Literature-based alignment of ontologies

Patrick Lambrix and He Tan and Wei Xu

Department of Computer and Information Science
Linköpings universitet, Sweden

Abstract. In this paper we propose and evaluate new strategies for aligning ontologies based on text categorization of literature using support vector machines-based text classifiers, and compare them with existing literature-based strategies. We also compare and combine these strategies with linguistic strategies.

1 Introduction

In recent years many ontologies have been developed and many of these ontologies contain overlapping information. A number of ontology alignment systems that support the user to find inter-ontology relationships exist (see overviews in e.g., [2, 5] and http://www.ontologymatching.org/). Recently, there is a growing interest in instance-based methods for ontology alignment. In this paper we slightly generalize the method for instance-based ontology alignment using literature that was proposed in [7]. Further, we propose a new instantiation of the method based on text categorization using support vector machines (SVMs). We evaluate these algorithms in terms of the quality of the alignment results for the five test cases used in [7]. We compare two SVM-based algorithms with each other and with the Naive Bayes text classification approach of [7]. Finally, we compare the algorithms with a good text-based approach and discuss the advantages and disadvantages of combining the approaches. For related work, more results and more details we refer to the longer version of this paper that is available from the SAMBO website (http://www.ida.liu.se/∼iislab/projects/SAMBO/).

2 Background

Many ontology alignment systems are based on the computation of similarity values between terms in different ontologies and can be described as instantiations of the general framework defined in [5]. An alignment algorithm receives as input two source ontologies. The algorithm can include several matchers. These matchers calculate similarities between the terms from the ontologies. Alignment suggestions are then determined by combining and filtering the results generated by one or more matchers. The suggestions are then presented to the user who accepts or rejects them.

A method for creating a matcher that uses scientific literature was proposed in [7]. It builds on the intuition that a similarity measure between concepts can

be computed based on relationships between the documents in which they are used. It contains the following basic steps (slightly generalized). (1) **Generate corpora.** For each ontology that we want to align we generate a corpus of documents. (2) **Generating classifiers.** For each ontology one or more document classifiers are generated. The corpus of documents associated to an ontology is used for generating its related classifiers. (3) **Classification.** Documents of one ontology are classified by the document classifiers of the other ontology and vice versa. (4) **Calculate similarities.** A similarity measure between concepts in the different ontologies is computed based on the results of the classification.

In [7] an instantiation (NB) of this method was implemented and evaluated using test cases involving biomedical ontologies. For step 1 a corpus was generated by querying PubMed (October 23, 2005) with each concept and retrieving the 100 most recent abstracts (if there were so many) for each concept. In step 2 one Naive Bayes classifier per ontology was generated. The classifiers return for a given document d the concept C in the ontology for which the posterior probability $P(C|d)$ results in the highest value. In step 3 the Naive Bayes classifier for one ontology was applied to every abstract in the abstract corpus of the other ontology and vice versa. Finally, in step 4 a similarity value between two concepts was computed using the numbers of abstracts associated with one concept that are also related to the other concept as found by the classifiers.

In general, in step 2 a document may be assigned to several concepts and thus we may regard the classification of documents to concepts as several binary classification problems, one for each concept in an ontology. In the next section we propose an instantiation of the method that does exactly this and is based on SVMs. SVMs [8] is a machine learning method that constructs a separating hyperplane in a feature space between two data sets (positive and negative examples) which maximizes the margin between the two sets. The setting can also be generalized to learning from positive and unlabeled examples (e.g. [6]).

3 Alignment algorithms

The basic algorithm implements the steps as follows. (1) **Generate corpora.** We used the same corpora as in [7]. (2) **Generating the classifiers.** For each concept in each ontology an SVM text classifier was generated. We used the LPU [6] system. LPU generates text classifiers based on positive and unlabeled examples. The abstracts retrieved when querying for a concept were used as positive examples for that concept. Further, for a given concept we used one abstract of each other concept in the same ontology as unlabeled examples. The SVM text classifier for a concept returns for a given document whether the document is related to the concept. It returns a value that is positive if the document is classified to the concept and negative otherwise. (3) **Classification.** The SVM text classifier for each concept in one ontology is applied to every abstract in the abstract corpus of the other ontology and vice versa. The classification was done by using the text classifiers generated by LPU within the SVM^{light} system [4]. Observe that a document can be classified to zero, one or more than one concept

in an ontology. (4) **Calculate similarities.** We define the similarity between a concept C_1 from the first ontology and a concept C_2 from the second ontology as:

$$\frac{n_{SVMC-C_2}(C_1, C_2) + n_{SVMC-C_1}(C_2, C_1)}{n_D(C_1) + n_D(C_2)}$$

where $n_D(C)$ is the number of abstracts originally associated with C, and $n_{SVMC-C_q}(C_p, C_q)$ is the number of abstracts associated with C_p that are also related to C_q as found by classifier $SVMC - C_q$ related to concept C_q.

