Using Information Quality for the Identification of Relevant Web Data Sources: A Proposal

Bernadette Farias Lóscio
Federal University of Pernambuco
50.732-970 Recife, PE, Brazil
+55 81 2126 8430
bfl@cin.ufpe.br

Maria C. M. Batista
Federal Rural University of Pernambuco
52171-900 Recife, PE, Brazil
+55 81 3320 6491
ceca@deinfo.ufrpe.br

Damires Souza
Federal Institute of Education, Science and Technology of Paraiba
50.732-970 João Pessoa, PB Brazil
+55 81 3453 9213
dysf@ifpb.edu.br

Ana Carolina Salgado
Federal University of Pernambuco
50.732-970 Recife, PE, Brazil
+55 81 2126 8430
acs@cin.ufpe.br

ABSTRACT

In the last decade, applications that make use of data sources available on the Web have experienced a huge growth. One of the main problems regarding that consists in finding the most relevant data sources for a given application. In a general way, a data source is considered relevant when it contributes for answering queries submitted to the application. However, it may happen that a specific data source contributes for answering an application query but the answer provided by the data source does not really meet the user requirements. This may occur because the data source has generic data and the user wants more specific data, for example. On the other hand, some data sources may have data of poor quality, i.e., the data may be outdated, incomplete or incorrect. In such cases, it is not enough just to find data sources that can answer to the application queries. It is also important to check if the available data also meet the user needs. In this paper, we discuss such problem and we propose an approach, based on Information Quality (IQ), to help the evaluation of the relevance of a Web data source for domain-specific applications. We also present an example illustrating how our proposal can be used to enhance this evaluation.

Categories and Subject Descriptors

H.4 [**Information System Applications**]: Miscellaneous;
H.2 [**Database Management**]: Miscellaneous

General Terms

Algorithms, Measurement.

Keywords

Web Application, Relevant Data Source, Information Quality.

1. INTRODUCTION

The huge volume of data available on the Web motivates the idea of developing applications able to offer access to one or even multiple heterogeneous Web data sources. In the context of this paper, a Web data source may be a linked data set, a structured data set obtained from HTML tables (or HTML lists) or a back-end Deep Web database [7]. Such huge volume of data makes hard the process of choosing the most suitable or relevant data sources for a given application. In such scenario, an interesting and relevant question arises: *Under what conditions a data source is considered relevant to an application that offers access to multiple heterogeneous Web data sources?*

Generally, a data source is considered relevant when contributes for answering queries posed to the application. Although, it may happen that a data source contributes for answering an application query but the obtained answer does not really meet the user requirements. To better illustrate this problem, suppose an application that aims publishing information about academic researchers in Computer Science. One relevant query for such application could be: *Return all researchers who published papers in 2012*. Considering that the application is specific for Computer Science, it is not worth mentioning the researchers' area when formulating the query. Therefore, in this case, any data source that has bibliographic information about scientific papers could be considered relevant for the application, once it is capable of answering the proposed query. This happens because only considering the application query as a criterion for evaluating the relevance of a data source is not enough, once that application queries may be generic and do not reflect precisely the user requirements. In this case, considering application queries as the unique criterion for identifying relevant data sources will lead to generic information too. Specifically, the application is interested in data sources as DBLP RKBExplorer[1] or DBLP L3S[2] that stores information about Computer Science bibliography. On the other hand, data sources like DBPedia[3] will also be considered relevant once they have bibliographic information about the most famous researchers. In a similar way, PubMed[4] and RAE[5] are considered relevant because they have bibliographic information about medicine researchers and researchers working in UK institutions,

iiWAS2012, 3-5 December, 2012, Bali, Indonesia

[1] http://dblp.rkbexplorer.com/
[2] http://dblp.l3s.de/d2r/
[3] http://dbpedia.org/
[4] http://pubmed.bio2rdf.org
[5] http://rae2001.rkbexplorer.com/

respectively. Therefore, in this case, it becomes necessary having more information about the contents of the data source in order to assure that it meets the user requirements and, therefore, should be considered as a relevant one.

In this paper, we discuss the problem of identifying relevant Web data sources for domain-specific applications. We propose the use of Information Quality in order to help such identification. Information quality (IQ) is a multidimensional aspect of information systems and it is based on a set of criteria, which are used to assess a specific IQ aspect [17]. To clarify matters, we present a case study illustrating how the proposed approach can be used to enhance the identification of relevant Web data sources.

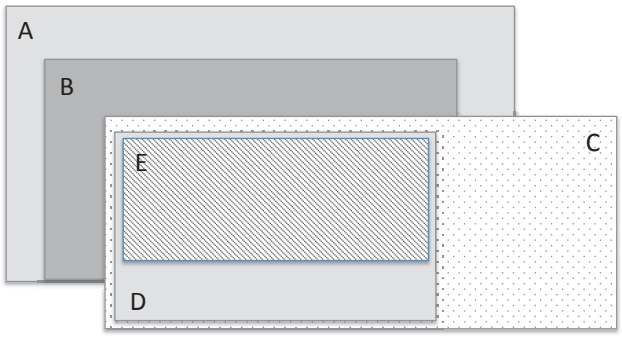
The remainder of this paper is organized as follows. Section 2 presents our definition for the problem of relevant data sources identification. Section 3 discusses some aspects of Information Quality. Section 4 describes our proposal for computing the degree of relevance of a data source. Section 5 presents an example to illustrate the proposed approach, while Section 6 discusses some related works. Finally, Section 7 points out some future works and concludes the paper.

2. PROBLEM DEFINITION

In general, conventional database applications are built on top of either a previously known database or a database that is created to meet specific application data requirements. In contrast to such scenario, when creating applications on top of Web data sources, it is very common to have applications, which are planned without taking into account a specific data source. In the same way, a huge volume of Web data is being generated and becoming available on the Web without considering specific application requirements. As an example, consider the huge volume of government open data actually available on the *Data.gov* site, which hosts more than 400,000 data sets. Such data sets may be viewed as data sources that are autonomously created and that may be used for several different applications. When someone wants to build an application on top of such data then it becomes necessary to search for the ones that meet the application requirements. Identifying relevant data sources on the Web for a specific application is a challenging task mainly due to the huge volume of available data sources and to the lack of knowledge about the quality of the available data.

