codifica o horizonte em que recompensas são relevantes e também assegura que a soma das recompensas recebidas é finita.

Contudo, no controle de GRN não existe interação com o ambiente, e sim um conjunto de amostras (lote) dessas interações. Para esses casos, uma ramificação do RL conhecida como aprendizado por reforço em lote (BRL) é mais indicada, pois visa encontrar a melhor política possível a partir de um conjunto de experiências $\mathscr{F} = \left\{(s^t, a^t, r^t, s^{t+1})\right\}$ fornecido a priori [Lange et al. 2011]. Um dos métodos de BRL mais utilizados é o Fitted Q-Iteration (FQI) [Ernst et al. 2005], que pode ser considerado a versão em lote do Q-Learning. O algoritmo FQI, aproxima $Q(s, a)$ iterativamente utilizando técnicas de regressão supervisionada em um conjunto de treinamento $\mathcal{TS} = \{(s^t, a^t, \bar{q}(s^t, a^t))\}$, onde $\bar{q}(s^t, a^t)$ representa a atualização de $Q(s^t, a^t)$ da experiência (s^t, a^t, r^t, s^{t+1}).

5. FITTED Q-ITERATION FATORADO

Como foi dito anteriormente, cada estado de uma PBN representa uma expressão gênica e em geral considera-se o seu equivalente em decimal. Entretanto, a representação decimal enfraquece a percepção de similaridade entre expressões. Por exemplo, a distância entre 2 e 34 não indica que representam estados similares, e possuem apenas um gene de diferença em suas expressões binárias equivalentes (000010 e 100010). Em outras palavras, a sua distância numérica (decimal) sempre será maior ou igual à distância de Hamming entre dois estados. Além disso, a noção de vizinhança, aspecto importante para funções de aproximação, é prejudicada. Levando isso em consideração, utilizou-se o conceito de MDP Fatorado [Boutilier et al. 2000], o qual decompõe os estados em um conjunto de fatores para representar o modelo de forma mais compacta. Logo, ao utilizar fatores no algoritmo de aproximação, é possível analisar as distâncias de forma mais eficaz. No MDP Fatorado, o conjunto de estados é descrito como uma variável aleatória fatorada $\mathbf{s} = \{s_1, .., s_n\}$, onde cada s_i toma valores do seu domínio $Dom(s_i)$. Assim, nas GRN, cada estado é formado por n genes $\mathbf{s} = \{x_1, .., x_n\}$, onde $Dom(x_i) \in \{0, 1\}$.

O método proposto neste trabalho, mostrado no algoritmo 1, considera o conjunto de experiências $\mathscr{F} = \left\{(x_1^t, .., x_n^t, a^t, r^t, x_1^{t+1}, .., x_n^{t+1})\right\}$ com o estado fatorado. Por isso, o algoritmo de aproximação da função Q consegue explorar a noção de vizinhança e possivelmente encontrar relações entre os genes. Na nossa proposta, usamos regressão por árvore [Breiman et al. 1984] no passo 11 do Algoritmo 1. Entre suas vantagens está a sua flexibilidade em modelar funções com forma não conhecida a priori, seleção automática de atributos e fácil interpretação. Além disso, a sua altura será menor na abordagem fatorada, conforme evidenciado na figura 1. Afinal, ela está limitada pela quantidade de genes e isso fará com que a predição seja mais rápida que na abordagem tradicional.

6. EXPERIMENTOS

Para avaliar a eficiência e eficácia do método proposto, foram feitos dois experimentos com diversas PBNs binárias geradas artificialmente pelo pacote Matlab disponível em [Yousefi and Dougherty 2013].

Algorithm 1 Algoritmo Fitted Q-Iteration Fatorado

Carrega $F = \left\{\left\langle x_1^t, ..., x_n^t, a^t, r^t, x_1^{t+1}, ..., x_n^{t+1}\right\rangle\right\}, t = 1, ..., g$;
Define $\bar{Q}^0(x_1, ..., x_n, a) = 0$ para todo $(x_1, ..., x_n, a) \in F$, com $(x_1, ..., x_n) \in S, a \in A$.
Define H como o horizonte.
h = 1.
enquanto $h \leq H$ **faça**
  Define $\mathcal{TS}^h = \emptyset$.
  para todo $\left\langle x_1^t, ..., x_n^t, a^t, r^t, x_1^{t+1}, ..., x_n^{t+1}\right\rangle \in F$ **faça**.
    $\bar{q}^h(x_1^t, ..., x_n^t, a) = r^t + \gamma max_a \bar{Q}^{h-1}(x_1^{t+1}, ..., x_n^{t+1}, a)$.
    $\mathcal{TS}^h \leftarrow \mathcal{TS}^h \cup \left\langle (x_1^t, ..., x_n, a^t, \bar{q}^h(x_1^t, ..., x_n, a))\right\rangle$.
  fim para
  Utilize um algoritmo de aprendizado supervisionado em $\mathcal{TS}$ para treinar um aproximador $\bar{Q}^h(x_1, ..., x_n, a)$
  $h \leftarrow h + 1$.
fim enquanto
para todo **s** $\in S$ **faça**
  $\pi(x_1, ..., x_n) = \arg\max_a \bar{Q}^H(x_1, ..., x_n, a)$.
fim para

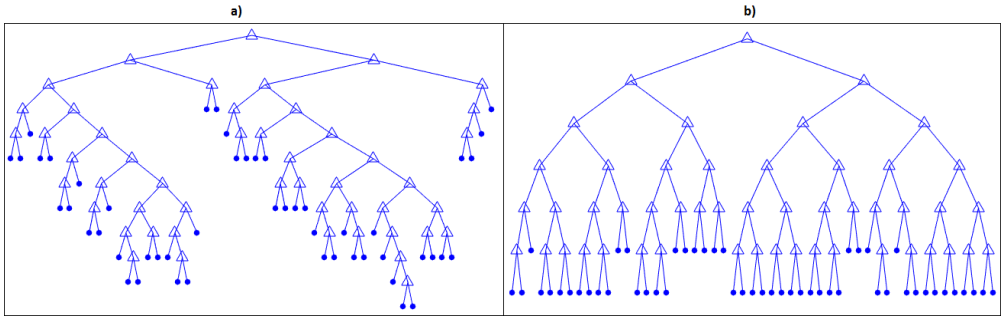


Fig. 1. Topologia da árvore de regressão gerada pelos algoritmos: a) FQI, b) FQI Fatorado

Para gerar o modelo da PBN, o algoritmo cria uma tabela verdade para cada gene i, que indica a saída f_i para cada combinação de entrada. Para todos os testes os parâmetros de criação das PBNs foram: $n = 8$ e $p = 0,01$, valores utilizados em [Yousefi and Dougherty 2013] e [Sirin et al. 2013]. Tendo o modelo gerado no simulador, pode-se avaliar a qualidade de cada algoritmo RL ao compará-los com o valor ótimo dado pelo modelo.

O primeiro experimento verificou o impacto na qualidade dos resultados dos algoritmos FQI Fatorado, FQI e LSFQI de acordo com o tamanho do conjunto de experiências. Já o segundo experimento avaliou seus resultados para diferentes PBNs. Em ambos os experimentos, calculou-se as métricas de qualidade e comparou-se com a métrica ótima (dada pelo simulador). As tuplas de experiências são transições de estados obtidas pelo modelo, no qual dado um estado s^t e ação a^t aleatoriamente escolhida, simula-se um passo no tempo para obter o novo estado s^{t+1}. Além disso, os algoritmos independentes de modelo processaram os mesmos conjuntos de experiências, evitando assim, diferenças nos resultados devido a diferentes entradas. No caso dos algoritmos FQI e FQI Fatorado, utilizou-se regressão por árvores e os parâmetros dos algoritmos: $\gamma = 0,9$ e $H = 100$.

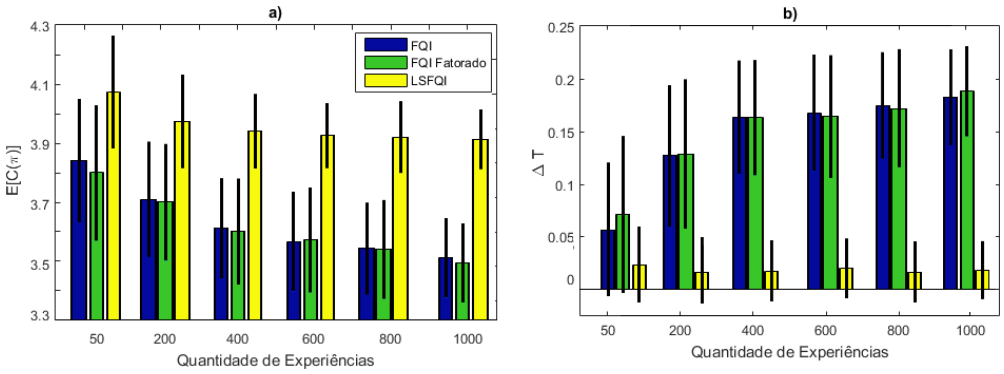


Fig. 2. Tendência com o aumento do conjunto de treinamento: a) ΔT (Eq. 2), b) $E\left[C(\pi)\right]$ (Eq. 1).

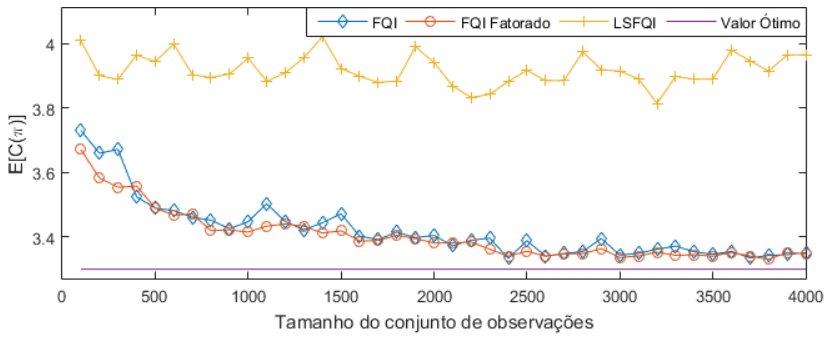


Fig. 3. Tendência de $E\left[C(\pi)\right]$ para conjuntos de treinamento de 100 a 4000 amostras.

6.1 Comparação por quantidade de tuplas de experiências

A fim de comparar a qualidade dos resultados dos métodos BRL em relação à quantidade de amostras disponíveis, foram realizados dois experimentos. O primeiro com 200 testes para cada algoritmo considerando conjuntos com tamanho de 50 a 1000 tuplas de experiências para medir o desvio padrão de $E\left[C(\pi)\right]$. O segundo com cinco testes para conjuntos com tamanho de 100 a 4000 experiências. A função de custo usada está presente nos trabalhos de [Pal et al. 2006] e [Sirin et al. 2013]. Ela é caracterizada por penalizar tanto a visita a estados indesejados como a aplicação de controle:

$$custo(\cdot, a, s^{t+1}) = \begin{cases} 0 & \text{se } s^{t+1} \text{ é um estado desejado e a=0} \\ 1 & \text{se } s^{t+1} \text{ é um estado desejado e a=1} \\ 5 & \text{se } s^{t+1} \text{ é um estado indesejado e a=0} \\ 6 & \text{se } s^{t+1} \text{ é um estado indesejado e a=1.} \end{cases}$$

O resultado do primeiro experimento, apresentado na figura 2, mostra as médias e os desvios padrões dos custos dos 200 testes realizados. Pode-se notar que a média do custo melhora conforme o conjunto aumenta, ou seja, o controle é mais eficaz quando mais experiências estão disponíveis para treinamento. O fato do desvio padrão diminuir indica resultados mais estáveis aos diferentes conjuntos de treinamento, pois tornam-se cada vez mais representativos em relação à distribuição real. O resultado do segundo experimento, apresentado na figura 3, mostra claramente que $E\left[C(\pi)\right]$ fica mais próximo do ótimo quando o conjunto de treinamento é maior.

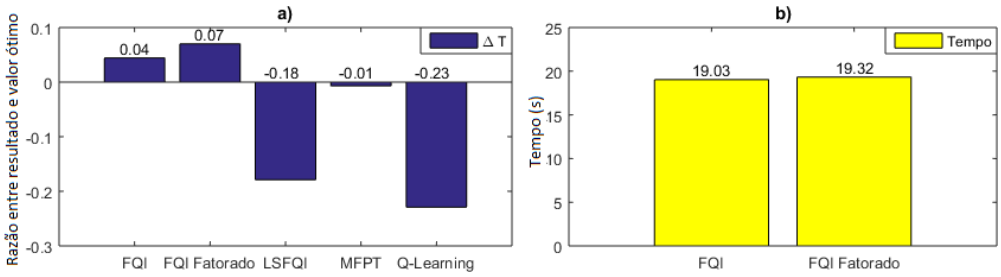


Fig. 4. a) Resultados dos algoritmos: a) ΔT (Eq. 2) b) Tempo de Processamento.

6.2 Comparação entre algoritmos

Neste experimento, os algoritmos de BRL foram comparados com os métodos Q-Learning e MFPT. Como o MFPT visa a maximização de ΔT, sem penalizar a quantidade de controles, neste experimento a função de custo foi alterada:

$$custo(\cdot, a, s^{t+1}) = \begin{cases} 0 & \text{se } s^{t+1} \text{ é um estado desejado,} \\ 5 & \text{se } s^{t+1} \text{ é um estado indesejado.} \end{cases}$$

Assim, todos os algoritmos consideram apenas a penalização de estados indesejados. Foram geradas 5000 PBNs artificiais para verificar os resultados com conjuntos com 50 observações. Desta vez, pretende-se investigar se o método proposto obtém bons resultados para qualquer PBN. A figura 4a) mostra a média das razões entre o resultado obtido e o ótimo. Já a figura 4b) mostra a média dos tempos de processamento do FQI e FQI Fatorado.

