Using Lexical Expansion to Learn Inference Rules from Sparse Data

Oren Melamud§, **Ido Dagan**§, **Jacob Goldberger**†, **Idan Szpektor**‡
§ Computer Science Department, Bar-Ilan University
† Faculty of Engineering, Bar-Ilan University
‡ Yahoo! Research Israel
{melamuo,dagan,goldbej}@{cs,cs,eng}.biu.ac.il
idan@yahoo-inc.com

Abstract

Automatic acquisition of inference rules for predicates is widely addressed by computing distributional similarity scores between vectors of argument words. In this scheme, prior work typically refrained from learning rules for low frequency predicates associated with very sparse argument vectors due to expected low reliability. To improve the learning of such rules in an unsupervised way, we propose to lexically expand sparse argument word vectors with semantically similar words. Our evaluation shows that lexical expansion significantly improves performance in comparison to state-of-the-art baselines.

1 Introduction

The benefit of utilizing template-based inference rules between predicates was demonstrated in NLP tasks such as Question Answering (QA) (Ravichandran and Hovy, 2002) and Information Extraction (IE) (Shinyama and Sekine, 2006). For example, the inference rule '*X treat Y* → *X relieve Y*', between the templates '*X* treat *Y*' and '*X* relieve *Y*' may be useful to identify the answer to "*Which drugs relieve stomach ache?*".

The predominant unsupervised approach for learning inference rules between templates is via distributional similarity (Lin and Pantel, 2001; Ravichandran and Hovy, 2002; Szpektor and Dagan, 2008). Specifically, each argument slot in a template is represented by an argument vector, containing the words (or terms) that instantiate this slot in all of the occurrences of the template in a learning corpus. Two templates are then deemed semantically similar if the argument vectors of their corresponding slots are similar.

Ideally, inference rules should be learned for all templates that occur in the learning corpus. However, many templates are rare and occur only few times in the corpus. This is a typical NLP phenomenon that can be associated with either a small learning corpus, as in the cases of domain specific corpora and resource-scarce languages, or with templates with rare terms or long multi-word expressions such as '*X* be also a risk factor to *Y*' or '*X* finish second in *Y*', which capture very specific meanings. Due to few occurrences, the slots of rare templates are represented with very sparse argument vectors, which in turn lead to low reliability in distributional similarity scores.

A common practice in prior work for learning predicate inference rules is to simply disregard templates below a minimal frequency threshold (Lin and Pantel, 2001; Kotlerman et al., 2010; Dinu and Lapata, 2010; Ritter et al., 2010). Yet, acquiring rules for rare templates may be beneficial both in terms of coverage, but also in terms of more accurate rule application, since rare templates are less ambiguous than frequent ones.

We propose to improve the learning of rules between infrequent templates by expanding their argument vectors. This is done via a "dual" distributional similarity approach, in which we consider two words to be similar if they instantiate similar sets of templates. We then use these similarities to expand the argument vector of each slot with words that were identified as similar to the original arguments in the vector. Finally, similarities between templates are computed using the expanded vectors, resulting in a *'smoothed'* version of the original similarity measure.

Evaluations on a rule application task show that our lexical expansion approach significantly improves the performance of the state-of-the-art DIRT algorithm (Lin and Pantel, 2001). In addition, our approach outperforms a similarity measure based on vectors of latent topics instead of word vectors, a common way to avoid sparseness issues by means of dimensionality reduction.

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 283–288,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics

2 Technical Background

The distributional similarity score for an inference rule between two predicate templates, *e.g.* '*X resign Y* → *X quit Y*', is typically computed by measuring the similarity between the argument vectors of the corresponding X slots and Y slots of the two templates. To this end, first the argument vectors should be constructed and then a similarity measure between two vectors should be provided. We note that we focus here on binary templates with two slots each, but this approach can be applied to any template.

A common starting point is to compute a co-occurrence matrix M from a learning corpus. M's rows correspond to the template slots and the columns correspond to the various terms that instantiate the slots. Each entry $M_{i,j}$, *e.g.* $M_{x\ quit,John}$, contains a count of the number of times the term j instantiated the template slot i in the corpus. Thus, each row $M_{i,*}$ corresponds to an argument vector for slot i. Next, some function of the counts is used to assign weights to all $M_{i,j}$ entries. In this paper we use pointwise mutual information (PMI), which is common in prior work (Lin and Pantel, 2001; Szpektor and Dagan, 2008).