In spite of the fact that some works have discussed the data source relevance problem [32, 33, 23] few efforts have been made to propose better ways to understand the relationship between application requirements and data sources capabilities. We argue that in order to propose a solution to such problem it is important to have a good understanding about the relationship between what the application intends to offer to its users and what the data sources contain. Figure 1 shows this idea focusing on the relationship between expected data and available data, as described in what follows.

Given an application ***P*** and a data source ***S***, set ***A*** represents the instances (i.e., values of data) expected to be available at ***S***, while set ***B*** represents the instances that are really available at ***S***. Concerning the application side, set ***C*** represents the instances expected to be available at ***P*** (from the data sources), while set ***D*** represents the instances that are really available at ***P***. Set ***E*** represents the instances that are available at ***P*** and that may be obtained from ***S***.



A: Data expected from the data source
B: Data available at the data source
C: Data expected from the application
D: Data available at the application
E: Data available at the application and available at the data source

Figure 1. Data Expectation and Data Availability

We make a distinction between what data is expected and what data is really available to better reflect the reality of the Web data sources and the applications defined over them. Different from conventional databases, which most of the times are created in a controlled way, Web data sources do not have a pre-defined schema or constraints that could be used to validate their data. Furthermore, Web data sources may result from data extraction processes. In this case, the data is retrieved out of unstructured or poorly structured data sources and, as a consequence, the extension of the data source may be incomplete or it may have erroneous or outdated data. Therefore, the set of instances expected to be available in a data source may be larger than the set of instances that are really available at a given moment. On the other hand, when building an application that uses Web data sources it may happen that the known or available data sources are not capable of meeting the whole set of application requirements. Therefore, the set of instances expected to be available at the application may be larger than the set of instances that may be obtained at a given moment according to the considered set of data sources. Moreover, considering that the application access data from more than one data source it is also possible having data available at the application, which is obtained from a data source different from the one that is being evaluated.

To measure the degree of relevance of a data source with respect to an application two aspects have to be considered: the intention and the extension of the data source. The first one concerns the data expectation and the second one concerns the data availability. Intuitively, we say that a data source is relevant for a given application if: (i) there is a big overlap between the data that is expected from the data source and the data that is expected from the application: it means that the data source has a big potential for providing instances that could meet the application requirements, and (ii) there is a big overlap between the data available at the data source and the data available at the application: it means that the data source really provides data that meet the application requirements. Given that, the data source relevance problem may be stated as follows.

*Given an application **P** and a data source **S** the problem of identifying the degree of relevance of **S** with respect to **P** may be defined as the problem of identifying how much **S** contributes to meet the application requirements both at the schema level and the data level.*

To calculate the relevance degree of a data source with respect to an application, the first problem to be solved is how to describe both data source and application requirements, and after that, the second problem is how to properly evaluate the data source relevance based on such description. Therefore, two interesting questions arise: (i) *How to describe both data source and application requirements in order to properly allow the evaluation of the degree of relevance of **S** with respect to **P**?*, and (ii) *What kind of criteria may be used in order to help the computation of the degree of relevance of **S** w.r.t. **P**?*

In this paper, we are particularly interested in the second question. Specifically, we propose an approach that uses Information Quality measures for the evaluation of the degree of relevance of a data source w.r.t. a given application.

In the next section, we describe some aspects related with Information Quality, which are relevant for the understanding of our proposal.

3. INFORMATION QUALITY

Information Quality (IQ) has become a critical aspect in organizations and, consequently, in Information Systems research [13]. The notion of IQ has emerged during the past years and shows a steadily increasing interest. IQ is based on a set of dimensions or criteria. The role of each one is to assess and measure a specific quality aspect [18]. In general, IQ researchers assume that there are some shared norms of quality, or quality expectations, and ways of measuring the extent of meeting those norms and expectations. For our purposes, we use the general definition of IQ – 'fitness for use' – which encompasses different aspects of quality [34].

It is important to distinguish the two concepts of Data Quality and Information Quality. IQ is a term to describe the quality of any element or content of information systems [34], not only the data. IQ assurance is the certainty that particular information meets some quality requirements. This leads us to think in a service-based perspective of quality, which focuses on the information consumer's response to his/her task-based interactions with the information system. The use of the term information rather than data implies that the use and delivery of the data must be considered in any quality judgments, i.e., the quality of delivered data represents its value to information consumers [27]. Thus, we use the definition of Information Quality as a set of criteria to indicate the overall quality degree associated with the information in the system [25].

Some interesting IQ approaches were aggregated and compared in the study presented by Scannapieco in [29] and [30]. Such comparative analysis was extended by Batista in [4] to add and analyze IQ approaches more related to data integration. The study has focused specifically on the IQ definitions created in the computer science field in the last years. The study is based on the following affirmatives:

- In the literature, there is no agreement on the set of the dimensions characterizing IQ. Many proposals have been made, but no one has emerged as a standard.
- Even if some dimensions are often considered as important, there is no agreement on their meanings. In different proposals, the same name is often used to indicate semantically different things (as well as different names are used for the same thing). The authors try to approximate some of the similar criteria into a unique definition.

