A Discriminative Approach to Ontology Mapping

Michael Wick, Khashayar
Rohanimanesh, Andrew McCallum
University of Massachusetts
140 Governor's Drive
Amherst, MA
mwick@cs.umass.edu

AnHai Doan
University of Wisconsin
210 W. Dayton St.
Madison, WI
anhai@cs.wisc.edu

ABSTRACT

Techniques for automatically performing ontology mapping are vital for many real-world applications. Unfortunately, the problem is difficult because many types of evidence must be integrated to make good alignment decisions, and these decisions are co-dependent. In this paper, we propose a conditional random field (CRF) for ontology mapping which combines probabilistic machine learning and dependencies among the prediction. We integrate multiple sources of evidence using clauses in first-order logic, and learn corresponding weights directly from labeled training data. Our experiments show examples of impressive gains when tested on a commonly used mapping corpus; our method achieves an average of 11% (absolute) improvement in F1 when compared to other systems. We also show that our CRF is capable of generalizing from one mapping domain to another—making our supervised approach applicable for domains that lack labeled training data.

Keywords

Data Integration, Ontology Mapping, Conditional Random Field, Weighted Logic

1. INTRODUCTION

Ontology mapping is the problem of finding equivalent concepts across several different ontological resources. This task is often viewed as an *n-ary* classification problem where each concept in the target ontology is classified as a concept in the source ontology. In practice, these classifiers are learned entirely in the source domain, treating each of its concepts as a class label and learning a corresponding distribution over words. Typically, several classifiers are learned in this manner, and combined using a *multi-strategy* approach [5, 14]. Although this technique does have some strengths, we also identify several weaknesses.

The first deficiency is that parameters are learned without a ground truth mapping between the ontologies. This is why weights in multi-strategy methods are often set by hand. Without automatic learning, restrictions are placed on the number of features that can cross ontology boundaries because manually setting a large number of weights places a significant burden on users.

A second is the limited expressivity of the classifiers. When classifying concepts in the target domain as ones in the source, we are implicitly modeling similarities between pairs of concepts (e.g., computing the probability of concept i given concept j requires only two concepts as arguments). This places a restriction on the types of features that can be implemented. For example, it may be useful to represent the fact that a node in a taxonomy tree is not likely to be mapped to both a child and its respective parent; however, this cannot be captured by a model that factorizes into pairs. In current practice, these types of constraints are usually enforced with heuristics or in the post-processing stage, rather than in a probabilistic fashion.

A third is that the *n-ary* classification paradigm is not well suited to complicated mappings, where (1) concepts in one domain lack an equivalent concept in the other and (2) concepts have more than one equivalent (i.e., an n:k mapping).

To address all three of these issues, we propose modeling ontology mapping with a conditional random field. Our model factorizes into sets of concepts and their pairs. Intuitively, a factor measures the affinity between sets of concepts; that is, how likely the concepts in those subsets map to each other. By factorizing the problem into sets we enable the model to reason with the full expressive power of first-order logic. For example, clauses in logic can tie information from multiple nodes in a tree: incorporating concept names, instance data, and structure (e.g., neighbors in a taxonomy tree), all within a single clause. Also, because this model is completely supervised, weights corresponding to each clause are learned discriminatively from data.

Once parameters are learned, the CRF is capable of producing the probability of a particular ontology mapping given the observed (and novel) input ontologies. The problem of MAP inference is to discover the mapping that maximizes this conditional probability. However, because there are an exponential number of subsets (and consequently, factors), exact inference is intractable. Therefore, we present a polynomial time approximation in Section 3.2.

A major contribution of our approach is that we learn a model that maximizes parameter likelihood under a true mapping, rather than learning a model that maximizes the generative probability of concepts in an ontology. The empirical results of our method are not directly comparable

VLDB '08, August 24-30, 2008, Auckland, New Zealand

to current systems (which are restricted to *1-1* mappings) because it is more expressive and the size of the prediction space explodes. However, we are able to achieve an average improvement of 11% absolute F1. We also show that our features are capable of generalizing across two domains, making our method applicable to domains with no labeled data.

2. RELATED WORK

Schema matching is a well studied task that has provided the foundation for research in ontology mapping [2, 11, 9, 4, 7]. This can be viewed as a special case of ontology mapping where database fields are considered the analog of concepts and schemas are treated as flat ontologies.

He and Chang [7] propose a generative statistical model for performing schema matching. In contrast, our model is discriminatively trained in a fully supervised setting. Furthermore, changes in the structure of our model are not needed to accommodate additional dependencies, making it easier for users who are not experts in statistics to adapt it to new domains.