Para verificar se o FQI Fatorado produz resultados melhores que os outros métodos, aplicou-se o teste estatístico teste t unicaudal. O teste t foi utilizado para avaliar se o FQI Fatorado tende a produzir resultados mais próximo do ótimo que os outros métodos. Assim, a hipótese nula é H_0 : *razão do* ΔT *entre FQI Fatorado e o ótimo* $\leq$ *razão do* ΔT *entre o método comparado e o ótimo.* Para os quatro métodos comparados a hipótese nula é rejeitada com 5% de significância.

7. CONCLUSÕES

O método proposto aplica a expressão gênica diretamente como representação de estado fatorado no algoritmo FQI. Essa abordagem aumenta a eficiência da regressão responsável por ajustar os valores na função de aproximação da função de custo. Por ser independente de modelo, o FQI pode ser aplicado diretamente nos dados temporais caso a informação dos controles (ações executadas) esteja disponível. Além disso, por usar função de aproximação em vez de armazenar o par estado-ação, é um método escalável que pode controlar GRNs maiores e mais complexas.

Os testes comprovaram que os algoritmos de BRL chegam próximo do custo ótimo quando o conjunto de treinamento é grande. Entre eles, o FQI Fatorado obteve os melhores resultados, demonstrando que a noção de similaridade entre expressões gênicas é um ponto importante. Vale notar que, apesar de considerar um estado fatorado, o tempo de processamento não teve um aumento expressivo, permitindo utilizar o FQI fatorado em vez do FQI tradicional. Além disso, é importante salientar que no segundo experimento os métodos LSFQI, MFPT e Q-Learning tiveram uma média menor que 0 para ΔT, ou seja, em geral o resultado é igual a não aplicar controle ou pior, aumentando a probabilidade de visitar estados indesejados. Apesar do experimento ter sido realizado com poucas amostras de experiências, esse é um resultado importante, pois é um cenário mais realístico que possuir muitas amostras disponíveis para execução [Vahedi et al. 2008].

A regressão por árvores mostrou resultados satisfatórios, principalmente por conseguir capturar o relacionamento entre genes. Esse fator é especialmente importante quando poucas amostras estão

disponíveis, pois é atribuído um custo mais assertivo a estados ainda não visitados. Outra vantagem é o fato do atributo mais discriminativo ficar na raiz da árvore, o que pode indicar o gene com a maior influência no controle da GRN. Essa informação pode auxiliar na descoberta de qual é o melhor gene de controle, quando essa informação não estiver disponível.

Como próximos passos, faremos uma análise se o gene na raiz da árvore de regressão pode ser considerado um gene de controle com forte influência na GRN. Também pretendemos verificar se outros algoritmos de regressão como Support Vector Machines [Vapnik 1995] e redes neurais artificiais conseguem resultados similares.

REFERENCES

Bittner, M., Meltzer, P., Khan, J., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., Radmacher, M., Simon, R., Yakhini, Z. Y. a. B., Dougherty, E., Wang, E., Marincola, F., Gooden, C., Lueders, J., Glatfelter, A., Pollock, P., Gillanders, E., Leja, A., Dietrich, K., Beaudry, C., Berrens, M., Alberts, D., Sondak, V., Hayward, N., and Trent, J. M. Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature* vol. 406, pp. 536–540, 2000.

Boutilier, C., Dearden, R., and Goldszmidt, M. Stochastic dynamic programming with factored representations. *Artificial Intelligence* 121 (1): 49–107, 2000.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. *Classification and regression trees*. CRC press, 1984.

Busoniu, L., Babuska, R., De Schutter, B., and Ernst, D. *Reinforcement learning and dynamic programming using function approximators*. Vol. 39. CRC press, 2010.

de Jong, H. Modeling and simulation of genetic regulatory systems: a literature review. *Journal of computational biology : a journal of computational molecular cell biology* 9 (1): 67–103, 2002.

El Samad, H., Khammash, M., Petzold, L., and Gillespie, D. Stochastic modelling of gene regulatory networks. *International Journal of Robust and Nonlinear Control* 15 (15): 691–711, 2005.

Ernst, D., Geurts, P., and Wehenkel, L. Tree-based batch mode reinforcement learning. In *Journal of Machine Learning Research*. pp. 503–556, 2005.

Faryabi, B., Datta, A., and Dougherty, E. On approximate stochastic control in genetic regulatory networks. *Systems Biology, IET* 1 (6): 361–368, November, 2007.

Lange, S., Gabel, T., and Riedmiller, M. Batch reinforcement learning. In *Reinforcement Learning*, M. Wiering and M. van Otterlo (Eds.). Vol. 12. Springer Berlin Heidelberg, pp. 45–73, 2011.

Pal, R., Datta, A., and Dougherty, E. Optimal infinite horizon control for probabilistic boolean networks. In *American Control Conference, 2006*. pp. 6 pp.–, 2006.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 2005.

Shmulevich, I. and Dougherty, E. R. *Probabilistic Boolean networks: the modeling and control of gene regulatory networks*. siam, 2010.

Shmulevich, I., Dougherty, E. R., Kim, S., and Zhang, W. Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics* 18 (2): 261–274, 2002.

Shmulevich, I., Dougherty, E. R., and Zhang, W. Gene perturbation and intervention in probabilistic boolean networks. *Bioinformatics* vol. 18, pp. 1319–1331, 2002.

Sirin, U. *Batch Mode Reinforcement Learning for Controlling Gene Regulatory Networks and Multi-model Gene Expression Data Enrichment Framework*. Ph.D. thesis, Middle East Technical University, 2013.

Sirin, U., Polat, F., and Alhajj, R. Employing batch reinforcement learning to control gene regulation without explicitly constructing gene regulatory networks. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. IJCAI '13. AAAI Press, Beijing, China, pp. 2042–2048, 2013.

Sutton, R. S. and Barto, A. G. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1998.

Vahedi, G., Faryabi, B., Chamberland, J.-F., Datta, A., and Dougherty, E. R. Mean first-passage time control policy versus reinforcement-learning control policy in gene regulatory networks. In *American Control Conference, 2008*. IEEE, pp. 1394–1399, 2008.

Vapnik, V. N. The nature of statistical learning theory, 1995.

Watkins, C. J. and Dayan, P. Q-learning. *Machine learning* 8 (3-4): 279–292, 1992.

Weeraratna, A. T., Jiang, Y., Hostetter, G., Rosenblatt, K., Duray, P., Bittner, M., and Trent, J. M. Wnt5a signaling directly affects cell motility and invasion of metastatic melanoma. *Cancer cell* 1 (3): 279–288, 2002.

Yousefi, M. R. and Dougherty, E. R. Intervention in gene regulatory networks with maximal phenotype alteration. *Bioinformatics* 29 (14): 1758–1767, 2013.

Identificação de Regiões Densas de Trajetórias Atômicas em Simulações de Dinâmica Molecular

A. M. Kronbauer[1], L. A. Schmidt[1], K. S. Machado[2], A. T. Winck[1]

[1] Universidade Federal de Santa Maria, Brasil
alyne.k@gmail.com, lschmidt@inf.ufsm.br, ana@inf.ufsm.br
[2] Centro de Ciências Computacionais, Universidade Federal do Rio Grande, Brasil
karina.machado@furg.br

Abstract.

Desenho racional de fármacos (RDD) é um campo da bioinformática que se preocupa com o desenvolvimento de novas drogas. Trata-se de um processo caro e custoso, onde testes *in silico* buscam reduzir tempo e custo. Para tanto, testes ocorrem sobre conformações de moléculas receptoras e de um dado ligante, buscando encontrar a melhor posição que um ligante deve estar para inibir a atividade do receptor, onde essa ligação é testada por meio de simulações de docagem molecular. Este trabalho apresenta uma abordagem baseada em agrupamento de dados por densidade para encontrar regiões densas de átomos de proteínas que melhor contribuam para esses bons resultados de docagem molecular. Foram realizados diferentes experimentos com parametrização do algoritmo, propondo-se uma metodologia capaz de identificar essas regiões densas. Esses resultados contribuem na redução do tempo de novos experimentos ao selecionar conformações promissoras, com base nos melhores grupos, bem como permite ao especialista obter um maior entendimento a respeito da proteína sendo analisada.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

1. INTRODUÇÃO

Aprendizagem de máquina é uma das áreas da Inteligência Artificial que visa o reconhecimento de padrões em conjuntos de dados. Diversas são as áreas de aplicação para problemas de aprendizado de máquina, dentre os quais destaca-se as bases de dados de ordem biológica. Esse tipo de dados pertencem a problemas de bioinformática, a qual corresponde a uma área multidisciplinar, envolvendo biologia e computação [Lesk 2005]. Dentre as áreas de investigação em bioinformática, destaca-se a relacionada ao Desenho Racional de Fármacos (*RDD* – Rational Drug Design) [Kuntz 1992] – objeto de estudo deste trabalho.

A área de RDD busca acelerar o processo de produção de novos medicamentos, por meio de quatro fases principais [Kuntz 1992]. Uma dessas fases é a predição *in-silico* da interação entre ligantes e receptores. Essa análise é feita por meio de simulações de docagem molecular, onde é analisado o encaixe de uma estrutura candidata a fármaco, ou ligante, em uma macromolécula receptora. Essa simulação de docagem molecular pode tornar-se dispendiosa, uma vez que as estruturas moleculares não são rígidas no ambiente celular, e simular a flexibilidade do receptor aparece como uma etapa importante. A flexibilidade do receptor pode ser obtida através de simulações por dinâmica molecular, a qual gera uma conformação da proteína para cada instante de tempo [van Gunsteren and Berendsen 1990][Karplus and McCammon 2002]. Assim, os experimentos de docagem são realizados sobre cada uma dessas estruturas.

Para contribuir com a redução do tempo nos experimentos de docagem molecular considerando a flexibilidade do receptor, este apresenta uma abordagem de agrupamento de dados baseado em densidade, por meio do algoritmo DBSCan [Ester et al. 1996], para selecionar conformações promissoras. Nossa abordagem contribui para a identificação das regiões atômicas densas, obtidas por meio de simulação de dinâmica molecular, que indiquem importante influência positiva em experimentos de docagem molecular. Assim, contribui-se não apenas com a redução do tempo desses experimentos, como também com o entendimento a respeito dos aspectos flexíveis da proteína.

2. MATERIAIS E MÉTODOS

O desenho racional de fármacos (*RDD*) [Kuntz 1992] é um processo que visa selecionar novos candidatos a fármacos, incluindo uma etapa de testes computacionais (testes *in silico*). Esse processo é baseado em análises das interações entre ligantes e receptores simuladas por docagem molecular [Lybrand 1995]. O objetivo da docagem molecular é indicar ligantes com maior potencial levando em consideração um receptor alvo. Para isso, estima-se a energia livre de ligação (*FEB – Free Energy of Binding*), sendo que quanto menor o FEB, melhor a interação receptor-ligante.

Um dos problemas envolvidos é a consideração das estruturas receptoras e ligantes como sendo flexíveis. Como as moléculas não são rígidas no ambiente celular é preciso que essa flexibilidade também seja considerada no processo *in silico*. Uma das possíveis estratégias é simular tal flexibilidade por meio de simulações por dinâmica molecular [van Gunsteren and Berendsen 1990][Karplus and McCammon 2002]. Tais simulações geram uma conformação do receptor para cada instante de tempo. Cada conformação obtida da simulação por dinâmica molecular é utilizada em experimentos de docagem molecular.

Esse processo, entretanto, torna-se computacionalmente custoso. Nesse sentido, é importante fazer uso de estratégias que selecionem aquelas conformações que tenham chance de resultar em bons resultados nos experimentos de docagem molecular. Além disso, conhecer essas conformações promissoras pode ser útil para os especialistas de domínio para entenderem a flexibilidade da proteína e, assim, identificar quais são as regiões flexíveis da mesma que contribuem para esses bons resultados de docagem.

Existem esforços em acelerar e entender esse processo, onde alguns deles são realizados por meio de técnicas de aprendizado de máquina. Este trabalho busca utilizar de uma abordagem de agrupamento de dados para agrupar conformações de proteínas, tendo como características as coordenadas tridimensionais dos seus átomos. Técnicas de agrupamento de dados normalmente não são genéricas, ou seja, não são aplicáveis de maneira satisfatória a todos os tipos de dados e em todos os contextos. Por isso, realizou-se testes com diferentes abordagens de agrupamento com o intuito de identificar aqueles algoritmos que melhor produzem resultados quando aplicados sobre datasets com representação tridimensional dos átomos. O estudo de [rodrigues Machado 2014] indica o algoritmo DBSCAN [Ester et al. 1996] como o mais apropriado para conjuntos de dados desse tipo.

O presente trabalho propõe uma abordagem de agrupamento baseado em densidade para encontrar regiões densas que indiquem a melhor concentração de instâncias que conduzem a bons resultados de FEB. Ao utilizar a técnica do DBSCAN, entende-se que ela é capaz de gerar a quantidade de grupos adequada para cada átomo, de acordo com a trajetória que o mesmo percorre quando de sua flexibilidade. Para tanto, é preciso também substituir a informação arbitrária do número de grupos (valor para k) pelos parâmetros admitidos pelo DBSCAN: Raio (Eps) e Número mínimo de pontos ($MinPts$).

