Uncertainty in data integration systems: automatic generation of probabilistic relationships

Sonia Bergamaschi, Laura Po, Serena Sorrentino and Alberto Corni [1]

Abstract. We propose a method for the automatic discovery of probabilistic relationships in the environment of data integration systems. Dynamic data integration systems extend the architecture of current data integration systems by modeling uncertainty at their core. Our method is a probabilistic word sense disambiguation (PWSD), which allows to automatically lexically annotate (i.e. annotation w.r.t. a thesaurus/lexical resource) the schemata of a given set of data sources to be integrated. From the annotated schemata we derived the probabilistic lexical relationships that are inserted in the Probabilistic Common Thesaurus (PCT) and are added together to the structural relationships.

1 Introduction

Traditional data integration systems are systems interconnecting a limited number of resources, which are relatively stable in time and which have been typically built with sophisticated designs that have taken several time. On the other hand, data applications broaden more and more and ask for flexibility and handling of uncertainty. Applications like Google Base or involving a large number of sources as in the deep web or tool dealing with biological data [9], require that the semantic mappings between the mediated schema and the data sources, may be approximate as they need to be automatically extracted.

Using a probabilistic view, our approach allows to insert potential matches and to assign a probability value to them. This significantly reduces the cost of schema integration by allowing it to be fully automated and thus scalable to a large number of data sources [8].

Starting from our previous works on automatic discovery of semantic mappings in the environment of the MOMIS data integration system [2] developed by our

[1] University of Modena and Reggio Emilia, Information Engineering Department, Modena, Italy, name.surname@unimore.it

research group[2], we propose a method for the automatic discovery of probabilistic relationships in the context of new dynamic data integration system, i.e. systems where semantic mappings among schemata of different sources have to be discovered on the fly without or with a minimal human intervention.

The PWSD method, introduced in this paper, will automatically annotate the labels of sources schemata and associate to each annotation a probability value. The probabilistic annotations generated by PWSD are used to derive probabilistic lexical relationships between local sources. PWSD has been implemented in the ALA (Automatic Lexical Annotator) tool [7] that is integrated in the MOMIS system. However, PWSD but can be easily generalized to other data integration systems. Moreover, our method can be used by ontology merging and data integration system, adopting OWL as conceptual language[3].

The paper is organized as follows: in Section 2, we describe the process of automatic annotation within the MOMIS system; in Section 3, we present our PWSD method and describe the application of the Dempster-Shafer theory for the management of uncertainty in disambiguation. In Section 4 we describe the generation of probabilistic relationships. In Section 5 we sketch out the evaluation of PWSD on a real scenario, comparing the results with other WSD approaches, and finally, Section 6 gives our concluding remarks.

2 Probabilistic Automatic annotation in a data integration system

The data integration methodology proposed by MOMIS in previous articles [2] has been modified to cope with the treatment of uncertainty. Instead of building a global schema, we focus on the automatic generation of probabilistic relationships. The process is organized in three steps.

(1) Source schema extraction
Specialized software (wrappers) logically convert the format of the source schemata into the internal object language ODL_I^3.

(2) Lexical knowledge extraction
The extraction of lexical knowledge from data source is performed by ALA. ALA allows the user to choose a set of WSD algorithms and a way to combine their outputs. ALA supports a sequential (or pipe) combination and a parallel combination of the outputs. The parallel combination is based on PWSD.

During the annotation process, ALA interacts with the lexical resource WordNet extended with WND (WordNet Domains[4]) and the WNEditor[1]. ALA sup-

[2] http://www.dbgroup.unimo.it/

[3] The MOMIS system uses ODL_I^3 as an internal language, but supports the translation of OWL/ ODL_I^3 and ODL_I^3 /OWL schemata

[4] http://wndomains.fbk.eu/

plies a set of probabilistic annotations of the source terms. From these annotations ALA calculates the probabilistic lexical relationships among source schema elements.

Definition 2.1-(Probabilistic Annotation). Let T be a schema and t be a label of an element e ∈ T. We define St = {t#1; ::; t#n} as the set of all meanings for t w.r.t. a lexical resource. The probabilistic annotation of the term t is the triple <T; t; At>, where At = {a1; :::; ak} is the set of annotations associated to t. In particular, ai is defined as the couple (t#i; P(t#i)), where t#i ∈ St is a meaning for the term t, and P(t#i) is the probability value assigned.

