Semantic Recognition of a Data Structure in Big-Data

Aïcha Ben Salem¹², Faouzi Boufares¹, Sebastiao Correia²

¹Laboratory LIPN-UMR 7030-CNRS, University Paris 13, Sorbonne Paris Cité, Villetaneuse, France
²Company Talend, Suresnes, France

Email: bensalem@lipn.univ-paris13.fr, boufares@lipn.univ-paris13.fr, abensalem@talend.com, scorreia@talend.com

Received April 2014

Abstract

Data governance is a subject that is becoming increasingly important in business and government. In fact, good governance data allows improved interactions between employees of one or more organizations. Data quality represents a great challenge because the cost of non-quality can be very high. Therefore the use of data quality becomes an absolute necessity within an organization. To improve the data quality in a Big-Data source, our purpose, in this paper, is to add semantics to data and help user to recognize the Big-Data schema. The originality of this approach lies in the semantic aspect it offers. It detects issues in data and proposes a data schema by applying a semantic data profiling.

Keywords

Data Quality, Big-Data, Semantic Data Profiling, Data Dictionary, Regular Expressions, Ontology

1. Introduction

The general management and business managers must have a unified vision and usable information to make the right decisions at the right time. The data quality governance has become an important topic in companies. Its purpose is to provide accurate, comprehensive, timely and consistent data by implementing understandable indicators, easy to communicate, inexpensive and simple to calculate. In the big-data era, the quality of the information contained in a variety of data sources, is becoming a real challenge.

Data quality and semantics aspects are rarely joined in the literature [1]-[3]. Our challenge is to use semantics to improve the data quality. Indeed, misunderstanding of the data schema is an obstacle to define a good strategy to correct any anomalies in the data. Very often metadata are not enough for understanding the meaning of data.

For a given data source S, we propose a semantic data profiling to get better understanding of the data definition and improve anomalies detection and correction. No schema available to understand the meaning of data and even less to correct them. There are currently no tools [4]-[8] that bring the strings “Pékin” to “Beijing” or even “Londres” to “London”. Additional semantic information is needed to know that these strings represent the same category and subcategory of information. Similarly, it is important to recognize semantically the meaning

of the string “16˚C” which is a city temperature in degree Celsius.

Let $S$ be an unstructured data source, result of integration of multiple heterogeneous data sources. $S$ can be seen as a set of strings, separated by semicolons (;). $S$ can then be described by the set $C$ of all its columns. One note $S(C)$ the data schema. Notice that the source $S$ has no defined structure, which can cause a problem for semantic data manipulation. $S$ may contain inconsistencies (Figure 1). Several questions arise such as: 1) what are the semantics of strings? 2) What are the languages used? 3) What is invalid and what is not?

Let us remark that this source has several columns. $S$ is defined by $(\text{Col}_i, i = 1;7)$.

In the data source $S$, the column Col4 should contain only cities given in English. London and Beijing are syntactically and semantically valid. While, “Pékin” and “Londres” are syntactically correct and semantically invalid. “Londre” is syntactically invalid. The Col2 column contains mostly dates. Therefore, the “13” value will be considered semantically invalid. This demonstrates the need of more semantics to understand and correct the data.

This paper is organized as follows. The second section presents the meta-information required for the semantic data structure. The semantic data profiling process is given in the third section. Our contribution and future works are given in conclusion.

2. Meta-Information

We discussed in the previous works [9] [10] various problems of data quality in particular the deduplication one. We started the development of a new kind of Big-Data ETL based on semantic aspects. It allows data profiling, data cleaning and data enrichment.

To assist the user in his quality approach, the originality of our work lies in: semantic recognition of descriptive data schema and hence fortification data themselves. We will focus, in this paper, to the data profiling step.

Data profiling presents the first step in the data quality process (DQM tool Figure 2). It is a quantitative analysis of the data source to identify data quality problems. It includes descriptive information such as schema, table, domain and data sources definitions. As a result, data profiling collects summaries of the data source (Number of records, attributes) [11] [12].

However, existing data profiling tools [13]-[16] provide a statistical data profiling and do not address the semantic aspects. For that, the purpose of this paper is to introduce some semantic indicators to enrich the data profiling process and propose a semantic one.

For the semantic data profiling, we propose for each input data source $S$, a bug report, log for updates and a new semantic structure using some meta-information.

The bug report contains the various existing anomalies in the data source: more than one category and language used for the same column, different data formats, duplicates, null values.