Nossa abordagem considera cada conformação da proteína como sendo um exemplo do *dataset*, e considera seus átomos como sendo as características do *dataset*, sendo que cada átomo é representado por sua conformação tridimensional e cada átomo é individualmente submetido à técnica do DBSCAN. Os experimentos são realizados sobre a proteína *Acriflavine resistance protein B (AcrB)*, composta

por 9.662 átomos. Os dados dessa proteína foram obtidos do trabalho de Prates [Prates 2014], que considera 1.001 conformações para experimentos de docagem molecular. Os arquivos resultantes da dinâmica molecular e da docagem molecular com a Proteína AcrB foram preprocessados para geração de um *dataset*. Considerando os 9.662 átomos da proteína, esse *dataset* é composto por 1.001 linhas (conformações) e por 28.987 colunas (três colunas para cada átomo, sendo que cada coluna representa a sua coordenada x, y, z respectivamente, mais uma coluna para o valor de FEB).

2.1 Casos de Teste

Elaborou-se uma rotina de teste para verificar quais seriam os melhores valores a serem informados para Eps e $MinPts$. Para tanto, adotou-se uma estratégia de criação de um *dataset* de teste, composto por 25 átomos aleatórios extraídos do *dataset* original. Primeiramente optou-se por assumir um valor de $MinPts$ único para todos os átomos, configurando um valor baixo, deixando o algoritmo percorrer os pontos dos átomos e decidir quantos são os pontos que devem estar presentes em cada grupo formado. Ao assumir o valor $MinPt$s entre 1 e 4, notou-se que foram gerados muitos grupos, mas muitos deles compostos por uma única instância. Além disso, muitas vezes o grupo considerado o melhor – aquele com menor valor médio de FEB – era justamente o que continha uma única instância. Como o objetivo é gerar um intervalo ideal de mobilidade do átomo, optou-se por manter o valor de $MinPts$ variando entre 5, 10 ou 20.

Em seguida passou-se a efetuar uma série de testes para avaliar a influência do valor de Eps na geração dos grupos. Foram adotados diferentes valores aleatórios, resultando em grupos muito grandes ou apenas um grupo. Como o número de conformações e instâncias é relativamente grande e como a proteína é flexível, a formação de apenas um grupo não é uma alternativa adequada. Então, passou-se a explorar a variância existente.

Uma vez calculadas as variâncias das coordenadas dos átomos nas conformações, analisou-se os valores da variância máxima, média e mínima, e estes valores foram configurados no parâmetro de Eps, ou seja, o raio de distância para considerar o grupo formado passou a ser dado pela variância dos valores constantes no *dataset*. Com o resultado dos primeiros testes, chegou-se a conclusão que a variância mínima era o valor inicial a ser trabalhado pois, em grande parte das execuções, foi possível observar a formação de diferentes números de grupos.

Nessa fase define-se que a qualidade de um agrupamento é dada pela quantidade de grupos formados. Assim, considera-se um átomo válido aquele cujo agrupamento resulta em mais de um grupo. Dessa forma, garante-se de antemão uma seleção dos átomos que apresentam flexibilidade relevante tal que ao representa-la em um agrupamento, obtem-se, pelo menos, duas regiões distintas e densas. Assim, os átomos não considerados válidos nessa etapa não são átomos candidatos para as etapas subsequentes do algoritmo.

Para tanto, identificou-se a menor variância dentre os eixos e aplicou-se uma taxa sobre essa variância para definir Eps. Isso é, considerando que o *dataset* contenha A átomos, onde $A = a_1, a_2, ..., a_n$, para cada átomo a obtem-se os vetores $X = x_1, x_2, ..., x_n$, $Y - y_1, y_2, ..., y_n$, e $Z = z_1, z_2, ..., z_n$, onde n é o número de instâncias sendo processadas. Em seguida, calcula-se a variância desses vetores, obtendo-se $VarX = Var(X_a)$, $VarY = Var(Y_a)$, e $Z = Var(Z_a)$. A menor variância dos eixos é então dada por $MenorVar_a = min(VarX, VarY, VarZ)$.

Primeiramente foram realizados 19 testes em que o valor de Eps varia com uma taxa t aplicada sobre $MenorVar_a$, onde t varia em um ponto percentual, de 1 a 9 e em dez pontos percentuais, de 10 a 100. Assim, $Eps = MenorVar_a \times t$. O objetivo em se aplicar taxas sobre o valor de de $MenorVar_a$ para determinar o valor de Eps_a é o de permitir ao DBSCAN observar a trajetória dos átomos considerando a sua própria flexibilidade. Nessa rotina de testes executou-se o DBSCAN parametrizando Eps_a para cada um dos 25 átomos de $DataTeste1$, com $MinPts = 10$, observando-se: a) o número de átomos válidos, ou seja, aqueles cujo resultado retornou mais de um grupo; b) o tempo

de execução; c) o valor de Eps para cada átomo; d) o número de grupos gerados; e) o FEB médio para o melhor grupo, onde $MelhorGrupo$ corresponde aquele com menor FEB médio; e f) o número de instâncias ($NumInstancias$) no melhor grupo.

Analisando as execuções sobre $DataTeste1$ com as 19 configurações descritas, não obteve-se nenhum átomo válido (com mais de um grupo gerado) para a taxa de Eps variando de 0,20 a 1,00. Sabendo que são gerados átomos válidos para as configurações de 0,01 à 0,10, optou-se por escolher apenas as configurações que aplicam uma taxa entre 0,05 e 0,10 para definição do Eps. Para cada configuração de Eps, varia-se o número mínimo de pontos ($MinPts$) em 5, 10 e 20. Essas configurações foram assim definidas com o objetivo de ilustrar o comportamento do algoritmo para esses 25 átomos.

3. RESULTADOS

Para analisar os resultados da metodologia proposta, foram definidas quatro configurações distintas dos parâmetros. Estas variam quanto à taxa sobre a variância aplicada ao Eps entre 0,05 ou 0,10; e o valor de $MinPts$, explicitadas na Tabela I.

Table I. Configurações dos experimentos

Experimento	Taxa Variância Eps	$MinPts$
1	0.05	5
2	0.05	10
3	0.10	5
4	0.10	10

Dado que os valores de configuração para o DBSCAN são os que influenciam na geração de átomos válidos, a primeira análise dos resultados diz respeito à identificação do número de átomos válidos gerados, isto é, aqueles cuja configuração para execução do DBSCAN resultou em mais de um grupo.

A Tabela II mostra, para cada Experimento, o número de átomos válidos e o percentual sobre o total de átomos do *dataset*. Essa informação busca ilustrar o percentual de seleção de átomos feita *a priori* pela execução do algoritmo. Percebe-se que $MinPts$ tem maior influência no total de átomos válidos gerados. O número de átomos válidos diz respeito àqueles átomos que tendem a ter uma maior flexibilidade durante os experimentos de Dinâmica Molecular, uma vez que se considera aqueles átomos cuja trajetória resultou em alguma concentração densa que pode ser distinguida de outra.

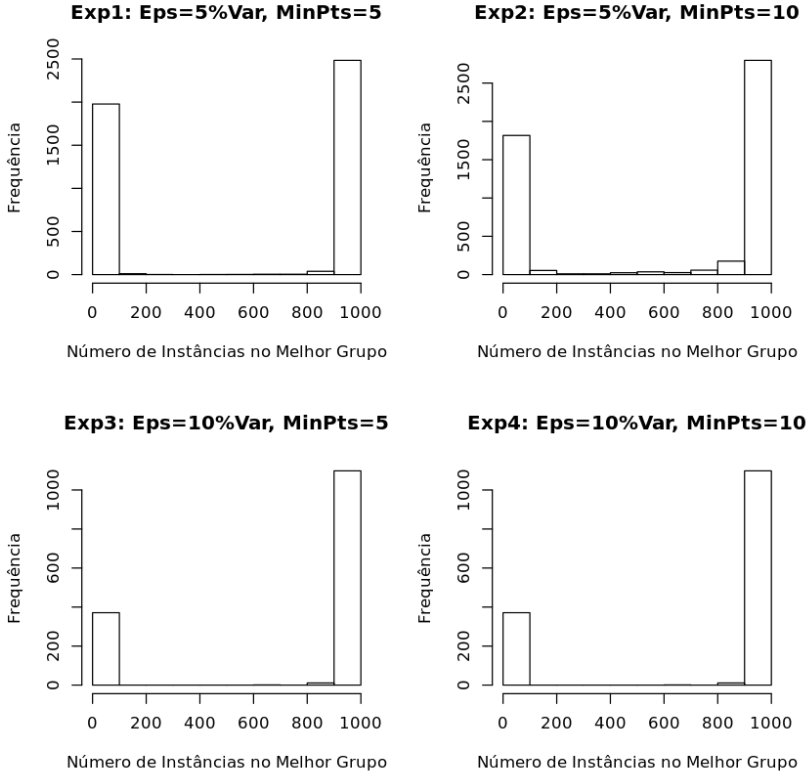
Table II. Átomos válidos por experimento

Experimento	Átomos Válidos	% de Seleção
1	5.422	56,1167
2	5.006	51,8112
3	1.482	15,3384
4	1.926	19,9338

Para os átomos válidos, é preciso olhar para as regiões densas formadas por meio do agrupamento e identificar quais são as instâncias que as compõem. Em especial, é preciso olhar para aquela região densa que forma o melhor grupo (aquele cujas instâncias resultam em um menor FEB médio). Para tanto, observou-se o número de instâncias pertencentes ao melhor grupo. Para entender essa distribuição, foram gerados histogramas das instâncias do melhor grupo para cada Experimento, apresentados na Figura 1.

Nota-se que há uma grande concentração de agrupamentos que reúnem, em seu melhor grupo, até 200 instâncias, contrastando com uma maior concentração de agrupamentos que reúnem, em seu melhor grupo, acima de 900 instâncias. Por entender-se que uma alta concentração de instâncias no melhor grupo não garante uma seleção de conformações, optou-se por realizar uma nova fase de

Fig. 1. Total de instâncias no melhor grupo



seleção de átomos, considerando apenas os átomos cujo melhor grupo seja composto por até 50% do total de instâncias do *dataset*, isso é, aqueles que contém até 500 conformações. Para visualizar tal comportamento, a Tabela III mostra, para cada Experimento, o total de átomos válidos gerados, o total de átomos selecionados (aqueles com até 500 instâncias), o percentual de átomos selecionados em relação aos átomos válidos para o Experimento e o percentual de átomos selecionados em comparação ao total de átomos.

Table III. Átomos selecionados por experimento

Experimento	Atomos Válidos	Atomos Selecionados	% Selecionados Experimento	%Selecionados Total
1	5.422	1.089	20,01	11,2710
2	5.006	1.914	38,23	19,8086
3	1.482	371	25,03	3,8398
4	1.926	365	18,95	3,7777

Ao realizar uma nova fase de seleção, a quantidade de átomos a ser observada reduz consideravelmente. Sendo assim, optou-se por realizar as análises subsequentes dos resultados em duas categorias distintas: a primeira considerando todos os átomos válidos descritos na Tabela II, aqui denominada *Átomos Válidos*; e a segunda considerando apenas os átomos selecionados após a execução do algoritmo, observados na Tabela III, aqui denominada *Átomos Selecionados*. Assim, percebe-se que, apesar de haver uma forte concentração de resultados cujo melhor grupo contém um pequeno número de instâncias, próximos de seus valores mínimos, também percebe-se que quanto maior o número de instâncias, menor é a frequência de resultados.

3.1 Análise do número de grupos

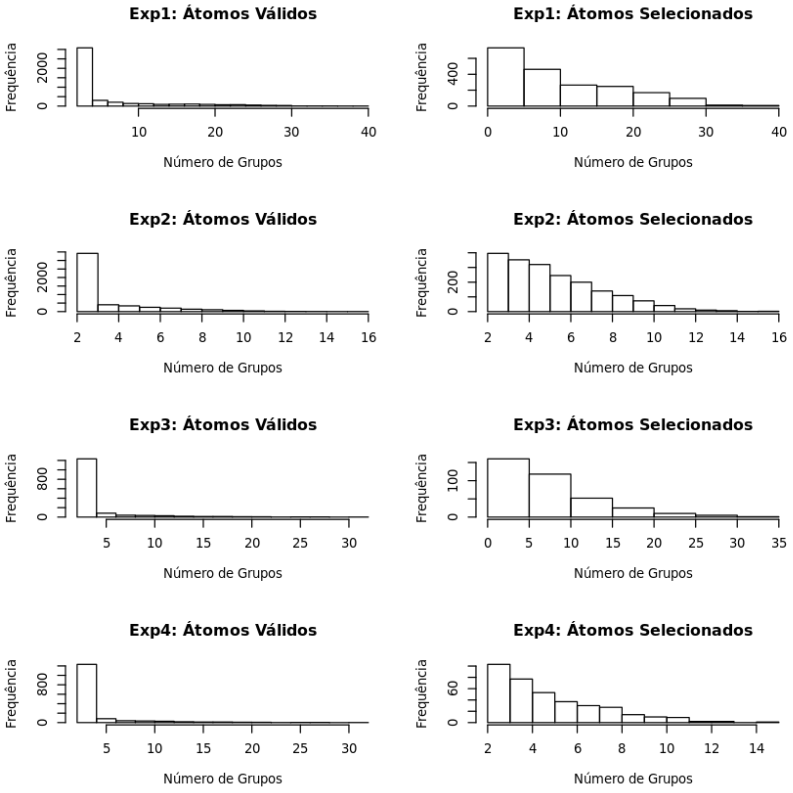
Dando sequência à análise dos resultados, são identificados o número de grupos gerados. Com essa informação objetiva-se observar o comportamento dos átomos em termos de sua flexibilidade. Acredita-se que quanto maior sua flexibilidade, mais regiões densas são formadas e, consequentemente, maior é o número de grupos. A Tabela IV mostra os valores mínimos, máximos e médios do número de grupos gerados para os átomos válidos e para os átomos selecionados.