Some of the IQ approaches discussed in [29] and [30] are summarized in the following:

- Wand and Wang [35]: IQ criteria are defined by considering mapping functions from the real world to an information system. For example, inaccuracy means that the information system represents a real world state different from the one of real world. Five dimensions are proposed: accuracy, completeness, consistency, timeliness, and reliability.
- Wang and Strong [34]: Wang and Strong have conceived one of the first set of structured and classified IQ dimensions, which has been considered as a strong reference for most of the studies in IQ area. They empirically identified fifteen IQ criteria under the perspective of a set of users. An empirical approach analyzed the information collected from the users and determined the characteristics of useful data for their tasks. The aspects were grouped into four broad information quality classes: *intrinsic*, *contextual*, *representational*, and *accessibility*. Intrinsic data quality denotes the quality of data itself. Contextual data quality enforces that data quality must be considered within the context of a task at hand, i.e., data must be relevant, timely, complete and appropriate in terms of amount. The Representational data quality category is related to the format and the meaning of data. Accessibility defines if data are available or obtainable for the user.
- Redman [28]: this work groups data quality dimensions into three categories, corresponding to the conceptual view of data, the data values and the data format, respectively. Five dimensions are proposed for the conceptual view, four dimensions for the data values and eight dimensions for the data format.
- Jarke [15]: this work handles the problem of quality in data warehousing. The objective is to establish foundations of data warehouse quality through linking semantic models of the data warehouse architecture to explicit models of data quality. To achieve this, it was produced a general multi-tier DW architecture modeling framework in three levels: source, data warehouse and client. Some quality criteria support this DW architecture.
- Naumann [20]: defines an IQ framework to address the query processing in a data integration system with a mediator-based architecture. This work proposes the interleaving of query planning with quality considerations. The approach distinguishes three classes of quality aspects, which are treated differently: i) Source-specific criteria: determine the overall quality of a data source; ii) Process criteria: determine quality aspects of specific queries that are computable by a source and iii) User query-specific criteria: denote the users preferences.

Only consistency and completeness are dimensions defined in all proposals. Besides these two specific dimensions, consistency-related dimensions and time-related dimensions are also taken into account by all proposals. Specifically, consistency is typically considered at instance level (consistency dimension) or at format level (representational consistency). Time-related quality aspects are mainly represented by the timeliness criterion. Also interpretability is considered by most of the proposals, both at data format and schema level. Each of the remaining dimensions is included only by a minority of proposals. In some cases there is a complete disagreement on a specific dimension definition. More details about the comparative study can be found in [30].

The classifications of quality dimensions discussed in this section have guided a number of other classifications including source selection IQ criteria sets.

4. USING IQ TO EVALUATE THE RELEVANCE OF A DATA SOURCE

In this section, we present our approach to identify relevant data sources by considering some IQ criteria. To this end, at first we introduce the matcher we have used to identify correspondences between data sources. These correspondences are used to measure the proposed IQ criteria. Then, we present a scenario for the IQ criteria definition and propose their specification.

4.1 SCHEMA MATCHING

Reconciling data sources schemas and finding correspondences between their elements (concepts or properties) is still a relevant research issue [12], mainly in distributed environments such as the Web. With respect to that, we have developed a semantic matcher, named *SemMatcher* [26] that considers, besides the traditional terminological and structural matching techniques, a semantic-based one. The matcher produces a set of semantic correspondences between two data sources schemas.

In our work, we consider domain ontologies (DO) as reliable references that are made available on the Web. We use them in order to bridge the conceptual differences or similarities between two data sources schemas. In this sense, first concepts and properties from the two schemas are mapped to equivalent concepts/properties in the DO and then their semantic correspondences are inferred based on the existing semantic relationship between the DO elements. To specify the correspondences, we take into account four aspects: (i) the semantic knowledge found in the DO; (ii) if the schema concepts share super-concepts in the DO; (iii) if these super-concepts are different from the root concept and; (iv) the depth of concepts measured in number of nodes.

We have defined seven kinds of semantic correspondences [31] which were formalized using a notation based on Distributed Description Logics (DDL) [6]. Considering two peer ontologies O_1 and O_2, the correspondences between their elements may be of the following types: *isEquivalentTo,* denoted as $O_1{:}x \equiv O_2{:}y$, *isSubConceptOf*, denoted as $O_1{:}x \sqsubseteq O_2{:}y$, *isSuperConceptOf,* denoted as $O_1{:}x \sqsupseteq O_2{:}y$, *isPartOf* denoted as $O_1{:}x \rhd O_2{:}y$, *isWholeOf* , denoted as $O_1{:}x \lhd O_2{:}y$, *isCloseTo* denoted as $O_1{:}x \approx O_2{:}y$, and *isDisjointWith*, denoted as $O_1{:}x \perp O_2{:}y$.

The *SemMatcher* approach for matching data sources schemas brings together a combination of already defined matching strategies [12]. In this approach, a linguistic-structural matcher and a semantic matcher are executed in parallel. The former may be any existing matching tool including linguistic and/or structural matchers, e.g. H-Match [8]. The latter uses the domain ontology (DO) as background knowledge and identifies the seven kinds of semantic correspondences, as described earlier. The obtained similarity values of both matchers are combined through a weighted average. Each matcher receives a particular weight according to its importance for the matching process.

Therefore, the *SemMatcher* identifies, besides the traditional types of correspondences (equivalence and subsumption), other ones such as closeness and disjointness [31]. Each generated semantic correspondence is ranked according to its level of confidence. We have assigned some weights to them, as follows:

- isEquivalentTo (1.0)
- isSubConceptOf (0.8)
- isSuperConceptOf (0.8)
- isCloseTo (0.7)
- isPartOf (0.3)
- isWholeOf (0.3)
- isDisjointWith (0.0)

The weights reflect the degree of closeness between the correspondence elements, from the strongest relationship (equivalence) to the weakest one (disjointness).

As an illustration, suppose a scenario composed by two data sources S_1 and S_2 which belong to the *Education* knowledge domain. In this scenario, data sources have complementary data about academic people and their works (e.g., Research) from different institutions. Since terminological normalization is a pre-matching step in which the initial representation of two schemas are transformed into a common format suitable for similarity computation, we have normalized both schemas S_1 and S_2 to a uniform representation format according to the DO we have used as background knowledge. The *SemMatcher* then received S_1 and S_2 as input and produced a set of semantic correspondences. We present examples of this set concerning the concept *Faculty* (from S_1) with some related concepts in S_2 in Table 1.

