



UNIVERSIDADE FEDERAL DE PERNAMBUCO

**CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO**

**TÉCNICAS DE APRENDIZAGEM DE MÁQUINA, APLICADAS NO
PROCESSO DE APRENDIZAGEM, EM DISCIPLINAS DE
PROGRAMAÇÃO**

FABRÍZIO BATISTA PEREIRA

Recife-PE
2018

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

**TÉCNICAS DE APRENDIZAGEM DE MÁQUINA, APLICADAS NO
PROCESSO DE APRENDIZAGEM, EM DISCIPLINAS DE
PROGRAMAÇÃO**

Trabalho apresentado ao Programa de Graduação em Ciências da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Bacharel em Ciências da Computação.

Orientadora: Patrícia Cabral de Azevedo Restelli Tedesco (pcart@cin.ufpe.br)
Aluno: Fabrício Batista Pereira (fbp@cin.ufpe.br)

Recife-PE
2018

UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

**TÉCNICAS DE APRENDIZAGEM DE MÁQUINA, APLICADAS NO
PROCESSO DE APRENDIZAGEM, EM DISCIPLINAS DE
PROGRAMAÇÃO**

Trabalho apresentado ao Programa de Graduação em
Ciência da Computação do Centro de Informática da
Universidade Federal de Pernambuco como requisito
parcial para obtenção do grau de Bacharel em Ciência da
Computação.

Aprovado em: ____/____/____

Orientadora: Patrícia Cabral de Azevedo Restelli Tedesco

Examinador:

Ricardo Bastos C. Prudêncio

PARECER -----

AGRADECIMENTOS

Aos meus pais, pelo amor, incentivo, confiança e oportunidades que depositaram em mim e aos sacrifícios realizados que possibilitaram a minha graduação. Serei eternamente grato.

À Érica Virginia e Franscisco, por todo carinho, conselhos e apoio que proporcionaram nessa jornada.

À Rosalina por seus puxões de orelha, incentivos e carinho.

À toda minha família pelo apoio e conselhos recebidos.

Aos meus amigos, que me acompanharam nessa caminhada, pelo apoio, companhia e amizade, em especial Lucas, Fernanda e Cristiano que me salvaram inúmeras vezes, das viradas noites para me ajudar e por me ensinaram muito.

À minha orientadora Patricia Tedesco pela confiança e direcionamento, que tornaram este trabalho possível.

Em especial, quero agradecer à Maria Eduarda, pelo amor, força, paciência, dedicação e companheirismo, por ser meu porto seguro e meu norte.

RESUMO

A busca de informações por meio dos sistemas computacionais, se tornou bastante eficaz com o avanço na capacidade de processamento dos computadores. Com isso, surge a motivação para elaboração de pesquisas e estudos em aplicações que usem a Inteligência Artificial, com o objetivo de facilitar o processo de ensino e aprendizagem. Os métodos tradicionais de avaliação de estudantes, muitas vezes causam no aluno um sentimento equivocado, onde a falha ou dificuldades encontradas no seu processo de aprendizagem, o leva a crer erroneamente que é incapaz de progredir. Esse sentimento acarreta em uma falta de esperança e estimula o aluno a desistir. Intrigado com esse cenário, são relatadas nesta revisão sistemática, diferentes estratégias de avaliação do aluno, utilizando técnicas de aprendizagem de máquina. Para isso foi realizada uma revisão sistemática da literatura sobre técnicas de aprendizagem de máquina usadas na avaliação de estudantes de programação com o objetivo de avaliar o processo de aprendizagem feito pelos alunos. Foi realizada uma busca nas bases: ACM Digital Libray e IEEE Xplore, gerando 8576 trabalhos selecionados, dos quais apenas 12 se encaixaram nos critérios de inclusão. Os resultados analisados nesta pesquisa nos levam a acreditar que as técnicas de aprendizagem de máquina são relevantes e que ajudam o aluno em seu processo de aprendizagem. Mostram como é possível mapear o desempenho e comportamento dos alunos, o que pode ser usado pelos educadores para melhorar as estratégias e políticas utilizadas nos cursos de Programação, diminuindo os índices de desistência dos alunos.

Palavras-chave: Aprendizagem de máquina; Avaliação do aluno; Estudantes de Programação.

ABSTRACT

The search for information through computer systems has become quite effective with advancing the processing power of computers. With this, the motivation to develop researches and studies in applications that use Artificial Intelligence appears, with the objective of facilitating the teaching and learning process. Traditional methods of student assessment often cause the student to misunderstand where the failure or difficulties encountered in their learning process leads him or her to mistakenly believe that they are incapable of progressing. This feeling leads to a lack of hope and encourages the student to give up. Intrigued by this scenario, different strategies of student assessment are reported in this systematic review using machine learning techniques. For this, a systematic review of the literature on machine learning techniques used in the evaluation of programming students was carried out in order to evaluate the learning process done by the students. A search was carried out at the bases: ACM Digital Library and IEEE Xplore, generating 8576 selected papers, of which only 12 fit the inclusion criteria. The results analyzed in this research lead us to believe that the machine learning techniques are relevant and that help the student in his learning process. They show how it is possible to map students' performance and behavior, which can be used by educators to improve the strategies and policies used in Programming courses, reducing the dropout rates of students.

Keywords: Machine learning; Student assessment; Programming Students.

SUMÁRIO

1 INTRODUÇÃO	07
1.1 Motivação	07
1.2 Objetivos	08
1.3 Estrutura do Trabalho	08
2 CONCEITOS FUNDAMENTAIS	09
2.1 Aprendizagem de Máquina	09
2.2 Avaliação de estudantes	10
3 REVISÃO SISTEMÁTICA	12
3.1 Metodologia de Pesquisa	12
3.1.1 Objetivo e Perguntas da Pesquisa	12
3.1.2 Estratégias de Busca e Fontes de Dados	12
3.1.3 Procedimento de Seleção dos Estudos	13
3.1.4 Critérios de Qualidade	14
4 APRESENTAÇÃO DOS RESULTADOS	15
4.1 Resultado do Procedimento de Busca	15
4.2 Análise e Discussão do Conteúdo dos Artigos Selecionados	17
4.3 Ameaças a Validade	22
5 CONCLUSÃO	23
REFERÊNCIAS	24
APÊNDICE	27

1 INTRODUÇÃO

1.1 Motivação

Sabe-se que é essencial a aprendizagem de programação de computadores para todas as carreiras relacionadas a Informática, sendo Programação a disciplina mais importante para a formação daqueles que terão que desenvolver softwares. Visto que a aprendizagem de programação se dá praticamente, durante todo o curso, o baixa índice de assimilação dos estudantes nas disciplinas, onde os requisitos exigem o conhecimento de programação, tem sido um entrave enfrentado em várias instituições. Alguns autores relataram que há um fracasso no ensino de programação, e um insucesso generalizado verificado na aprendizagem de programação (ROCHA, 1991; GOMES 2000).