Finally, rules are assessed using some similarity measure between corresponding argument vectors. The state-of-the-art DIRT algorithm (Lin and Pantel, 2001) uses the highly cited *Lin* similarity measures (Lin, 1998) to score rules between binary templates as follows:

$$Lin(v, v') = \frac{\sum_{w \in v \cap v'} [v(w) + v'(w)]}{\sum_{w \in v \cup v'} [v(w) + v'(w)]} \quad (1)$$

$$\begin{aligned} &DIRT(l \rightarrow r) \\ &= \sqrt{Lin(v_{l:x}, v_{r:x}) \cdot Lin(v_{l:y}, v_{r:y})} \end{aligned} \quad (2)$$

where v and v' are two argument vectors, l and r are the templates participating in the inference rule and $v_{l:x}$ corresponds to the argument vector of slot X of template l, etc. While the original DIRT algorithm utilizes the *Lin* measure, one can replace it with any other vector similarity measure.

A separate line of research for word similarity introduced directional similarity measures that have a bias for identifying generalization/specification relations, *i.e.* relations between predicates with narrow (or specific) semantic meanings to predicates with broader meanings inferred by them (unlike the symmetric *Lin*). One such example is the *Cover* measure (Weeds and Weir, 2003):

$$Cover(v, v') = \frac{\sum_{w \in v \cap v'} [v(w)]}{\sum_{w \in v \cup v'} [v(w)]} \quad (3)$$

As can be seen, in the core of the *Lin* and *Cover* measures, as well as in many other well known distributional similarity measures such as Jaccard, Dice and Cosine, stand the number of shared arguments vs. the total number of arguments in the two vectors. Therefore, when the argument vectors are sparse, containing very few non-zero features, these scores become unreliable and volatile, changing greatly with every inclusion or exclusion of a single shared argument.

3 Lexical Expansion Scheme

We wish to overcome the sparseness issues in rare feature vectors, especially in cases where argument vectors of semantically similar predicates comprise similar but not exactly identical arguments. To this end, we propose a three step scheme. First, we learn lexical expansion sets for argument words, such as the set {*euros, money*} for the word *dollars*. Then we use these sets to expand the argument word vectors of predicate templates. For example, given the template '*X* can be exchanged for *Y*', with the following argument words instantiating slot X {*dollars, gold*}, and the expansion set above, we would expand the argument word vector to include all the following words {*dollars, euros, money, gold*}. Finally, we use the expanded argument word vectors to compute the scores for predicate inference rules with a given similarity measure.

When a template is instantiated with an observed word, we expect it to also be instantiated with semantically similar words such as the ones in the expansion set of the observed word. We "blame" the lack of such template occurrences only on the size of the corpus and the sparseness phenomenon in natural languages. Thus, we utilize our lexical expansion scheme to synthetically add these expected but missing occurrences, effectively smoothing or generalizing over the explicitly observed argument occurrences. Our approach is inspired by query expansion (Voorhees, 1994) in Information Retrieval (IR), as well as by the recent lexical expansion framework proposed in (Biemann and Riedl, 2013), and the work by

Miller et al. (2012) on word sense disambiguation. Yet, to the best of our knowledge, this is the first work that applies lexical expansion to distributional similarity feature vectors. We next describe our scheme in detail.

3.1 Learning Lexical Expansions

We start by constructing the co-occurrence matrix M (Section 2), where each entry $M_{t:s,w}$ indicates the number of times that word w instantiates slot s of template t in the learning corpus, denoted by *'t:s'*, where s can be either X or Y.

In traditional distributional similarity, the rows $M_{t:s,*}$ serve as argument vectors of template slots. However, to learn expansion sets we take a "dual" view and consider each matrix column $M_{*:*,w}$ (denoted v_w) as a feature vector for the argument word w. Under this view, templates (or more specifically, template slots) are the features. For instance, for the word *dollars* the respective feature vector may include entries such as '*X can be exchanged for*', '*can be exchanged for Y*', '*purchase Y*' and '*sell Y*'.

We next learn an expansion set per each word w by computing the distributional similarity between the vectors of w and any other argument word w', $sim(v_w, v_{w'})$. Then we take the N most similar words as w's expansion set with degree N, denoted by $L_w^N = \{w'_1, ..., w'_N\}$. Any similarity measure could be used, but as our experiments show, different measures generate sets with different properties, and some may be fitter for argument vector expansion than others.