Table 1. Some semantic correspondences between S_1 and S_2

Semantic Correspondences	
S_1:Faculty $\equiv$ S_2:Faculty S_1:Faculty $\sqsubseteq$ S_2:Worker S_1:Faculty $\sqsupseteq$ S_2:Professor	S_1:Faculty $\sqsupseteq$ S_2:PostDoc S_1:Faculty $\approx$ S_2:Assistant S_1:Faculty $\approx$ S_2:AdministrativeStaff

4.2 PROPOSED IQ CRITERIA

In determining the data relevance of a Web data source ***S*** for a specific application ***P***, we consider three IQ criteria – correctness, schema completeness and data completeness. The scenario for IQ criteria specification is detailed in the following.

Let us suppose that an application ***P*** has a set of queries $Q = \{q_1, q_2, \ldots, q_n\}$ representing the users requirements, and it is associated to a set of data sources $DS = \{s_1, s_2, \ldots, s_r\}$, which are candidates to answer the queries in Q. Each query $q_i \in Q$ includes a number of concepts represented by the set $C(q_i) = \{q_i.c_1, q_i.c_2, \ldots, q_i.c_l\}$. Each data source $s_j \in S$ has a source description represented by a set of concepts $C(s_j) = \{s_j.c_1, s_j.c_2, \ldots, s_j.c_m\}$. We also define a set $C(q_i s_j) = \{(q_i.c_1, s_j.c_x, sim(q_i.c_1, s_j.c_x)), (q_i.c_2, s_j.c_{x+1}, sim(q_i.c_2, s_j.c_{x+1})), \ldots, (q_i.c_p, s_j.c_o, sim(q_i.c_p, s_j.c_o))\}$, as a set of triples. Each triple $(q_i.c_p, s_j.c_o, sim(q_i.c_p, s_j.c_o))$ contains a pair of concepts: the first is a concept queried by q_i ($q_i.c_p \in C(q_i)$) and the second is a similar concept that exists in s_j ($s_j.c_o \in C(s_j)$). The third element of the triple is the similarity degree $sim(q_i.c_p, s_j.c_o)$ that indicates how the concept $q_i.c_p$ is similar to the concept $s_j.c_o$. The similarity degree is a value in [0,1] interval, where 0 denotes no similarity and 1 denotes equivalence between the pair of concepts. The set $C(q_i s_j)$, contains only combinations of concepts of query q_i that are also present in the data source s_j ($C(q_i s_j) = C(q_i) \cap C(s_j)$).

Furthermore, we only consider pairs of concepts with similarity degrees that are greater than a minimum threshold.

The similarity values are generated by the *SemMatcher* that analyzes the correspondences between the query concepts and the data source schema and gives as output the set $C(q_i s_j)$. For example, the *SemMatcher* generates the similarity value 1 for two equivalent concepts, 0.8 for subconcepts/superconcepts, 0 for non-similar concepts, among other values. Given that, our intention is to determine if a given data source s_j from *DS* is relevant for ***P*** in terms of its completeness and correctness values, as explained in the following.

Schema Completeness: the schema completeness states that the more concepts queried by the queries in Q are present in a data source s_j, the better is s_j as a relevant data source for the application ***P***. In order to enrich such definition, we also consider the similarity degree $sim(q_i.c_y, s_j.c_x)$ between concepts of q_i and concepts of s_j as a positive weight to be applied over the schema completeness degree. Thus, we extend the completeness criterion presented in [21] and define the schema completeness criterion of a data source s_j as a metric obtained by the quotient between the sum of the number of concepts of $C(q_i s_j)$ and the sum of the number of concepts of $C(q_i)$ (calculated for each q_i) pondered by the similarity degree between the concepts of Q and s_j, represented by the K_{ij} element in the Formula 3.1.

$$SC(Ps_j) = \frac{\sum_{i=1}^{n} |C(q_i s_j)|}{\sum_{i=1}^{n} |C(q_i)|} \cdot \frac{\sum_{i=1}^{n} K_{ij}}{n} \qquad (3.1)$$

where $|C(q_i)|$ is the number of concepts queried by q_i;
$|C(q_i s_j)|$ is the number of concepts queried by q_i that are present in S_j;
n is the number of queries in Q;
K_{ij} is the overall similarity degree of a query q_i w.r.t. a data source s_j, obtained by the following formula:

$$K_{ij} = \begin{cases} \frac{\sum_{r=1}^{|C(q_i s_j)|} sim(q_i.c_r, s_j.c_o)}{|C(q_i s_j)|}, \text{ if } |C(q_i s_j)| > 0 \\ 0, \text{ if } |C(q_i s_j)| = 0 \end{cases} \qquad (3.2)$$

Data Completeness: according to its original definition, data completeness is the ratio of the query answer set size to the total amount of known data [21]. Considering that in Web applications the data returned by querying Web data sources may be incomplete, the data completeness of a data source may be defined as the ratio between the existing suitable set of instances belonging to the data source s_j at query answering time and the set of instances received as results for each query q_i from Q when evaluated over the set of all data sources *DS*. Thus, we define that the data completeness of a data source s_j with respect to Q is calculated by the formula:

$$DC(Ps_j) = \frac{\sum_{i=1}^{n} |q_i s_j instances|}{\sum_{j=1}^{r} \sum_{i=1}^{n} |q_i s_j instances|} \qquad (3.3)$$

where $|q_i s_j instances|$ is the number of instances returned by data source s_j for query q_i;
$\sum_{j=1}^{r} \sum_{i=1}^{n} |q_i s_j instances|$ is the number of instances returned by all data sources in *DS* for all the queries in Q;
n is the number of queries q_i in Q;
r is the number of data sources s_i in S.

Correctness: the correctness is the degree in which the data is free of errors. Considering that a query over a data source returns as answer a set of instances, the correctness of the data source can be measured by the quotient of the number of incorrect instances and the overall number of instances returned by a query, e.g., is the percentage of data without errors [24]. The correctness of a data source s_j with respect to Q is:

$$C(Ps_j) = 1 - \frac{\sum_{i=1}^{n} |q_i s_j instancesWithError|}{\sum_{i=1}^{n} |q_i s_j instances|} \qquad (3.4)$$

where $|q_i s_j instances|$ is the number of instances returned by the data source s_j for query q_i and;
$|q_i s_j instancesWithError|$ is the number of incorrect instances returned by the data source s_j for query q_i.