Vale lembrar que alunos não são iguais, ou seja, tem origens, habilidades e experiências diferentes. O que pode justificar fatos de alunos da mesma classe, que são submetidos as mesmas condições de ensino, possuem resultados distintos, reforçando a necessidade do uso de técnicas variadas que permitam aumentar os resultados de ensino. No entanto, mesmo cada aluno aprendendo de forma diferente, há pontos comuns a certos grupos de alunos, o que seria chamado de estilos de aprendizagem (CARDOSO E JANDL 1998).

A personalização do ensino é uma atividade impraticável para o professor, no ensino presencial, mesmo que haja um número reduzido de estudantes na turma. Uma alternativa para esse problema, seria o uso de técnicas computacionais que, quando aplicadas a um conjunto de dados, pode gerar informações úteis para tomada de decisões pedagógicas que contribua com a aprendizagem dos alunos (DE FRANÇA E AMARAL, 2013). Portanto este trabalho visa mostrar variadas técnicas de aprendizagem em máquina, voltada para o ensino de programação.

A busca de informações por meio dos sistemas computacionais, se tornou bastante eficaz com o avanço na capacidade de processamento dos computadores. Com isso, surge a motivação para elaboração de pesquisas e estudos em aplicações que usem a Inteligência Artificial, com o objetivo de facilitar o processo de ensino e aprendizagem (PEREIRA 2005, apud GAMBI, 2013).

Embora a aprendizagem de máquina (ramo da Inteligência artificial) seja uma poderosa ferramenta para a aquisição automática de conhecimento, deve-se salientar que: não há um único algoritmo que apresente o melhor desempenho para todos os problemas. Por isso, deve-se compreender o poder e as limitações dos diversos algoritmos de aprendizagem de máquina utilizando alguma metodologia, que permita avaliar os conceitos induzidos por esses algoritmos em determinados problemas (REZENDE, 2005).

1.2 Objetivos

Nesse contexto, o principal objetivo desta pesquisa é averiguar o que existe disponível no meio acadêmico sobre técnicas de aprendizagem de máquina que possam auxiliar no processo de ensino, nas disciplinas de programação. Para isso será realizada uma revisão sistemática sobre essas técnicas de aprendizagem.

1.3 Estrutura do Trabalho

São apresentados 5 capítulos neste trabalho de graduação, incluindo este capítulo introdutório. Enquanto o capítulo 2 discorre sobre conceitos básicos os quais são necessários para o entendimento das técnicas a serem descritas. O capítulo 3 apresenta uma revisão sistemática sobre o tema, onde é descrito o protocolo criado para a aplicação da revisão e são apresentados os principais resultados obtidos. De acordo com estes resultados, foi exposto no capítulo 4, as análises sobre os estudos selecionados, e ameaças a validade do estudo. Por fim no capítulo 5, estão presentes as conclusões do trabalho.

2 CONCEITOS FUNDAMENTAIS

2.1 Aprendizagem de Máquina

A aprendizagem de máquina é um setor da Inteligência artificial que tem por finalidade o desenvolvimento de técnicas computacionais sobre o aprendizado, assim como a construção de sistemas capazes de adquirir conhecimento de modo automático. Se trata de um programa de computador que toma decisões influenciado por experiências acumuladas por meio da solução bem sucedida de problemas anteriores (REZENDE, 2005).

Os variados sistemas de aprendizado de máquina possuem características particulares e comuns que permitem sua classificação no que se refere a linguagem de descrição, modo, paradigma e forma de aprendizado utilizado. Embora a aprendizagem de máquina seja um artifício poderoso para aquisição automática de conhecimento, observa-se que não há um único algoritmo que apresente o melhor desempenho para todos os problemas. Então é imprescindível compreender o poder e limitações dos algoritmos de aprendizagem de máquina, com uso de alguma metodologia que permita avaliar os conceitos induzidos por esses algoritmos (REZENDE, 2005).

Uma forma que permite obter conclusões genéricas sobre um conjunto particular de exemplos, é a indução, que se caracteriza como raciocínio que origina um conceito específico e o generaliza. É na indução que um conceito é aprendido efetuando-se interferência indutiva sobre os exemplos apresentados. A interferência indutiva é um dos principais métodos utilizados para derivar conhecimento novo e prever eventos futuros (REZENDE, 2005).

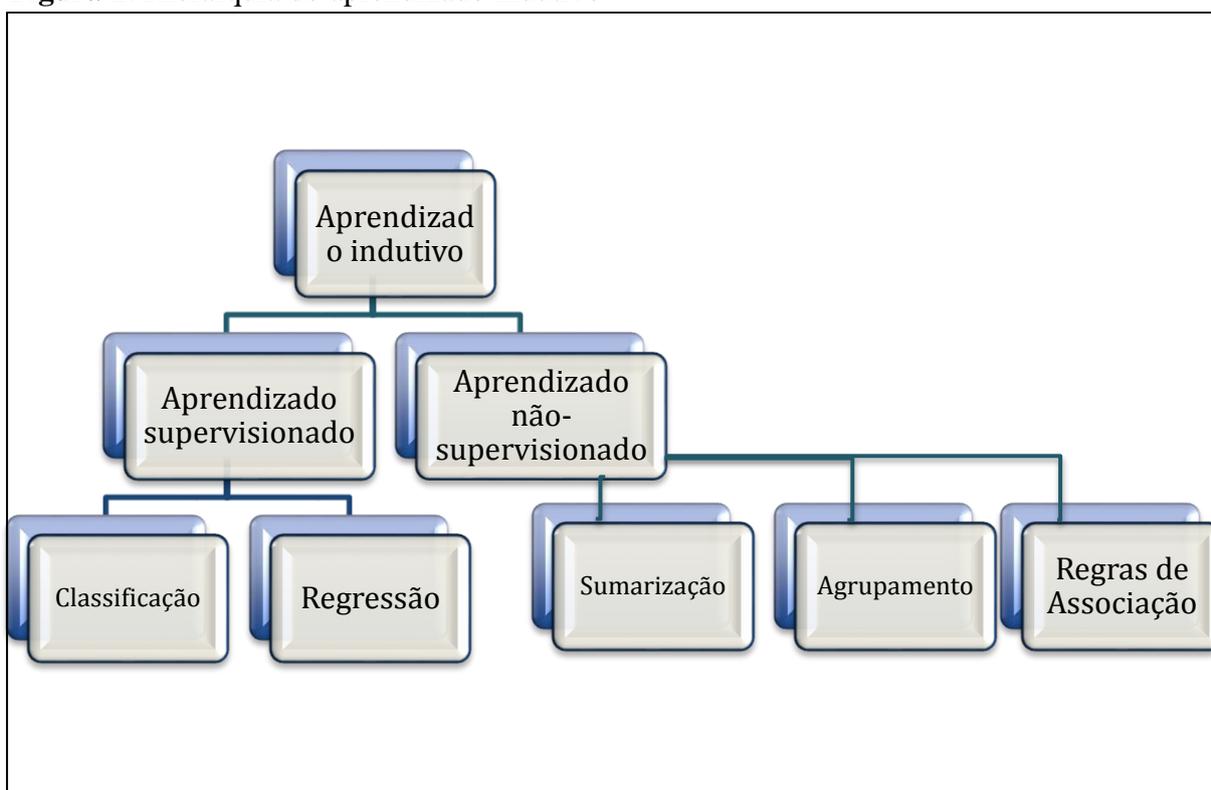
A principal finalidade das técnicas de aprendizado de máquina é descobrir automaticamente regras gerais em grandes conjuntos de dados, que possibilitem extrair informações implicitamente representadas. De forma geral, essas técnicas de aprendizagem de máquina (aprendizado indutivo) podem ser divididas em: aprendizado supervisionado e aprendizado não supervisionado (STONE, DUNPHY, SMITH, 1966).