GLUE [5] is a system that produces mappings across two taxonomy trees. The system contains features for modeling both concepts and structures, but combines them with post-processing rather than modeling the uncertainty jointly. Additionally, GLUE is a less supervised approach in the sense that only part of the model is learned from labeled data. The meta classifier is hand-tuned and relaxation labeling, the approach that combines additional structural constraints from neighbors, is an unsupervised method. In contrast our model is entirely supervised and is capable of learning all parameters from labeled mappings (including those tied to structural constraints).

RiMOM [14] is another system that discovers mappings across ontologies. Mapping is performed by finding a configuration that minimizes Bayesian risk. However, weights for various pieces of evidence are hand tuned in advance rather than learned from the data. An extension, iRiMOM [15] learns to select a strategy from multiple sub-strategies, but treats the possibly overlapping sub-strategies as independent of each-other. In contrast, our discriminatively trained method makes it easy to model dependencies between overlapping pieces of evidence without any independence assumptions.

APFEL [6] is a system that learns parameters through user interaction with the alignment process. Although supervised machine learning is explored, the problem of ontology mapping itself is not formalized as a statistical model.

Conditional random fields [8] have been successfully applied to many structured prediction problems in natural language processing [8], image analysis [10], and other areas. We use first-order features to tie parameters in our CRF combining logic and probability. Others have explored forms of weighted logic including research on Markov logic [12] and Bayesian logic (BLOG) [1]. Our method is in the same family as the former, since both are undirected, discriminatively trained models with weights on logical clauses.

3. SOLUTION OVERVIEW

3.1 Ontology Mapping with a CRF

In this section we briefly discuss how to model ontology mapping with a conditional random field. We begin by factorizing the mapping into sets of concepts where each concept in the set is predicted to be semantically equivalent (mapped). Let x^i be a set of edges each of which maps a concept in one ontology to a concept in another. Let the binary random variable y_i be true if and only if all edges in x^i correctly map the concepts they span. Let $\Lambda = \{\lambda_i\}$ be a set of real-valued parameters and $\mathcal{F} = \{f(x^k, y_k, y_{k'})_i\}$ be a set of real valued feature functions (or clauses in first order logic). Then we model the probability that mappings of concepts in x^i are correct as

$$P(y_i|x^i) \propto exp \sum_k \lambda_k f_k(x^k, y_k, y_{k'}) \tag{1}$$

where $y_{k'}$ are labels in the neighborhood of concepts in x^k. Notice how feature functions ($\mathcal{F}$) take the target and neighbor labels as well as the observed ontologies as arguments. This allows the model to probabilistically reason over nearly any function from a simple string match to a complicated sub-strategy in a multi-strategy approach.

Intuitively, we can think of the above model as an affinity metric describing how compatible a set of concepts are. Highly compatible concepts obtain higher scores than those that are not likely to be mapped. However, we would like our model to handle more than a single set of concepts, so we extend the model to represent an entire mapping as follows:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_x} \prod_{y_i \in \mathbf{y}} \psi(y_i, y_{i'}\mathbf{x}) \tag{2}$$

where Z_x is an input dependent normalizing constant, and ψ are factors that represents affinity of concepts in a set (given by Equation 1).

3.2 Learning and Inference

Unfortunately exact inference and learning is intractable in this model. Therefore, to learn the parameters Λ, we use a piecewise approximation [13]. This is done by sampling positive and negative examples of sets x^i and corresponding labels y_i. Weights are assigned by performing gradient descent on these data-label pairs with the usual gaussian prior. This can be viewed as a regularized maximum entropy (or logistic regression) model for binary classification.

Inference is currently a greedy agglomerative approximation to the *maximum a posteriori* (MAP) setting. The process initializes the configuration to an empty mapping and then iteratively adds the highest scoring edges until all scores are below a stopping threshold τ. This method does not require computing the intractable partition function $\frac{1}{Z_x}$, because it is constant given the observed input.

These training and testing procedures have been shown to work well in practice, especially when exact methods are not feasible [16, 3].

4. EXPERIMENTS

4.1 Dataset

We use the Illinois Semantic Integration Archive datasets to evaluate our methods. It consists of taxonomy trees from two domains: course catalog and company profiles.

The course catalog contains a hierarchical representation of classes from Cornell and University of Washington (see

Features for concept compatability	
Description	**real/bool**
TFIDF Cos distance between concepts	real
TFIDF Cosine distance between instances	real
Substring match	boolean

Features for structural dependencies	
Description	**real/bool**
Concepts are within n tree-levels	boolean
Parents are mapped	boolean
Number of children mapped	real
Number of siblings mapped	real

Table 1: pairwise feature extractors

Figure 1). There are a total of 104 concepts in this domain. Cornell contributes 54 concepts and 4360 instances while Washington contributes 50 concepts and 6957 instances.