After measuring the three IQ criteria, we calculate the relevance of a data source s_j to an application ***P***, as described by the following formula:

$$Rel(Ps_j) = [(SC(Ps_j) \times DC(Ps_j)) + C(Ps_j)]/2 \qquad (3.5)$$

The formula combines the data and schema completeness values through a product and calculates a mean between the combined completeness and the correctness score. The final relevance score $Rel(Ps_j)$ is also in [0,1] interval and gives the preference ranking position of the data source s_j w.r.t. *DS*.

5. AN EXAMPLE

To illustrate the IQ criteria assessment, consider four data sources S_1, S_2, S_3 and S_4, belonging to the Bibliographic domain. This domain concerns data (e.g., publications) compiled upon some common principle, as authorship, subject, place of publication, or editor. It is important to note that these data sources are hypothetical and merely illustrative used to show how we can apply our approach. Thus, we have $DS = \{s_1, s_2, s_3, s_4\}$ and $C(s_1) = \{s_1.Researcher, s_1.Journal\}$, $C(s_2) = \{s_2.Paper, s_2.Journal\}$, $C(s_3) = \{s_3.Article, s_3.Publication\}$ and $C(s_4) = \{s_4.Author, s_4.Publisher\}$.

In order to represent the Web application requirements, we have some queries, which were formulated using the Description Logics language ALC (Attribute Language with Complement) [9]. In ALC, the constructors are: $\neg \boldsymbol{C}$ (negation), $\boldsymbol{C} * \boldsymbol{D}$ (conjunction), $\boldsymbol{C} + \boldsymbol{D}$ (disjunction), $\forall \boldsymbol{R}.\boldsymbol{C}$ (universal restriction) and $\exists \boldsymbol{R}.\boldsymbol{C}$ (limited existential restriction) where ***C*** and ***D*** are concepts and ***R*** *is* a role. Since we are regarding only concepts, we use the constructors underlying negation, conjunction and disjunction. Then, we have $Q = \{q_1, q_2, q_3, q_4\}$, where: q_1: *Author* $*$ *Researcher*, q_2: *Publication* $*$ *Article*, q_3: *Book + Journal* and q_4: *Author + Researcher*. For each considered data source, a matching between each $C(q_i)$ and $C(s_i)$ was accomplished. An example of such matching is depicted in Table 2. This example shows the resulting set of semantic correspondences between the application queries and the data source s_1.

After the matching between each $C(q_i)$ and $C(s_i)$, the following similarity scores were obtained:

$C(q_1 s_1)$ = {(q_1.*Author*, s_1.*Researcher*, 0.8), (q_1.*Researcher*, s_1.*Researcher*, 1.0); {(q_1.*Author*, s_1.*Journal*, 0.3), (q_1.*Researcher*, s_1.*Journal*, 0,3)}
$C(q_2 s_1)$ = {(q_2.*Publication*, s_1.*Researcher*, 0.3), (q_2.*Article*, s_1.*Researcher*,0.3); {(q_2.*Publication*, s_1.*Journal*, 0.3), (q_2.*Article*, s_1.*Journal*,0.3)}
$C(q_3 s_1)$ = {(q_3.*Book*, s_1.*Researcher*, 0.3), (q_3.*Journal*, s_1.*Researcher*,0.3); (q_3.*Book*, s_1.*Journal*, 0.8), {(q_3.*Journal*, s_1.*Journal*, 1.0)}
$C(q_4 s_1)$ = {(q_4.*Author*, s_1.*Researcher*, 0.8), (q_4.*Researcher*, s_1.*Researcher*,1.0); (q_4.*Author*, s_1.*Journal*, 0.3), q_4.*Researcher*, s_1.*Journal*, 0.3)}

$C(q_1s_2)$ = {(q_1.Author, s_2.Paper, 0.3), (q_1.Researcher, s_2.Paper,0.3); {(q_1.Author, s_2.Journal, 0.3), (q_1.Researcher, s_2.Journal,0.3)}
$C(q_2s_2)$ = {(q_2.Publication, s_2.Paper, 0.8), (q_2.Article, s_2.Paper,0.8); (q_2.Publication, s_2.Journal, 0.3), (q_2.Article, s_2.Journal,0.3)}
$C(q_3s_2)$ = {(q_3.Book, s_2.Paper, 0.3), {(q_3.Journal, s_2.Paper, 0,3); q_3.Book, s_2.Journal, 0.8), {(q_3.Journal, s_2.Journal, 1.0)}
$C(q_4s_2)$ = {(q_4.Author, s_2.Paper, 0.3),(q_4.Researcher, s_2.Paper, 0.3); (q_4.Author, s_2.Journal, 0.3),(q_4.Researcher, s_2.Journal, 0.3)}

$C(q_1s_3)$ = {(q_1.Author, s_3.Article, 0.3), (q_1.Researcher, s_3.Article, 0.3); (q_1.Author, s_3.Publication, 0.3), (q_1.Researcher, s_3.Publication, 0.3)}
$C(q_2s_3)$ = {(q_2.Publication, s_3.Article, 0.8), (q_2.Article, s_3.Article, 1.0); (q_2.Publication, s_3.Publication, 1.0), (q_2.Article, s_3.Publication, 0.8)}
$C(q_3s_3)$ = {(q_3.Book, s_3.Article, 0.3), {(q_3.Journal, s_3.Article, 0.3); (q_3.Book, s_3.Publication, 0.3), {(q_3.Journal, s_3.Publication, 0.3)}
$C(q_4s_3)$ = {(q_4.Author, s_3.Article, 0.3), (q_4.Researcher, s_3.Article, 0.3); (q_4.Author, s_3.Publication, 0.3), q_4. Researcher, s_3.Publication, 0.3)}

