Topic Modeling for Selectional Preferences

Anonymous

Abstract

Computation of *selectional preferences*, the admissible argument values for a relation, is a well-studied NLP task with wide applicability. We present LDA-SP, a novel approach to computing selectional preferences using a generative topic model called LinkLDA. Topic models are a natural fit for our task, since they generate the relation arguments using a two step process, first generating a hidden topic (an abstract representation for the type of the argument) based on the relation and then generating the argument based on the topic. We compare LDA-SP to the state of the art approaches achieving a 11% increase in the area under precision-recall curve compared to a mutual information based algorithm on a pseudodisambiguation evaluation. We evaluate LDA-SP's effectiveness on an end-task of identifying inapplicable inferences in a textual inference system, where we also show a large improvement in performance.

1 Introduction

Selectional Preferences maintain the set of admissible argument values for a relation. For example, locations are likely to appear in the second argument of the relation *X is headquartered in Y*. A large, high-quality database of preferences has the potential to improve the performance of a wide range of NLP tasks including semantic role labeling (Gildea and Jurafsky, 2002), pronoun resolution (Bergsma et al., 2008), textual inference (Pantel et al., 2007), word-sense disambiguation (Resnik, 1997), and many more. Therefore, much attention has been focused on automatically computing them based on a corpus of relation instances. Resnik (1996) presented the earliest work in this area, describing an information-theoretic approach that generalized selectional preferences based on the WordNet classes. Recent work (Erk, 2007; Bergsma et al., 2008) has moved away from generalization to known classes, instead utilizes the distributional similarity between nouns to generalize beyond observed relation-argument combinations. This avoids problems like WordNet's poor coverage of proper nouns and is shown to perform much better (Erk, 2007; Bergsma et al., 2008). However, they no longer produce the generalized class for an argument.

Unsupervised topic models, such as latent Dirichlet allocation (LDA) and its variants (Blei et al., 2003), have become immensely popular in recent times. They are characterized by a set of hidden topics, which represent the underlying semantic structure of a document collection. Because they are able to discover patterns of word use and connect documents that exhibit similar patterns in a completely unsupervised fashion, topic models have been applied to a wide variety of NLP applications such as summarization (Daumé III and Marcu, 2006), document allignment and segmentation (Chen et al., 2009), infering class-attribute hierarchies (Reisinger and Pasca, 2009) and many more.

In this paper we describe the application of topic models to the task of computing selectional preferences. These topics have a natural interpretation for us – they represent the (latent) set of classes that store the preferences for the different relations. Thus, these models are a great fit for modeling our relation data. Moreover, our approach is able to combine benefits of both kinds of existing methods: it retains the generalization and human-interpretability of class-based methods as well as the lexical coverage of the direct methods.

In particular, our system, called LDA-SP, uses LinkLDA (Erosheva et al., 2004), which is a variant of LDA and simultaneously models two sets of multinomials for each topic. These two sets represent the two arguments for the relations. Thus, LinkLDA is able to capture and exploit the correlations between the topics, which is missed by other existing approaches.

We run LDA-SP on a massive dataset of relations extracted by TEXTRUNNER from the Web (Banko and Etzioni, 2008). Our experiments demonstrate that LDA-SP outperforms state of the art approaches to selectional preferences by a wide margin obtaining an 11% increase in the area under precision-recall curve on the standard pseudo-disambiguation task. We additionally test the effectiveness of the output of LDA-SP on an end-task of textual inference introduced by Pantel *et al* (2007). LDA-SP outperforms Pantel's system on this task.

Additionally, we note that because of the topic-based nature of our output, manually labeling this small set of topics automatically results in a large repository of human-interpretable selectional preferences for the thousands of relations extracted from the Web. Our preliminary experiments detrmine the precision of our repository to be about 76%. We plan to release this resource for the benefit of the NLP community.

2 Related Work

Previous work on selectional preferences can be broken into four categories: class-based approaches (Resnik, 1996; Li and Abe, 1998; Clark and Weir, 2002), similarity based approaches (Dagan et al., 1999; Erk, 2007), discriminative (Bergsma et al., 2008), and generative probabilistic models (Rooth et al., 1999).