3.2 Expanding Argument Vectors

Given a row count vector $M_{t:s,*}$ for slot s of template t, we enrich it with expansion sets as follows. For each w in $M_{t:s,*}$, the original count in $v_{t:s}(w)$ is redistributed equally between itself and all words in w's expansion set, i.e. all $w' \in L_w^N$, (possibly yielding fractional counts) where N is a global parameter of the model. Specifically, the new count that is assigned to each word w is its remaining original count after it has been redistributed (or zero if no original count), plus all the counts that were distributed to it from other words.

Next, PMI weights are recomputed according to the new counts, and the resulting expanded vector is denoted by $v_{t:s}^+$. Similarity between template slots is now computed over the expanded vectors instead of the original ones, *e.g.* $Lin(v_{l:x}^+, v_{r:x}^+)$.

4 Experimental Settings

We constructed a relatively small learning corpus for investigating the sparseness issues of such corpora. To this end, we used a random sample from the large scale web-based ReVerb corpus[1] (Fader et al., 2011), comprising tuple extractions of predicate templates with their argument instantiations. We applied some clean-up preprocessing to these extractions, discarding stop words, rare words and non-alphabetical words that instantiated either the X or the Y argument slots. In addition, we discarded templates that co-occur with less than 5 unique argument words in either of their slots, assuming that such few arguments cannot convey reliable semantic information, even with expansion. Our final corpus consists of around 350,000 extractions and 14,000 unique templates. In this corpus around one third of the extractions refer to templates that co-occur with at most 35 unique arguments in both their slots.

We evaluated the quality of inference rules using the dataset constructed by Zeichner et al. (2012)[2], which contains about 6,500 manually annotated template rule applications, each labeled as correct or not. For example, '*The game develop eye-hand coordination $\nrightarrow$ The game launch eye-hand coordination*' is a rule application in this dataset of the rule '*X develop Y $\rightarrow$ X launch Y*', labeled as incorrect, and '*Captain Cook sail to Australia $\rightarrow$ Captain Cook depart for Australia*' is a rule application of the rule '*X sail to Y $\rightarrow$ X depart for Y*', labeled as correct. Specifically, we induced two datasets from Zeichner et al.'s dataset, denoted *DS-5-35* and *DS-5-50*, which consist of all rule applications whose templates are present in our learning corpus and co-occurred with at least 5 and at most 35 and 50 unique argument words in both their slots, respectively. *DS-5-35* includes 311 rule applications (104 correct and 207 incorrect) and *DS-5-50* includes 502 rule applications (190 correct and 312 incorrect).

Our evaluation task is to rank all rule applications in each test set based on the similarity scores of the applied rules. Optimal performance would rank all correct rule applications above the incorrect ones. As a baseline for rule scoring we

[1] http://reverb.cs.washington.edu/

[2] http://www.cs.biu.ac.il/nlp/downloads/annotation-rule-application.htm

used the DIRT algorithm scheme, denoted *DIRT-LE-None*. We then compared between the performance of this baseline and its expanded versions, testing two similarity measures for generating the expansion sets of arguments: *Lin* and *Cover*. We denote these expanded methods *DIRT-LE-SIM-N*, where *SIM* is the similarity measure used to generate the expansion sets and *N* is the lexical expansion degree, *e.g. DIRT-LE-Lin-2*.

We remind the reader that our scheme utilizes two similarity measures. The first measure assesses the similarity between the argument vectors of the two templates in the rule. This measure is kept constant in our experiments and is identical to DIRT's similarity measure (*Lin*). [3] The second measure assesses the similarity between words and is used for the lexical expansion of argument vectors. Since this is the research goal of this paper, we experimented with two different measures for lexical expansion: a symmetric measure (Lin) and an asymmetric measure (Cover). To this end we evaluated their effect on DIRT's rule ranking performance and compared them to a vanilla version of DIRT without lexical expansion.

As another baseline, we follow Dinu and Lapata (2010) inducing LDA topic vectors for template slots and computing predicate template inference rule scores based on similarity between these vectors. We use standard hyperparameters for learning the LDA model (Griffiths and Steyvers, 2004). This method is denoted *LDA-K*, where *K* is the number of topics in the model.

5 Results

We evaluated the performance of each tested method by measuring Mean Average Precision (MAP) (Manning et al., 2008) of the rule application ranking computed by this method. In order to compute MAP values and corresponding statistical significance, we randomly split each test set into 30 subsets. For each method we computed Average Precision on every subset and then took the average as the MAP value. We varied the degree of the lexical expansion in our model and the number of topics in the topic model baseline to analyze their effect on the performance of these methods on our datasets. We note that in our model a greater degree of lexical expansion corresponds to more aggressive smoothing (or generalization) of the explicitly observed data, while the same goes for a lower number of topics in the topic model. The results on *DS-5-35* and *DS-5-50* are illustrated in Figure 1.