$C(q_1s_4)$ = {(q_1.Author, s_4.Author, 1.0), (q_1.Researcher, s_4.Author, 0.8); (q_1.Author, s_4.Publisher, 0.0), (q_1.Researcher, s_4.Publisher, 0.0)}
$C(q_2s_4)$ = {(q_2.Publication, s_4.Author, 0.3), (q_2.Article, s_4.Author, 0.3); (q_2.Publication, s_4.Publisher, 0.3), (q_2.Article, s_4.Publisher, 0.3)}
$C(q_3s_4)$ = {(q_3.Book, s_4.Author, 0.3), {(q_3.Journal, s_4.Author, 0.3); q_3.Book, s_4.Publisher, 0.3), (q_3.Journal, s_4.Publisher, 0.3)}
$C(q_4s_4)$ = {(q_4.Author, s_4.Author, 1.0), (q_4.Researcher, s_4.Author, 0.8); q_4.Author, s_4.Publisher, 0.0), (q_4.Researcher s_4.Publisher, 0,8)}

Table 2: Similarity values between queries q_1, q_2, q_3 and q_4 and data source s_1

Concept		Similarity Degree Value
Query	**Data Source**	
Q_1.Author	S_1. Researcher	0,8
Q_1.Author	S_1.Journal	0,3
Q_1.Researcher	S_1.Researcher	1,0
Q_1.Researcher	S_1.Journal	0,3
Q_2.Publication	S_1.Researcher	0,3
Q_2.Publication	S_1.Journal	0,3
Q_2.Article	S_1.Researcher	0,3
Q_2.Article	S_1.Journal	0,3
Q_3.Book	S_1.Researcher	0,3
Q_3.Book	S_1.Journal	0,8
Q_3.Journal	S_1.Researcher	0,3
Q_3.Journal	S_1.Journal	1,0
Q_4.Author	S_1.Researcher	0,8
Q_4.Author	S_1.Journal	0,3
Q_4.Publisher	S_1.Researcher	0,8
Q_4.Publisher	S_1.Journal	0,3

As mentioned in the Section 4.2 we only consider pairs of concepts with similarity degrees greater than a threshold, established in 0.7. Scores less than this value are considered absent. In order to calculate the relevance degree of each s_j, consider the values of Tables 3 to 6, which shows the number of concepts queried by q_i; the number of concepts queried by q_i and present in s_j; the overall similarity degree K_{ij} between the concepts of q_i and s_j; the number of instances returned from s_j and the number of incorrect instances returned from s_j. The Tables 7 to 10 show the values of $SC(Ps_j)$, $DC(Ps_j)$, $C(Ps_j)$ and $Rel(Ps_j)$ calculated with formulas (3.1) to (3.5) for each one of the data sources s_j.

Table 3. Intermediate values to calculate IQ criteria for data source s_1

Query	$\|C(q_i)\|$	$\|C(q_is_1)\|$	K_{i1}	q_is_1 instances	$q_is_1$instances with error
q_1	2	2	0.900	574	76
q_2	2	0	0.000	0	0
q_3	2	2	0.900	670	0
q_4	2	2	0.900	1690	334

Table 4. Intermediate values to calculate IQ criteria for data source s_2

Query	$\|C(q_i)\|$	$\|C(q_is_2)\|$	K_{i2}	q_is_2 instances	$q_is_2$instances with error
q_1	2	0	0.000	378	12
q_2	2	2	0.800	533	67
q_3	2	2	0.900	899	118
q_4	2	0	0.000	356	167

Table 5. Intermediate values to calculate IQ criteria for data source s_3

Query	$\|C(q_i)\|$	$\|C(q_is_3)\|$	K_{i3}	q_is_3 instances	$q_is_3$instances with error
q_1	2	0	0.000	9887	300
q_2	2	2	1.000	5455	987
q_3	2	0	0.000	1299	345
q_4	2	0	0.000	577	20

Table 6. Intermediate values to calculate IQ criteria for data source s_4

Query	$\|C(q_i)\|$	$\|C(q_is_4)\|$	K_{i4}	q_is_4 instances	$q_is_4$instances with error
q1	2	2	0.900	829	324
q2	2	0	0.000	2123	155
q3	2	0	0.000	458	11
q4	2	2	0.900	189	185

Table 7. IQ criteria scores for data source s_1

$SC(Ps_1)$	0.5063
$DC(Ps_1)$	0.1132
$C(Ps_1)$	0.8603
***Rel*(Ps_1)**	**0.4588**

Table 8. IQ criteria scores for data source s_2

$SC(Ps_2)$	0.2125
$DC(Ps_2)$	0.0836
$C(Ps_2)$	0.8319
***Rel*(Ps_2)**	**0.4249**

Table 9. IQ criteria scores for data source s_3

$SC(Ps_3)$	0.0625

$DC(Ps_3)$	0.6644
$C(Ps_3)$	0.9041
***Rel*(Ps_3)**	**0.4728**

Table 10. IQ criteria scores for data source s_4

$SC(Ps_4)$	0.2250
$DC(Ps_4)$	0.1389
$C(Ps_4)$	0.8124
***Rel*(Ps_4)**	**0.4218**

At the end of the IQ criteria calculation process, we have obtained that s_3 is the most relevant source to $\boldsymbol{P}$ with a relevance score of approximately 47%. The data sources s_2 and s_4 are the less relevant ones with 42% of relevance.

6. RELATED WORK

In this section we briefly present some of the research literature related to our work. The work presented in [22], for example, has the goal of identifying relevant sources for data linking. They propose an approach, which utilizes keyword-based search to find initial candidate sources for data linking, and ontology matching technologies as a way to assess the relevance of these candidates. Their approach has two main steps: (i) the searching for potentially entities in external data sources and (ii) the filtering of these sources using ontology matching techniques to filter out the irrelevant ones. They also apply a similarity measure between classes of the different sources in order to filter out the ones with low scores. One drawback of this proposal is that, in the filtering stage, only classes with stronger degree of semantic similarity are confirmed. In other words, many relevant classes may be filtered out because they are not considered as exact matches. Our work differs from this one in the sense that their approach focus on finding relevant data sets for linking instead of querying.

