CONTEXT ANALYSIS FOR SEMANTIC MAPPING OF DATA SOURCES USING A MULTI-STRATEGY MACHINE LEARNING APPROACH

Youssef Bououlid Idrissi and Julie Vachon
DIRO, University of Montreal
Montreal (Quebec), Canada

Keywords: context analysis, semantic mapping, data sources alignment, machine learning, multi-strategy, semantic web.

Abstract: Be it on a webwide or inter-entreprise scale, data integration has become a major necessity urged by the expansion of the Internet and of its widespread use for communication between business actors. However, since data sources are often heterogeneous, their integration remains an expensive procedure. Indeed, this task requires prior semantic alignment of all the data sources concepts. Doing this alignment manually is quite laborious especially if there is a large number of concepts to be matched. Various solutions have been proposed attempting to automatize this step. This paper introduces a new framework for data sources alignment which integrates context analysis to multi-strategy machine learning. Although their adaptability and extensibility are appreciated, actual machine learning systems often suffer from the low quality and the lack of diversity of training data sets. To overcome this limitation, we introduce a new notion called "informational context" of data sources. We therefore briefly explain the architecture of a context analyser to be integrated into a learning system combining multiple strategies to achieve data source mapping.

1 INTRODUCTION

Machine learning systems using a multi-strategy approach are composed of a set of basic learners. Although independent, these learners are coordinated by a special unit called meta-learner. These machine learning systems are adaptive since they can deploy learners able to specialize in the processing of a specific type of information (e.g. field names, data types, etc.). These systems are also easily extensible since new learners, developed independently, can naturally be integrated under the control of a meta-learner. Each basic learner is responsible for automatically mapping the elements of two given data sets (source data onto target data) according to its specific knowledge. The main tasks of basic learners are the following:

- learning data mappings from training examples provided by a user.
- generating mappings between new data sets by using the classification model settled during the training stage.

As for the meta-learner, its task is to combine all the mapping proposals issued by the basic learners and to compute a final matching for each concept present in the source data set.

It is well-known that the precision of the mapping directly depends on the quantity and quality of the information used in the training set. Most of the time, this training set is composed of data and their corresponding data scheme (XSD, DTD, RDFS, etc.). This selection appears to be too thin and restrictive if one hopes to unveil semantic ambiguities which makes it hard to identify implicit relations between concepts. Achieving accurate semantic analysis is thus a major challenge. Here are some examples of problems one can meet:

- The attribute `date1` of an entity `Command` does not indicate if it represents the invoicing date, the delivery date or the reception date.
- The attribute `name` of an entity `Employee` does not specify if the content is about the first name or the full name of the employee.
- The attribute `amount` of an entity `Invoice` does not allow one to know if indicated values include tax or not, neither does it specify the used currency type.
- The attribute `cl` of a entity `Command` uses an ab-

Bououlid Idrissi Y. and Vachon J. (2005).
CONTEXT ANALYSIS FOR SEMANTIC MAPPING OF DATA SOURCES USING A MULTI-STRATEGY MACHINE LEARNING APPROACH.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems*, pages 445-448
DOI: 10.5220/0002539804450448
Copyright © SciTePress

breviated form which makes it hard to automatically identify that it denotes a command line.

Taken alone, a specialized learner can prove inefficient or inadequate. For example, a learner specialized in the mapping of field names does not prove particularly outstanding when it comes to match names which are not well-known synonyms (e.g. "'comment"' and "'outline"'), or names which abbreviate a concept (e.g. the name "'home"' used instead of "'telephone at home"') or names whose broad meaning would allow them to be matched with almost everything (e.g. "'thing"' or "'entity"'). Similarly, a content learner, that bases its semantic deduction on the frequency of lexical units appearing in fields, would prove quite inadequate to analyze numerical fields! Moreover, a learner relying on a naive bayesian approach (Berlin and Motro, 2002; Pedro Domingos, 1997; Kohavi, 1996) would not be profitable for analyzing fields accepting values of numerical or enumerated types (e.g. "'gender"').

Hence, this article proposes to broaden and diversify both data sets and training sets by extending them with documents coming from, what we call, the *informational context* of data sources. Indeed, isolating a data source from its context (as it is the case when solely considering its XML schema) reduces beforehand the set of usable cognitive information which underlies the conceptualization of the represented data. In practice, the context within which the data source lies, constitutes a precious fount of information calling for a more systematic exploration so as to better define the semantics of concepts.

In the sequel, the notion of *informational context* is defined and the architecture of a context analyzer is presented.

2 INFORMATIONAL CONTEXT

The informational context of a data source is composed of all the information, saved in electronic format, which belongs to the data source's environment and shares the same domain.