Log for updates is the set of update actions to be applied to a data source such as translation in the same language, homogenization in the same format. These updates cover one column at a time. In order to make corrections
between columns, the concept of functional dependencies has to be applied.
This meta-information can be enriched over the time (more details will be presented in the Section 3.3).
In the following, we will be interested in the semantic data profiling process details (presented in Figure 3) and in particular to the meta-information.
The meta-information consists of three components: the Meta-Schema-Ontology (MSO), the Meta-Repository (MR) composed by the DD and RE and the list I of indicators.
Several tables (T_k, k = 1,7) are used to store the different artefacts corresponding to the results of the semantic data profiling process.
Let us start by defining the first component, the Meta-Schema-Ontology (MSO).

2.1. Meta-Schema-Ontology
A database, as a set of information, can be described in many different ways. The difference is mainly in the name of concepts and attributes.
The idea with the MSO is to store all these equivalent descriptions in a meta-structure. The Meta is presented with the UML [17] (Unified Modeling Language) class diagram (Figure 4).
MSO is a set of knowledge that can be managed as ontologies [18]-[22]. Ontology is a formal language. It is a grammar that defines how terms may be used together. Ontologies allow sharing a common understanding of the information structure among people.
Many instances (knowledge) can be created from the MSO. For instance, Person, Organization and Invoice are three Concepts. Each of them may have several synonyms.
For instance, the concept Person can have many synonyms such as Client, Student and Customer. The concept Person is defined by some Attributes like FirstName, Address, City, Country and BirthDate. This implies that each synonym of the concept Person can be defined in a similar manner. The ontology is viewed with the

![Figure 3. The semantic data profiling process.](image)

![Figure 4. The Meta-Schema-Ontology UML class diagram.](image)
open source Protégé tool [23] (Figure 5).

This knowledge can evolve over the time according to different descriptions of the databases and it can be represented as a meta-repository.

2.2. Meta-Repository

The meta-repository is a set of knowledge describing the data dictionary (set of categories in different languages), regular expressions and a list of indicators (statistical, syntactic and semantic).

2.2.1. Data Dictionary

Valid strings (syntactically and semantically) can be grouped into categories. Categories describe concepts. These descriptions (strings) can be in several languages. They may also contain sub-categories. The set of categories Cat\textsubscript{ext} can be seen as a data dictionary. For example, the monument category will contain all valid strings describing the airports, universities, hospitals, museums and castles names. The names of cities, countries and continents where are these monuments, are also part of data dictionary (DD).

Let Cat\textsubscript{ext} be the set of categories defined by extension: \text{Cat}_{\text{ext}} = \{\text{Cat}_i, i = 1:n\} with \text{Cat}_i belongs to \{\text{FirstName, Country, City, Civility, Gender, Email, Web Site, Phone Number}\}. For each \text{Cat}_i, a set of sub-categories Sub\text{Cat} = \{\text{Cat}_{ij}, j = 1:m\} can be defined. In this study, language is used as a sub-category. The set of languages used is Lang = \{English, French, German, Italian, Portuguese, Spanish\}.

We define the DD as a set of triplets of (Category, Information, Language). A category \text{Cat}_i is then defined by extension where Information is a valid string, Category \in \text{Cat}_{\text{ext}} and Language \in \text{Lang}.

Note that, as mentioned in the Figure 6, the information “France” can refer to two categories in the same time: Country and FirstName. Other exceptions may exist.

![Figure 5. An instance of the Meta-Schema-Ontology under Protégé.](image)

![Figure 6. A sample of the data dictionary.](image)
2.2.2. Regular Expressions
A category $K_\text{ati}$ can also be defined by intention using regular expressions (RE). These are used to validate the syntactic and semantic of strings. Let $K_\text{ati}_\text{set}$ be the set of these categories.

RE can be defined as a set of pairs $\text{Catregex} (\text{Category, Regular-Expressions})$.

$$RE = \left\{\text{Catregex}_i (K_\text{ati}, \text{Regex}_ij); i = 1...p, j = 1...q\right\}$$

Some instances of categories are presented in Figure 7.

2.2.3. Indicators
The semantic data profiling is based on a set $I$ of $p$ indicators applied to the data source. Most of the existing tools are interested only in quantitative summaries of the source data. Few tools focus on semantic analysis. For that, we propose semantic indicators. $I$ is composed of three types of indicators (Figure 8): statistic indicators $\{I_{\text{stati}}, i = 1;p\}$, two syntactic indicators ($I_{\text{SYN1,2}}$) and two semantic ones ($I_{\text{SEM1,2}}$).

After presenting in this paragraph, the input data for semantic data profiling, we will outline below, the process itself.