O aprendizado supervisionado acontece quando antes do processo de aprendizado (Figura 1), o indutor recebe um conjunto de exemplos pré-classificados, cada exemplo sendo formado por um conjunto de atributos de entrada e um conjunto de atributos de saída (rótulos com classe conhecida). Se as classes possuem valores discretos o problema é conhecido como classificação, e se possuem valores contínuos, o problema é conhecido como regressão (SOUTO et al, 2003; MARTINS, 2003).

Em contrapartida, no aprendizado não supervisionado, para cada exemplo, apenas os atributos de entrada estão disponíveis, ou seja, os exemplos não possuem uma classe correspondente. Nesse caso, as tarefas podem estar relacionadas com o *agrupamento* dos

exemplos; com uma descrição compacta de um subconjunto de dados – chamado *sumarização* – ou com a caracterização, por meio de regras de associação, do quanto a presença de um conjunto de atributos implica na presença de algum outro conjunto distinto de atributos nos mesmo exemplos. Essas técnicas mencionadas, são usadas quando a finalidade for encontrar em um conjunto de dados, padrões ou tendências que auxiliam o entendimento desses dados (SOUTO et al, 2003; MARTINS,2003).

Figura 1: Hierarquia do aprendizado indutivo



Fonte: Próprio autor, adaptado de REZENDE, 2005.

2.2 Avaliação de estudantes

Os estudantes das universidades brasileiras apresentam conhecimentos e habilidades bastante heterogêneos, e mesmo assim recebem o mesmo tratamento, independente da sua capacidade cognitiva, sobre um conteúdo específico. Esse quadro condena vários estudantes a uma falta de aprendizagem adequada, podendo gerar sucessivas reprovações (PIMENTEL et al., 2003).

Portanto é preciso diversificar, rompendo com a pedagogia frontal que impõe a mesma lição e mesmos exercícios para todos os alunos (PERRENOUD, 2000a). Perrenoud (2000b) defende que praticar uma pedagogia diferenciada, faz com que, quando necessário, o aluno seja recolocado ou reorientado para uma atividade fecunda para ele.

Tendo em vista essa necessidade mencionada acima, e os avanços tecnológicos, que são grandiosos e fornecem novas possibilidades de representação, interação, comunicação e processamento das informações, são promissoras as perspectivas no âmbito escolar. Estas mudanças apresentam como consequências: transformações econômicas, sociais, políticas e culturais, numa sociedade denominada “sociedade informática” (GAMBIM, 2013).

Recentemente, técnicas de Mineração de Dados (*Data Mining-DM*) tem sido utilizadas como forma de predição do desempenho dos alunos. Essas técnicas de mineração consistem na aplicação de algoritmos específicos para extrair padrões a partir de dados (FAYYAD et al, 1996). O conhecimento gerado a partir desses dados, pode ser útil para o aprimoramento dos sistemas de ensino.

O termo Mineração de Dados Educacionais (EDM) se refere ao desenvolvimento de métodos para explorar informações coletadas de ambientes educacionais, possibilitando compreender os alunos de forma mais eficaz e adequada, gerando melhores benefícios educacionais. Basicamente, é possível minerar dados de alunos para averiguar a relação entre uma abordagem pedagógica e o seu aprendizado, e por meio disso, o professor pode compreender se sua abordagem está realmente contribuindo para o estudante desenvolver métodos de ensino mais eficazes (BAKER et al. 2011). É importante salientar que através da mineração de dados, é possível fomentar as técnicas de aprendizagem de máquina.

3 REVISÃO SISTEMÁTICA

Com o propósito de tornar os resultados mais auditáveis, confiáveis e propensos a reprodução por outros pesquisadores, neste capítulo é apresentada uma revisão sistemática da literatura seguindo um protocolo que é uma versão adaptada das recomendações propostas por Kitchenham & Charters (2007).

3.1 Metodologia de Pesquisa

3.1.1 Objetivo e Perguntas da Pesquisa

O objetivo desta revisão sistemática foi adquirir conhecimentos sobre o estado da arte e as práticas no uso de técnicas de aprendizagem de máquina durante o processo de avaliação de estudantes em cursos de programação. Dessa forma, buscou-se obter informações que podem auxiliar as tomadas de decisão sobre a escolha de estratégias que auxiliem os estudantes em seu processo de aprendizagem.

Quatro questões de pesquisa (P) foram definidas para guiar a seleção dos estudos e a extração dos dados:

- P1 – Como ilustrar uma correlação entre as características individuais de cada estudante e seu desempenho, usando atributos gerais?
- P2 – Como auxiliar o professor a planejar estratégias e políticas que melhorem o engajamento dos alunos, a motivação e suas notas?
- P3 – Quais técnicas de aprendizado de máquina podem ser usadas para descobrir regras gerais em grandes conjuntos de dados, que possibilitem extrair informações implicitamente representadas sobre os alunos?
- P4 – Como utilizar as informações obtidas através das técnicas de aprendizado de máquina para gerar um feedback personalizado e mais preciso, que ajude no engajamento do aluno?

3.1.2 Estratégias de Busca e Fontes de Dados

Este estudo foi delimitado a publicações do período de janeiro de 2010 a novembro de 2018. Foram selecionados os estudos a partir de bases de busca automática (Tabela 1-pág 13), as quais foram escolhidas por sua relevância no meio acadêmico e disponibilidade dos estudos completos a partir do domínio da Universidade Federal de Pernambuco.