1. The *descriptive context* of a data source gathers all the specification files describing the data or their application environment. These files document the data according to various abstraction levels. For example, if the data source is a database the descriptive context could be composed of the following documents:
   - A requirements document describing data and services which the user calls for in applications using the database. A test plan for instance, might be practical to establish the link between input and output data what could hide relevant complex concepts.
   - Analysis and design specifications including the various formal and semi-formal models elaborated for applications relying on the database. Data dictionnaries are worth citing under this category. It describes in a formal fashion, among others, data flows, data structures et data deposits. As an example, consider a structure description of the concept "Order", using regular expressions:

     $$Order = O\_Header + O\_Item^* + O\_Footer$$
     $$O\_Header = O\_Number + Date + CustAdress$$
     $$O\_Item = ItemNum + Descr + Qty + Price$$
     $$O\_Footer = TaxAmount + TotalAmount$$

     This provides relevant information about compositions and dependencies of "Order" and "Items" concepts. Furthermore, detailed description of each data element can be obtained from a data description deposit.
   - User manuals. In the same way as for dictionaries, one can think of the numerous formula linking concepts present in a such resource.
2. The *operational context* of a data source is composed of all the data management and processing files. Among others, these files can be
   - programs written in any known programming paradigm and language. The way concepts are manipulated could hide valuable information about how they are linked to each other.
   - Files containing SQL-type requests.

For each data source, the important is to list all the documents which may compose the descriptive and operational contexts of this data set. The analysis of these documents (in addition to the analysis of the data themselves and their schema definition) will help enhancing the knowledge required by the learners to deduce the best semantic mapping between the given data sources.

3 CONTEXT ANALYSIS

The main objective of context analysis consists in expanding data sources with semantic information and hints drawn from the context. Among other, this information is intented to be used by learners during their training stage to increase the precision of the mapping they are asked to compute.

In particular, context analysis offers an interesting opportunity for resolving complex mappings which are pairing *combinations* of concepts (e.g. `(street, zip code, city)` → `employee_address`). To the best of our knowledge, there is still no satisfactory solution addressing this problem although it is frequently encountered. For example,

we can imagine a context analyzer detecting the presence of a function call "'`concat(address, zip_code, city)`"' in a program file. The analyzer should therefore be able to add the definition of a new concept `complete_address` to the concerned data source and present it as a new candidate to be mapped (hence if `complete_address` → `employee_address` then our problem is solved by transitivity). As mentioned earlier, learners specialized in field name analysis may often prove ineffective when processing, for example, abbreviated names or names with a very broad meaning. Making the most of the context, it is hence possible to tag field names with additional information to make them more significant. From this viewpoint, a simple documentation text file may turn out to be a valuable source of information to analyze if it contains a complete data description table (with two columns: one for data names, the other for their description).

Furthermore, the analysis of formal annotations in models may also contribute to refine mapping solutions. Among others, OCL is a formal language (Botting, 2004) which allows the expression of constraints in object-oriented models. In fact, OCL is a standard of the OMG[1] and can be used together with the UML to constrain models. Let's consider the UML class diagram shown on Figure 1.

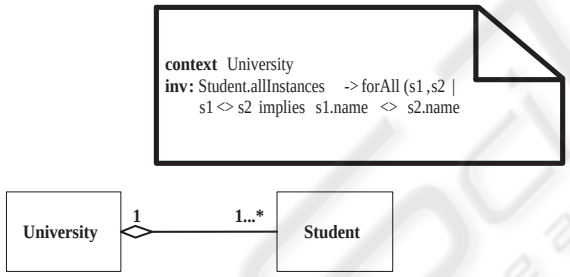

Figure 1: OCL invariant

Using OCL invariant constraints, one can easily specify that there cannot be two students with the same name within the same university. This OCL invariant (specified in the attached note) must be verified by all data sets constructed on this model. One must therefore ensure that no computed mapping produces target data violating the invariant. The analysis of such invariants contributes de facto to the accuracy of computed mappings.

To summarize, context analysis (c.f. Figure 2) is a process applied to a data source and its context and producing:

- an enriched data source, which can be used by learners as a training set or as a candidate data set for mapping.

[1]Object Management Group

Figure 2: Context analysis

- a set of integrity constraints, that computed mappings must ensure.

Data sources enhancement consists, among others, in replacing ambiguous or vaguely defined concept names with more significant ones. It may also enrich the source with new names to define composite concepts.