The work presented in [33] propose to find "dirty" sources using functional dependencies with probabilities (pFD) in the context of pay-as-you-go data integration systems. During the addition of a new data source, it is possible to decide if the data source is good enough for the system based on the quality of the functional dependencies. It is important to mention that this approach just considers the relational model; in addition it also supposes the presence of a mediated schema.

In [32] is presented an approach to guide the addition of new sources in keyword search-based data integration systems. This process builds a search graph from the sources and its relationships. The search is performed over the graph and the results are returned in a top-k view with the most relevant answers to the user. The graph maintenance is made incrementally through user feedback and when new data sources are discovered the graph is realigned.

The work proposed in [16] uses the user feedback to rank mappings in pay-as-you-go systems. The approach uses the concept of VPI (value of perfect information) as a metric to rank. VPI provides a means of estimating the benefit to the pay-as-you-go system in such a way that it is possible to evaluate the correctness of a candidate matching based on the user feedback. This concept is based on the utility function that quantifies the quality of query's results. Similarly to VPI, in our approach we generate a relevance value. However, we are interested in rank data sets instead of ranking mappings.

Concerning IQ, some works uses IQ criteria to select data sources for a specific task. Naumann *et al.* in [19] proposes three IQ criteria for source selection: availability, understandability and extent. The source selection is executed to determine which sources are more suitable to answer queries. The approach uses the Data Envelopment Analysis (DEA) [9] to create an efficiency ranking of the data sources in terms of the three criteria. More recently, Bleiholder and Naumann [5] have proposed the IQ criteria – completeness and conciseness – to be applied in data integration and as a base to data fusion process. Batista in [3] proposed the criteria of availaibility, response time, timeliness and access frequency of a data source. These criteria are used to decide if some content of a data source should be locally materialized in a data integration system. Zhu and Buchmann [36] use Multi-Criteria Decision Making (MCDM) methods to select the best sources to extract content to a data warehouse. They define the criteria of availability, accessibility, correctness, completeness, relevance and presentation, calculate IQ scores using several MCDM methods and at the end they compare the results.

Dustdar *et al*. in [11] emphasize the importance of source selection depending on user needs. They suggest the use of quality aspects to guide the selection and propose a quality-aware data service as a data source model for data integration tasks and mashups. Daas and Ossen [10] have compiled a large set of metadata quality aspects and use them to select data sources for research purposes. They also proposed a set of indicators and methods to evaluate each IQ aspect. Their proposed list of IQ criteria includes the relevance criterion.

All the works discussed in this section address the source selection problem. Some of them apply concepts of IQ to guide the source selection process [19, 5, 3, 36, 11, 10]. Specifically, [2, 11, 10] include the relevance criterion as one of the IQ aspects to be considered, while [5, 36] use completeness aspects to guide source evaluation. None of the works discussed in this section uses similarity weights in the IQ assessment to explore the semantic aspects between data sources and queries.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a discussion about the problem of identifying relevant Web data sources for domain-specific applications. We focused our discussion on the distinction between what data is expected and what data is really available both at the Web data sources and the applications defined over them. In order to help the identification of relevant Web data sources, we proposed an approach to calculate the degree of relevance of a given data source based on the following IQ criteria: correctness, schema completeness and data completeness. We argue that using these criteria it is possible to evaluate how much a data source contributes to meet the application requirements both at the schema level and the data level.

As future work, we plan to conduct some experiments with real data sources and applications to validate our proposal. Moreover, there are still some interesting questions to investigate, for example, the correlation between the scores of completeness and correctness, and how to describe both data sources and application requirements in order to properly allow the evaluation of the degree of relevance of a data source with respect to a given application.

8. ACKNOWLEDGEMENT

This work was partially supported by the National Institute of Science and Technology for Software Engineering (INES[6]), funded by CNPq and FACEPE, grants 573964/2008-4 and APQ-1037-1.03/08.

9. REFERENCES

[1] David Aumueller, Hong-Hai Do, Sabine Massmann, and Erhard Rahm. 2005. Schema and ontology matching with COMA++. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data* (SIGMOD '05). ACM, New York, NY, USA, 906-908. DOI=10.1145/1066157.1066283 http://doi.acm.org/10.1145/1066157.1066283

[2] Baader, F., Calvanese, D., McGuinness, D., Nardi D., and Patel-Schneider P. 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.

[3] Batista, M. C. M. Otimização de Acesso em um Sistema de Integração de Dados através do uso de Caching e Materialização de Dados, Master Thesis, Federal University of Pernambuco, 2003.

[4] Batista, M. C. M. Schema Quality Analysis in a Data Integration System. PhD Thesis, Federal University of Pernambuco, 2008.

[5] Bleiholder, J. and Naumann, F. Data fusion. ACM Comput. Surv., 41, 1, Article 1, 2008. DOI = 10.1145/1456650.1456651 http://doi.acm.org/10.1145/1456650.1456651

[6] Borgida A. and Serafini L. 2003. Distributed description logics: Assimilating information from peer sources. Journal of Data Semantics, 1:153–184. LNCS 2800, Springer Verlag, 2003.

[7] Cafarella, M. J., Halevy, A. Y., Madhavan J. 2011. Structured data on the web. Commun. ACM. 54, 2 (January 2011), 72-79. DOI = http://doi.acm.org/10.1145/1897816.1897839

[8] Castano S., Ferrara, A., and Montanelli, S. 2006. Matching ontologies in open networked systems: Techniques and applications. Journal on Data Semantics, 3870: 25-63, 2006.

[9] Charnes, A., Cooper, W. and Rhodes, E. Measuring the efficiency of decision-making units. European Journal of Operational Research, 2: 429–444, 1978

[10] Daas, P.J.H. and Ossen, S.J.L. 2011. Metadata Quality Evaluation of Secondary Data Sources. In *Proceedings of 5th International Quality Conference*, Center for Quality, Faculty of Mechanical Engineering, University of Kragujevac. 2011.