Class-based approaches, first proposed by Resnik (1996), are the most studied of the four. They make use of a pre-defined set of classes, either manually produced (e.g. WORDNET), or automatically generated (Pantel, 2003). For each relation, some measure of the overlap between the classes and observed arguments is used to identify the classes that best describe selectional preference. These techniques produce a human-interpretable output, but often suffer in quality or coverage due to an incoherent taxonomy, inability to map arguments to a class (poor lexical coverage), or word sense ambiguity.

Because of these limitations researchers have investigated non-class based approaches, which attempt to directly classify a given noun-phrase as plausible/implausible for a relation. Of these *similarity based approaches* make use of a distributional similarity measure between arguments and evaluate the following heuristic scoring function:

$$S_{\mathrm{rel}}(\mathrm{arg}) = \sum_{\mathrm{arg}' \in \mathrm{Seen}(\mathrm{rel})} \mathrm{sim}(\mathrm{arg}, \mathrm{arg}') \cdot \mathrm{wt}_{\mathrm{rel}}(\mathrm{arg})$$

Erk (2007) showed the advantages of this approach over Resnik's information-theoretic class-based method on a pseudodisambiguation evaluation. As we show in Section 4 our novel method achieves a 62% increase in recall at 90% precision over the strongest performing similarity based system which uses the Mutual-Information based similarity measure proposed by Lin (1998).

Our solution fits into the general category of *generative probabilistic models*, which model each relation/argument combination as being generated by a latent class variable. These classes are automatically learnt from the data. This retains the class-based flavor of the problem, without the knowledge limitations of the explicit class-based approaches. Probably the closest to our work is a model proposed by Rooth (1999), in which each class corresponds to a multinomial over relations and arguments and EM is used to learn the parameters of the model. In contrast, we use a Linked Latent Dirichlet Allocation framework in which each predicate is associated with a corresponding multinomial distribution over classes, and each argument is drawn from a class-specific distribution over words; LinkLDA models correlations between classes in both arguments. Additionally we perform full Bayesian inference using Collapsed Gibbs Sampling, in which parameters are integrated out, in contrast to EM (a maximum likelihood approach).

Recently, Bergsma et. al. (2008) proposed the first *discriminative approach* to selectional preferences. Their novel insight that pseudo-negative examples could be used as training data allows the application of an SVM classifier, which makes use of many features in addition to the relation-argument co-occurrence frequencies used by other methods. They automatically generated positive and negative examples by selecting arguments having high and low Mutual Information with the relation.

3 Topic Modeling for Selectional Preferences

We are given a set R of binary[1] relations and a corpus $D = \{r(a_1, a_2)\}$ of extracted instances for these relations. Our task is to compute, for each argument a_i of each relation r, a set of usual argument values (noun phrases) that it takes. For example, for the relation *is headquartered in* the selectional preference for the first argument will include companies like *Microsoft, Intel, General Motors* and second argument will favor locations like *New York, California, Seattle*.

A scalable and robust method to compute selectional preferences needs to be able to deal with hundreds of thousands of relations found on the Web, noisy extractors and data sparsity issues. The class-based methods are a natural fit to this task, but they suffer from the inconsistent quality of the class-taxonomy (in most cases, Wordnet) as well as the inability to match an argument to a class. Automatically computed classes (in the form of clusters) have performed better than Wordnet-based, however, they assume an additional set of clusters input to the problem (Pantel et al., 2007). In this paper, we retain the class-based flavor of the problem, but relax the need for a known taxonomy or a set of clusters by the use of *topic models*.

We present a novel formulation of our problem as an unsupervised generative topic model called *Linked Latent Dirichlet Allocation* (Erosheva et al., 2004), or in short, LinkLDA. LinkLDA is able to simultaneously learn the classes as topics and determine the selectional preferences of all relations as a distribution over these computed topics.

3.1 Applying the LinkLDA Model

All LDA based models postulate a set of latent "topic" variables, which generate the observed data. A topic in our case has a direct interpretation, and represents a semantically coherent class of nouns, like companies, or people. Indeed, it is the simultaneous computation of these topics that eliminates the need for a known taxonomy.

