Latent Relation Representations for Universal Schemas

Sebastian Riedel
Department of Computer Science
University College London
S.Riedel@cs.ucl.ac.uk

Limin Yao, Andrew McCallum
Department of Computer Science
University of Massachusetts at Amherst
{lmyao,mccallum}@cs.umass.edu

1 Introduction

Supervised relation extraction uses a pre-defined schema of relation types (such as *born-in* or *employed-by*). This approach requires labeling textual relations, a time-consuming and difficult process. This has led to significant interest in distantly-supervised learning. Here one aligns existing database records with the sentences in which these records have been "rendered", and from this labeling one can train a machine learning system as before [1, 2]. However, this method relies on the availability of a large database that has the desired schema.

The need for pre-existing databases can be avoided by not having any fixed schema. This is the approach taken by OpenIE [3]. Here surface patterns between mentions of concepts serve as relations. This approach requires no supervision and has tremendous flexibility, but lacks the ability to generalize. For example, OpenIE may find FERGUSON–*historian-at*–HARVARD but does not know FERGUSON–*is-a-professor-at*–HARVARD.

One way to gain generalization is to cluster textual surface forms that have similar meaning [4, 5, 6, 7]. While the clusters discovered by all these methods usually contain semantically related items, closer inspection invariably shows that they do not provide reliable implicature. For example, a cluster may include *historian-at*, *professor-at*, *scientist-at*, *worked-at*. However, *scientist-at* does not necessarily imply *professor-at*, and *worked-at* certainly does not imply *scientist-at*. In fact, we contend that any relational schema would inherently be brittle and ill-defined—having ambiguities, problematic boundary cases, and incompleteness.

In response to this problem, we present a new approach: implicature with *universal schemas*. Here we embrace the diversity and ambiguity of original inputs. This is accomplished by defining our schema to be the union of all source schemas: original input forms, e.g. variants of surface patterns similarly to OpenIE, as well as relations in the schemas of pre-existing structured databases. But unlike OpenIE, we learn asymmetric implicature among relations and entity types. This allows us to probabilistically "fill in" inferred unobserved entity-entity relations in this union. For example, after observing FERGUSON–*historian-at*–HARVARD, our system infers that FERGUSON–*professor-at*–HARVARD, but not vice versa.

At the heart of our approach is the hypothesis that we should concentrate on predicting source data—a relatively well defined task that can be evaluated and optimized—as opposed to modeling semantic equivalence, which we believe will always be illusive.

To reason with a universal schema, we learn latent feature representations of relations, tuples and entities. These act, through dot products, as natural parameters of a log-linear model for the probability that a given relation holds for a given tuple. We show experimentally that this approach significantly outperforms a comparable baseline without latent features, and the current state-of-the-art distant supervision method.

2 Model

We use $\mathcal{R}$ to denote the set of relations we seek to predict (such as *works-written* in Freebase, or the X–*heads*–Y pattern), and $\mathcal{T}$ to denote the set of input tuples. For simplicity we assume each relation to be binary. Given a relation $r \in \mathcal{R}$ and a tuple $t \in \mathcal{T}$ the pair $\langle r, t\rangle$ is a *fact*, or relation instance. The input to our model is a set of observed facts $\mathcal{O}$, and the observed facts for a given tuple $\mathcal{O}_t := \{\langle r, t\rangle \in \mathcal{O}\}$.

Our goal is a model that can estimate, for a given relation r (such as X–*historian-at*–Y) and a given tuple t (such as <FERGUSON,HARVARD>) a score $c_{r,t}$ for the fact $\langle r, t\rangle$. This matrix completion problem is related to collaborative filtering. We can think of each tuple as a customer, and each relation as a product. Our goal is to predict how the tuple rates the relation (rating 0 = false, rating 1 = true), based on observed ratings in $\mathcal{O}$. We interpret $c_{r,t}$ as the probability $p\left(y_{r,t} = 1\right)$ where $y_{r,t}$ is a binary random variable that is true iff $\langle r, t\rangle$ holds. To this end we introduce a series of exponential family models inspired by generalized PCA [8], a probabilistic generalization of Principle Component Analysis. These models will estimate the confidence in $\langle r, t\rangle$ using a *natural parameter* $\theta_{r,t}$ and the logistic function: $c_{r,t} := p\left(y_{r,t}|\theta_{r,t}\right) := \frac{1}{1+\exp(-\theta_{r,t})}$.

We follow[9] and use a ranking based objective function to estimate parameters of our models.

Latent Feature Model One way to define $\theta_{r,t}$ is through a latent feature model F. We measure compatibility between relation r and tuple t as a dot product of two latent feature representations of size K^{F}: $\mathbf{a}_r$ for relation r, and $\mathbf{v}_t$ for tuple t. This gives $\theta^{\mathrm{F}}_{r,t} := \sum_k^{K^{\mathrm{F}}} a_{r,k} v_{t,k}$ and corresponds to the original generalized PCA that learns a low-rank factorization of $\Theta = (\theta_{r,t})$.

Neighborhood Model We can interpolate the confidence for a given tuple and relation based on the trueness of other similar relations for the same tuple. In Collaborative Filtering this is referred as a *neighborhood-based* approach [10]. We implement a neighborhood model N via a set of *weights* $w_{r,r'}$, where each corresponds to a directed association strength between relations r and r'. Summing these up gives $\theta^{\mathrm{N}}_{r,t} := \sum_{r' \in \mathcal{O}_t \setminus \{r\}} w_{r,r'}$.[1]

Entity Model Relations have selectional preferences: they allow only certain types in their argument slots. To capture this observation, we learn a latent entity representation from data. For each entity e we introduce a latent feature vector $\mathbf{t}_e \in R^l$. In addition, for each relation r and argument slot i we introduce a feature vector $\mathbf{d}_i$. Measuring compatibility of an entity tuple and relation amounts to summing up the compatibilities between each argument slot representation and the corresponding entity representation: $\theta^{\mathrm{E}}_{r,t} := \sum_{i=1}^{\mathrm{arity}(r)} \sum_k^{K^{\mathrm{E}}} d_{i,k} t_{t_i,k}$.