The most dramatic improvement over the baselines is evident in *DS-5-35*, where *DIRT-LE-Cover-2* achieves a MAP score of 0.577 in comparison to 0.459 achieved by its *DIRT-LE-None* baseline. This is indeed the dataset where we expected expansion to affect most due the extreme sparseness of argument vectors. Both *DIRT-LE-Cover-N* and *DIRT-LE-Lin-N* outperform *DIRT-LE-None* for all tested values of N, with statistical significance via a paired t-test at $p < 0.05$ for *DIRT-LE-Cover-N* where $1 \leq N \leq 5$, and $p < 0.01$ for *DIRT-LE-Cover-2*. On *DS-5-50*, improvement over the *DIRT-LE-None* baseline is still significant with both *DIRT-LE-Cover-N* and *DIRT-LE-Lin-N* outperforming *DIRT-LE-None*. *DIRT-LE-Cover-N* again performs best and achieves a relative improvement of over 10% with statistical significance at $p < 0.05$ for $2 \leq N \leq 3$.

The above shows that expansion is effective for improving rule learning between infrequent templates. Furthermore, the fact that *DIRT-LE-Cover-N* outperforms *DIRT-LE-Lin-N* suggests that using directional expansions, which are biased to *generalizations* of the observed argument words, *e.g. vehicle* as an expansion for *car*, is more effective than using symmetrically related words, such as *bicycle* or *automobile*. This conclusion appears also to be valid from a semantic reasoning perspective, as given an observed predicate-argument occurrence, such as *'drive car'* we can more likely infer that a presumed occurrence of the same predicate with a *generalization* of the argument, such as *'drive vehicle'*, is valid, i.e. '*drive car* $\rightarrow$ *drive vehicle*'. On the other hand while '*drive car* $\rightarrow$ *drive automobile*' is likely to be valid, '*drive car* $\rightarrow$ *drive bicycle*' and '*drive vehicle* $\rightarrow$ *drive bicycle*' are not.

Figure 1 also depicts the performance of LDA as a vector smoothing approach. *LDA-K* outperforms the *DIRT-LE-None* baseline under *DS-5-35* but with no statistical significance. Under *DS-5-50 LDA-K* performs worst, slightly outperforming *DIRT-LE-None* only for K=450. Furthermore, under both datasets, *LDA-K* is outperformed by *DIRT-LE-Cover-N*. These results indicate that LDA is less effective than our expansion approach.

[3] Experiments with *Cosine* as the template similarity measure instead of *Lin* for both DIRT and its expanded versions yielded similar results. We omit those for brevity.

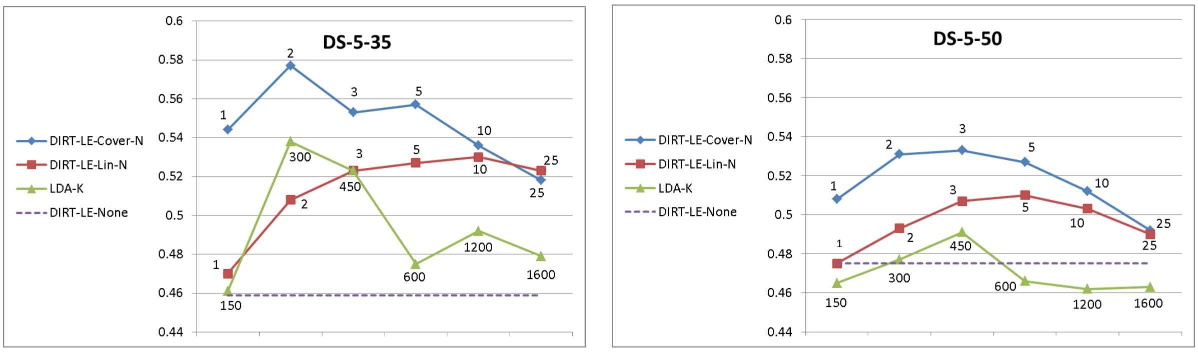

Figure 1: MAP scores on *DS-5-35* and *DS-5-50* for the original DIRT scheme, denoted *DIRT-LE-None*, and for the compared smoothing methods as follows. DIRT with varied degrees of lexical expansion is denoted as *DIRT-LE-Lin-N* and *DIRT-LE-Cover-N*. The topic model with varied number of topics is denoted as *LDA-K*. Data labels indicate the expansion degree (*N*) or the number of LDA topics (*K*), depending on the tested method.