3. Semantic Data Profiling Process
Let us give some notations and definitions used in the algorithm of the semantic data profiling process.

Each column $C_i$ belonging to the data source $S$, has a set of values $v_i (i = 1...n)$. Each $v_i$ has a data type such as $\{\text{String, Number, Date, Boolean, list or range of values}\}$.

Definition 1: Syntactic validity of a value $v$
A value $v$ is syntactically valid if and only if (iff) $v \in RE$ or $v \approx w \in DD$. ($\approx$ means similar using similarity distances [5] [6]).

Definition 2: Syntactic invalidity of a value $v$
A value $v$ is syntactically invalid iff $v \notin RE$ and $v \notin DD$.

Definition 3: Dominant Category
Let $C_\text{ati}(v)$ be the number of syntactically correct values for a given attribute.

A category is a dominant category iff $C_\text{ati}(v) > C_\text{ati}(v)$ with $i \neq j$.

The “Number of categories” indicator defines the number of categories detected.

![Figure 7. A set of regular expressions.](image_url)

![Figure 8. A set of indicators.](image_url)
Definition 4: Semantic validity of a value $v$
A value $v$ is semantically valid iff $v \in \text{Cat}_i$, and $\text{Cat}_i$ is the dominant category.

Definition 5: Semantic invalidity of a value $v$
A value $v$ is semantically invalid iff $v \notin \text{Cat}_i$, and $\text{Cat}_i$ is the dominant category.

3.1. Profiling Algorithm

The principle of semantic data profiling algorithm (Figure 9) is to check if a value $v$ belongs to the meta-repository. The aim is to verify the syntactic and semantic validity of $v$.

Given the data source $S$ and the meta-information as inputs, the algorithm returns several tables ($T_k$, $k = 1,7$). These contain indicators results, invalid syntactic data, valid syntactic data, invalid semantic category-data, invalid semantic language-data and the new semantic structure.

The statistic Indicators function consists on applying different statistical indicators for a general summary (total number of values, number of duplicate values, pattern frequency) or according to the data type such as year Frequency, Maximum Length, Minimum Length.

The role of the semantic Recognition Structure function is trying to find a category and language for each data ($v$) using RE or DD. The three steps below will describe the principle of this function. Note that if $v$ is a string, several possibilities are considered. Two types of research are used according to the presence or absence of keywords.

The first step is to check if $v$ satisfies the definition 1. $v$ is then considered syntactically valid. Then, we check the semantic validity (definition 4) using the dominant category concept (definition 3). This step allows obtaining the category and language for each column.

The second step deals, in one hand, with semantically invalid values (definition 5), remind that they are syntactically correct. In the other hand, this step processes with syntactically invalid ones (definition 2).

In the third step, the syntactically correct and semantically incorrect values are handled in several ways. According to their membership to the dominant category and the selected language, updates are automatically proposed such as homogenization, translation and standardization.

Whenever, the syntactically invalid values are well spelled (satisfy some regular expressions), they can be used to enrich the DD.

As there may be several languages for each column, not only one has to choose the dominant language column but also the dominant language of the source studied. The principle is presented in the semantic Language function.

The details of these functions (statistic Indicators, semantic Recognition Structure, semantic Language) are presented in Appendix (Figure A1).

The following paragraph will present the intermediate results.

3.2. Profiling Results

Several tables are used to store the different artefacts corresponding to the results of the semantic data profiling process.

![Figure 9. Semantic data profiling algorithm.](image-url)
The first one contains indicators results. For each column, we have some statistical summaries (e.g. percentage of null values), the number of invalid syntax values, the number of valid syntax values, the number of detected categories and number of detected languages.

The misspelled values are automatically added to the invalid syntax table (second table).

The third table contains the values, syntactically correct, which do not belong to Meta-Repository. They will be designated unknown categories.

For each column of the data source, we can have more than one category. So, to validate the dominant category, we choose the one with the greater percentage. The percentage is calculated based on the number of values that belong to this category. If we have two categories with the same percentage, we choose another sample from the data source and apply the semantic data profiling.

The values that do not belong to the dominant category are stored in the table T4 as semantic invalid category-value. In the same way, values that do not belong to the dominant language are stored in the table T5 as semantic invalid language-value.

Note that each column $C_i$ of the source S is seen initially as a string. The goal is to recognize its semantic meaning (Figure 10). The dominant category and language are used to define the semantic structure for a data source.

Data source may contain similar columns, noted $Col_i \leq Col_j$. For instance, Temperature_1 and Temperature_2 columns are similar categories ($Col_i \leq Col_j$). When two columns $Col_i$ and $Col_j$ belong to the same semantic category and have the same content ($Col_i = Col_j$), one of the two columns should be deleted.