Definition 2.2-(Ordinary Annotation). An ordinary annotation for t is a probabilistic annotation where there is only an annotation associated to t (||At|| =1) and the probability value assigned is equal to "1".

(3) Probabilistic Common Thesaurus generation

The PCT is a set of ODL_I^3 relationships describing inter- and intra-schema knowledge among the source schemata. ODL_I^3 relationships can be structural or lexicon derived, and ordinary or probabilistic.

Definition 2.3-(Structural ODL_I^3 relationship).The structural ODL_I^3 relationships are:

- *SYN_{EXT} (t1 is equivalent to t2 iff extension(t1) = extension(t2));*
- *BT_{EXT} (t1 subsumes t2 iff extension(t2) ⊆ extension(t1));*

Definition 2.4-(Lexical ODL_I^3 relationship).The lexical ODL_I^3 relationships are defined on the basis of thesaurus relationships:

- *SYN: (Synonym-of),defined between two terms that are synonymous;*
- *BT: (Broader Term),defined between two terms where an hypernym relationship holds between their meanings (the opposite of BT is NT, Narrower Term);*
- *RT: (Related Term) defined between two terms when a holonym or meronym relationships holds between their meanings.*

Structural relationships are automatically extracted by the MOMIS wrapper and ODB-Tools [3]. Lexical relationships are automatically extracted on the basis of the probabilistic annotations obtained (see section 4).

Definition 2.5-(Probabilistic ODL_I^3 relationship). A probabilistic ODL_I^3 relationship is a pair (Rel ODL_I^3 ; P(Rel ODL_I^3)), where Rel ODL_I^3 is a ODL_I^3 relationship and P(Rel ODL_I^3) is a probability value, in the interval [0, 1].

Definition 2.6-(Ordinary ODL_I^3 relationship). An ordinary ODL_I^3 relationship is a probabilistic ODL_I^3 relationship with probability value equal to "1".

Lexical ODL_I^3 relationships can be both probabilistic and ordinary; structural ODL_I^3 relationships are only ordinary. In addition to these relationships, other ordinary ODL_I^3 relationships can be supplied directly by the designer, interacting with the MOMIS Ontology Builder. MOMIS exploits description logic tech-

niques[3] to infer new relationships by applying subsumption computation to "virtual schemata" obtained by interpreting BT and NT as subclass relationships and RT as domain attributes.

3 PWSD

PWSD is based on a probabilistic combination of different WSD algorithms. In our previous works [4, 6], we have developed and tested on a real data scenario different types of WSD algorithms. These algorithms constitute an evolution of the ones proposed in the area of Natural Language Processing to disambiguate text, because they have been adapted to the case of structured and semi-structured data sources. At present, we have developed five algorithms[5]: Structural Disambiguation, WordNet Domain Disambiguation, WordNet first sense, Gloss similarity, Iterative gloss similarity. All this algorithms need to be configured about their reliability, although each algorithm has a default reliability based on its precision evaluated on a benchmark.

Example 1

As a case in point, let us consider the term "name". In WordNet we found six different meanings for "name" (*name#1*, *name#2*, .., *name#6*). Suppose we have to combine three algorithms that give different outputs: WSD1 that chooses a set of meanings formed by *name#1*, *name#2*, WSD2 that provides *name#1* as the correct meaning and WSD3 that does not give any result. What we want to obtain is a rate of confidence to be assigned to each possible meaning of the term "name".

3.1 Uncertainty in disambiguation - The use of the Dempster-Shafer theory

The set of WSD algorithms defines a type of evidence that can be consistent or arbitrary. These types of evidence cannot be handled by the traditional probability theory without resorting to further assumptions. That is why we decided to support the use of the Dempster-Shafer theory [11,12]. This theory allows us to model ignorance through lack of knowledge.