Tabela 1: Bases de busca automáticas utilizadas no processo da revisão sistemática

Base de Dados	Endereço Eletrônico
ACM Digital Library IEEE Xplore	https://dl.acm.org/ https://ieeexplore.ieee.org/

Fonte: Próprio autor

Os artigos foram identificados a partir da definição de strings de busca derivadas das questões de pesquisa (Tabela 2). As strings foram calibradas pela realização de um estudo piloto, através do qual possíveis adaptações que poderiam aperfeiçoar as strings foram identificadas. A busca pelos estudos foi realizada nos meses de abril à outubro de 2018.

Tabela 2: Strings de busca utilizadas na revisão sistemática da literatura

(Machine Learning) AND (Techniques) AND ((Student) AND (Assessment)) OR ((Programming) AND (Students))
--

Fonte: Próprio autor

3.1.3 Procedimento de Seleção dos Estudos

Para selecionar os artigos, foi desenvolvido um processo em três etapas: avaliação de títulos, dos resumos e dos textos completos, segundo os critérios de inclusão e exclusão. Foram incluídos na revisão os artigos identificados que satisfizessem todos os critérios de inclusão (I) listados abaixo:

- I1: Estudos que respondem pelo menos um dos questionamentos de pesquisa;
- I2: Estudos primários;
- I3: Estudos acadêmicos ou da indústria;
- I4: Estudos que se caracterizam como artigos completos ou resumos estendidos publicados em periódicos ou conferências.

Foram excluídos os estudos que fazem parte dos seguintes critérios (E):

- E1: Estudos em outro idioma, que não esteja em inglês ou português;
- E2: Estudos que tratam de outros assuntos não relacionados às questões de pesquisa;
- E3: Estudos cujo objetivo não esteja relacionado com técnicas de aprendizagem de máquina e/ou avaliação de estudantes;
- E4: Estudos duplicados;

- E5: Estudos que não possuem discussões e resultados pautados em métodos e técnicas de pesquisa científica ou experimentação;

- E6: Estudos indisponíveis para consulta em versão completa e gratuita pela rede institucional da UFPE;

- E7: Estudos que não sejam artigos completos ou resumos estendidos publicados em periódicos ou conferências, como por exemplo: capítulos de livros, produção artística, patentes, resumos, textos em jornais de notícias ou revistas.

3.1.4 Critérios de Qualidade

Para que fossem selecionados os estudos com maior relevância para pesquisa, foram criados 10 critérios de qualidade (Q). Esses critérios tomaram como base a qualidade do trabalho em relação à estruturação, metodologia, objetivos e resultados e ao grau de atendimento às questões de pesquisa:

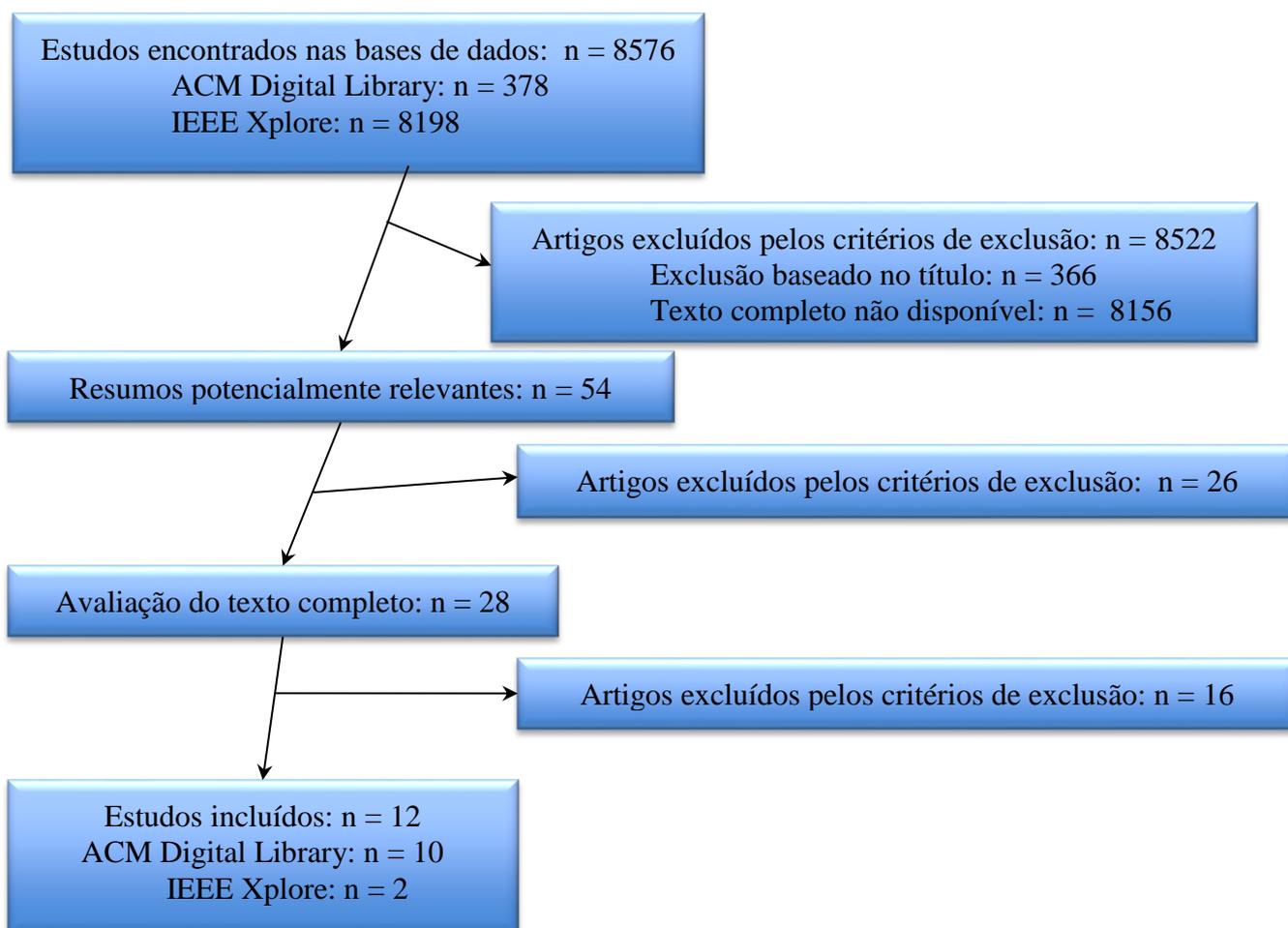
- Q1: Descreve abordagens para melhoria do processo de aprendizagem?
- Q2: Aborda o contexto de Aprendizagem de Máquina?
- Q3: Aborda o contexto de Avaliação do Estudante?
- Q4: Aborda o contexto de Cursos de Programação?
- Q5: Os resultados condizem com os objetivos?
- Q6: Existe uma descrição adequada do contexto?
- Q7: Os dados coletados foram exibidos de forma clara e objetiva?
- Q8: A metodologia usada foi descrita de forma objetiva?
- Q9: Os dados coletados foram exibidos de forma clara e objetiva?
- Q10: Os resultados contribuem para o estado da arte e estado da prática?