Figure 1 illustrates the LinkLDA model in the plate notation, which is analogous to the model in (Erosheva et al., 2004). The key difference in LinkLDA (versus LDA) is that instead of one, it maintains *two* sets of topics (latent distributions over words) denoted by β and γ, one for classes of each argument. A topic id k represents a pair of topics, say β_k and γ_k, that often co-occur in many relations. Common examples include *(Person, Location), (Politician, Political issue), etc.* The hidden variable $z_1 = k$ indicates that the noun phrase for the first argument was drawn from the multinomial β_k, whereas $z_2 = k$ indicates that the second argument was drawn from γ_k. The per-relation distribution θ_r is a multinomial over the topics and represents the selectional preferences, both for arg1s and arg2s of a relation r.

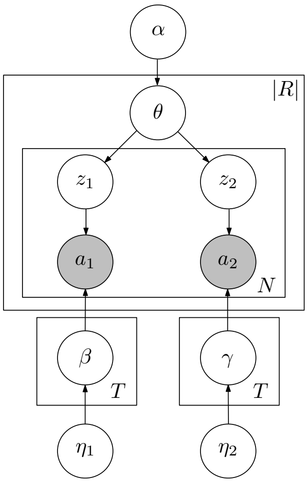

Figure 1: LinkLDA model for generating the corpus of extractions for all relations. The topics β_k and γ_k represent the common classes found automatically after learning.

Formally, LinkLDA generates the corpus as follows:

1. For each topic $t = 1 \ldots T$, generate β_t and γ_t according to symmetric Dirichlet distribution Dir(η_1) and Dir(η_2) respectively.
2. For each relation $r = 1 \ldots |R|$
   - generate θ_r according to Dirichlet distribution Dir(α).
   - for each tuple $i = 1 \ldots N_r$
     - generate two topics $z_{r,i,1}$ and $z_{r,i,2}$ from multinomial θ_r
     - generate the arguments $a_{r,i,1}$ and $a_{r,i,2}$ from multinomials $\beta_{z_{r,i,1}}$ and $\gamma_{z_{r,i,2}}$ respectively.

To facilitate learning correlated topic pairs for a topic id we employ sparse prior over the per-relation topic distributions. This makes it more likely that the same topic id will be drawn for the two arguments.

Similar models have recently been applied to other natural language tasks like simultaneously modeling the words appearing in blog posts and

[1] We focus on binary relations, though the techniques presented in the paper are easily extensible to n-ary relations.

users who will likely respond to them (Yano et al., 2009), and modeling topic-aligned articles in different languages (Mimno et al., 2009). In all these applications the data being modeled by two sets of topics is of different types – users and words, or words in different languages. In contrast, our noun phrases for the two arguments are governed by the same vocabulary. Still, use of LinkLDA is advantageous because it allows the correlations between different topics to guide inference. For example, if a particular relation has a politician as it's first argument, then it's more likely that the second argument will be a political issue than a type of computer software.

3.2 Advantages of LinkLDA

Modeling our corpus via LinkLDA has several advantages, both theoretical as well as empirical. First, we argue that it is a right fit for our problem of computing selectional preferences.

LinkLDA naturally models the class-based nature of selectional preferences, but doesn't take the classes as input – rather computes the classes automatically. This leads to better lexical coverage since the issue of matching a new argument to a known class is side-stepped. Because the computation of classes looks at statistical regularities in the data, LinkLDA is very robust to noise. Moreover, it easily handles the ambiguity of text by automatically identifying the probable topic that generates a specific argument. This knowledge can further enable the disambiguation of that argument from another usage generated by a different topic.

LinkLDA (as opposed to LDA) couples the generation of two arguments by having a common distribution θ for generating the two topics of a relation. In effect, commonly occurring pairs of topics get assigned unique topic ids. In this way, the distribution of one argument affects the topics of the other argument, leading to reduced sparsity by an effective use of available data and better generalization. Finally, LinkLDA is a generative model and has all advantages of a generative model, *e.g.*, it outputs the complete probability distribution for the data, and hence, can easily be integrated as part of a larger system.