The company profile domain contains a hierarchical representation of companies, industries, and sectors (see Figure 1 for an example). The company profile domain is larger than the catalog set having 219 concepts total. Yahoo contributes 102 concepts and and the Standard contributes 117.

4.2 Features

We use first-order logic clauses to express our features. These clauses allow us to aggregate pairwise comparisons into representations of entire sets (mappings). Most of our extractors take two concepts as arguments and produce a binary or real-valued result. Binary results are aggregated universally and existentially whereas real-valued features are aggregated with functions such as max, min, and average.

Our features can be organized into features over concepts, instances, and taxonomy tree structure. Features involving concepts are binary valued substring matches or real-valued TFIDF cosine distances. Similarly, cosine distances are used between instances. We also incorporate a variety of features that examine the labels (mappings) of parents, children, and siblings. See Table 1 for a complete list.

4.3 Implementation Details

In the following experiments we employ the learning and inference methods described in Section 3.2. For learning we sample $n * 30$ training examples (where n is the number of instances in the training set) in such that 25% are positive examples of matchings and the remaining 75% are negative. We justify this based on the observation that by randomly sampling concepts, the number of negative examples would far exceed positive ones; yet, we still wish the model to be somewhat biased towards negative examples. Features (see Section 4.2) are extracted from the training instances and weights are set as described in Section 3.2.

For inference we use the greedy agglomerative approximation with stopping threshold $\tau = 0.5$. This is a natural threshold choice since it represents the decision boundary of the binary random variable y_i from Section 3.2.

4.4 Results

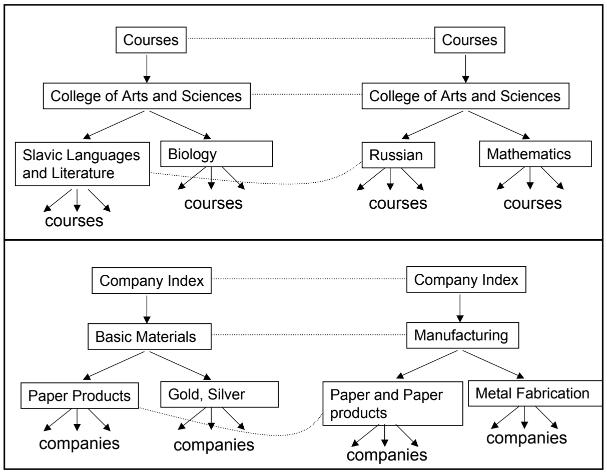

Figure 1: Examples of both ontology alignment domains. The top half gives an example of an alignment for a portion of the taxonomy trees of University of Washington (left) and Cornell University (right). Instances in this domain are courses such as "Russian Literature 100". The bottom half of the figure is an example of an alignment from the Yahoo (left) and The Standard (right). In this domain instances are names of companies that deal with the corresponding concepts, for example "HammerMill" makes paper.

We evaluate our system with two types of experiments: the first tests performance on a single domain, while the second tests generalization across the domains. Both experiments are important since the former allows comparison to previous work while the latter shows applicability in real-world problems where labeled training data may be unavailable.

For single-domain experiments, we perform two-fold cross validation and report the average. Essentially, this entails dividing the data into two disjoint sets: training on one and testing on the other (and vice versa). In this fashion we are able to evaluate the entire mapping for each domain.

For our second experiment, we train the CRF on one domain and then test on the other. We evaluate the result of training on course catalog and testing on company profile as well as the opposite: training on company profile and testing on course catalog.

All experiments are evaluated in terms of precision, recall and F1. In our problem setup, we do not assume a bijection mapping, and so we present only one set of numbers per domain. In contrast, previous systems evaluated on these datasets have had the crutch of a *1-1* mapping and presented two sets of accuracies per domain (one for the $A \rightarrow B$ direction and another for the $B \rightarrow A$ direction) [5, 14].

Analysis and Comparison of Results

When training and testing on the same domain we obtain results that are competitive with current systems. For example, GLUE [5] with the relaxation labeler configuration achieves roughly 66 to 80% accuracy for the course catalog II dataset. The best configuration of GLUE achieve between 68 and 80% accuracy on the company profile dataset. Our numbers are not directly comparable since our evaluation metrics differ (we present a global mapping score rather than two separate bijection evaluations); however, they do

	F1	Precision	Recall
courses	89.1	93.8	84.9
company	82.1	88.9	76.3
A(course to company)	65.2	79.7	55.2
B(company to course)	76.4	94.4	64.1

Table 2: Shows the precision, recall and F1 of the taxonomy tree alignment task with A) the model trained on the course catalog mapping and tested on the company profile mapping and B) the model trained on company profile and tested on course catalog. Results for 2-fold cross validation on both company profile and course catalog are shown as well.

provide a loose basis for comparison.