Table IV. Número de grupos gerados por experimento e categoria

Experimento	Categoria	Mínimo	Máximo	Média
1	Átomos Válidos	2	39	5,8931 ± 5,6915
	Átomos Selecionados	2	39	10,6552 ± 8,3231
2	Átomos Válidos	2	16	3,4674 ± 2,2951
	Átomos Selecionados	2	16	5,6332 ± 2,4182
3	Átomos Válidos	2	31	3,5436 ± 3,7891
	Átomos Selecionados	2	31	7,9623 ± 5,5548
4	Átomos Válidos	2	15	2,6629 ± 1,6395
	Átomos Selecionados	2	15	5,2548 ± 2,3404

A Figura 2 ilustra a distribuição do número de grupos para cada categoria avaliada em cada experimento. A distribuição é apresentada na forma de histogramas, os quais estão dispostos em ordem de experimento de cima para baixo, sendo que os gráficos da esquerda representam todos os átomos válidos e os da direita apenas os átomos selecionados.

Fig. 2. Número de grupos gerados por categorias e por experimento



Nota-se que, apesar de o número de grupos apresentar os mesmos valores mínimos e máximos para os átomos válidos e para os selecionados, a distribuição parece mais linear nos átomos selecionados. Isto é, quando analisados todos os átomos válidos, há uma grande concentração de resultados que geram apenas dois grupos, que é o número mínimo de grupos aceito pelo algoritmo. Por outro lado, ao analisar os átomos selecionados, é possível identificar que apesar da distribuição ser mais concentrada no número mínimo, ela decresce linearmente para as demais conformações. Esses resultados sugerem que a seleção dos átomos é pertinente para representar a flexibilidade dos mesmos.

3.2 Análise dos valores médios de FEB

Para os experimentos também são observadas a distribuição dos valores médios de FEB nos melhores grupos. A Tabela V mostra seus valores mínimos, máximos e médios, por experimento e por categoria.

Table V. FEB médio dos melhores grupos por experimento e categoria

Experimento	Categoria	Mínimo	Máximo	Média
1	Átomos Válidos	-8,9250	-7,7200	-7,9545 ± 0,2830
	Átomos Selecionados	-8,9250	-7,7200	-8,2278 ± 0,2324
2	Átomos Válidos	-8,7500	-7,3500	-7,8500 ± 0,1850
	Átomos Selecionados	-8,7500	-7,3500	-8,0300 ± 0,1934
3	Átomos Válidos	-9,2000	-7,7236	-7,8553 ± 0,2288
	Átomos Selecionados	-9,2000	-7,7400	-8,2018 ± 0,2208
4	Átomos Válidos	-8,8667	-7,3882	-7,7991 ± 0,1358
	Átomos Selecionados	-8,8667	-7,3882	-8,0151 ± 0,1878

A Figura 3 apresenta histogramas que representam o menor FEB médio para o melhor grupo, por experimento e por categoria. Nos quatro experimentos, há uma forte concentração de resultados cujo FEB médio do melhor grupo faz parte dos menores FEBs médios obtidos. Em contrapartida, é interessante observar a distribuição dos valores médios de FEB quando analisados apenas os átomos selecionados, pois seus histogramas formam quase que uma distribuição normal.

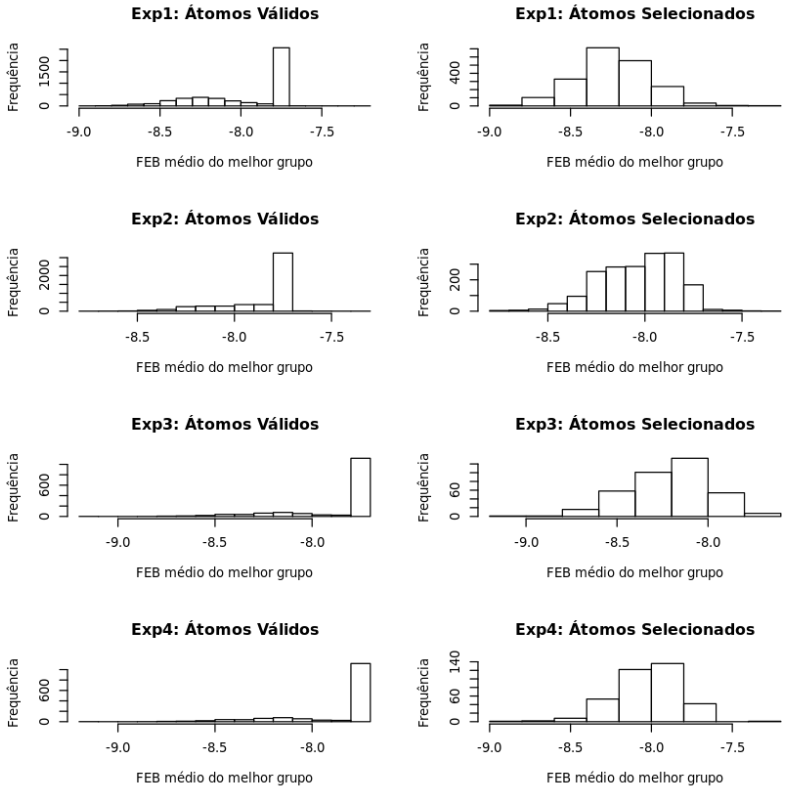
Com os resultados pode-se dizer que a metodologia proposta é capaz de encontrar regiões densas promissoras para átomos que tendem a ser relevantes de se inspecionar por um especialista de domínio. Isto é, os resultados obtidos até a análise dos agrupamentos já permite selecionar os átomos e suas respectivas conformações que podem resultar em bons valores de FEB. Essa informação pode ajudar um especialista a compreender o comportamento dos átomos da proteína.

4. CONSIDERAÇÕES

Este trabalho apresenta uma estratégia para seleção de conformações geradas a partir de Dinâmica Molecular para futuros experimentos de docagem molecular, fazendo uso de uma abordagem de agrupamento de dados baseado em densidade. A adoção da estratégia apontada por este trabalho é eficaz na identificação de regiões densas que representam a flexibilidade da proteína AcrB. Com o método, busca reduzir, através de diferentes parametrizações do algoritmo proposto, a porcentagem de átomos considerados promissores. A estratégia proposta torna possível que o especialista tenha um número reduzido de átomos válidos a serem considerados em sua análise, o que vai permitir que a demanda de tempo e custo para esse tipo de análise seja minimizada.

Como trabalhos futuros propõe-se pesquisas sobre medidas de similaridade que possam ser úteis no agrupamento desse tipo de dados, diferentes módulos para parametrização do algoritmo e incorporação de novos gráficos e medidas que visem auxiliar na análise dos agrupamentos formados.

Fig. 3. FEB Médio do melhor grupo por experimento e categoria



AGRADECIMENTO

Este trabalho teve apoio financeiro do CNPq por meio do Edital Universal processo nº 476764/2013-0 para Ana T. Winck e do processo nº 477462/2013-8 para Karina S. Machado, e da CAPES pelo Edital Biologia Computacional CAPES nº 051/2013 para Karina S. Machado. Aline M. Kronbauer realizou seu mestrado com apoio de bolsa FAPERGS. Leonardo A. Schmidt é bolsista PET.

REFERENCES

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. AAAI Press, pp. 226–231, 1996.

Karplus, M. and McCammon, A. Molecular Dynamics Simulation of Biomolecules. *Nat. Struct. Biol.* vol. 9, pp. 646–652, 2002.

Kuntz, I. D. Structure-based Strategies for Drug Design and Discovery. *Science* vol. 257, pp. 1078–1082, 1992.

Lesk, A. M. *Introduction to bioinformatics*. Oxford University Press, Oxford, 2005.

Lybrand, T. Ligand-Protein Docking and Rational Drug Design. *Curr. Opin. Struct. Biol.* vol. 5, pp. 224–228, 1995.

Prates, N. S. *Simulações por Dinâmica Molecular e Agrupamento de Estruturas da Proteína da Bomba de Efluxo AcrB*. M.S. thesis, Universidade Federal de Rio Grande, Rio Grande, 2014.

rodrigues Machado, O. Um estudo comparativo de algoritmos de agrupamento de dados para dados de docagem molecular. Tech. rep., Curso de Ciência da Computação. Universidade Federal de Santa Maria., Santa Maria, 2014.

van Gunsteren, W. F. and Berendsen, H. J. C. Computer simulation of molecular dynamics: Methodology, applications, and perspectives in chemistry. *Angewandte Chemie International Edition in English* 29 (9): 992–1023, 1990.

Classificação de Relações Abertas Utilizando Features Independentes do Idioma

George C. G. Barbosa, Rafael Glauber, Daniela Barreiro Claro

Formalismos e Aplicações Semânticas (FORMAS)
LaSiD - Departamento de Ciência da Computação - Universidade Federal da Bahia
Av. Adhemar de Barros, s/n, Ondina, Salvador - Bahia - Brasil
gcgbarbosa@gmail.com, rglauber@dcc.ufba.br, dclaro@ufba.br

Abstract. A quantidade de textos em linguagem natural disponível na Internet vem aumentando os desafios do seu processamento automatizado. Diversas abordagens de Extração da Informação estão sendo propostas, principalmente no que cerne as entidades e as suas relações. Extrair relações abertas, ou seja, sem um conhecimento pré-determinado, tem sido um dos importantes desafios do processamento de textos na Internet. A abordagem aberta pode ser dividida em duas etapas: (i) extração e (ii) classificação. Porém, os trabalhos relacionados apresentam a dependência do idioma em ambas as etapas. Assim, este trabalho propõe um conjunto de *features* usado na etapa de classificação que não utilizam termos presentes em um idioma específico, tornado o método de classificação independente de idioma. Experimentos foram realizados em três diferentes corpora com sentenças extraídas da Web, Wikipédia e do New York Times (em Inglês) e os resultados apresentados neste artigo foram promissores para o direcionamento da pesquisa.

Categories and Subject Descriptors: H.3.3 [**Information systems**]: Information extraction

Keywords: Extração de Informação Aberta, Extração de Relação, Features, Independência de Idioma

1. INTRODUÇÃO

Há um grande conjunto de dados textuais disponível na Internet escrito em linguagem natural nos mais diferentes idiomas. Aproximadamente 50% do conteúdo disponível em web sites é escrito em Inglês[1]. Os demais idiomas somam a outra metade do conteúdo disponível e muitos esforços têm sido realizados no sentido de extrair informação útil desses dados [Fader et al. 2011]. A Extração da Informação (IE, do Inglês *Information Extraction*) é a tarefa de aquisição de informação a partir de dados não estruturados ou semi-estruturados. Embora a pesquisa para extrair informação em textos em Inglês na Web esteja avançando nos últimos anos [Banko et al. 2007] [Wu and Weld 2010] [Fader et al. 2011] [Schmitz et al. 2012] [Del Corro and Gemulla 2013] [Angeli et al. 2015], as pesquisa em diferentes idiomas têm recebido pouca atenção [Gamallo et al. 2012].

A tarefa de IE pode ser classificada em aberta ou fechada. A IE fechada, também conhecida como tradicional, tem como objetivo a extração de relações em um domínio específico, geralmente um conjunto pré-especificado de expressões [Schmitz et al. 2012]. Já a IE aberta (OIE, do Inglês *Open Information Extraction*) tem como principais objetivos: (i) independência de domínio, (ii) extração não supervisionada e (iii) escalabilidade para grandes bases de dados [Del Corro and Gemulla 2013]. A OIE pode ser classificada pelas técnicas empregadas na extração, que podem ser (por ordem de complexidade): (i) análise rasa, (ii) análise de dependência e (iii) anotação de papéis semânticos [Mesquita et al. 2013]. Na análise rasa, o método de extração é realizado em duas etapas, sendo a

[1]https://w3techs.com/technologies/overview/content_language/all

primeira etapa a de extração propriamente dita e, posteriormente, é feita a classificação das relações extraídas. Esta segunda etapa, foco deste trabalho, define se uma extração realizada é válida ou inválida com o objetivo de conferir ao método uma melhor precisão nos resultados. Embora a OIE apresente uma melhor cobertura nas extrações realizadas, não se limitando a um conjunto específico de relações, um *trade-off* é esperado nos resultados de precisão para estes métodos.

A IE em textos utiliza tarefas realizadas no Processamento de Linguagem Natural (NLP, do Inglês *Natural Language Processing*) tais como: *Tokenization*, *Sentence Splitting* e *Part-of-Speech tagging - POS* [Manning et al. 2014]. Estas tarefas são altamente dependentes do idioma no qual o texto foi escrito. Os trabalhos encontrados na literatura que utilizam classificadores baseados em *features* [Fader et al. 2011] [Xu et al. 2013] [Pereira and Pinheiro 2015] também utilizam funções linguísticas dependentes de idioma. O objetivo deste trabalho é contribuir nesta área avançando a pesquisa para a independência de idioma. A hipótese é de que *features* independentes de características linguísticas podem apresentar resultados similares a *features* dependentes dessas características.

Entende-se por dependência de idioma a utilização de funções linguísticas que estão presentes no idioma alvo do estudo, mas não fazem parte de outros idiomas. Por exemplo, o Português não apresenta nenhum recurso similar ao *genitive marker* ('s) do Inglês. Com isso, a utilização dessa função linguística em alguma *feature* tornaria difícil a adaptação do método para o Português.

Assim, o presente trabalho está estruturado como segue: a Seção 2 apresenta os trabalhos relacionados, a Seção 3 descreve a proposta deste trabalho. Na Seção 4 são descritos os experimentos e a Seção 5 apresenta os resultados. A Seção 6 apresenta as conclusões seguida da definição dos trabalhos futuros que encerra este artigo.

