Automatic annotation of local data sources for data integration systems *

Sonia Bergamaschi, Laura Po, Antonio Sala, and Serena Sorrentino[1]

Dipartimento di Ingegneria dell'Informazione
Universitá di Modena e Reggio Emilia
bergamaschi.sonia@unimore.it, po.laura@unimore.it,
sala.antonio@unimore.it, serena.sorrentino@dbgroup.unimo.it

Abstract. In this article we present CWSD (Combined Word Sense Disambiguation) a method and a software tool for enabling automatic annotation of local structured and semi-structured data sources, with lexical information, in a data integration system. CWSD is based on the exploitation of WordNet Domains, structural knowledge and on the extension of the lexical annotation module of the MOMIS data integration system. The distinguishing feature of the algorithm is its low dependence of a human intervention. Our approach is a valid method to satisfy two important tasks: (1) the source annotation process, i.e. the operation of associating an element of a lexical reference database (WordNet) to all source elements, (2) the discover of mappings among concepts of distributed data sources/ontologies.

1 Introduction

The growth of information available on the Internet has required the development of new methods and tools to automatically manage information available on Web site or Web-based applications. The aim of the Semantic Web is to build a web of data by providing a common framework that enables data sharing and reuse across application, enterprise, and community boundaries. The Semantic Web relies on the use of shared schemas and ontologies, which should provide a well-defined basis of shared meanings for data integration and reuse. On the other end, the database community relies on the use of shared database schemas to be integrated in a global view.

However, we observe that several methods and tools developed to address the two problems rely, in different ways, on the use of lexical information. The reason is simple: beyond the syntactic and semantic heterogeneity of schemas and ontologies, it is a fact that their elements and properties are named using natural language expressions, and that this is done precisely because they bring in useful (but often implicit) information on the intended meaning and use of

* This work was partially supported by MUR FIRB Network Peer for Business project (http://www.dbgroup.unimo.it/nep4b) and by the IST FP6 STREP project 2006 STASIS (http://www.dbgroup.unimo.it/stasis).

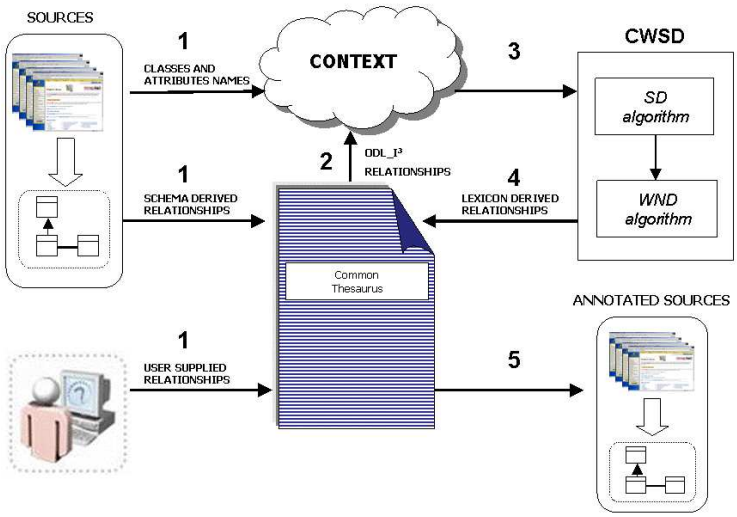

Fig. 1. Automatic annotation of local data sources with CWSD

the schema/ontology under construction. Therefore, it should not come as a surprise that a large number of tools for ontology learning and schema/ontology matching include some lexical resource (mainly WordNet) as a component, and use it in some intermediate step to annotate schema elements and ontology classes/properties with lexical annotation seems to be a critical task to develop smart methods for ontology learning and matching.

Combination methods are an effective way of improving the Word Sense Disambiguation process performance. The idea of combining the results of different methods of word sense disambiguation is not new and was used in almost any approach to word sense disambiguation in literature [1, 2].

WordNet Domains has been proven a useful resource for WSD. In fact, it has been used in different WSD composed algorithm as presented in [3] and in [4].