The pairs of concepts with a similarity measure greater or equal than a predefined threshold are then presented to the user as candidate alignments.

In NB a document was classified to exactly one concept. We wanted to evaluate whether this has a real influence in the similarity computation. Therefore, we also developed an alternative to the basic SVM algorithm where in step 3 a document can be classified to only one concept. We assign a document only to the concept for which its SVM classifier generated the highest positive value for that document. In the case more than one classifier produces the highest positive value, then one of the associated concepts is chosen.

4 Evaluation

We evaluate the proposed algorithms with respect to the quality of the suggestions they generate. We also compare them to NB as well as to the best text-based matcher (TermWN) implemented in SAMBO [5]. Further, we investigate the combination of the proposed algorithms and TermWN.

We used the following set-up. We use the same five **test cases** as in [7]. For the first two cases we use a part of a Gene Ontology (GO) ontology together with a part of Signal Ontology (SigO). The first case, *B* (behavior), contains 57 terms from GO and 10 terms from SigO. The second case, *ID* (immune defense), contains 73 terms from GO and 17 terms from SigO. The other cases are taken from the anatomy category of Medical Subject Headings (MeSH) and the Adult Mouse Anatomy (MA): *nose* (containing 15 terms from MeSH and 18 terms from MA), *ear* (containing 39 terms from MeSH and 77 terms from MA), and *eye* (containing 45 terms from MeSH and 112 terms from MA). Golden standards for these cases were developed by domain experts. Further, we use the same **corpus** as in [7]. We use SVM-based **matchers** based on sets of maximum 100 documents per concept. These matchers are denoted as SVM-P and SVM-S where P and S stand for Plural (a document can be classified to several concepts) and Single (a document can be classified to only one concept), respectively.

The results are given in table 1. The first column represents the cases and the number of expected alignments for each case based on the golden standards. The expected alignments are a minimal set of suggestions that matchers are expected to generate for a perfect recall. The second column represents threshold values. The cells in the other columns contain quadruplets a/b/c/d which represent the number of a) suggestions, b) correct suggestions, c) wrong suggestions and d) inferred suggestions, for a given case, matcher and threshold.

Comparison of single and plural assignment. The recall for the plural assignment is much higher than the recall for the single assignment. This comes, however, at a cost. The precision for the single assignment algorithm is much higher than for the plural assignment algorithm. We see a real trade-off here: find many expected alignments, but also get many wrong suggestions, or, find few expected alignments, but receive almost no wrong suggestions.

Comparison of NB and SVM-S. These two single assignment algorithms give relatively few suggestions but have high precision. However, NB gives always more suggestions than SVM for the same threshold. NB also always gives suggestions, except for case ID and threshold 0.8, while SVM-S often does not give suggestions. It is clear that SVM-S does not perform well with high thresholds. In general, NB has slightly better recall than SVM-S, while SVM-S has slightly higher precision than NB.

	Th	SVM-P	SVM-S	NB	TermWN	TermWN+ SVM-S	TermWN+ SVM-P
B 4	0.4	387/4/258/125	0/0/0/0	4/2/1/1	58/4/22/32	4/4/0/0	156/4/84/68
	0.5	306/4/203/99	0/0/0/0	2/2/0/0	35/4/13/18	4/4/0/0	52/4/19/29
	0.6	225/4/148/73	0/0/0/0	2/2/0/0	13/4/4/5	0/0/0/0	21/4/7/10
	0.7	130/3/79/48	0/0/0/0	2/2/0/0	6/4/0/2	0/0/0/0	7/4/1/2
	0.8	36/0/22/14	0/0/0/0	1/1/0/0	4/4/0/0	0/0/0/0	4/4/0/0
ID 8	0.4	672/8/592/72	2/2/0/0	9/6/3/0	96/7/66/23	8/6/2/0	302/8/262/32
	0.5	490/8/433/28	0/0/0/0	5/5/0/0	49/7/25/17	6/6/0/0	155/7/127/21
	0.6	336/8/300/28	0/0/0/0	2/2/0/0	16/5/5/6	2/2/0/0	71/7/48/16
	0.7	222/6/191/25	0/0/0/0	1/1/0/0	7/5/2/0	1/1/0/0	19/7/7/5
	0.8	108/5/93/10	0/0/0/0	0/0/0/0	6/4/0/2	0/0/0/0	7/5/2/0
nose 7	0.4	155/7/124/24	5/5/0/0	6/5/1/0	48/7/37/4	9/7/2/0	80/7/66/7
	0.5	120/7/91/22	4/4/0/0	6/5/1/0	28/7/18/3	7/7/0/0	58/7/47/4
	0.6	85/7/60/18	2/2/0/0	5/5/0/0	8/6/2/0	6/6/0/0	31/7/47/4
	0.7	58/6/45/7	0/0/0/0	5/5/0/0	6/6/0/0	4/4/0/0	11/7/4/0
	0.8	34/6/27/1	0/0/0/0	3/3/0/0	6/6/0/0	1/1/0/0	6/6/0/0
ear 27	0.4	1224/24/1056/144	14/12/2/0	18/16/2/0	155/26/110/19	34/25/8/1	585/27/481/77
	0.5	957/23/822/112	11/10/1/0	15/14/1/0	99/26/65/8	27/23/4/0	203/26/146/31
	0.6	696/22/590/84	1/1/0/0	12/11/1/0	47/26/19/2	17/17/0/0	96/24/64/8
	0.7	478/22/392/64	0/0/0/0	11/10/1/0	34/26/8/0	12/12/0/0	55/23/28/4
	0.8	278/21/223/34	0/0/0/0	3/3/0/0	28/25/3/0	1/1/0/0	29/21/6/2
eye 27	0.4	2055/25/1926/104	7/7/0/0	25/18/7/0	135/26/100/9	28/23/5/0	643/25/568/50
	0.5	1481/25/1366/90	4/4/0/0	18/17/1/0	74/23/44/7	21/20/1/0	272/25/221/26
	0.6	957/25/860/72	0/0/0/0	14/14/0/0	33/22/10/1	16/16/0/0	138/24/101/13
	0.7	612/24/539/49	0/0/0/0	10/10/0/0	24/21/3/0	7/7/0/0	54/21/27/6
	0.8	344/23/290/31	0/0/0/0	3/3/0/0	22/20/2/0	0/0/0/0	25/21/4/0