2. TRABALHOS RELACIONADOS

O primeiro trabalho a introduzir a OIE foi o TextRunner [Banko et al. 2007] que fazia uso de etiquetação gramatical (*Part-of-Speech*) e etiquetação de frases nominais (NP, do inglês *Noun Phrase*) e um classificador *Naïve Bayes* treinado usando exemplos gerados a partir do *Penn Tree Bank*. Trabalhos posteriores mostraram que a utilização de uma cadeia linear CRF [Banko et al. 2008] melhorava a qualidade das extrações. Por fim, [Wu and Weld 2010] demonstraram com o WOE^{Parse} que era possível usar as tabelas de informação presentes nas páginas da Wikipédia como fonte de treinamento, o que resulta em uma melhora significativa na cobertura em decorrência da disponibilidade de uma grande base de treino.

Os sistemas de extração de relações precursores em OIE obtém extratos na forma (e_1, frase relacional, e_2) em três etapas [Wu and Weld 2010]:

(1) **Etiquetação**: As sentenças são etiquetadas automaticamente através de heurísticas ou a partir de supervisão distante (treinamento semi-supervisionado);
(2) **Aprendizado**: Um extrator de frases relacionais é treinado utilizando um modelo de etiquetação sequencial (ex: CRF);
(3) **Extração**: Um conjunto de argumentos (e_1, e_2) é identificado na sentença de teste. Em seguida o extrator treinado na etapa 2 é utilizado para etiquetar as palavras contidas entre os argumentos e compor a frase relacional (caso ela exista), extraindo a relação no formato (e_1, $frase relacional$, e_2).

As abordagens mais recentes têm sido desenvolvidas por meio de modificações na metodologia e, consequentemente, nas estrategias adotadas nas etapas de extração [Banko et al. 2007] [Banko et al. 2008] [Fader et al. 2011]. Assim, é realizada primeiramente a etapa de extração, seguida pelo aprendizado necessário à posterior classificação das relações conforme descrito abaixo:

(1) **Extração**: Inicialmente, um extrator baseado em padrões linguísticos (ex: padrões verbais) seleciona uma sequencia de palavras que representa a relação semântica entre e_1 e e_2, identificando frases relacionais que casam com esses padrões. Em seguida, se um conjunto de argumentos (e_1, e_2) for identificado na sentença de teste, então é gerada a relação na forma (e_1, frase relacional, e_2);
(2) **Aprendizado**: Um classificador de extrações é treinado por meio de um conjunto de *features* linguísticas;
(3) **Classificação**: O classificador treinado na etapa 2 é utilizado para distinguir as relações válidas das inválidas geradas na etapa 1.

Essa nova abordagem tornou-se mais sólida com o *ReVerb* [Fader et al. 2011]. Ela substitui o aprendizado, na etapa de extração, pelo processamento de regras baseadas em padrões morfológicos. Após a extração das relações, um classificador é utilizado na remoção das extrações inválidas do conjunto que contém todas as relações extraídas. Esta metodologia permite uma redução significativa na cardinalidade do conjunto de treinamento, já que a complexidade do aprendizado para classificação das relações é inferior à do aprendizado para a identificação das mesmas. Por outro lado, a construção de conjuntos de treinamento a partir de *features* linguísticas eleva o custo de classificação, pois a identificação de *features* representativas requer uma análise mais aprofundada das características da língua no contexto do problema.

Recentemente, trabalhos que fazem uso de técnicas de análise de dependência demostraram uma melhora na quantidade de relações extraídas. O *OLLIE* [Schmitz et al. 2012] faz uso de um conjunto de padrões aprendidos a partir de uma base de extrações de alto grau de confiança obtidos pelo ReVerb para, em seguida, extrair relações de forma aberta. Uma abordagem similar é utilizada no *ClausIE* [Del Corro and Gemulla 2013], na qual, padrões identificados manualmente a partir da árvore de dependência das sentenças são utilizados para extrair relações. Em [Angeli et al. 2015] uma abordagem similar ao *ClausIE* é utilizada, porém, antes da etapa de extração são aplicadas técnicas para separar as sentenças em núcleos semânticos, de forma que as relações extraídas possuam a menor quantidade de *tokens* necessária para que a frase relacional seja informativa, facilitando a utilização das extrações resultantes para outros fins (e.g. criação ontologias).

3. NOSSA PROPOSTA

O presente trabalho utiliza extrações feitas a partir do ClausIE, OLLIE, ReVerb, WOE e TextRunner no idioma Inglês para avaliar a eficiência do nosso conjunto de *features* apresentado na Tabela I. Este novo conjunto é resultado da alteração de algumas das *features* proposta em [Fader et al. 2011]. As *features* que antes utilizavam palavras do Inglês agora utilizam classes morfológicas que estão presentes nos principais idiomas utilizados no mundo na atualidade, tais como: Inglês, Espanhol, Francês, Português e Alemão. Com isso, não há a necessidade de adaptação do método de classificação para que ele possa ser aplicado a outros idiomas.

Na Tabela II é apresentada a lista de *features* utilizadas em [Fader et al. 2011]. A maioria delas faz referência as características especificas do Inglês. Por exemplo: as *features* 2-4 possuem palavras do Inglês que podem não ter correspondentes em outros idiomas ('for', 'on', 'of'). A *feature* 6 diz respeito as palavras "WH" (e.g. 'What', 'Why', 'Where'), que são específicas tornando bastante difícil o uso direto deste conjunto, por exemplo, em Português.

4. EXPERIMENTOS

Este trabalho se concentra na etapa de classificação binária (válida ou inválida) de extrações de relação realizadas em sentenças escritas em linguagem natural. É considerado em nossos experimentos que a primeira etapa da tarefa de OIE (extração) foi realizada previamente por algum sistema e que

	Feature
1	Tamanho de S - Tamanho de E1+FR+E2
2	Número de verbos na FR
3	Tamanho de FR
4	Existe uma pergunta a esquerda da FR em S
5	A sentença tem 10 palavras ou menos
6	Distância entre E1 e FR
7	Existe uma preposição a esquerda de E1
8	Tamanho de E2
9	Distância entre E2 e FR
10	Número de preposições na FR
11	Número de substantivos a direta de E2
12	Tamanho de E1
13	Tamanho de S
14	Número de nomes próprios em E1
15	Número de nomes próprios em E2

Table I: Features propostas neste trabalho.

S: sentença na qual é feita a extração
$E1$ e $E2$: entidades nominais da tripla da relação
FR: frase relacional da extração

	Feature
1	Extração cobre todas as palavras da sentença
2	A ultima preposição na relação é 'for'
3	A ultima preposição na relação é 'on'
4	A ultima preposição na relação é 'of'
5	A sentença tem 10 palavras ou menos
6	Existe uma palavra com 'WH' a esquerda da relação na sentença
7	A relação corresponde ao padrão VW*P
8	A ultima preposição na relação é 'to'
9	A ultima preposição na relação é 'in'
10	A sentença tem entre 10 e 20 palavras
11	A sentença começa com E1
12	E1 é um nome próprio
13	E2 é um nome próprio
14	Existe uma frase nominal a esquerda de E1 na sentença
15	A sentença tem mais de 20 palavras
16	A relação corresponde ao padrão V
17	Existe uma preposição a esquerda de E1 na sentença
18	Existe uma frase nominal a direita de E2 na sentença
19	Existe uma conjunção coordenativa a esquerda da relação na sentença

Table II: Features utilizadas no ReVerb [Fader et al. 2011].

nossa proposta pode ser utilizada por qualquer sistema que realiza extrações em textos dando a estes extratores maior confiabilidade aos seus resultados (melhorando sua precisão e confiança). Os experimentos foram organizados visando analisar duas diferentes dimensões para o problema: a) capacidade preditiva da nossa proposta e b) verificar qual a necessidade de exemplos de treinamento é requerido para que o modelo alcance resultados satisfatórios.

4.1 Conjunto de dados

Nos experimentos foram utilizados três conjuntos de dados obtidos a partir do trabalho de [Del Corro and Gemulla 2013][2]. Os pesquisadores disponibilizaram os arquivos, os quais possuem 200 sentenças extraídas aleatoriamente do New York Times (NYT-200) e 200 sentenças extraídas aleatoriamente da Wikipédia (Wiki-200). O terceiro arquivo é o conjunto de dados apresentado em [Fader et al. 2011] (ReVerb-500). Todos os arquivos estão organizados da seguinte forma: (i) uma linha contendo a sentença original S_1 e N linhas subsequentes contendo as relações extraídas a partir de S_1, sendo este padrão repetido para as sentenças S_2 até S_n. Cada arquivo possui extrações de cinco dos mais importantes trabalhos em OIE presentes na literatura (ClausIE, OLLIE, ReVerb, WOE e TextRunner). As 2094 extrações realizadas em NYT-200, as 1769 em Wiki-200 e as 6443 de ReVerb-500 foram avaliadas manualmente por especialistas de acordo com os pesquisadores.

4.2 Pré-processamento

Apesar de considerar que a etapa de extração foi realizada por outros sistemas de OIE, para realização dos experimentos foi necessário refazer os passos iniciais da extração referentes às tarefas de NLP empregadas. Para isso foi utilizado o OpenNLP[3] que permite a extração das *features*, principalmente as do ReVerb considerando que estas utilizam informação de POS. Observando a Tabela II é possível verificar que a *feature* 6 informa qual a distancia (em número de palavras) entre e_1 e a frase relacional. Para isso é necessário que se indique onde começa e onde termina cada frase nominal e frase relacional

[2]https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/software/clausie/
[3]https://opennlp.apache.org

dentro das sentenças. Por exemplo, na frase "George believes in God", a frase relacional começa na posição 2 e termina na posição 3 ([George=1][believes=2][in=3][God=4]). Outras *features* necessitam de informações adicionais, como por exemplo as *features* 12 e 13, as quais usam etiqueta POS das entidades nominais extraídas. De acordo com a etiqueta, é possível saber se a palavra corresponde a um nome próprio ou não. Por esse motivo é necessário adicionar metadados com a etiquetação POS à base de dados antes da extração das *features*.

Algumas ferramentas utilizadas na extração das relações (e.g. OLLIE e ClausIE) que compõem os bancos de dados descritos na Subseção 4.1 criam relações denominadas "sintéticas". Essas extrações contém palavras necessárias para que a relação extraída seja coerente, mas que não estão presentes na sentença. Por exemplo, a partir da sentença "US president Obama" a relação (Obama, president of, US) é extraída pelo OLLIE. A frase relacional "president of" contém a palavra (*of*) que não está presente na sentença original. Isso torna impossível extrair a *feature* 1 (Tabela II) simplesmente porque a relação possui mais palavras do que a própria sentença. Por esta razão, relações sintéticas, ou seja, que apresentam palavras que não estão presentes nas sentenças, foram removidas do conjunto de dados. Após a etapa de pré-processamento, das 1769 extrações do Wiki-200, restaram 680 extrações que não possuíam relações sintéticas, enquanto que das 2094 extrações do NYT-200, 743 permaneceram no arquivo final e considerando o ReVerb-500 restaram 2918.

4.3 Fluxo da experimentação

A Figura 1 apresenta a descrição geral das etapas propostas em nossa metodologia experimental. Avaliando a capacidade preditiva desta proposta, a primeira etapa da experimentação foi responsável por comparar os conjuntos de *features* através de validação cruzada (10-fold *cross-validation*) utilizando todas as extrações feitas em NYT-200 e Wiki-200. Na segunda etapa foram verificadas as mesmas medidas de Acurácia, Precisão, Revocação e Medida-F (F1) através de *holdout* treinando as *features* com as extrações feitas em Wiki-200 e testando com NYT-200 (Wiki-200 → NYT-200) e em seguida o caminho inverso (NYT-200 → Wiki-200). Em uma terceira combinação, o treinamento foi realizado com Wiki-200 + NYT-200 e o teste foi feito utilizando ReVerb-500. Apesar das proporções feitas aqui no método *holdout* não apresentarem exatidão (diante das diferentes quantidades de extrações resultantes de cada conjunto de dados), foi considerado que isolar as extrações feitas em domínios diferentes, melhor generaliza a avaliação realizada. Considerando a avaliação para verificar o comportamento dos modelos, as bases de dados NYT-200 e ReVerb-500 foram utilizadas como conjunto de treinamento variante e a Wiki-200 como conjunto fixo de teste. A medida utilizada nesta etapa foi a Acurácia calculada através de média aritmética de 20 execuções diferentes para cada modelo com a mesma quantidade de extrações de treinamento.

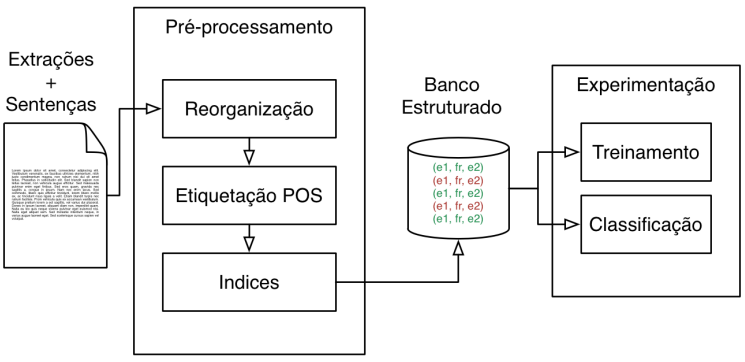


Fig. 1: Fluxograma de execução do método proposto

Para execução da experimentação foi utilizada a ferramenta Scikit[4] que disponibiliza uma implementação de alguns dos algoritmos de Aprendizado de Máquina (ML, do Inglês *Machine Learning*) mais populares nos dias atuais. Na tentativa de apresentar maior generalização, as *features* com diferentes algoritmos foram testadas para observar o comportamento em diferentes abordagens de ML.