In [5], we have development a software tool for enabling an incremental process of automatic annotation of local schemas. MELIS exploits knowledge provided by the initial annotation. Differently, CWSD method does not need initial annotations to disambiguate the source terms.

In this context, we developed CWSD (Combined Word Sense Disambiguation), a method and a tool for the automatic annotation of structured and semi-structured data sources. Instead of being targeted to textual data sources like most of the traditional WSD (Word Sense Disambiguation) algorithms, CWSD exploit the structure of data sources together with the lexical knowledge associated with schema elements (terms in the following).

We integrated CWSD in the I^3 framework designed for the integration of data sources, MOMIS (Mediator EnvirOment for Multiple Information Sources) [6, 7], to overcome the heavy user involvement in manual lexical annotation of

data source terms. CWSD combines a structural disambiguation algorithm, that starts the disambiguation process by using the structural relationships extracted from the data source schemata, with a WordNet Domains based disambiguation algorithm, which refines terms disambiguation by using lexical knowledge.

CWSD tries to couple WSD approaches with the results obtained exploiting structural knowledge by the database and the semantic web communities [8, 9].

The outline of the paper is the following: section 2 describes the CWSD tool and its components. In section 3 we evaluate its performance. Finally we sketch out some conclusions and future works.

2 The Combined Word Sense Disambiguation method

CWSD is composed of two algorithms: SD (Structural Disambiguation) and WND (WordNet Domains Disambiguation).

SD tries to disambiguate source terms by using semantic relationships inferred from the structure of data sources and WND tries to disambiguate the terms using domains information supplied by WordNet Domains.

In order to disambiguate the sense of an ambiguous word, any WSD algorithm receives as input (and works in) a *context*. According to [10], many algorithms in literature represent the context as a "bag-of-word", a set of words that must be disambiguated, and sometime they insert in the context the information of the word positions in the text. Others approaches [11], consider a "window-of-context" around every target word, and submit all the words in this window as input to the disambiguation algorithm.

In CWSD the context is composed by: a set of terms (classes and attributes names) to be disambiguated, and a set of structural relationships among these terms included in a Common Thesaurus (CT) (as shown in figure 1). The CT is a set of ODL_{I^3} relationships describing inter- and intra-schema knowledge among a set of data source schemas: SYN (Synonym-of), defined between two terms that are equivalent/ synonymous; BT (Broader Term), defined between two terms where the first generalized the second (the opposite of BT is NT, Narrower Term); RT (Related Term) defined between two terms that are related in an aggregation hierarchy.

The default context for a data integration system is given by the data sources to be integrated and the local data sources ODL_{I^3} relationships.

2.1 The Structural Disambiguation algorithm

The SD algorithm exploits the structural ODL_{I^3} relationships of a data source to infer new CT relationships on the basis of a lexical database. As described in [6] the following ODL_{I^3} relationships are automatically extracted:

- For an ISA relationship between two classes (like `T1` ISA `T2`) we extract a BT relationship: `T2` BT `T1` (`T1` NT `T2`)