Table 1. Results.

Comparison with and combination with other matchers. The table also shows the quality of the suggestions of TermWN (from [5]), and the combinations

(sum, equal weight) of TermWN with SVM-P and SVM-S. TermWN has higher recall than SVM-S and NB. It also has better recall than SVM-P for the case ear, but for the other cases the recall is similar. TermWN has better precision than SVM-P, but worse than SVM-S and NB. Almost all expected alignments were found by at least one SVM or NB matcher and threshold at least 0.4. TermWN with threshold 0.4 missed 1 expected alignment for ID, 1 for ear and 1 for eye.

The combination of TermWN and SVM-S gave perfect results for B and thresholds 0.4 and 0.5. Otherwise, when it gave suggestions, the precision was high. For thresholds 0.4 and 0.5, SVM-S worked as a filter on TermWN by removing many wrong suggestions at the cost of no or few correct suggestions. For higher thresholds too many correct suggestions were removed. For most cases and thresholds the combination of TermWN and SVM-P gave better recall than TermWN and SVM-P. The precision of the combination was higher than the precision for SVM-P, but lower than the precision for TermWN. As shown in the longer version of the paper, the precision for the combination could become better than the precision for TermWN by using the double threshold filtering technique of [1] while keeping the recall at the same level for most cases.

5 Conclusion

We have proposed SVM-based algorithms for aligning ontologies using literature. We have shown that there is a trade-off between the single and plural assignment methods regarding precision and recall. Further, SVM-S and NB obtained similar results. The combinations of TermWN with SVM-S and with SVM-P lead to a large gain in precision compared to TermWN and SVM-P, with still a high recall.

References

1. Chen B, Tan H and Lambrix P. 2006. Structure-based filtering for ontology alignment. *Proceedings of the IEEE WETICE Workshop on Semantic Technologies in Collaborative Applications*, pp 364-369.
2. Euzenat J and Shvaiko P. 2007. *Ontology Matching.* Springer.
3. Joachims T. 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *Proceedings of the European Conference on Machine Learning*, LNCS 1398, 137-142.
4. Joachims T. 1999. Making Large-Scale SVM Learning Practical. *Advances in Kernel Methods - Support Vector Learning*, B Schölkopf and C Burges and A Smola (eds), MIT-Press. http://svmlight.joachims.org/
5. Lambrix P and Tan H. 2006. SAMBO - A System for Aligning and Merging Biomedical Ontologies. *Journal of Web Semantics*, 4(3):196-206.
6. Liu B, Dai Y, Li X, Lee WS and Yu Ph. 2003. Building Text Classifiers Using Positive and Unlabeled Examples. *Proceedings of the Third IEEE International Conference on Data Mining*, 179-188. http://www.cs.uic.edu/∼liub/LPU/LPU-download.html
7. Tan H, Jakoniene V, Lambrix P, Aberg J and Shahmehri N. 2006. Alignment of Biomedical Ontologies using Life Science Literature. *Proceedings of the International Workshop on Knowledge Discovery in Life Science Literature*, LNBI 3886, 1-17.
8. Vapnik V. 1995. *The Nature of Statistical Learning Theory.* Springer.