4 APRESENTAÇÃO DOS RESULTADOS

4.1 Resultado do Procedimento de Busca

Seguindo a estratégia descrita na seção 3.1.2 foram realizadas as buscas nas bases de dados, em seguida os artigos encontrados passaram por um processo de avaliação em três etapas, com o procedimento descrito na seção 3.1.3. A Figura 2 ilustra o processo de seleção do qual os artigos foram submetidos. Foram encontrados 8576 artigos, dos quais, 8156 não possuíam o texto completo disponível, pois para isso era cobrada uma taxa e por esse motivo não fizeram parte desse estudo. Em seguida foi feita a análise dos títulos dos artigos que restaram, e ao final foram considerados potencialmente relevantes 54 estudos. Depois foi feita a análise dos resumos, dos quais 28 foram selecionados para que o texto completo fosse avaliado. Ao final do processo de seleção, 12 estudos foram selecionados por preencherem os critérios preestabelecidos.

Figura 2: Fluxograma do processo de seleção dos estudos.



Fonte: Próprio autor.

Foi observado que para avaliação inicial, a pesquisa feita nas bases de dados retornou uma quantidade considerável de estudos e que para responder as questões de pesquisa poucos foram de fato relevantes. Muito provável que isso tenha ocorrido por que certas bases de dados não são tão precisas em relação a string de busca escolhida, em especial a IEEE Xplore. Ela retornou uma enorme quantidade de artigos, que na sua grande maioria foram considerados irrelevantes à pesquisa. Outras strings de busca foram testadas, algumas retornavam quantidades enormes de artigos em todas as bases escolhidas, outras mais restritivas também não foram utilizadas por retornarem quase nenhum ou nenhum artigo.

No Apêndice, são listados os artigos selecionados, estes organizados em três categorias: Base de Dados, Ano de Publicação e Tipo de Publicação.

Os artigos selecionados por base de dados, são apresentados na Tabela 3, sendo a maioria dos estudos, extraída da base de dados ACM Digital (83,33%).

Tabela 3: Proporção de artigos selecionados por base de dados.

Base de Dados	Número de artigos n(%)	ID*
ACM Digital	10 (83,33)	A02, A03, A04, A05, A06, A07, A08, A09, A10, A11
IEEE Xplore	2 (16,67)	A01, A12
Total	12 (100)	--

*ID: Identificação do estudo.

Fonte: Próprio autor.

Na Tabela 4 se encontram os dados relacionados ao ano de publicação dos artigos selecionados. Apesar de ter feito uma pesquisa buscando artigos entre os anos de 2010 e 2018, foram encontrados artigos relevantes para o estudo somente entre 2015 e 2018, sendo 2016 o ano em que foi encontrado mais artigos relevantes.

Tabela 4: Proporção de artigos selecionados por ano de publicação.

Ano de Publicação	Número de Artigos n (%)	ID
2015	2 (16,67)	A01, A02
2016	5 (41,66)	A03, A04, A05, A06, A07
2017	3 (25)	A08, A09, A10
2018	2 (16,67)	A11, A12
Total	12 (100)	--

*ID: Identificação do estudo.

Fonte: Próprio autor

Os dados referentes ao tipo da publicação dos artigos selecionados são apresentados na Tabela 5. Sendo os tipos Periódico e Conferência os escolhidos para os artigos do estudo e sua proporção foi 1 Periódico para 5 Conferências.

Tabela 5: Proporção de artigos selecionados por tipo de publicação.

Tipo de publicação	Número de Artigos n (%)	ID
Periódico	2 (16,67)	A01, A12
Conferência	10 (83,33)	A02, A03, A04, A05, A06, A07, A08, A09, A10, A11
Total	12 (100)	--

*ID: Identificação do estudo.

Fonte: Próprio autor

4.2 Análise e Discussão do Conteúdo dos Artigos Selecionados

Nesta sessão serão discutidos os trabalhos selecionados, os métodos utilizados e resultados positivos ou negativos provenientes das pesquisas.

Sorour e Mine (2016) desenvolveram um método a partir dos comentártios que descreviam claramente o comportamento e situações após cada aula, escritos pelos alunos, ao responderem cinco questões preparadas por seu professor. Os autores ilustram a correlação entre as características individuais de cada estudante e seu desempenho, usando atributos gerais, e chamaram essa ilustração de Método Baseado em Atributo (AM). Depois compararam os resultados de predição do AM com os de um método estatístico existente usando o modelo Alocação de Dirichlet Latente (LDA), e chamaram este método de Método Baseado em Tópico (TM).

Também usaram modelos de Floresta Aleatória (RF), Rede Neural Artificial (ANN) e Máquina de Vetor de Suporte (SVM) para construir modelos de predição da nota fina do aluno em cada aula usando AM e TM. Para acompanhar a situação de aprendizado de cada aluno, os métodos Aprendizagem Multi-Instância (MIL) e Votação Maioritária (MV) foram apresentados (Sorour e Mine, 2016).

Seus resultados mostraram que o método baseado em atributos (AM) teve melhor previsão dos resultados do que o método baseado em tópicos (TM). Também foram aplicados os métodos MV e MIL para melhorar os resultados de previsão e obter um acompanhamento contínuo do desempenho dos alunos. Os resultados mostram que o MV e MIL melhoraram os resultados de predição mais do que o método que considera os resultados de predição em cada lição. Eles observaram que houve uma variação entre os métodos MV e MIL e concluíram que

o método MIL é mais estável para prever a nota do aluno do que o método MV (Sorour e Mine, 2016). Responde as questões de pesquisa (P): P1, P3 e P4.

Já a pesquisa de Fonseca e colaboradores (2016) propôs o uso de uma ferramenta de monitoramento, capaz de capturar e exibir para os professores informações em tempo real sobre o desempenho dos alunos, para que os professores através dessas informações, identifiquem problemas/desvios e possam agir de acordo. Essa ferramenta de monitoramento toma como entrada os snapshots do código-fonte das soluções dos alunos para as tarefas propostas pelos professores.

Esses snapshots são processados e então uma série de informações sobre o desempenho dos alunos fica imediatamente disponível para os professores. Após identificar o aluno e a tarefa em questão, a ferramenta verifica se existem erros de compilação, caso existam são coletadas algumas métricas sobre o código-fonte e armazenadas junto com informações sobre o erro de compilação e o processo é interrompido. Já quando o código compila bem, o sistema coleta informações sobre a saída produzida e algumas métricas adicionais usando a ferramenta radon. Assim os professores têm acesso imediato a uma série de informações sobre as turmas e alunos individuais. Dessa forma o professor deve ser capaz de identificar possíveis desvios e fornecer feedback ou tomar medidas mais adequadas (Fonseca et al, 2016).