4 ARCHITECTURE OF THE CONTEXT ANALYZER

The informational context of a data source may include a wide range of document types. As mentioned, they can differ in their style, which can either be descriptive or operational. They can also differ in the formality degree of their contents. Some may have a formal content (e.g. formal models such as transition systems, programs, etc.), some may rely on a semi-formal notation(e.g. decision tables, UML diagrams, etc.), while others are written without any formality consideration (e.g. textual documents, drawings, etc.). Of course, informal documents are the most demanding regarding analysis. For efficient results, it is important that the analysis addresses each type of contents individually, taking into account its specificities.

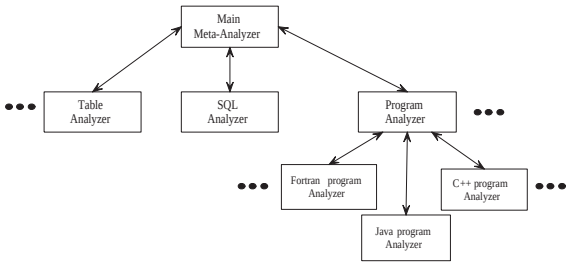

Figure 3: Architecture of a context analyzer

Hence, each type of contents is treated by a analyzer trained for this specific kind of information. Following a multi-strategy approach, we advocate an architecture (c.f. Figure 3) composed of many specialized analyzers coordinated by a main meta-analyzer.

The role of the meta-analyzer consists in running through the context documents, identifying the various contents types and determining which specialized analyzer is the most appropriate to take care of it. Results returned by specialized analyzers are then filtered (redundancy elimination) and combined by the meta-analyzer. The context analyzer has a hierarchical architecture of arbitrary depth. Indeed, a child analyzer can itself supervise, as a meta-analyzer, a set of more specialized analyzers. As illustrated on Figure 3, a program analyzer can itself coordinate the processing of many specialized analyzers, each one being an expert in the parsing of a specific programming language.

Finally, it is possible for analyzers to collaborate with each other without being related in the coordination hierarchy. For example, a program analyzer could resort to a SQL analyzer for the analysis of a request inserted in the program it is parsing.

5 CONCLUSIONS

Exploitation of contextual resources is a promising approach to solve the problem of semantic alignment of data sources (Tierney and Jackson, 2004). Documents composing the context of a data source offer valuable information which can help resolving major problems such as the identification of complex mappings (e.g. mapping a combinination of source concepts onto a single target concept).

As an extension to machine learning architectures using a multi-strategy approach for semanctic mapping (Doan et al., 2003; Berlin and Motro, 2002; Doan et al., 2002; Kurgan et al., 2002), we propose a context analyzer based on multiple specialized analyzers which are themselves coordinated by a meta-analyzer. This architecture has both the advantage of being adaptative and extensible. The hierarchical organization of analyzers allows a more efficient repartition of tasks between units. It also facilitates analyzing the context according to different abstraction levels. Hence, the context analyzer can enhance data sources with a range of details going from the most general (with low time cost) to the most specific (with higher time cost).

Moreover, this way of exploiting the context can prove to pay off for the effort put into data documentation. It also reduces the importance of user intervention in the mapping, which can be tedious, costly and error prone.

Although only two enrichment types have been brought up in this paper (i.e. expliciting abbreviated field names and identifying composite concepts), many other types of data source enrichment can be achieved from context analysis. Among others, our project aims at identifying and experimenting various enrichment strategies (based on context information) which could help improve the quality of data source mappings.

REFERENCES

Berlin, J. and Motro, A. (2002). Database schema matching using machine learning with feature selection. In *Proceedings of the 14th International Conference on Advanced Information Systems Engineering*, pages 452–466.

Botting, R. J. (2004). The object constraint language. Web site. http://www.csci.csusb.edu/dick/samples/ocl.html.

Doan, A., Domingos, P., and Halevy, A. (2003). Learning to match the schemas of databases- a multistrategy approach. *Journal of Machine Learning*, 50(3):279–301.

Doan, A., Madhavan, J., Domingos, P., and Halevy, A. (2002). Learning to map between ontologies on the semantic web. In *Proceedings of the 11th international conference on World Wide Web*, pages 662–673.

Kohavi, R. (1996). Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In Press, A., editor, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 202–207, Portland, OR.

Kurgan, L., Swiercz, W., and Cios, K. J. (2002). Semantic mapping of xml tags using inductive machine learning. In *Proceedings of the 2002 International Conference on Machine Learning and Applications (ICMLA'02)*, pages 99–109, Las Vegas, NV.

Pedro Domingos, M. P. (1997). On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29:103–130.

Tierney, B. and Jackson, M. (2004). Contextual semantic integration for ontologies. In *Doctoral Consortium of the 21rst Annual British National Conference on Databases*.