One reason may be that in our model, every expansion set may be viewed as a cluster around a specific word, an outstanding difference in comparison to topics, which provide a global partition over all words. We note that performance improvement of singleton document clusters over global partitions was also shown in IR (Kurland and Lee, 2009).

In order to further illustrate our lexical expansion scheme we focus on the rule application '*Captain Cook sail to Australia* → *Captain Cook depart for Australia*', which is labeled as correct in our test set and corresponds to the rule '*X sail to Y* → *X depart for Y*'. There are 30 words instantiating the X slot of the predicate *'sail to'* in our learning corpus including {*Columbus, emperor, James, John, trader*}. On the other hand, there are 18 words instantiating the X slot of the predicate *'depart for'* including {*Amanda, Jerry, Michael, mother, queen*}. While semantic similarity between these two sets of words is evident, they share no words in common, and therefore the original DIRT algorithm, *DIRT-LE-None*, wrongly assigns a zero score to the rule.

The following are descriptions of some of the argument word expansions performed by *DIRT-LE-Cover-2* (using the notation L_w^N defined in Section 3.1) for the X slot of *'sail to'* L^2_{John} = {*mr., dr.*}, L^2_{trader} = {*people, man*}, and for the X slot of *'depart for'*, $L^2_{Michael}$ = {*John, mr.*}, L^2_{mother} = {*people, woman*}. Given these expansions the two slots now share the following words {*mr. ,people, John*} and the rule score becomes positive.

It is also interesting to compare the expansions performed by *DIRT-LE-Lin-2* to the above. For instance in this case L^2_{mother} = {*father, sarah*}, which does not identify *people* as a shared argument for the rule.

6 Conclusions

We propose to improve the learning of inference rules between infrequent predicate templates with sparse argument vectors by utilizing a novel scheme that lexically expands argument vectors with semantically similar words. Similarities between argument words are discovered using a dual distributional representation, in which templates are the features.

We tested the performance of our expansion approach on rule application datasets that were biased towards rare templates. Our evaluation showed that rule learning with expanded vectors outperformed the baseline learning with original vectors. It also outperformed an LDA-based similarity model that overcomes sparseness via dimensionality reduction.

In future work we plan to investigate how our scheme performs when integrated with manually constructed resources for lexical expansion, such as WordNet (Fellbaum, 1998).

Acknowledgments

This work was partially supported by the Israeli Ministry of Science and Technology grant 3-8705, the Israel Science Foundation grant 880/12, and the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 287923 (EXCITEMENT).

References

Chris Biemann and Martin Riedl. 2013. Text: Now in 2d! a framework for lexical expansion with contextual similarity. *Journal of Language Modeling*, 1(1).

Georgiana Dinu and Mirella Lapata. 2010. Topic models for meaning similarity in context. In *Proceedings of COLING: Posters*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of EMNLP*.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Thomas L Griffiths and Mark Steyvers. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(4):359–389.

Oren Kurland and Lillian Lee. 2009. Clusters, language models, and ad hoc information retrieval. *ACM Transactions on Information Systems (TOIS)*, 27(3):13.

Dekang Lin and Patrick Pantel. 2001. DIRT – discovery of inference rules from text. In *Proceedings of KDD*.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL*.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge.

Tristan Miller, Chris Biemann, Torsten Zesch, and Iryna Gurevych. 2012. Using distributional similarity for lexical expansion in knowledge-based word sense disambiguation. *Proceedings of COLING, Mumbai, India*.

Deepak Ravichandran and Eduard Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of ACL*.

Alan Ritter, Oren Etzioni, et al. 2010. A latent dirichlet allocation method for selectional preferences. In *Proceedings of ACL*.

Yusuke Shinyama and Satoshi Sekine. 2006. Preemptive information extraction using unrestricted relation discovery. In *Proceedings of NAACL*.

Idan Szpektor and Ido Dagan. 2008. Learning entailment rules for unary templates. In *Proceedings of COLING*.

Ellen M Voorhees. 1994. Query expansion using lexical-semantic relations. In *Proceedings of SIGIR*.

Julie Weeds and David Weir. 2003. A general framework for distributional similarity. In *Proceedings of EMNLP*.

Naomi Zeichner, Jonathan Berant, and Ido Dagan. 2012. Crowdsourcing inference-rule evaluation. In *Proceedings of ACL (short papers)*.