5. RESULTADOS

O primeiro experimento mostra que nossa proposta apresenta resultado promissor para todas as dimensões testadas, exceto pela Revocação que apresentou uma pequena superioridade à proposta em [Fader et al. 2011] no conjunto de dados testado. Considerando a medida harmônica entre Precisão e Revocação (F1) observou-se que nossa proposta teve desempenho semelhante ao outro conjunto de *features* proposto no estado-da-arte. Além disso, o algoritmo *Logit*, representante da Regressão Logística, domina os resultados nesta primeira etapa do experimento com ligeira vantagem.

Table III: Comparação entre as features utilizando validação cruzada

	Algoritmo	Acurácia	Precisão	Revocação	F1
Features Independentes	Logit	0.689 ± 0.042	0.707 ± 0.034	0.851 ± 0.060	0.772 ± 0.033
	SVM	0.685 ± 0.025	0.698 ± 0.019	0.864 ± 0.053	0.771 ± 0.022
	C5.0	0.648 ± 0.049	0.727 ± 0.038	0.701 ± 0.062	0.718 ± 0.039
Features ReVerb	Logit	0.653 ± 0.037	0.672 ± 0.027	0.853 ± 0.036	0.752 ± 0.026
	SVM	0.643 ± 0.023	0.639 ± 0.014	0.967 ± 0.017	0.770 ± 0.013
	C5.0	0.607 ± 0.036	0.659 ± 0.025	0.743 ± 0.039	0.698 ± 0.025

5.1 Treinamento com Wiki-200 e teste em NYT-200

A segunda comparação observa o comportamento do sistema através do treinamento utilizando as extrações feitas em Wiki-200 e o teste do sistema através das extrações feitas em NYT-200. É observado uma queda nos resultados de Precisão em todas os algoritmos e conjuntos de *features* testados. Isso se dá pela menor quantidade de extrações utilizadas no treinamento, porém ainda é possível observar comportamento similar entre os modelos testados.

Table IV: Treinamento com Wiki-200 e teste em NYT-200

	Algoritmo	Acurácia	Precisão	Revocação	F1
Features Independentes	Logit	0.664	0.692	0.819	0.750
	SVM	0.646	0.664	0.860	0.750
	C5.0	0.576	0.667	0.622	0.644
Features ReVerb	Logit	0.626	0.657	0.823	0.731
	SVM	0.630	0.628	0.983	0.766
	C5.0	0.585	0.640	0.747	0.690

5.2 Treinamento com NYT-200 e teste em Wiki-200

A terceira comparação da experimentação mantém o mesmo comportamento da segunda etapa: queda dos resultados de Precisão, mantendo as *features* independente de idioma ainda com resultados compatíveis com a proposta do estado-da-arte comparado. O modelo gerado pelo algoritmo *Logit* ainda apresenta melhor resultado de Precisão e de Medida-F que os demais modelos de classificação testados considerando esta configuração do experimento.

[4]http://scikit-learn.org/

Table V: Treinamento com NYT-200 e teste em Wiki-200

	Algoritmo	Acurácia	Precisão	Revocação	F1
Features Independentes	Logit	0.722	0.748	0.879	0.808
	SVM	0.685	0.705	0.907	0.793
	C5.0	0.575	0.694	0.647	0.670
Features ReVerb	Logit	0.659	0.699	0.857	0.770
	SVM	0.674	0.672	0.996	0.802
	C5.0	0.622	0.705	0.744	0.724

5.3 Treinamento com NYT-200 + Wiki-200 e teste em Reverb-500

Completando a segunda etapa de experimentação testamos mais uma combinação dos conjuntos de dados com objetivo de melhor generalizar os resultados. Nesta comparação, foi avaliada a combinação dos conjuntos Wiki-200 junto com NYT-200 para realizar o treinamento do modelo e o teste foi realizado com o ReVerb-500. Há uma pequena queda dos resultados, mesmo considerando um pequeno incremento no conjunto de treinamento. Porém, observa-se que a nossa proposta mantém os resultados similares para todas as medidas utilizadas com relação as *features* de [Fader et al. 2011] exceto para Revocação com o algoritmo SVM.

Table VI: Treinamento com Wiki-200+NYT-200 e teste em Reverb-500

	Algoritmo	Acurácia	Precisão	Revocação	F1
Features Independentes	Logit	0.607	0.586	0.759	0.661
	SVM	0.561	0.546	0.774	0.640
	C5.0	0.550	0.549	0.614	0.580
Features ReVerb	Logit	0.521	0.516	0.862	0.645
	SVM	0.502	0.504	0.955	0.659
	C5.0	0.520	0.517	0.784	0.623

5.4 Comportamento na fase de treinamento

Ao observar a queda de Precisão dos modelos testados no segunda etapa de avaliação, apresentados nas subseções 5.1, 5.2 e 5.3, foi necessário observar o comportamento do melhor modelo de classificação para esta tarefa (*Logit*) em termos da quantidade de exemplos de treinamento para o modelo. Assim, a Figura 2 descreve o comportamento de cada um dos modelos à medida que cresce a quantidade de extrações usadas para o treinamento.

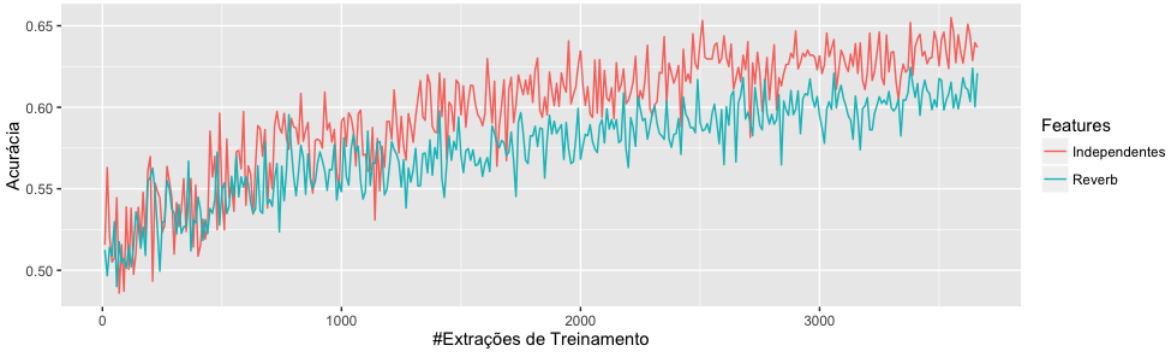


Fig. 2: Comportamento do treinamento utilizando regressão logística para os dois conjuntos de features testados.

É possível observar que mesmo com um outro conjunto de dados adicionado, o modelo preditivo proposto com *features* independentes de idioma permanece com resultados similares ao estado do

arte. O comportamento observado é de que a medida que o volume de dados de treinamento cresce, o modelo apresenta melhores resultados. Esse comportamento é similar ao apresentado em [Fader et al. 2011] que realiza o treinamento de seu modelo com milhares de exemplos.

6. CONCLUSÃO E TRABALHOS FUTUROS

Grande parte das soluções atuais realizam OIE exclusivamente para o Inglês. Esse idioma possui ferramentas linguísticas mais sofisticadas, como por exemplo: etiquetadores morfossintáticos, analisadores de FN(Frases Nominais), além de analisadores de árvores de dependência e por fim, grandes conjutos de dados de treino. Infelizmente, muitas dessas ferramentas não estão disponíveis para outros idiomas. Esse fato eleva a necessidade de desenvolver métodos que forneçam essa independência de idiomas. Neste trabalho foi proposto um conjunto de *features* que não utilizam funções específicas de uma língua, o que pode viabilizar seu uso para textos em outros idiomas. As *features* propostas e avaliadas através da nossa metodologia experimental, evidenciaram que os resultados obtidos são promissores para a avanço na independência do idioma na OIE.

Como trabalhos futuros espera-se avaliar os métodos apresentados em outros idiomas além do Inglês. A expectativa é que a utilização dessa metodologia apresente resultados similares aos encontrados na literatura, com a vantagem da independência de idioma. Entretanto, para realização desses novos experimentos há várias limitações das quais destacam-se: (i) desenvolvimento de um novo conjunto de dados para treinamento e testes, (ii) a criação de algoritmos para realização das extrações em diferentes idiomas e (iii) esforço manual para a avaliação dos resultados das etapas de extração e classificação.

REFERENCES

Angeli, G., Premkumar, M. J., and Manning, C. D. Leveraging linguistic structure for open domain information extraction. *Linguistics* (1/24, 2015.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. Open information extraction for the web. In *IJCAI*. Vol. 7. pp. 2670–2676, 2007.

Banko, M., Etzioni, O., and Center, T. The tradeoffs between open and traditional relation extraction. In *ACL*. Vol. 8. pp. 28–36, 2008.

Del Corro, L. and Gemulla, R. Clausie: clause-based open information extraction. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, pp. 355–366, 2013.

Fader, A., Soderland, S., and Etzioni, O. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 1535–1545, 2011.

Gamallo, P., Garcia, M., and Fernández-Lanza, S. Dependency-based open information extraction. In *Proceedings of the joint workshop on unsupervised and semi-supervised learning in NLP*. Association for Computational Linguistics, pp. 10–18, 2012.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*. pp. 55–60, 2014.

Mesquita, F., Schmidek, J., and Barbosa, D. Effectiveness and efficiency of open relation extraction. *New York Times* vol. 500, pp. 150, 2013.

Pereira, V. and Pinheiro, V. Report-um sistema de extração de informações aberta para língua portuguesa. In *Proceedings of Symposium in Information and Human Language Technology*. Sociedade Brasileira de Computação, pp. 191–200, 2015.

Schmitz, M., Bart, R., Soderland, S., Etzioni, O., et al. Open language learning for information extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, pp. 523–534, 2012.

Wu, F. and Weld, D. S. Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, pp. 118–127, 2010.

Xu, Y., Kim, M.-Y., Quinn, K., Goebel, R., and Barbosa, D. Open information extraction with tree kernels. In *HLT-NAACL*. pp. 868–877, 2013.

Um Método de Discretização Supervisionado para o Contexto de Classificação Hierárquica

Valter Hugo Guandaline, Luiz Henrique de Campos Merschmann

Universidade Federal de Ouro Preto, Brasil
vhguandaline@gmail.com,luizhenrique@iceb.ufop.br

Abstract. A discretização é uma das etapas do pré-processamento de dados que tem sido objeto de pesquisas em diversos trabalhos relacionados com classificação plana. Apesar da importância da discretização de dados para a tarefa de classificação, até onde se tem conhecimento, para o cenário de classificação hierárquica, onde as classes a serem preditas estão organizadas de acordo com uma hierarquia, não existem na literatura métodos de discretização que levem em consideração a hierarquia de classes. O desenvolvimento de métodos de discretização capazes de lidar com a hierarquia de classes é de fundamental importância para viabilizar a utilização de classificadores hierárquicos globais que necessitem de dados discretizados. Portanto, neste trabalho, preenchemos essa lacuna propondo e avaliando um método de discretização supervisionado para o contexto de classificação hierárquica. Experimentos realizados com nove bases de dados de bioinformática utilizando um classificador hierárquico global mostraram que o método proposto permitiu ao classificador alcançar desempenho preditivo superior àqueles obtidos quando outros métodos de discretização não supervisionados foram utilizados.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications; I.2.6 [**Artificial Intelligence**]: Learning

Keywords: Discretization, hierarchical classification, CAIM.

1. INTRODUÇÃO

A mineração de dados é apenas uma das etapas de um processo maior denominado Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Database – KDD*), que também inclui o pré-processamento dos dados e o pós-processamento da informação minerada [Fayyad et al. 1996].

O principal objetivo da etapa de pré-processamento é preparar o conjunto de dados para que ele possa ser utilizado por alguma técnica de mineração de dados. Um dos processos que podem ser realizados nesta etapa é a discretização. O seu objetivo é transformar atributos contínuos em atributos discretos. Essa transformação é feita associando intervalos de valores contínuos à novos valores categóricos. Assim, os métodos de discretização reduzem e simplificam os dados, tornando o aprendizado mais rápido e os resultados mais compactos [Garcia et al. 2013].

A classificação é uma das principais tarefas de mineração de dados. O seu objetivo é, a partir de uma base de dados contendo instâncias com características e classes conhecidas, gerar modelos capazes de predizer a classe de novas instâncias a partir de suas características. A maioria dos problemas de classificação abordados na literatura são considerados problemas de classificação plana, onde as classes não possuem relacionamentos entre si. No entanto, existem problemas de classificação mais complexos, conhecidos como problemas de classificação hierárquica, onde as classes a serem preditas estão estruturadas de acordo com uma hierarquia [Freitas and de Carvalho 2007].

Apesar de as aplicações do mundo real geralmente envolverem atributos contínuos, alguns algo-

ritmos de classificação lidam somente com atributos discretos. Além disso, para alguns métodos de classificação, ainda que sejam capazes de lidar com os atributos contínuos, o seu desempenho preditivo melhora quando os atributos contínuos são previamente discretizados [Kurgan and Cios 2004].

Até onde se tem conhecimento, não existem na literatura métodos de discretização que levem em consideração os relacionamentos entre classes existentes em problemas de classificação hierárquica. Trabalhos que abordaram problemas de classificação hierárquica e necessitaram realizar a discretização dos dados, tais como [Merschmann and Freitas 2013] e [Silla Jr and Freitas 2009], tiveram que utilizar métodos de discretização não supervisionados, uma vez que métodos não supervisionados podem ser utilizados para o contexto de classificação plana ou hierárquica.

A hipótese levantada neste trabalho é que métodos de discretização supervisionados, pelo fato de levarem em consideração o atributo classe no momento da discretização, poderiam proporcionar melhoria no desempenho preditivo de classificadores hierárquicos. Isso nos motivou a propor um método de discretização supervisionado para o contexto da classificação hierárquica.