A ferramenta do estudo acima, foi testada em uma sala de aula real com 101 estudantes, e como resultado fica claro que muitas informações sobre o comportamento e o desempenho dos alunos podem ser extraídas de snapshots de seu código-fonte. O sistema se mostrou uma ferramenta importante para ajudar os professores a melhorar o desempenho dos alunos (notas) e diminuir o número de desistências e falhas (Fonseca et al, 2016).

Responde as questões de pesquisa (P): P1 e P4.

No trabalho intitulado “Automated Data-Driven Hints for Computer Programming Students”, os autores sugerem que o feedback formativo é essencial para aprender programação de computadores, e que é um desafio para automatizá-lo, pois para um exercício de programação pode existir muitas soluções. Então é apresentada uma abordagem para gerar sugestões usando dados de alunos anteriores. Para isso os autores utilizam uma variedade de técnicas, como filtragem, agrupamento e mineração de padrões, e assim são geradas quatro tipos diferentes de dicas: sugestão de entrada, baseada em código, conceito e dicas preventivas (Chow et al, 2017).

Essa abordagem foi testada com 5529 alunos usando a plataforma Grok Learning, e os resultados mostram que pode ser gerado vários tipos de dicas para mais de 90% dos alunos, a partir dos dados de apenas 10 alunos e assim reduzir o problema do coldstart. Além da

avaliação quantitativa, também foi feita uma qualitativa, onde especialistas foram solicitados para avaliar a qualidade das sugestões geradas. Os especialistas avaliaram positivamente a utilidade e a relevância das dicas, em diferentes níveis de habilidade do aluno (Chow et al, 2017). Responde as questões de pesquisa (P): P3 e P4.

Em seu estudo, Ruvo e colaboradores (2018), propuseram o conceito de indicadores de estilo semânticos que podem indicar uma falta de conhecimento de algum aspecto de programação. Eles desenvolveram uma ferramenta para detectar esses indicadores analisando sua Abstract Syntax Tree (AST) e aplicaram a ferramenta em mais de 19000 submissões de 928 alunos de graduação, que resolveram mais de 665 exercícios distintos.

Seus resultados mostraram que um grande número de submissões continham os indicadores, como também que a maioria dos alunos em todos os níveis exibiu um ou mais indicadores, demonstrando que estes indicadores de estilo semântico, além de serem encontrados nos alunos, são na verdade bastante comuns. A presença desses indicadores aponta para uma confusão feita pelos alunos em relação a alguns conceitos de programação. Logo que detectada a presença de um indicador, pode ser fornecido um feedback sobre onde o indicador está, qual o problema em potencial e como o código pode ser melhorado. Como encontrar esses indicadores requer pouco poder computacional, esse método pode ser integrado em Ambientes de Desenvolvimento Integrado (IDEs) e assim facilmente usado (Ruvo et al, 2018).

Responde as questões de pesquisa (P): P1 e P4.

Enquanto Annamaa e colaboradores (2017), descreveram a experiência que obtiveram com quatro estilos diferentes de feedback automático (AF), aplicados em um curso de introdução a programação com mais de 300 alunos. A turma foi dividida em 4 grupos de tamanho igual e cada grupo recebeu uma política de feedback diferente. Os resultados mostraram que a experiência com feedback automático (AF) foi positiva. Eles chegaram a conclusão de que, apesar de um grupo apresentar notas finais ligeiramente melhores que os outros não se pode declarar nenhum estilo de feedback claramente superior aos outros.

Responde as questões de pesquisa (P): P4.

Dewey et al (2017), usando Programação Lógica de Restrições (CLP) desenvolveram uma técnica para avaliar um conjunto de testes gerados manualmente. Os resultados mostraram que: uma abordagem baseada em CLP, que gera milhões de casos de teste, pode ser usada para expor os pontos fracos em um conjunto de testes feitos manualmente, para uma atribuição de programação. Peneirar os resultados de milhões de casos de teste pode parecer assustador, mas os autores mostram que, concentrando-se nos pontos corretos, se obtém uma percepção considerável nas deficiências do teste e nos erros dos alunos.

Responde as questões de pesquisa (P): P1 e P3.

Já Rubio (2015) utiliza técnicas de agrupamento para categorizar automaticamente o estágio de aprendizado dos alunos, ao fim de um curso de introdução a programação. Ele utilizou o exame final para avaliar os resultados de aprendizado dos alunos. Fez isso, através das técnicas de agrupamento para analisar o comportamento dos alunos, lendo e escrevendo notas de código. Os resultados mostraram, com sucesso, que pode-se inferir o estágio de aprendizagem de alguns alunos. Responde as questões de pesquisa (P): P1 e P3.

Ahmad et al (2018) apresentam em seu estudo, um método de medição de habilidades cognitivas multicamada (CS), que depende da quantificação das Características Relacionadas ao Estudo (SRC) do aluno. Primeiro o método dividiu o SRC em cinco fatores, para desenvolver uma abordagem multicamada, foi atribuído um intervalo específico a cada fator que iniciou um modelo de cinco camadas de SRC. Em seguida o método dividiu o CS em 21 intervalos periódicos. A divisão feita no SRC e no CS foi para melhorar a exatidão e precisão do método. Por último, o método usou o Algoritmo de Gauss-Newton (GNA) para simular a relação entre as camadas SRC e os intervalos CS. Como resultado, foram obtidas seis equações matemáticas para o modelo multicamada do SRC que simulou com sucesso a relação não linear entre CS e SRC. Os resultados revelaram que o método proposto de medição multicamadas de CS, superou as técnicas de medição de CS existentes, porque alcançou valores precisos e melhores de precisão, recordação, pontuação F1 e medidas de precisão.

Responde as questões de pesquisa (P): P3.

Enquanto que no trabalho de Liu et al (2015), foi utilizado um sistema analítico de aprendizagem chamado Tracer, que deriva medidas comportamentais de engajamento e cria visualizações de padrões comportamentais dos alunos. A ferramenta registra os estágios intermediários de desenvolvimento, e utiliza os dados para medir o engajamento comportamental dos alunos. Os resultados sugeriram que: a intensidade da visualização pode ser útil para distinguir os diferentes comportamentos dos alunos, ao abordar uma tarefa.

Responde as questões de pesquisa (P): P1.

Kaleeswaran et al (2016) propuseram uma nova metodologia chamada de geração de feedback verificada semi-supervisionada. Eles fizeram o uso de técnicas de agrupamento para categorizar por estratégia de solução as submissões dos alunos, e em cada agrupamento, eles pediram para ao professor identificar ou adicionar uma submissão correta. Em seguida, foi verificado cada envio em um agrupamento em relação a sua submissão modelo. Ao detectar falhas, um feedback sugerindo as correções, será gerado. O agrupamento reduz a sobrecarga

do instrutor e também as variações que precisam ser manipuladas durante a geração de feedback.