LinkLDA is also very efficient – it is linear in both the size of the corpus as well as the number of relations. There are several scalability enhancements such as SparseLDA (Yao et al., 2009) that can easily be adapted to LinkLDA. Moreover, once a topic distribution has been learnt over a set of training relations, one can efficiently parallelize inference to unseen relations (Yao et al., 2009).

3.3 Implementation

We use collapsed Gibbs sampling for inference in which each of the hidden variables ($z_{r,i,1}$ and $z_{r,i,2}$) are sampled sequentially conditioned on a full-assignment to all others, and integrating out parameters (Griffiths and Steyvers, 2004). This approach produces more robust parameter estimates than EM, as it allows us to compute expectations over the posterior distribution as opposed the maximum likelihood parameter values. In addition, the integration allows the use of sparse priors, which are typically more appropriate for natural language. In all experiments we use hyperparameters $\alpha = \eta_1 = \eta_2 = 0.1$. To perform inference, we generate a sampler for the LinkLDA model using the Hierarchical Bayes Compiler (Daume III, 2007).

4 Experiments

We perform two main experiments to assess the quality of the preferences obtained by LDA-SP. The first is a task-independent evaluation using a pseudo-disambiguation experiment (Section 4.2), which is a standard way to evaluate the quality of selectional preferences (Rooth et al., 1999; Erk, 2007; Bergsma et al., 2008). Secondly, we also show significant improvements to performance at an end-task (Textual Inference) in Section 4.3. Finally, we also report on the quality of a large database of Wordnet based preferences obtained after manually associating our topics with Wordnet classes (Section 4.4).

4.1 Data

For all experiments we make use of the corpus of $r(a_1, a_2)$ tuples, which was automatically extracted by TEXTRUNNER (Banko and Etzioni, 2008) from 500 million high quality (as ranked by Google) webpages. To convert this massive corpus into a manageable sized dataset we performed some data selection and preprocessing using Yahoo's PIG query language (Olston et al., 2008) running on HADOOP.

We selected 3,000 relations from the middle of the tail (we used the 2,000-5,000 most frequent ones[2]). We created a dataset by collecting all instances for this set of relations and removing any duplicate tuples. To reduce sparsity, we discarded any tuple containing an NP that occurred fewer than 50 times in the data. This resulted in a final set of about 2.4 million tuples in which each relation as well as noun phrase had a minimum level of support. We call this dataset the *generalization corpus*.

We infered topic-argument and predicate-topic multinomials (β, γ, and θ) on the generalization corpus by taking 5 samples at a lag of 50 after a burn in of 750 iterations. Table 1 lists some sample topics and high ranked words for each topic (for both arguments) as well as relations favoring those topics. We notice that topics learnt by the method are quite coherent and so are the relations for them.

4.2 Task Independent Evaluation

We compare our LinkLDA-based approach to the two state of the art similarity based systems (mutual information and Jaccard) described by Erk (2007), which are also known to outperform the generative model of Rooth (1999), as well as class-based methods such as Resnik's. In this pseudo-disambiguation experiment an original tuple is paired with a pseudo-negative, which has both arguments randomly generated from the whole vocabulary (according to the corpus-wide distribution over arguments). The task is, for each relation-argument pair, to determine whether it is observed, or a random distractor.

4.2.1 Test Set

For this experiment we gathered a primary corpus by first randomly selecting 100 high-frequency relations *not* in the generalization corpus. For these 100 we held out 500 randomly selected tuples as the test set. For each of the tuples in the held-out set, we removed all those from the training set containing either *rel-arg* pair, that is any tuple matching $r(a_1, *)$ or $r(*, a_2)$. Next we used Gibbs Sampling to infer a distribution over topics, θ_r, for each of the relations in the primary corpus based on the topics from the generalization corpus.

For each of the 500 observed tuples in the test-set we generated a pseudo-negative tuple by randomly sampling two noun phrases from the distribution of NPs in both corpora.

4.2.2 Prediction

Our prediction system needs to determine whether a specific relation-argument pair is admissible according to the selectional preferences or is a random distractor (D). We perform this experiment independently for the two relation-argument pairs (r, a_1) and (r, a_2) for a fair comparison to other systems, since they do not model the two arguments simultaneously.