[11] Dustdar, S., Pichler, R., Savenkov, V. and Truong, H.L. Quality-aware service-oriented data integration: requirements, state of the art and open challenges. *SIGMOD Rec.* 41, 1, 2012. DOI=10.1145/2206869.2206873 http://doi.acm.org/10.1145/2206869.2206873

[12] Euzenat, J., Shvaiko, P. Ontology Matching. Springer, Heidelberg (2007).

[13] Ge, M. and Helfert, M. 2007. A Review of Information Quality Research - Develop a Research Agenda, In Proceedings of 12th ICIQ, 2007.

[14] Herden, O. 2001. Measuring Quality of Database Schema by Reviewing - Concept, Criteria and Tool. In Proceedings of 5th International Workshop on Quantitative Approaches in Object-Oriented Software Engineering, p. 59-70, 2001.

[15] Jarke, M., Jeusfeld, M.A., Quix, C. and Vassiliadis, P. Architecture and Quality in Data Warehouses: An Extended Repository Approach, in: Information Systems 24 (3), p. 229-253, 1999.

[16] Jeffery S. R., Franklin M. J., Halevy A. Y. 2007. Soliciting User Feedback in a Dataspace System. Electrical Engineering and Computer Sciences University of California at Berkeley, 2007.

[17] Keeton, K., Mehra, P., Wilkes, J. 2009. Do you know your IQ? A research agenda for information quality in systems. SIGMETRICS Performance Evaluation Review. 37, 3 (December 2009). 26–31. DOI = http://doi.acm.org/10.1145/1710115.1710121

[18] Lee, Y. W., Strong, D. M., Khan, B. K. and Wang, R. Y. AIMQ: A Methodology for Information Quality Assessment, in: Information & Management 40 (2), p. 133 - 146, 2002.

[19] Naumann, F. Leser, U. and Freytag, J. C. Quality-driven integration of heterogenous information systems. In Proc. of the 25th Int. Conf. on Very Large Data Bases (VLDB'99), pages 447–458, 1999.

[20] Naumann, F. Quality-Driven Query Answering for Integrated Information Systems, in: LNCS 2261, 2002.

[21] Naumann, F., Freytag, J.C. and Leser, U. 2004. Completeness of Integrated Information Sources, *Information Systems,* 29, 7 (October 2004), 583-615. DOI = http://dx.doi.org/10.1016/j.is.2003.12.005

[22] Nikolov, A., d'Aquin, M. 2011. Identifying Relevant Sources for Data Linking using a Semantic Web Index. In *Proceedings of the Linked Data on the Web*, LDOW 2011, Hyderabad, India.

[23] Oliveira, H. R., Tavares, A. T., Lóscio, B. F. 2012. Feedback-based Data Set Recommendation for Building Linked Data Applications. In *Proceedings of 8th Int. Conference on Semantic Systems* (Vienna, Austria, September 5-7, 2012) I-SEMANTICS 2012.

[24] Olson, J.E. 2003. *Data Quality: The Accuracy Dimension*. Morgan Kaufmann.

[25] Pipino, L. L., Lee, Y. and Wang, R. Data Quality Assessment, in: ACM 45 (4), p. 211 - 218, 2002.

[26] Pires C. E, Souza D., Pacheco T., and Salgado A. C. 2009. A Semantic-Based Ontology Matching Process for PDMS. In *Proceedings of 2nd International Conference on Data Management in Grid and P2P Systems* (Linz, Austria, September 1-2, 2009), GLOBE 2009, Springer-Verlag, Berlin, Heidelberg, 124-135. DOI = http://dx.doi.org/10.1007/978-3-642-03715-3_11

[27] Price, R. and Shanksa, G. Semiotic Information Quality Framework, in: DSS, p.658 - 672, 2004.

[28] Redman, T. C. Data Quality for the Information Age, in: Artech House, 1996.

[6] www.ines.org.br

[29] Scannapieco, M. and Catarci, T. Data Quality under a Computer Science Perspective, Journal of Archivi & Computer, 2003.

[30] Scannapieco, M. Data Quality at a Glance, in: Datenbank-Spektrum 14, p. 6-14, 2005.

[31] Souza D., Arruda T., Salgado A. C., Tedesco P., and Kedad, Z.: Using Semantics to Enhance Query Reformulation in Dynamic Environments. In: 13th East European Conference on Advances in Databases and Information Systems (ADBIS'09), Riga, Latvia, pp. 78-92 (2009).

[32] Talukdar, P. P., Ives, Z. G., Pereira, F. 2010. Automatically incorporating new sources in keyword search-based data integration. In *Proceedings of the ACM SIGMOD International Conference on Management of Data* (Indianapolis, Indiana, June 6-11, 2010) SIGMOD 2010, ACM, New York, NY, 387-398. DOI = , http://doi.acm.org/10.1145/1807167.1807211

[33] Wang, D. Z., Dong, X. L., Das Sarma, A., Franklin, M. J., Halevy, A. Y. 2009. Functional Dependency Generation and Applications in *Pay-as-you-go* Data Integration Systems. In *Proceedings of the 12th International Workshop on the Web and Databases* (Providence, Rhode Island, June 28, 2009) WebDB 2009.

[34] Wang, R. Y. and Strong, D. M. Beyond Accuracy: What Data Quality Means to Data Consumers, in: JMIS 12 (4), p. 5 - 33, 1996.

[35] Wand, Y and Wang, R.Y. Anchoring Data Quality Dimensions in Ontological Foundations, in: ACM, 39 (11), p. 86-95, 1996.

[36] Zhu, Y. and Buchmann, A. Evaluating and Selecting Web Sources as External Information Resources of a Data Warehouse. In Proceedings of the 3rd International Conference on Web Information Systems Engineering (WISE' 2002), Singapore, 2002.