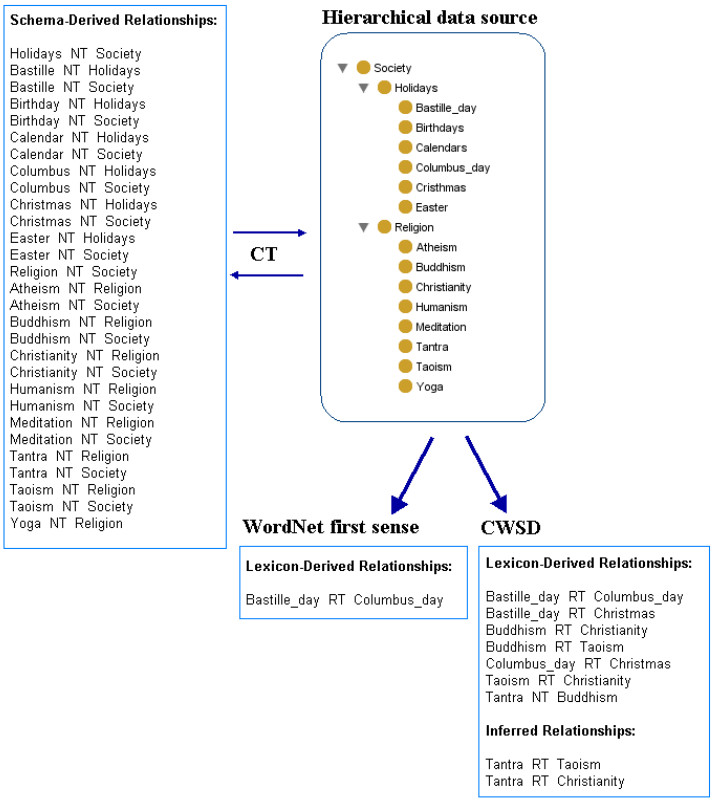

Fig. 2. Enrichment of the CT with relationships extracted by CWSD applied to a hierarchical data source

– For a foreign key (FK) between two relations:
`T1(A1,A2...AN) T2(B1,B2...BM) FK: B1 REFERENCES T1(A1)`
we infer `A1` SYN `B1`
and if B1 is a key on table T2: `T1` BT `T2` (`T2` NT `T1`)
else: `T1` RT `T2`

The extracted relationships are stored in the CT and are used in the disambiguation process according to a lexical database (in our approach we use WordNet [12]).

SD tries to find a corresponding lexical relationship when a relationship holds among two terms. In practice, if we have a direct/chain of relationship between two terms, we try to find the semantically related meanings and annotate the terms with these meanings. A chain of relationship is obtained navigating through the lexical database relationships.

Figure 2 shows an example of the application of the SD algorithm on a hierarchical data source, i.e. a portion of the first three level of the "society" subtree

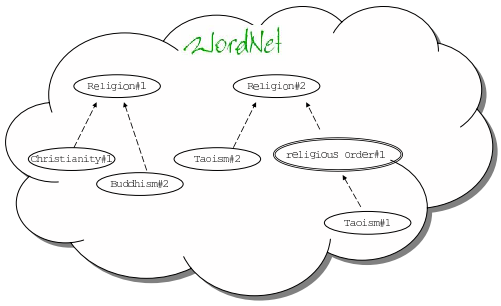

Fig. 3. Hyponym relationships in WN extracted by SD applied to a hierarchical data source

in the Google directory. First of all, all the ISA relationships in the schemata are extracted from the source and inserted in the CT as NT relationships, then, SD finds the corresponding hyponym relationships in WordNet. The annotations generated using SD enrich the CT of new ODL_{I^3} relationships (all the lexicon-derived relationships shown in figure). Using ODB-Tool (a component of the MOMIS system) the CT inferred new relationships.

Figure 3 shows some hyponym relationships found in WordNet, and the correspondent chosen synsets. In particular, for the terms "religion" and "Taoism", SD chooses two correct synsets, because two different hyponym relationships exist between the terms.

2.2 The WordNet Domains algorithm

WordNet Domains [13] [14] can be considered an extended version of WordNet, (or a lexical resource) in which synsets have been annotated with one or more domain labels. The information brought by domains is complementary with the one already present in WordNet. Besides, domains may group senses of the same word, into a thematic cluster, which has the important side effect of reducing the level of ambiguity of polysemic words.

The WND algorithm takes inspiration from the domain-based one proposed in [15]. First, we examine all the possible synsets connected to a term and extract the domains associated to these synsets, with this information we calculate a list of the *prevalent domains* in the chosen context. Then, we compare these list of domains with the ones associated to each term. For a term we choose as the correct synsets all the synsets associated to the prevalent domains.