We first compute the probability of observing a_1 for first argument of relation r given that it is not a distractor, $P(a_1|r, \neg D)$, which we approximate by it's probability given an estimate of the parameters inferred by our model, marginalizing over hidden topics t. Note that analysis for second argument is similar.

$$\begin{aligned} P(a_1|r,\neg D) \approx P_{LDA}(a_1|r) &= \sum_{t=0}^{T} P(a_1|t)P(t|r) \\ &= \sum_{t=0}^{T} \beta_t(a_1)\theta_r(t) \end{aligned}$$

A simple application of Bayes Rule gives the probability that a particular argument is not a distractor:

$$\begin{aligned} P(\neg D|r, a_1) &= \frac{P(\neg D|r)P(a_1|r,\neg D)}{P(a_1|r)} \\ &\approx \frac{P(\neg D)P_{LDA}(a_1|r)}{P(D)P(a_1|D) + P(\neg D)P_{LDA}(a_1|r)} \end{aligned}$$

The distractor-related probabilities such as $P(D|r)$ are independent of r. We estimate $P(a_1|D)$ according to the distribution of NPs in the corpus.

4.2.3 Results

Figure 2 plots the precision-recall curve for the pseudo-disambiguation experiment comparing LDA-SP with mutual information and Jaccard similarities using both the generalization and primary corpus for computation of similarities. We find LDA-SP significantly outperforms other methods, obtaining an 11% increase in the area under precision-recall curve over Mutual Information. All 3 system's AUC are shown in table 2; both differences are statistically significantly with a significance level less than 0.01 using a paired t-test.

In addition to a superior performance in selectional preference evaluation LinkLDA also produces a set of coherent topics, which can be useful in their own right. For instance, one could

[2] In our corpus the very highly frequent relations are often ill-formed due to systemic extraction erors.

Topic t	Arg1	Relations which assign highest probability to t	Arg2
18	The residue - The mixture - The reaction mixture - The solution - the mixture - the reaction mixture - the residue - The reaction - the solution - The filtrate - the reaction - The product - The crude product - The pellet - The organic layer - Thereto - This solution - The resulting solution - Next - The organic phase - The resulting mixture - C.)	was treated with, is treated with, was poured into, was extracted with, was purified by, was diluted with, was filtered through, is disolved in, is washed with	EtOAc - CH2Cl2 - H2O - CH.sub.2Cl.sub.2 - H.sub.2O - water - MeOH - NaHCO3 - Et2O - NHCl - CHCl.sub.3 - NHCl - dropwise - CH2Cl.sub.2 - Celite - Et.sub.2O - Cl.sub.2 - NaOH - AcOEt - CH2C12 - the mixture - saturated NaHCO3 - SiO2 - H2O - N hydrochloric acid - NHCl - preparative HPLC - to0 C
151	the Court - The Court - the Supreme Court - The Supreme Court - this Court - Court - The US Supreme Court - the court - This Court - the US Supreme Court - The court - Supreme Court - Judge - the Court of Appeals - A federal judge	will hear, ruled in, decides, upholds, struck down, overturned, sided with, affirms	the case - the appeal - arguments - a case - evidence - this case - the decision - the law - testimony - the State - an interview - an appeal - cases - the Court - that decision - Congress - a decision - the complaint - oral arguments - a law - the statute
211	President Bush - Bush - The President - Clinton - the President - President Clinton - President George W. Bush - Mr. Bush - The Governor - the Governor - Romney - McCain - The White House - President - Schwarzenegger - Obama	hailed, vetoed, promoted, will deliver, favors, denounced, defended	the bill - a bill - the decision - the war - the idea - the plan - the move - the legislation - legislation - the measure - the proposal - the deal - this bill - a measure - the program - the law - the resolution - efforts - the agreement - gay marriage - the report - abortion
224	Google - Software - the CPU - Clicking - Excel - the user - Firefox - System - The CPU - Internet Explorer - the ability - Program - users - Option - SQL Server - Code - the OS - the BIOS	will display, to store, to load, processes, cannot find, invokes, to search for, to delete	data - files - the data - the file - the URL - information - the files - images - a URL - the information - the IP address - the user - text - the code - a file - the page - IP addresses - PDF files - messages - pages - an IP address

Table 1: Example argument lists from the inferred topics. For each topic number t the k most probable values are listed according to the multinomial distributions for each argument (β_k and γ_k). In addition a few relations whose inferred topic distributions θ_r assign highest probability to t are listed.