A proposta aqui apresentada corresponde a uma adaptação realizada no método CAIM [Kurgan and Cios 2004] para torná-lo capaz de considerar a hierarquia de classes existente em problemas de classificação hierárquica. Os resultados mostram que o método proposto permitiu a um classificador hierárquico alcançar desempenho preditivo superior àqueles alcançados quando a base de dados foi pré-processada por métodos não supervisionados.

O restante deste artigo encontra-se organizado como descrito a seguir. A Seção 2 apresenta uma breve revisão da literatura sobre classificação hierárquica e discretização de dados. Em seguida, o método proposto é detalhado na Seção 3 e os experimentos computacionais com os resultados obtidos são descritos na Seção 4. Por fim, a Seção 5 apresenta as conclusões deste trabalho.

2. REFERENCIAL TEÓRICO

2.1 Classificação Hierárquica

Em um problema de classificação hierárquica, os relacionamentos entre as classes são representados por uma estrutura hierárquica, que pode ser uma árvore ou um grafo acíclico direcionado (*Directed Acyclic Graph – DAG*). A principal diferença entre essas estruturas é que, enquanto em uma árvore um nó (classe) está associado a no máximo um nó (classe) pai, em um DAG um nó pode ter mais do que um nó pai.

De acordo com [Freitas and de Carvalho 2007], os métodos de classificação hierárquica diferem em uma série de aspectos. O primeiro aspecto refere-se ao tipo de estrutura com a qual o método é capaz de lidar. No caso deste trabalho, a estrutura hierárquica das classes corresponde a uma árvore.

O segundo aspecto está relacionado à profundidade da execução da classificação na hierarquia. Um método pode realizar predições usando somente classes dos nós folha da hierarquia (*Mandatory Leaf-Node Prediction – MLNP*) ou classes referentes a qualquer nó (interno ou folha) da hierarquia (*Non-Mandatory Leaf-Node Prediction – NMLNP*). Neste trabalho, considera-se o cenário *NMLNP*.

O terceiro aspecto está relacionado ao número de classes (ramos da estrutura hierárquica) que um método é capaz de atribuir a uma instância. Um método pode ser capaz de predizer múltiplas classes para uma determinada instância (multirrótulo), desse modo envolvendo múltiplos ramos da hierarquia de classes (*multiple paths of labels*), ou somente uma classe (monorrótulo), a qual estará vinculada a um único ramo da hierarquia de classes (*single path of labels*). O método proposto neste trabalho lida com a classificação monorrótulo.

Por fim, o quarto aspecto está relacionado ao tipo de abordagem que o classificador utiliza para explorar a estrutura hierárquica. Segundo [Silla Jr and Freitas 2011] existem três tipos de abordagens: (i) abordagem por classificação plana, na qual a hierarquia de classes é ignorada e as predições são

realizadas considerando-se somente as classes dos nós folha da estrutura hierárquica; (ii) abordagem local, onde são utilizados diversos classificadores planos tradicionais, cada um com uma visão local da estrutura hierárquica; (iii) abordagem global, onde um único modelo de classificação é construído levando em consideração toda a hierarquia de classes de uma só vez. O método de discretização proposto neste trabalho tem como objetivo adequar as bases de dados para serem utilizadas por classificadores hierárquicos globais, dado que para a abordagem local, podem ser utilizados métodos de discretização supervisionados projetados para o cenário de classificação plana.

2.2 Discretização de Dados

Discretização é uma estratégia de redução de dados amplamente utilizada na etapa de pré-processamento dos dados [Garcia et al. 2013]. O processo de discretização transforma atributos contínuos em atributos discretos dividindo o atributo contínuo em intervalos de valores e associando cada um desses intervalos a um valor discreto.

De acordo com [Garcia et al. 2013], os métodos de discretização podem ser categorizados como supervisionados ou não supervisionados. O método é denominado supervisionado quando ele leva em consideração os valores do atributo classe. Por outro lado, se o atributo classe não é considerado no processo de discretização, o método é dito não supervisionado.

Diferentes critérios podem ser usados para avaliar os algoritmos de discretização, tais como o número de intervalos gerados, o nível de inconsistência e a taxa de acerto de classificadores. Neste trabalho, os métodos de discretização foram avaliados a partir do método de classificação hierárquica global denominado *Global Model Naive Bayes – GMNB*, proposto em [Silla Jr and Freitas 2009].

Em [Garcia et al. 2013] os autores avaliaram 30 discretizadores sobre 40 bases de dados utilizando 6 classificadores planos. Essa avaliação mostrou que o CAIM foi um dos métodos de discretização mais eficientes. Por isso, esse foi o método escolhido neste trabalho para ser adaptado para o contexto hierárquico. Além disso, dada a inexistência de métodos supervisionados para o contexto hierárquico, os métodos não supervisionados *EqualWidth* e *EqualFrequency* (adotado em [Merschmann and Freitas 2013] e [Silla Jr and Freitas 2009]) foram utilizados como base de comparação com o método aqui proposto. A seguir, serão apresentados mais detalhes do método de discretização (CAIM) que foi adaptado para o contexto de classificação hierárquica.

2.2.1 *CAIM. Class-Attribute Interdependency Maximization*

é um método de discretização supervisionado que independe de outros métodos de aprendizagem. Ele utiliza uma métrica para avaliar a interdependência entre o atributo classe e o atributo em processo de discretização [Kurgan and Cios 2004].

Considere uma base de dados com um conjunto de instâncias M, um conjunto de atributos numéricos F e atributo classe S, onde $|M|$, $|F|$ e $|S|$ são, respectivamente, o número de instâncias, número de atributos e o número de classes. Além disso, cada instância M_k está associada à uma classe S_i, onde $k \in \{1, 2, ..., |M|\}$ e $i \in \{1, 2, \ldots, |S|\}$.

Para cada atributo contínuo F_j, $j \in \{1, 2, \ldots, |F|\}$, o CAIM ordena os seus valores em ordem crescente e, em seguida, divide-os em n intervalos da seguinte forma: $D = \{[d_0, d_1], (d_1, d_2], \ldots, (d_{n-1}, d_n]\}$, onde d_0 e d_n são, respectivamente, os valores mínimo e máximo do atributo F_j e $d_i < d_{i+1}$ para $i \in \{0, 1, \ldots, n-1\}$. Cada par de valores (d_i, d_{i+1}) define um intervalo do atributo F_j, sendo que o resultado da discretização D, chamado de esquema de discretização do atributo F_j, define o seguinte conjunto de pontos de corte $P = \{d_1, d_2, \ldots, d_{n-1}\}$.

A interdependência entre o atributo classe S e um esquema de discretização D de um atributo F_j é calculada utilizando-se a métrica CAIM (Equação 1), que faz uso de uma matriz de frequência denominada matriz de contingência, apresentada na Figura 1. Nessa matriz, considerando-se um esquema de discretização $D = \{[d_0, d_1], (d_1, d_2], \ldots, (d_{n-1}, d_n]\}$ do atributo em processo de discretização, q_{ir}

é a quantidade de instâncias pertencentes à i-ésima classe que estão contidas no r-ésimo intervalo, M_{i+} é a quantidade total de instâncias pertencentes à i-ésima classe e M_{+r} é a quantidade total de instâncias contidas no r-ésimo intervalo.

$$CAIM(S, D|F_j) = \frac{\sum_{r=1}^{n} \frac{max_r^2}{M_{+r}}}{n}, \quad (1)$$

onde n é o número de intervalos e max_r é o número máximo de instâncias contidas no intervalo r pertencentes a uma mesma classe. Essa equação é utilizada pelo método CAIM para se escolher o melhor ponto de corte a ser inserido em um determinado esquema de discretização. Quanto maior o valor retornado por essa métrica, maior é dependência entre o atributo F_j (discretizado segundo o esquema D) e o atributo classe S.

Classes	Intervalos					Instâncias por Classe
	$[d_0, d_1]$	...	$(d_{r-1}, d_r]$	...	$(d_{n-1}, d_n]$	
C_1	q_{11}	...	q_{1r}	...	q_{1n}	M_{1+}
...	...	...	...	...	...	...
C_i	q_{i1}	...	q_{ir}	...	q_{in}	M_{i+}
...	...	...	...	...	...	...
C_s	q_{s1}	...	q_{sr}	...	q_{sn}	M_{s+}
Instâncias por Intervalo	M_{+1}	...	M_{+r}	...	M_{+n}	M

Fig. 1. Matriz de contingência para o atributo F_j e esquema de discretização D

3. MÉTODO PROPOSTO

O principal problema na utilização de métodos de discretização supervisionados tradicionais (utilizados em conjunto com classificadores planos) para bases de dados relacionadas com o contexto de classificação hierárquica está no fato de esses discretizadores não serem capazes de considerar as informações dos relacionamentos entre as classes do problema. Neste trabalho, parte-se da hipótese de que esse tipo de informação, se considerada ao longo do processo de discretização, pode contribuir na geração de uma base de dados discretizada de melhor qualidade para a tarefa de classificação.

Portanto, o método de discretização aqui proposto, denominado HCAIM (Hierarchical CAIM), considera a hierarquia de classes enquanto realiza o processo de discretização. O HCAIM corresponde a uma adaptação do método de discretização CAIM para o contexto hierárquico, cuja principal diferença está na métrica de avaliação utilizada pelo método para a definição dos pontos de corte de um esquema de discretização.

3.1 Métrica de avaliação

Para avaliar um esquema de discretização $D = \{[d_0, d_1], (d_1, d_2], \ldots, (d_{n-1}, d_n]\}$ para um atributo F_j, o método CAIM verifica o quão bons são os intervalos contidos nesse esquema. Por meio da métrica também denominada CAIM (ver Equação 1), cada intervalo contido em D é avaliado medindo-se a correlação entre os valores do atributo F_j existentes naquele intervalo e as classes nele contidas. Essa correlação é dada por $(\frac{max_r^2}{M_{r+}})$, onde max_r é o número de ocorrências da classe mais frequente no intervalo r e M_{r+} é a quantidade de instâncias contidas nesse mesmo intervalo. Esse cálculo permite ao método CAIM: (i) considerar o grau de pureza do intervalo (quanto mais próximo max_r for de M_{r+}, mais puro é o intervalo) e (ii) priorizar intervalos com maior número de instâncias.

Entretanto, essa métrica CAIM não leva em consideração a hierarquia de classes existente em um problema onde as classes estão hierarquicamente organizadas. Por exemplo, dado um intervalo r contendo 9 instâncias ($M_{r+} = 9$), sendo 3 instâncias da classe *R.2* e 6 da classe *R.2.1*, a avaliação

desse intervalo segundo a métrica CAIM é dada por $6^2/9 = 4$, uma vez que a classe majoritária $R.2.1$ é considerada como completamente distinta da $R.2$. No entanto, no contexto hierárquico, instâncias da classe *R.2.1* também pertencem à classe *R.2*, já que $R.2.1$ é uma classe filha da $R.2$.

Portanto, neste trabalho, a métrica CAIM foi adaptada para calcular o grau de pureza de cada intervalo considerando a hierarquia de classes. Na adaptação proposta, denominada HCAIM (*Hierarchical CAIM*), o cálculo do grau de pureza de um intervalo é realizado para cada nível hierárquico, sendo o seu valor final uma média ponderada dos valores calculados para cada um dos níveis. Desse modo, a dependência entre o atributo classe S e o esquema de discretização D para um dado atributo F_j levando em consideração a hierarquia de classes é dada por:

$$HCAIM(S, D|F_j) = \frac{\sum_{r=1}^{n} \sum_{l=1}^{H_r} \frac{max_{r,l}^2}{M_{+r}} . W_{l,r}}{n}, \quad (2)$$

onde n é o número de intervalos, H_r é a profundidade da hierarquia de classes referente ao intervalo r, $max_{r,l}$ é o número de ocorrências da classe mais frequente no intervalo r considerando-se a hierarquia de classes até o nível l, M_{+r} é o total de instâncias contidas no intervalo r e $W_{l,r}$ é peso associado ao nível l da hierarquia de classes referente ao intervalo r.

No cálculo da métrica HCAIM para um determinado intervalo r, além das matrizes de contingência para cada nível hierárquico, há necessidade de se calcular os pesos $W_{l,r}$ que serão aplicados de acordo com o nível hierárquico l e a profundidade da hierarquia naquele intervalo H_r. O valor do peso para cada um dos níveis hierárquicos é dado por $W(l, r) = (H_r - l + 1)\frac{2}{H_r \times (H_r+1)}$, sendo que $\sum_{l=1}^{H_r} W_{l,r} = 1$.

Retomando o exemplo anterior, onde consideramos um único intervalo r contendo 9 instâncias ($M_{r+} = 9$), sendo 3 instâncias da classe *R.2* e 6 da classe *R.2.1*, a avaliação desse intervalo segundo a métrica HCAIM é dada por $9^2/9 \times W_{1,r} + 6^2/9 \times W_{2,r}$. A primeira parcela ($9^2/9$) deve-se ao fato de considerarmos todas as classes presentes no intervalo r somente até o primeiro nível da hierarquia, ou seja, todas as instância estão associadas à classe $R.2$. Já a segunda parcela ($6^2/9$) é calculada considerando-se todas as classes até o segundo nível hierárquico, onde passamos a ter 3 instâncias da classe *R.2* e 6 da classe *R.2.1*. Considerando que os pesos associados aos níveis hierárquicos são $W_{1,r} = 2/3$ e $W_{2,r} = 1/3$, o valor final da métrica para o exemplo em questão é $HCAIM = 7,33$.

3.2 Algoritmo HCAIM

O algoritmo HCAIM pode ser divido em três etapas: Inicialização, Avaliação e Verificação. Essas etapas são aplicadas a cada atributo contínuo F_j. A seguir, tem-se o detalhamento de cada uma delas.