The theory deals with the so-called *frame of discernment*, the set of base elements θ in which we are interested (in our case, θ is the set of all possible meanings for the term under consideration), and its power set 2^{θ} , which is the set of all subsets of the interesting elements (in our case, all the possible subsets of the possible meanings). The basic of the measure of uncertainty is a *probability mass function m(·).* The mass function is defined for every element *A* of 2^{θ}, it assigns

[5] More details available http://www.dbgroup.unimo.it/publication/d2_1.pdf

zero mass to empty set and a value in the range [0,1] to each A of 2^{θ}. The total mass distributed being 1 so that:

$$\sum_{A \subseteq 2^{\theta}} m(A) = 1 \tag{1}$$

We can apportion the probability mass exactly as we wish, ignoring assignment to those levels of detail that we know nothing about. In our case, we derive the mass functions from the output and the precision of each WSD algorithm. To combine several algorithms we use the *Dempster's rule of combination* [11,12]:

$$m(a) = K \sum_{\bigcap A_i = a} \prod_{i=1}^{n} m_i(A_i) \tag{2}$$

$$K = \sum_{\bigcap A_i = \varnothing} \prod_{i=1}^{n} m_i(A_i) \tag{3}$$

where n is the number of algorithms that supplied a disambiguation output for the term under analysis.

In the end, to obtain the probability assigned to each meaning we split the belief mass function concerning a set of meanings.

$$P(a_i) = \sum_{a_i \in A} \frac{m(A)}{\|A\|} \tag{4}$$

where a_i is a meaning and A are all the sets of meanings that contain a_i .

Let us see the application of PWSD to the element “name” in Example 1. In order to combine different outputs, PWSD does not consider the algorithms that do not supply any annotations for the term. In the example, PWSD will be executed only on the outputs of WSD1 and WSD2. Each algorithm has a reliability value and an ignorance value, (the complementary value of the reliability), i.e. the mass function assigned to the entire set of possible meanings. Let us suppose that WSD1 has a reliability of 70% and WSD2 a reliability of 50%.

The application of the Dempster's rule of combination is shown in Figure 1. As WSD1 supplies a set composed of two meanings, the probability will be assigned to this set.

mass function	WSD 1	WSD 2	Dempster combination
m{name#1}		0.5	0.5
m{name#1,name#2}	0.7		0.35
m{name#1,name#2, ...name#6} = m{ignorance}	0.3	0.5	0.15

probability function	PWSD
P{name#1}	0.67
P{name#2}	0.17
P{ignorance}	0.15

Fig. 1. Application of the Dempster-Shafer theory on the WSD algorithms output and generation of the probabilistic annotations

The results obtained after the application of the Dempster's rule of combination show the probability assigned to different sets of meanings. In order to compute lexical relationships, we have to bring back to the case of probabilities assigned to individual meanings. As shown in Figure 1 on the right, the probability assigned to the set of meanings *{name#1,name#2}* will be split in two probabilities assigned to *name#1* and *name#2*.

4 From probabilistic annotation to probabilistic relationship discovering

MOMIS derives lexical ODL_I^3 relationships between local sources terms from the semantic relationships defined in WordNet between meanings, by using the following WordNet constructors:

- *synonymy* (similar relation) corresponds to a SYN ODL_I^3 relationship;
- *hyponymy* (sub-name relation) corresponds to an NT ODL_I^3 relationship;
- *hypernymy* (super-name relation) corresponds to a BT ODL_I^3 relationship;
- *holonymy* (whole-name relation) corresponds to an RT ODL_I^3 relationship;
- *meronymy* (part-name relation) corresponds to an RT ODL_I^3 relationship.
- *correlation* (two terms that share the same hypernym) corresponds to a RT ODL_I^3 relationship.

The application of PWSD associates a set of probabilistic meanings to a term in a source. So, a term t is described by a meaning *t#i* with a certain probability. When we assign the meaning *t#i* to the term t, t will inherit the same lexical relationships that occur for the synset *t#i* within the WordNet relationships network.

We restrict to the sub-network of relationships that branch off from *t#i*, in the context of analysis of the sources to be integrated. From the sub-net of lexical relationships between meanings we derive lexical ODL_I^3 relationships among schemata terms. Thanks to the formula of the *join probability*, the probability value associated with an ODL_I^3 relationship holding among *t#i* and *s#j* is defined as:

$$P(\mathrm{REL}_{\mathrm{ODL}_1{}^3}(t_i, s_j)) = P(t_i) \times P(s_j) \quad (5)$$

5 Evaluation: experimental result

We experimented PWSD over real data sources, for sake of simplicity, we considered only three data sources, but the process is scalable and applicable to a large set of data sources. We used three ontologies from the benchmark 2008 of the OAEI project[6] to run the automatic annotation.