	LDA-SP	MI-Sim	Jaccard-Sim
AUC	0.807	0.727	0.711

Table 2: Area under the precision recall curve. LDA-SP's AUC is significantly higher than both similarity-based methods according to a paired t-test with a significance level below 0.01.

use them for tasks like tasks such as set-expansion (Carlson et al., 2010) or automatic thesaurus induction (Etzioni et al., 2005; Kozareva et al., 2008).

4.3 End Task Evaluation

We now evaluate LDA-SP's ability to improve performance at an end-task. We choose the task of improving textual entailment by learning selectional preferences for inference rules and filtering inferences that do not respect these. For now we stick to inference rules of the form $r_1(a_1, a_2) \implies r_2(a_1, a_2)$, though our ideas are more generally applicable to more complex rules. This application of selectional preferences was introduced by (Pantel et al., 2007). As an example the rule (X *defeats* Y) ⇒ (X *plays* Y) holds when X and Y are both sports teams, however fails to produce a reasonable inference if X and Y are *Britain* and *Nazi Germany* respectively.

4.3.1 Filtering Inferences

In order for an inference to be plausible, both relations must have similar selectional preferences, and further the arguments must obey the selectional preferences of both the antecedent r_1 and the consequent r_2. Pantel et al. (2007) made use of these intuitions by producing a set of class-based selectional preferences for each relation, then filtering out any inferences where the arguments were incompatible with the intersection of these preferences. In contrast, we take a probabilistic approach, evaluating the quality of a specific inference by measuring the probability that the arguments in both the antecedent and the consequent were drawn from the same hidden topic in our model. Note that this probability captures both the requirement that the antecedent and consequent of the rule have similar selectional preferences, and that the arguments from a particular instance of the rule's application match their overlap.

We use $z_{r_i,j}$ to denote the topic that generates the j^{th} argument of relation r_i. Thus the probability that the two arguments a_1, a_2 were drawn

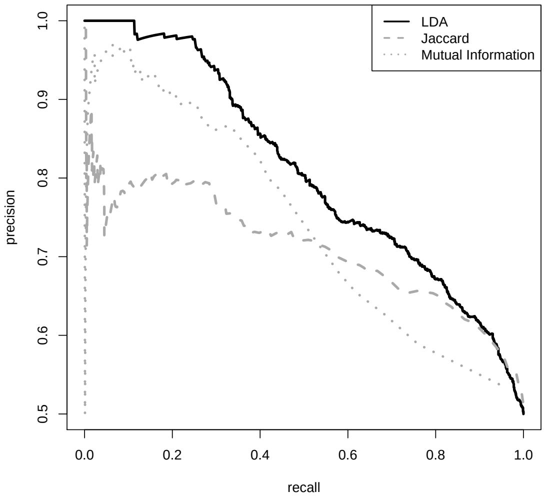

Figure 2: Precision and Recall on Pseudo-Disambiguation Experiment. LDA-SP performs significantly better than existing state of the art systems by obtaining a 11% increase in area under precision recall curve compared to Mutual Information.

from the same hidden topic factorizes as follows due to the conditional independences based on our model:[3]

$$P(z_{r_1,1} = z_{r_2,1}, z_{r_1,2} = z_{r_2,2}|a_1, a_2) = P(z_{r_1,1} = z_{r_2,1}|a_1)P(z_{r_1,2} = z_{r_2,2}|a_2)$$