Our method is able to outperform GLUE achieving a 50% reduction in error on the course catalog domain (or 10% absolute improvement); we also obtain a 2% absolute increase in F1 on the company profile set. On average for both domains, our method achieves 85% matching accuracy in comparison with 74% for GLUE.

For the generalization experiments we were pleased to see a relatively small decrease in F1. The precision is particularly high (80 to 94%) implying predicted links are fairly reliable. In fact, the generalization capabilities exceeded our expectations. Our model properly learns to place emphasis on both the concept-name-features and instance-features even though these weights are learned from an entirely separate domain. For example, we correctly map "Recreational Activities" (Yahoo) and "Miscellaneous Entertainment" (The Standard) even though no words in the concept names overlap.

5. CONCLUSION AND FUTURE WORK

We have presented a fully supervised statistical model for ontology mapping based on conditional random fields. Our model accounts for uncertainty in both the data and the data's structure. We evaluated our results on two domains and showed that our supervised model is able to generalize across them. This is promising since mappings can be automatically predicted even for domains with no labeled data. However, experiments on more domains should be conducted to verify results further. Future work can also help identify classes of features with greater generalization capabilities.

From a machine learning perspective, exact learning and inference in the model is intractable and improvements may also be achieved by investigating other approximation solutions to these tasks. For example, rank-based training or stochastic inference methods. Finally, conditional random fields provide a convenient framework for modeling many tasks together. Ontology alignment can be modeled jointly with related tasks (such as ontology integration) to improve the performance of both tasks.

6. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval, in part by Lockheed Martin through prime contract #FA8650-06-C-7605 from the Air Force Office of Scientific Research, in part by The Central Intelligence Agency, the National Security Agency and National Science Foundation under NSF grant #IIS-0326249, in part by The Central Intelligence Agency, the National Security Agency and National Science Foundation under NSF grant #IIS-0427594, and in part by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under contract number NBCHD030010, and AFRL #FA8750-07-D-0185. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsor.

7. REFERENCES

[1] *BLOG: Relational Modeling with Unknown Objects*, 2003.

[2] C. Batini, M. Lenzerini, and S. B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM Comput. Surv.*, 18(4):323–364, 1986.

[3] A. Culotta, M. Wick, R. Hall, and A. McCallum. First-order probabilistic models for coreference resolution. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT/NAACL)*, pages 81–88, 2007. (24% accepted).

[4] R. Dhamankar, Y. Lee, and A. Doan. imap: Discovering complex semantic matches between database schemas. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 383 – 394, 2004.

[5] A. Doan, J. Madhavan, R. Dhamankar, P. Domingos, and A. Halevy. Learning to match ontologies on the semantic web. *The VLDB Journal*, 12(4):303–319, 2003.

[6] M. Ehrig, S. Staab, and Y. Sure. Supervised learning of an ontology alignment process. In *In KMTools workshop at Konferenz Professionelles Wissensmanagement*, 2005.

[7] B. He and K. C.-C. Chang. Statistical schema matching across web query interfaces. In *SIGMOD '03: Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 217–228, New York, NY, USA, 2003. ACM.

[8] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[9] R. J. Miller, L. M. Haas, and M. A. Hernández. Schema mapping as query discovery. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 77–88, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[10] S. Nicolas, J. Dardenne, T. Paquet, and L. Heutte. Document image segmentation using a 2d conditional random field model. In *International Conference on Document Analysis and Recognition*, pages 407–411, Washington, DC, USA, 2007. IEEE Computer Society.

[11] R. A. Pottinger and P. A. Bernstein. Merging models based on given correspondences. In *vldb'2003: Proceedings of the 29th international conference on Very large data bases*, pages 862–873. VLDB

Endowment, 2003.

[12] M. Richardson and P. Domingos. Markov logic networks. *Machine Learning*, 62:107–136, 2006.

[13] C. Sutton and A. McCallum. Piecewise training of undirected models. In *21st Conference on Uncertainty in Artificial Intelligence*, 2005.

[14] J. Tang, J. Li, B. Liang, X. Huang, Y. Li, and K. Wang. Using bayesian decision for ontology mapping. *Web Semant.*, 4(4):243–262, 2006.

[15] J. Tang, B.-Y. Liang, and J.-Z. Li. Toward detecting mapping strategies for ontology interoperability. In *WWW 2008*, 2005.

[16] M. Wick, A. Culotta, and A. McCallum. Learning field compatibilities to extract database records from unstructured text. In *EMNLP*, 2006.