Combined Models In practice all the above models can capture important aspects of the data. Hence we also use various combinations, such as $\theta^{\mathrm{N,F,E}}_{r,t} := \theta^{N}_{r,t} + \theta^{\mathrm{F}}_{r,t} + \theta^{\mathrm{E}}_{r,t}$.

3 Experiments

Does reasoning jointly across a universal schema help to improve over more isolated approaches? In the following we seek to answer this question empirically.

Data Our experimental setup is roughly equivalent to previous work [2], and hence we omit details. To summarize, we consider each pair $\langle t_1, t_2\rangle$ of Freebase entities that appear together in a corpus. Its set of observed facts $\mathcal{O}_t$ correspond to: Extracted surface patterns (in our case *lexicalized dependency paths*) between mentions of t_1 and t_2, and the relations of t_1 and t_2 in Freebase. We divide all our tuples into approximately 200k training tuples, and 200k test tuples. The total number of relations (patterns and from Freebase) is approximately 4k.

[1] Notice that the neighborhood model amounts to a collection of local log-linear classifiers, one for each relation r with weights $\mathbf{w}_r$.

Predicting Freebase and Surface Pattern Relations For evaluation we use two collections of relations: Freebase relations and surface patterns. In either case we compare the competing systems with respect to their ranked results for each relation in the collection.

Our first baseline is MI09, a distantly supervised classifier based on the work of [1]. We also compare against YA11, a version of MI09 that uses preprocessed pattern cluster features according to [7]. The third baseline is SU12, the state-of-the-art Multi-Instance Multi-Label system by [11]. The remaining systems are our neighborhood model (N), the factorized model (F), their combination (NF) and the combined model with a latent entity representation (NFE).

The results in terms of mean average precision (with respect to pooled results from each system) are in the table below:

Relation	#	MI09	YA11	SU12	N	F	NF	NFE
Total Freebase	334	0.48	0.52	0.57	0.52	0.66	0.67	0.69
Total Pattern	329				0.28	0.56	0.50	0.46

For Freebase relations, we can see that adding pattern cluster features (and hence incorporating more data) helps YA11 to improve over MI09. Likewise, we see that the factorized model F improves over N, again learning from unlabeled data. This improvement is bigger than the corresponding change between MI09 and YA11, possibly indicating that our latent representations are optimized directly towards improving prediction performance. Our best model, the combination of N, F and E, outperforms all other models in terms of total MAP, indicating the power of selectional preferences learned from data.

MI09, YA11 and SU12 are designed to predict structured relations, and so we omit them for results on surface patterns. Look at our models for predicting tuples of surface patterns. We again see that learning a latent representation (F, NF and NFE models) from additional data helps substantially over the non-latent N model.

All our models are fast to train. The slowest model trains in just 30 minutes. By contrast, training the topic model in YA11 alone takes 4 hours. Training SU12 takes two hours (on less data). Also notice that our models not only learn to predict Freebase relations, but also approximately 4k surface pattern relations.

4 Conclusion

We represent relations using universal schemas. Such schemas contain surface patterns as relations, as well as relations from structured sources. We can predict missing tuples for surface pattern relations and structured schema relations. We show this experimentally by contrasting a series of popular weakly supervised models to our collaborative filtering models that learn latent feature representations across surface patterns and structured relations. Moreover, our models are computationally efficient, requiring less time than comparable methods, while learning more relations.

Reasoning with universal schemas is not merely a tool for information extraction. It can also serve as a framework for various data integration tasks, for example, schema matching. In future work we also plan to integrate universal entity types and attributes into the model.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by an award from Google, in part by UPenn NSF medium IIS-0803847 and in part by DARPA under agreement number FA8750-13-2-0020. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

References

[1] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (ACL '09)*, pages 1003–1011. Association for Computational Linguistics, 2009.

[2] Sebastian Riedel, Limin Yao, and Andrew McCallum. Modeling relations and their mentions without labeled text. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '10)*, 2010.

[3] Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. Open information extraction from the web. *Commun. ACM*, 51(12):68–74, 2008.

[4] Dekang Lin and Patrick Pantel. DIRT - discovery of inference rules from text. In *Knowledge Discovery and Data Mining*, pages 323–328, 2001.

[5] Patrick Pantel, Rahul Bhagat, Bonaventura Coppola, Timothy Chklovski, and Eduard Hovy. ISP: Learning Inferential Selectional Preferences. In *Proceedings of NAACL HLT*, 2007.

[6] Alexander Yates and Oren Etzioni. Unsupervised methods for determining object and relation synonyms on the web. *Journal of Artificial Intelligence Research*, 34:255–296, 2009.

[7] Limin Yao, Aria Haghighi, Sebastian Riedel, and Andrew McCallum. Structured relation discovery using generative models. In *Proceedings of the Conference on Empirical methods in natural language processing (EMNLP '11)*, July 2011.

[8] Michael Collins, Sanjoy Dasgupta, and Robert E. Schapire. A generalization of principal component analysis to the exponential family. In *Proceedings of NIPS*, 2001.

[9] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of UAI*, 2009.

[10] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 426–434, New York, NY, USA, 2008. ACM.

[11] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. Multi-instance multi-label learning for relation extraction. In *Proceedings of EMNLP-CoNLL*, 2012.