Essa metodologia garante apenas que o feedback correto seja gerado. Por isso os autores implementaram uma ferramenta, denominada CoderAssist. Depois foi projetado um novo algoritmo de geração de feedback guiado de contra-exemplo. A ferramenta foi testada com 2226 submissões de 4 problemas, ela pôde gerar um feedback verificado para 85% dos casos. Os resultados sugerem que a combinação de técnicas de agrupamento e de verificação abrem o caminho para ferramentas práticas de geração de feedback (Kaleeswaran et al, 2016).

Responde as questões de pesquisa (P): P1, P2, P3 e P4.

No trabalho Sajjadi e colaboradores (2016) foi abordado a classificação por pares, que é o processo onde um aluno revisa a tarefa do outro, que se tornou um mecanismo popular usado em cursos on-line massivos abertos (MOOCs). Intrigados com essa ideia, os autores coletaram durante um semestre, os dados sobre submissões dos alunos, avaliadas tanto por assistentes de ensino (TA) como por um mecanismo de classificação por pares. Então aplicaram diferentes métodos estatísticos e de aprendizagem de máquina, para agregar as notas de pares, afim de obter notas finais precisas para as submissões. Os resultados surpreenderam os autores, pois não houve melhoria na linha de base do uso da nota média, como a nota final. O ponto positivo é que, as notas dadas pelos alunos não se distanciaram das verdadeiras, embora tendam a ser melhores que as notas dadas pelo TA. Do ponto de vista estatístico ou de aprendizado de máquina, os resultados mostram que nenhum dos modelos testados superou o estimador de média simples para o conjunto de dados que foi trabalhado.

Responde as questões de pesquisa (P): P1 e P3.

Enquanto Färnqvist e Heintz (2016) investigaram diferentes formas de feedback por meio de avaliação automatizada. Eles utilizaram em sua investigação o Kattis que é um sistema de julgamento automatizado. Com o Kattis foi possível avaliar automaticamente as submissões dos alunos e gerar também, automaticamente, feedback sobre correção e eficiência, que de imediato era passado para os alunos. Foi investigado como esse feedback influenciava as formas de trabalho do aluno, seu desempenho e suas relações com a equipe examinadora.

Os resultados do trabalho acima, mostraram que o feedback por meio de avaliação automatizada gerou melhorias no desempenho dos alunos estimulando os mesmo a se esforçarem mais durante o curso. Verificou-se que 80% dos alunos são favoráveis ao uso da avaliação e tiveram sua forma de trabalhar melhorada. E 50% dos alunos se esforçaram mais devido ao julgamento automatizado. A avaliação foi considerada mais objetiva, por ser

executada da mesma forma para todos, onde 73% dos alunos julgaram a classificação automatizada mais justa que a classificação manual.

Responde as questões de pesquisa (P): P2 e P4.

Estas foram as análises dos trabalhos que se enquadraram nos critérios desta revisão, onde pode-se perceber as técnicas usadas e os resultados positivos ou negativos que os pesquisadores obtiveram. Essas técnicas podem servir de parâmetro para os educadores, e assim o ensino de Programação pode ser melhorado, através de variadas técnicas de aprendizagem de máquina.

4.3 Ameaças a Validade

Algumas limitações foram observadas ao realizar essa revisão sistemática da literatura descrita no Capítulo 3, embora houve esforço para seguir um protocolo bem fundamentado, essas limitações podem significar uma ameaça à validade do estudo. Uma das ameaças seria que só um avaliador foi destinado a realizar a seleção dos estudos. Outro ponto seria a subjetividade na interpretação do avaliador, que pode ter gerado exclusão de estudos relevantes, visto que não foi corrigido por um revisor. E de acordo com a Figura 2, não houve a análise de muitos artigos, pois não se encontravam disponíveis na rede da UFPE.

5 CONCLUSÃO

Neste trabalho de graduação foi realizada uma revisão sistemática da literatura, buscando encontrar estudos relacionados ao uso de técnicas de aprendizagem de máquinas na avaliação de estudantes, de cursos de Programação.

Os estudos de Sorour, e de outros autores, revelam que é viável a aplicação de técnicas de aprendizagem de máquina para automatizar a avaliação do comportamento e do desempenho dos alunos, durante o desenvolvimento de suas atividades em cursos de introdução a Programação. Desta forma, pode-se devolver um feedback personalizado e mais preciso, que ajuda no engajamento do aluno. Em outro trabalho, técnicas de agrupamento foram usadas para separar as submissões dos alunos por estilo de solução, e com isso observou-se que houve redução da sobrecarga do instrutor, além de reduzir as variações que precisam ser manipuladas durante a geração de feedback.

Desta revisão sistemática, pode-se extrair resultados que sirvam de parâmetro para que os educadores, em posse dos dados sobre o comportamento e desempenho dos alunos, possam planejar estratégias e políticas que melhorem o engajamento dos alunos, a motivação e suas notas, e assim consigam diminuir o índice de desistências em cursos de programação.

REFERÊNCIAS

AHMAD, S; et al. **A Multilayer Prediction Approach for the Student Cognitive Skills Measurement**. IEEE Access, 2018

AMARAL, H.J.C.; FRANÇA, R.S. **Mineração de Dados na Identificação de Grupos de Estudantes com Dificuldades de Aprendizagem no Ensino de Programação**. CINTED-UFRGS, 2013.

ANNAMAA, A.; SUVISTE, R.; VENE, V. **Comparing different styles of automated feedback for programming exercises**. Proceedings of the 17th Koli Calling International Conference on Computing Education Research, 2017.

BAKER, R.; ISOTANI, S.; CARVALHO, A. (2011) “Mineração de Dados Educacionais: Oportunidades para o Brasil”. **Revista Brasileira de Informática na Educação**, v. 19, n. 02, pp. 3-13.

CARDOSO, S. M. V.; JANDL, P. (1998). **Estilos de Aprendizagem: Aprender a Aprender**.

CHOW, S.; et al. **Automated Data-Driven Hints for Computer Programming Students**. Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization, 2017.

DE RUVO, G.; TEMPERO, E.; ROWE, G.B. **Understanding Semantic Style by Analysing Student Code**. Proceedings of the 20th Australasian Computing Education Conference, 2018.

DEWEY, K.; et al. **Evaluating Test Suite Effectiveness and Assessing Student Code via Constraint Logic Programming**. Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education, 2017.

FAYYAD, U.; PIATETSKY-SHAPIO, G.; SMYTH, P. (1996) “From data mining to knowledge discovery in databases”. **AI magazine**, v. 17, n. 3, pp. 37-54.