Inicialização: a primeira inicialização é a do conjunto dos possíveis pontos de corte B. Considerando-se que o atributo a ser discretizado F_j encontra-se ordenado, os pontos de corte inseridos no conjunto B correspondem à média dos valores do atributo F_j para cada par de instâncias vizinhas que encontram-se associadas a classes diferentes e possuem valores distintos para o atributo em questão. Em seguida, o esquema de discretização D é inicializado com um único intervalo, $D = \{[-\infty, +\infty]\}$. Por fim, as variáveis $GlobalHCaim$ (armazena os melhores valores da métrica HCAIM ao longo do processo de discretização) e k (número de intervalos no esquema D) são inicializadas com os valores 0 e 1, respectivamente. Após isso, o método passa para a etapa seguinte (etapa de avaliação).

Avaliação: é uma etapa iterativa que consiste em avaliar todos os pontos de corte contidos em B enquanto o critério de parada não for satisfeito. Para cada possível ponto de corte p do conjunto B, o método cria um novo esquema de discretização D' inserindo o ponto de corte p no esquema de discretização D. Em seguida, o esquema D' é avaliado pela métrica HCAIM(S, $D'|F_j$). Após avaliar todos os possíveis pontos de corte contidos em B, o método armazena o ponto de corte $p*$ que obteve o maior valor para o critério de avaliação HCAIM. Essa informação é utilizada na etapa seguinte (etapa de verificação).

Verificação: nessa etapa é realizada a verificação do critério de parada. O algoritmo encerra a sua execução quando as duas condições a seguir são falsas: i) se o número de intervalos gerados até o momento (k) é menor do que o número de classes ($|S|$); ii) se o valor da métrica HCAIM para o ponto de corte $p*$ é maior do que o obtido na iteração anterior ($GlobalHCaim$). Caso o contrário, o algoritmo remove o ponto de corte $p*$ do conjunto B e adiciona-o ao esquema D, incrementa o número de intervalos criados ($k = k + 1$), atualiza o valor da métrica HCAIM para o esquema D ($GlobalHCaim = HCAIM$) e, por fim, volta para a etapa de avaliação, onde a inserção de novos pontos de corte será avaliada.

4. EXPERIMENTOS COMPUTACIONAIS

4.1 Bases de Dados

Todos os experimentos foram conduzidos a partir de nove bases de dados relacionadas com a classificação de funções de genes. Nessas bases, os atributos preditores incluem diversos tipos de dados da área de bioinformática, tais como: estrutura secundária da sequência, fenótipo, homologia, estatísticas da sequência e outros. Essas bases de dados, inicialmente utilizadas em [Clare and King 2003], são multirrótulo. Como o foco deste trabalho é a classificação hierárquica monorrótulo, essas bases de dados foram transformadas para o contexto monorrótulo selecionando-se, para cada instância, a classe mais frequente na base de dados original.

A partir das bases de dados monorrótulo, o pré-processamento descrito a seguir foi realizado para substituição dos valores ausentes de atributos. Ao identificar um valor ausente para um determinado atributo F_j de uma instância associada à classe C_i, calcula-se a média dos valores conhecidos do atributo F_j de todas as demais instâncias da base associadas à classe C_i e, em seguida, utiliza-se essa média para substituir o valor ausente. Se para a classe C_i nenhuma instância possuir valor conhecido para o atributo F_j, calcula-se a média dos valores conhecidos do atributo F_j de todas as instâncias da base associadas a classes descendentes de C_i na hierarquia e, em seguida, utiliza-se essa média para substituição do valor ausente. Em último caso, se a classe C_i não possuir classes descendentes ou se para as classes descendentes de C_i nenhuma instância possuir valor conhecido para o atributo F_j, então substitui-se o valor ausente utilizando-se a média global do atributo F_j.

A Tabela I mostra as principais características das bases de dados após o pré-processamento. Essa tabela apresenta, para cada base de dados, o seu número de instâncias, de atributos preditores, de classes e a distribuição das mesmas pelos níveis da hierarquia (1°|2°|3°|4°|5°|6°).

4.2 Configuração Experimental

Os métodos de discretização não supervisionados *EqualFrequency* e *EqualWidth* foram utilizados como referência para comparação com método proposto, o HCAIM. Esses métodos foram escolhidos para

Table I. Características das bases de dados

Bases	# Instâncias	# Atributos Contínuos / Categóricos	# Classes	# Classes por Nível
Church	3755	26 / 1	190	7|37|72|47|25|2
Eisen	2424	79 / 0	143	4|26|55|34|22|2
Cellcycle	3757	77 / 0	190	7|37|73|46|25|2
Gasch2	3779	52 / 0	191	7|37|73|46|26|2
Gasch1	3764	173 / 0	191	7|37|73|46|26|2
Derisi	3725	63 / 0	190	7|37|72|47|25|2
Spo	3703	77 / 3	191	7|37|73|46|26|2
Seq	3919	473 / 5	192	7|37|73|47|26|2
Expr	3779	547 / 4	191	7|37|72|47|26|2

as comparações pelo fato de já terem sido adotados em trabalhos de classificação hierárquica, uma vez que não há métodos de discretização supervisionados para esse contexto.

Os métodos *EqualFrequency* e *EqualWidth* possuem um parâmetro k, que define o número de intervalos a serem criados no processo de discretização. Os experimentos foram executados para diferentes valores de k, a saber, 5, 10, 15 e 20. Esses métodos foram executados a partir das suas implementações disponíveis na ferramenta *WEKA* [Hall et al. 2009].

Para avaliar a qualidade da discretização realizada por cada um dos métodos foi utilizado o classificador hierárquico *Global Model Naive Bayes – GMNB* [Silla Jr and Freitas 2009]. Para expressar o desempenho preditivo do classificador hierárquico *GMNB* adotou-se a métrica *F-measure* hierárquica (hF) posposta em [Kiritchenko et al. 2005]. Além disso, o método k-validação cruzada (k=10) foi utilizado na avaliação do desempenho do *GMNB*. Vale ressaltar também que a discretização dos dados ocorreu somente após o particionamento da base pelo método 10-validação cruzada, ou seja, para cada base, ela foi aplicada considerando-se cada uma das 10 partições de treinamento.

4.3 Resultados

A Tabela II apresenta o hF médio (com desvio padrão entre parênteses) obtido pelo classificador *GMNB* para cada base de dados discretizada utilizando-se o HCAIM e os outros dois métodos utilizados como referência, a saber, *EqualFrequency* (EF) e *EqualWidth* (EW). No caso dos métodos de discretização EF e EW, o nome da coluna é formado pelo nome do método acrescido, entre parênteses, do valor do parâmetro k utilizado. Para cada base de dados, com objetivo de verificar se há diferença com significância estatística entre os desempenhos (hF) do classificador *GMNB* ao processar a base de dados discretizada pelo HCAIM e por outro método de referência, utilizou-se o teste estatístico de Wilcoxon com a correção de Bonferroni devido às múltipas comparações entre o HCAIM e cada método de referência [Japkowicz and Shah 2011]. Esse teste estatístico foi executado com nível de confiança de 95%. Os valores em negrito indicam o melhor resultado obtido para cada base de dados. Além disso, o símbolo • indica que há diferença com significância estatística entre o método de referência em questão e o HCAIM. Por fim, a última linha dessa tabela resume o resultado do teste estatístico, ou seja, para cada método de referência, mostra-se o número de vezes em que o HCAIM o superou apresentando um melhor desempenho preditivo do classificador *GMNB*.

Os resultados apresentados na Tabela II mostram que o método de discretização proposto neste trabalho (HCAIM) proporcionou o maior desempenho preditivo ao GMNB para 6 das 9 bases de dados utilizadas nos experimentos (valores em negrito). Além disso, os testes estatísticos revelam que,

Table II. Valores médios de hF obtidos pelo GMNB após discretização das bases.

Base	EF(5)	EF(10)	EF(15)	EF(20)	EW(5)	EW(10)	EW(15)	EW(20)	HCAIM
Cellcycle	20.83 • (1.39)	24.56 • (2.69)	26.03 • (2.21)	26.56 • (2.13)	15.41 • (1.75)	17.08 • (1.37)	18.96 • (1.63)	19.10 • (2.03)	**31.85** (**1.69**)
Church	10.07 • (1.25)	11.57 • (1.31)	12.00 • (1.63)	13.14 • (1.45)	10.90 • (1.01)	11.85 • (1.51)	11.76 • (1.41)	12.63 • (1.58)	**18.63** (**1.13**)
Derisi	9.36 • (1.00)	10.98 (1.20)	11.73 (1.07)	11.52 (1.07)	8.92 • (0.75)	9.31 • (1.28)	9.69 (1.30)	9.91 • 1.14)	**12.42** (**0.82**)
Eisen	**22.56** (**1.33**)	22.52 (2.10)	21.86 (2.23)	21.78 (1.70)	20.74 (2.35)	22.17 (1.42)	22.40 (1.27)	22.49 (1.97)	21.28 (1.43)
Expr	43.91 • (1.39)	45.82 (1.88)	45.67 (2.35)	45.54 (2.49)	26.82 • (1.69)	29.62 • (1.88)	32.60• (1.49)	34.46 • (1.27)	**46.41** (**1.66**)
Gasch1	18.64 • (1.97)	21.75 • (2.33)	22.39 • (1.51)	22.84 • (1.75)	16.80 • (1.47)	18.37 • (1.61)	19.62 • (2.27)	19.36 • (2.06)	**26.86** (**1.16**)
Gasch2	16.48 • (1.70)	17.59 • (1.45)	19.37 • (1.90)	19.69 • (1.68)	14.75 • (1.17)	16.11 • (1.47)	15.77 • (1.32)	16.61 • (1.47)	**25.51** (**1.74**)
SPO	13.62 (1.41)	14.31 (1.25)	**14.84** (**1.37**)	14.30 (0.78)	13.77 (1.43)	13.17 (1.19)	13.02 (1.63)	13.62 (0.85)	13.14 (1.73)
Seq	21.39 • (0.87)	19.58 (1.03)	18.83 (1.32)	18.75 (1.15)	24.02 • (1.67)	**24.91** • (**1.47**)	24.05 • (1.11)	24.05 • (1.26)	18.10 (1.08)
# Vitórias do HCAIM	6	4	4	4	6	6	5	6	

para quatro base de dados (Cellcycle, Church,Gasch1 e Gasch2), o HCAIM superou todos os métodos de referência utilizados na avaliação comparativa. Para apenas uma base de dados (Seq) o HCAIM obteve desempenho inferior, com significância estatística, ao de alguns métodos de referência.

Os testes estatísticos também mostram que na comparação do HCAIM com cada um dos outros métodos utilizados nos experimentos, ele apresenta um desempenho estatisticamente superior ou igual ao dos demais métodos para a maioria das bases de dados avaliadas. Por exemplo, quando comparado com o método *EqualFrequency* com $k = 5$ (EF(5)), o HCAIM é superior para 6 bases de dados e equivalente para as 3 bases restantes.

5. CONCLUSÃO

Apesar da importância dos métodos de discretização para o pré-processamento das bases de dados utilizadas por técnicas de classificação, até onde se tem conhecimento, não existem na literatura propostas de métodos de discretização supervisionados que possam ser utilizados em conjunto com classificadores hierárquicos globais. Portanto, este trabalho propôs um método de discretização supervisionado para o contexto de classificação hierárquica monorrótulo. A proposta apresentada, denominada HCAIM, corresponde a uma adaptação do método de discretização supervisionado CAIM.

Os experimentos computacionais realizados mostraram que o método HCAIM, para a maioria das bases avaliadas, permitiu ao classificador hierárquico GMNB alcançar desempenho preditivo superior àqueles alcançados quando as bases de dados foram pré-processadas pelos métodos não supervisionados *EqualWidth* e *EqualFrequency*. Esse resultado confirma o potencial de aplicação do método proposto para a realização da discretização de bases utilizadas em trabalhos de classificação hierárquica.

AGRADECIMENTOS

Os autores agradecem a UFOP, a FAPEMIG e o CNPq pelo apoio financeiro concedido.

REFERENCES

Clare, A. and King, R. D. Predicting gene function in saccharomyces cerevisiae. *Bioinformatics* 19 (suppl 2): ii42–ii49, 2003.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. From data mining to knowledge discovery in databases. *AI magazine* 17 (3): 37, 1996.

Freitas, A. A. and de Carvalho, A. C. A tutorial on hierarchical classification with applications in bioinformatics. In *D. Taniar (Ed.) Research and Trends in Data Mining Technologies and Applications*. Idea Group, pp. 175–208, 2007.

Garcia, S., Luengo, J., Sáez, J. A., López, V., and Herrera, F. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering* 25 (4): 734–750, 2013.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. The weka data mining software: An update. *SIGKDD Explorations* 11 (1), 2009.

Japkowicz, N. and Shah, M. *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press, New York, NY, USA, 2011.

Kiritchenko, S., Matwin, S., and Famili, A. F. Functional annotation of genes using hierarchical text categorization. In *Proceedings of the ACL workshop on linking biological literature, ontologies and databases: mining biological semantics*, 2005.

Kurgan, L. A. and Cios, K. J. Caim discretization algorithm. *IEEE Transactions on Knowledge and Data Engineering* 16 (2): 145–153, 2004.

Merschmann, L. H. C. and Freitas, A. A. An extended local hierarchical classifier for prediction of protein and gene functions. In *Data Warehousing and Knowledge Discovery*. pp. 159–171, 2013.

Silla Jr, C. N. and Freitas, A. A. A global-model naive bayes approach to the hierarchical prediction of protein functions. In *IEEE International Conference on Data Mining*. pp. 992–997, 2009.

Silla Jr, C. N. and Freitas, A. A. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery* 22 (1-2): 31–72, 2011.