In WordNet Domains there is a particular domain called "factotum" which is the domain associated to synsets that do not belong to a specific domain and, as described in [14], in most cases the more frequent domain in a context. Unlike [15] we choose to use the "factotum" domain only when no domain in *prevalent domains* is related to the meanings of a term. WND results depends on the context and on the *configuration* chosen. The configuration is the maximum number

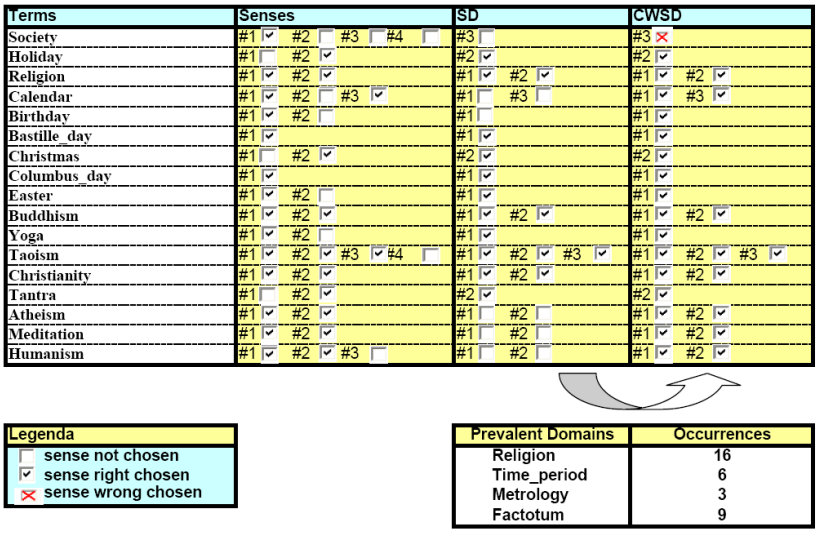

Fig. 4. Evaluation of the CWSD algorithm on a hierarchical data source

of domains we select for the disambiguation. The choice of the configuration and of the context is delegated to the user.

In Figure 4 we show the final result of the application of CWSD to the hierarchical data source. In particular, we compare the result obtained with CWSD with the result obtained using only the SD algorithm. If we disambiguate by using only SD, we obtain the correct senses for only some terms. With the CWSD algorithm we improved the results in two directions:(1) the disambiguation of the terms is more accurate; polysemy leads to have more than one synset associated to a terms, thanks to CWSD we can assign to these terms more than one sense; (2) morever, CWSD enriches the CT of new relationships: this is particularly important for the integration task (like showed in Figure 2). The unique term annotated in a wrong way is "Society", this is because it is associated, by the WND algorithm, to the "factotum" domain, but the correct sense is associated to the "anthropology" domain that is not present in the prevalent domains.[1]

3 Evaluation: experimental result

We experimented CWSD over a real data sources. In particular, we selected the first three levels of a subtree of the Yahoo and Google directories ("society and culture" and "society", respectively), which amounts to 327 categories for Yahoo and 408 for Google.

[1] A detailed description of the CWSD algorithm is available at http://www.dbgroup.unimo.it/momis/CWSD

WSD approach	Recall	Precision
SD	8.00%	97.00%
WND	66.62%	69.97%
CWSD	74.18%	74.18%
MELIS	53.03%	58.85%

Table 1. Comparing the different WSD approaches on the Google and Yahoo directories

In table 1 we compare the disambiguation of the subtree of the Google and Yahoo directories obtained with different algorithm: only SD, only WND, CWSD and MELIS.

The MELIS algorithm is incremental, so the evaluation is done after a number of runs until a fixed point has been reached. We compared CWSD results with the ones in MELIS that start with no annotations at all.

The annotation results have been evaluated in terms of recall (the number of correct annotations made by the algorithm divided by the total number of annotations, i.e. one for each category, as defined in a golden standard) and precision (the number of correct annotations retrieved divided by the total number of annotations retrieved). In the table, the recall and precision values are obtained by considering an element as properly annotated if the annotation given by the user is included in the set of annotations calculated by the WSD approach evaluated.

The application of SD over the web directories exploits the 792 ISA relationships and allows to obtain 60 annotations of which 58 are correct annotations, so we deduce an high precision but a very low recall. For our experience this is caused by the scarcity on relationships in WordNet.

The results remark that a combined algorithm outperforms the single algorithm of which it is composed[2]. Moreover the results gained by CWSD improve the ones obtained by MELIS.