Table 5.2. PWSD comparison with other WSD method

	Accuracy	Error	Precision	Recall	Fmeasure
CWSD	0.78%	0.22%	0.66%	0.55%	0.60%
PWSD	0.75%	0.25%	0.56%	0.76%	0.65%
PWSD with Threshold=0.2	0.84%	0.16%	0.80%	0.70%	0.75%
WordNet First Sense heuristic	0.83%	0.17%	0.81%	0.53%	0.64%

The golden standard for the benchmark is the annotation selected by an expert. The expert may select more than one meaning for each term and the evaluation is done on each possible selected meaning. We calculate statistics of accuracy, error, precision, recall and F-measure. All these measures are express as percentage, with 100% being the best score, except of error measure where 0% is the best.

We compared the results of PWSD with our previous combined algorithm, CWSD [6], and with the WordNet first sense heuristic (this heuristic is often used as baseline for WSD systems and often outperforms many of these systems which take surrounding context into account [10]).

As table 5.2 shows, precision and recall of PWSD do not increase with respect to CWSD (a method that combines only 2 WSD algorithms), this is due to a high number of annotations performed by PWSD with a very low probability value. Filtering the PWSD annotation output refines the annotation results. The threshold chosen was quite low (the average probability value of PWSD was 0.34), this al-

[6] http://oaei.ontologymatching.org/2008/

lowed to filter out only the annotations that were not supported by a lot of WSD algorithms (the annotations that can introduce noise) without decreasing the recall.

6 Conclusions and Future Work

We presented a method for the automatic discovery of probabilistic relationships in the environment of data integration systems. We proposed PWSD, a probabilistic method to automatically annotate the terms of source schemata w.r.t a lexical resource. PWSD associates a probability value to each annotation that is determined combining the results of many WSD algorithms through the application of the Dempster-Shafer theory. The PWSD has been implemented in ALA tool and integrated in the MOMIS system; the annotations are used in MOMIS to derive probabilistic lexical relationships between among sources.

We noticed that, to improve the relationships discovery of this process, it is crucial that the probabilistic annotations are as much accurate and robust as possible. Future work will be devoted to improve the annotation process inserting techniques able to deal with acronyms and abbreviations [5].

References

1. Benassi R., Bergamaschi S., Fergnani A., Miselli D. (2004), Extending a lexicon ontology for intelligent information integration. ECAI, 278-282. IOS Press.
2. Beneventano D., Bergamaschi S., Guerra F., Vincini M. (2003), Synthesizing an integrated ontology. IEEE Internet Computing, 42-51.
3. Beneventano D., Bergamaschi S., Sartori C. (2003), Description logics for semantic query optimization in object-oriented database systems. ACM Trans. Database Syst., 28:1-50.
4. Bergamaschi S., Bouquet P., Giacomuzzi D., Guerra F., Po L., Vincini M. (2007), An incremental method for the lexical annotation of domain ontologies. Int. J. Semantic Web Inf. Syst., 3(3):57-80.
5. Sorrentino S., Bergamaschi S., Gawinecki M., Po L. (2009), Schema Normalization for Improving Schema Matching, A.H.F. Laender et al. (Eds.): ER 2009, LNCS 5829, 280–293.
6. Bergamaschi S., Po L., Sorrentino S. (2007), Automatic annotation in data integration systems. OTM Workshops (1), LNCS 4805, 27-28, Springer.
7. Bergamaschi S., Po L., Sorrentino S., Corni A. (2010) Dealing with Uncertainty in Lexical Annotation. ER 2009, to appear at Journal of Theoretical and Applied Informatics
8. Dalvi N. N., Suciu D. (2007), Management of probabilistic data: foundations and challenges, PODS, 1-12, ACM.
9. Louie B., Detwiler L., Dalvi N. N., Shaker R., Tarczy-Hornoch P., Suciu D. (2007), Incorporating uncertainty metrics into a general-purpose data integration system, SSDBM, 19, IEEE Computer Society
10. McCarthy D., Carroll J. (2003), Disambiguating nouns, verbs, and adjectives using automatically acquired selectional preferences. Computational Linguistics, 29(4):639-654.
11. Parsons S., Hunter A. (1998), A review of uncertainty handling formalisms, Applications of Uncertainty Formalisms, LNCS 1455: 8-37, Springer.
12. Shafer G. (1976), A Mathematical Theory of Evidence, *Princeton University Press*.