3.3. Semantic Enrichment

As mentioned before, the meta-information must be enriched with new information. Both the data dictionary and the Meta-Schema-Ontology can be enriched.

The content of the DD may evolve using the values in T3, which must exist in some lexical databases such as WordNet [24] and WOLF [25]. Similarly, when new categories are discovered after the semantic data profiling, the Meta-Schema-Ontology is expanded using new Attributes and their synonyms synAttributes.

Users can also enrich the meta-information with new regular expressions.

4. Conclusions and Contribution

Big data often have even less metadata than usual databases and that's a problem when the data scientist wants to perform analyses on these data. The use of our DQM tool would help the data scientist in recognizing data types (integer, dates, strings) and data semantics (Email, FirstName, Phone). The semantics would then be useful to automatically suggest views on data with a semantic meaning or to find matches between heterogeneous structures in big data.

DQM tool that we are currently developing is a contribution to new generation of Big-Data ETL based on semantics. Our goal is to guide the user in his quality approach.

In the case of the absence of the data structure, we help the user:

1) To understand more the definition of manipulated data. Indeed, during the integration process for the union or the join operations, it is essential to differentiate synonyms and homonyms to succeed semantic data integration. Existing tools [14]-[16] [26] do not take into account semantic aspects. Only the syntactic ones are considered. For instance, in the case of the data integration process, user can choose to join two columns syntactically equivalent but semantically not $S1.Col1$ and $S2.Col1$ can be synonyms or homonyms (Figure 11). The union of $S1$ and $S2$ is semantically meaningless, while existing tools allow this operation. DQM tool alerts users to
incorporate semantic integration operations.

2) Throughout the laborious cleaning step. Transformation and homogenization that we propose will allow better elimination of duplicate or similar tuples. In fact, recalling that no method of calculating similarity distance permits the approximation between Pékin and Beijing, for example, because information on the language used is not taken into account. Our approach allows this reconciliation. The originality of our approach is to infer the semantics of the data source structure using on one hand, the data itself and on the other hand, instances of the Meta-Schema-Ontology. Furthermore, our approach allows us to automatically propose cleaning actions on unstructured data. This constitutes part of our current and future work using MapReduce concepts [13] [27].

The results of the data profiling process are: 1) a data structure for better understanding of the semantic content of Big Data, 2) a set of updates for the correction of invalid data.

The semantic structure of the Big-Data source is:

\begin{align*}
S & \left( \text{Col1\_FirstName: String, Col2\_Date: Date,} \\
& \text{Col3\_Address: String, Col4\_City: String,} \\
& \text{Col5\_Country: String, Col6\_Temperature\_1: Number,} \\
& \text{Col7\_Temperature\_2: Number} \right)
\end{align*}

The target data after the cleaning actions should be for instance (Figure 12).

**References**


Appendix

Function statisticIndicators (Column C)
//return statistical indicators results
Begin
For each Id from 1 do //d=1..18
Add(Id (C), T1c)
//statistic indicators: total number of values, number of null values…
end for
End statisticIndicators

Function semanticLanguage (Data Source S')
//return the dominant language
Begin
For each Languagei from T7 (i=1..n) //T7 is the semantic structure
ni:= Count the number of occurrences (Languagei)
End for
DominantLanguage := Language where Max(ni)
End semanticCategories

Function semanticCategories (Column C)
//return syntactic and semantic indicators results and semantic structure
Begin
For each vj from C do //j=1..m (m number of tuples)
If vj∈ RE then add(vj, Catj, Langj) //vj∈Catj and vj∈Langj
elseif vjcheckSpelling=true
//verifies some regular expressions for strings
then if vj≈ w ∈ DD //w a value from DD
then add(vj, Catj, Langj) //vj∈Catj and vj∈Langj; j'≠j
else add(vj, CatUNKNOWN)
//vj∈Unknown Category
add(vj, T3c) //vj is a candidate to enrich DD
else add(vj, T2c)
end if
end if
end For
add(Isem1(C), T1c) //number of used categories
add(Isem2(C), T1c) //number of used languages
add(Isyn1(C), T1c) //number of valid syntax value
add(Isyn2(C), T1c) //number of invalid semantic value
add((Catdom, Langdom), T7c) where %Catdom =Max(%Catp) //p=1..x
and %Langdom =Max(%Langq) //q=1..y
add(Catp', T4c) where p' ≠ p
add(Langq', T5c) where q' ≠ q
End semanticCategories

Figure A1. Functions of the semantic data profiling algorithm.