FÄRNQVIST, T.; HEINTZ, F. **Competition and Feedback through Automated Assessment in a Data Structures and Algorithms Course**. Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education, 2016.

FONSECA, N.G.; MACEDO, L.; MENDES, A.J. **CodeInsights - Monitoring programming students' progress**. Proceedings of the 17th International Conference on Computer Systems and Technologies 2016, 2016.

GAMBIM, G. **Inteligência artificial: Utilização de jogos eletrônicos no campo educacional**. In: vii colóquio internacional, 2013, São Luiz Gonzaga. VII colóquio internacional, 2013. *apud* LEAL A.S. **Aplicação do método de aprendizagem por reforço Q-Learning na adaptatividade dinâmica de dificuldade de um jogo digital ortográfico**. 2016. 102 páginas. Trabalho de Conclusão de Curso (Bacharel em Sistemas de Informação) – Universidade Federal dos Vales do Jequitinhonha e Mucuri, Diamantina-MG.

- GOMES, A. J. (2000). **Ambiente de Suporte à Aprendizagem de Conceitos Básicos de Programação**. Dissertação de Mestrado. Universidade de Coimbra.
- KALEESWARAN, S.; et al. **Semi-supervised Verified Feedback Generation**. Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, 2016.
- KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. Technical Report EBSE 2007-001, Department of Computer Science Keele University, Keele, 2007
- LIU, M.; et al. **Measuring and Visualizing Students' Behavioral Engagement in Writing Activities**. IEEE Transactions on Learning Technologies, 2015.
- MARTINS, C. A. **Uma abordagem para pré-processamento de dados textuais em algoritmos de aprendizado**. 2003. 208f. Tese (Doutorado em Ciências da Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos.
- PEREIRA, G.V. **A Inteligência artificial aplicada na educação**. Disponível em: <http://www.ceavi.udesc.br/arquivos/id_submenu/387/guilherme_vota_pereira.pdf>.
- ROCHA, H. V. (1991). **Representações Computacionais Auxiliares ao Entendimento de Conceitos de Programação**. Unicamp.
- PERRENOUD, P.. **Dez Novas Competências para Ensinar**. Porto Alegre: Artes Médicas Sul, 2000a.
- PERRENOUD, P. **Pedagogia diferenciada: das intenções à ação**. Porto Alegre: Artes Médicas, 2000b.
- PIMENTEL, E. P.; FRANÇA, V. F.; OMAR, N.. **A identificação de grupos de aprendizes no ensino presencial utilizando técnicas de clusterização**. In: XIV Simpósio Brasileiro de Informática na Educação, 2003, Rio de Janeiro. Anais do XIV SBIE, 2003.
- RUBIO, M.A **Automatic Categorization of Introductory Programming Students using Cluster Analysis**. Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education, 2015.
- SAJJADI, M.S.M.; ALAMGIR, M.; LUXBURG U.V. **Peer Grading in a Course on Algorithms and Data Structures: Machine Learning Algorithms do not Improve over Simple Baselines**. Proceedings of the Third (2016) ACM Conference on Learning @ Scale, 2016.
- SOLANGE O. R. **Sistemas Inteligentes**. 2003 . 550 páginas **Editora:** Editora Manole (2005).
- SOROUR, S.E.; MINE, T. **Exploring Students' Learning Attributes in Consecutive Lessons to Improve Prediction Performance**. Proceedings of the Australasian Computer Science Week Multiconference, 2016.

SOUTO, et al. **Técnicas de Aprendizado de Máquina para Problemas de Biologia Molecular**. Instituto de Ciências Matemáticas e de Computação Universidade de São Paulo – São Carlos.

STONE, P. J.; DUNPHY, D. C.; AND SMITH, M. S. (1966). The general inquirer: A computer approach to content analysis.

APÊNDICE - DESCRIÇÃO DAS INFORMAÇÕES GERAIS DOS ARTIGOS PARA ANÁLISE EM PROFUNDIDADE

ID*	Título	BD	Autoria	Ano	Fonte	Tipo
A01	Measuring and Visualizing Students' Behavioral Engagement in Writing Activities	IEEE Xplore	Ming Liu, Rafael A. Calvo, Abelardo Pardo, and Andrew Martin	2015	IEEE Transactions on Learning Technologies	Periódico
A02	Automatic Categorization of Introductory Programming Students using Cluster Analysis	ACM Digital Library	Miguel A. Rubio	2015	Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education	Conferência
A03	Exploring Students' Learning Attributes in Consecutive Lessons to Improve Prediction Performance	ACM Digital Library	Shaymaa E. Sorour Tsunenori Mine	2016	Proceedings of the Australasian Computer Science Week Multiconference	Conferência
A04	Peer Grading in a Course on Algorithms and Data Structures: Machine Learning Algorithms do not Improve over Simple Baselines	ACM Digital Library	Mehdi S. M. Sajjadi, Morteza Alamgir, Ulrike von Luxburg	2016	Proceedings of the Third (2016) ACM Conference on Learning @ Scale	Conferência
A05	CodeInsights - Monitoring programming students' progress	ACM Digital Library	Nuno Gil Fonseca, Luís Macedo, António José Mendes	2016	Proceedings of the 17th International Conference on Computer Systems and Technologies 2016	Conferência
A06	Competition and Feedback through Automated Assessment in a Data Structures and Algorithms Course	ACM Digital Library	Tommy Färnqvist, Fredrik Heintz	2016	Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education	Conferência
A07	Semi-supervised Verified Feedback Generation	ACM Digital Library	Shalini Kaleeswaran, Anirudh Santhiar, Aditya Kanade, Sumit Gulwani	2016	Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering	Conferência
A08	Evaluating Test Suite Effectiveness and Assessing Student Code via Constraint Logic Programming	ACM Digital Library	Kyle Dewey, Phill Conrad, Michelle Craig, Elena Morozova	2017	Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education	Conferência
A09	Automated Data-Driven Hints for Computer Programming Students	ACM Digital Library	Sammi Chow, Kalina Yacef, Irena Koprinska, James Curran	2017	Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization	Conferência
A10	Comparing different styles of automated feedback for programming exercises	ACM Digital Library	Aivar Annamaa, Reelika Suviste, Varmo Vene	2017	Proceedings of the 17th Koli Calling International Conference on Computing Education Research	Conferência
A11	Understanding Semantic Style by Analysing Student Code	ACM Digital Library	Giuseppe De Ruvo, Ewan Tempero, Gerard B. Rowe	2018	Proceedings of the 20th Australasian Computing Education Conference	Conferência
A12	A Multilayer Prediction Approach for the Student Cognitive Skills Measurement	IEEE Xplore	Sadique Ahmad, Kan Li, Adnan Amin, Muhammad Shahid Anwar, Wahab Khan	2018	IEEE Access	Periódico

*ID: Identificação do estudo.

Fonte: Próprio autor.