4 Conclusion and future work

In this paper we presented a combined algorithm for the automatic annotation of structured and semi-structured data sources. CWSD exploits structural knowledge of a set of data sources together with the lexical information supplied by WordNet & WordNet Domains lexical database, to automatically annotate sources schemata.

We automatically extracted schema-derived relationships from the sources using the ODB-Tool component of the MOMIS system and inserted then in a Common Thesaurus. In the first step, the SD algorithm infers lexical meanings

[2] In this evaluation we do not discuss about the configuration chosen, because in general this is delegated to the user; however the showed results have been obtained on a limited context that considers together the terms of the classes that are correlated with an ISA relationship and the number of chosen domains is the best.

for terms from the structural ODL_{I^3} relationships stored in the Common Thesaurus. In the second step, the WND algorithm refines terms disambiguation using domain information supplied by WordNet Domains. The experimental results show how CWSD permit to obtain good results, moreover, structural knowledge of data sources is shown to significantly improve the disambiguation results obtained by applying only the WND algorithm.

Future work will be devoted to investigate the role of the context choice in our algorithm and to determine a criteria to choose the best number of domains during the configuration of the WordNet Domains disambiguation algorithm.

References

1. Rigau, G., Atserias, J., Agirre, E.: Combining unsupervised lexical knowledge methods for word sense disambiguation. CoRR **cmp-lg/9704007** (1997)
2. Mihalcea, R., Moldovan, D.I.: An iterative approach to word sense disambiguation. In Etheredge, J.N., Manaris, B.Z., eds.: FLAIRS Conference, AAAI Press (2000) 219–223
3. Gliozzo, A.M., Giuliano, C., Strapparava, C.: Domain kernels for word sense disambiguation. In: ACL, The Association for Computer Linguistics (2005)
4. Novischi, A.: Combining methods for word sense disambiguation of wordnet glosses. In Barr, V., Markov, Z., eds.: FLAIRS Conference, AAAI Press (2004)
5. Bergamaschi, S., Bouquet, P., Giacomuzzi, D., Guerra, F., Po, L., Vincini, M.: Melis: an incremental method for the lexical annotation of domain ontologies. IJSWIS **3(3)** (2007) 57–80
6. Bergamaschi, S., Castano, S., Beneventano, D., Vincini, M.: Semantic integration of heterogeneous information sources. Journal of Data and Knowledge Engineering **36** (2001) 215–249
7. Bergamaschi, S., Castano, S., Vincini, M.: Semantic integration of semistructured and structured data sources. SIGMOD Record **28(1)** (1999) 54–59
8. Rahm, E., Bernstein, P.A.: A survey of approaches to automatic schema matching. VLDB J. **10** (2001) 334–350
9. Noy, N.F.: Semantic integration: A survey of ontology-based approaches. SIGMOD Record **33** (2004) 65–70
10. Pahikkala, T., Pyysalo, S., Ginter, F., Boberg, J., Järvinen, J., Salakoski, T.: Kernels incorporating word positional information in natural language disambiguation tasks. In Russell, I., Markov, Z., eds.: FLAIRS Conference, AAAI Press (2005) 442–448
11. Banerjee, S., Pedersen, T.: An adapted lesk algorithm for word sense disambiguation using wordnet. In Gelbukh, A.F., ed.: CICLing. Volume 2276 of Lecture Notes in Computer Science., Springer (2002) 136–145
12. Fellbaum, C., Miller, G., eds.: WordNet: An electronic lexical database. The MIT Press (1998)
13. Gliozzo, A.M., Strapparava, C., Dagan, I.: Unsupervised and supervised exploitation of semantic domains in lexical disambiguation. Computer Speech & Language **18** (2004) 275–299
14. Magnini, B., Strapparava, C., Pezzulo, G., Gliozzo, A.: The role of domain information in word sense disambiguation. In: Natural Language Engineering, special issue on Word Sense Disambiguation. (2002) 359–373
15. Magnini, B.: Experiments in word domain disambiguation for parallel texts (2000)