To compute each of these factors we simply marginalize over the hidden topics:

$$P(z_{r_1,j} = z_{r_2,j}|a_j) = \sum_{t=1}^{T} P(z_{r_1,j} = t|a_j)P(z_{r_2,j} = t|a_j)$$

where a $P(z = t|a_j)$ can be computed using Bayes rule. For example,

$$\begin{aligned} P(z_{r_1,1} = t|a_1) &= \frac{P(a_1|z_{r_1,1} = t)P(z_{r_1,1} = t)}{P(a_1)} \\ &= \frac{\beta_{r_1}(a_1)\theta_{r_1}(t)}{P(a_1)} \end{aligned}$$

4.3.2 Experimental Conditions

In order to evaluate LDA-SP's ability to filter inferences based on selectional preferences we need a set of inference rules between the relations in our corpus. We therefore mapped the DIRT Inference rules (Lin and Pantel, 2001), (which consist of pairs of dependency paths) to TEXTRUNNER relations as follows. We first gathered all instances in the generalization corpus, and for each $r(a_1, a_2)$ tuple created a corresponding simple sentence by concatenating the arguments with the relation string between them. Each such simple sentence was parsed using Minipar. We then extracted all dependency paths between nouns that contain only words present in the TEXTRUNNER relation string. These dependency paths were then matched against each pair in the DIRT database, and all pairs of associated relations were collected producing about 26,000 inference rules.

Following Pantel (2007) we randomly sampled 100 inference rules. We then automatically filtered out any rules which contained a negation, or for which the antecedent and consequent contained a pair of antonyms found in WordNet (this left us with 85 rules). For each rule we collected 10 random instances of the antecedent, and generated the consequent. We randomly sampled 300 of these inferences to hand-label.

4.3.3 Results

In figure 3 we compare the precision and recall of LDA-SP against the top two performing systems described by Pantel et al. (ISP.IIM-$\vee$ and ISP.JIM, both using the CBC clusters (Pantel, 2003)). We find that LDA-SP achieves both higher precision and recall than ISP.IIM-$\vee$. It is also able to achieve the high-precision point of ISP.JIM and can trade precision to get much more recall.

In addition we demonstrate LDA-SP's ability to rank inference rules by measuring the KL Divergence between the topic-distributions of the antecedent and consequent, θ_{r_1} and θ_{r_2} respectively. Table 3 shows the top 10 and bottom 10 rules out of the 26,000 ranked by KL Divergence after automatically filtering antonyms (using WordNet) and negations. For slight variations in rules (*e.g.*, symmetric pairs) we mention only one example to show more variety.

4.4 Building a Repository of Human-Interpretable Preferences

Finally we explore LDA-SP's ability to produce a human-interpratable repository of selectional preferences. As an example for the relation *was born in* we would like to infer that plausable pairs of arguments include (*person*, *location*) and (*person*,

[3] Note that all probabilities are conditioned on an estimate of the parameters θ, β, γ from our model, which are omitted for compactness.

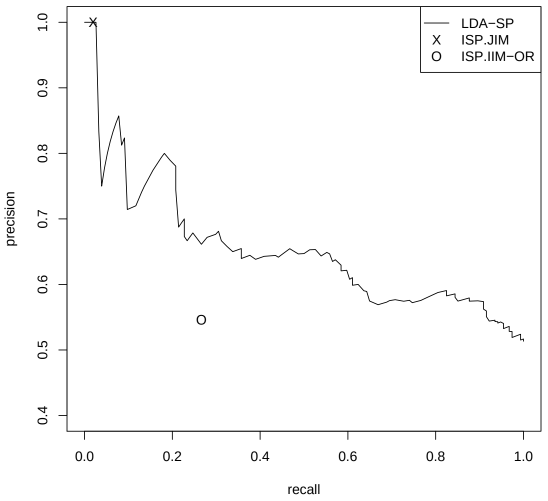

Figure 3: Precision and Recall on Inference Filtering Task.

Top 10 Inference Rules Ranked by LDA-SP		
antecedent	consequent	KL-div
will begin at	will start at	0.014999
shall review	shall determine	0.129434
may increase	may reduce	0.214841
walk from	walk to	0.219471
consume	absorb	0.240730
shall keep	shall maintain	0.264299
shall pay to	will notify	0.290555
may apply for	may obtain	0.313916
copy	download	0.316502
should pay	must pay	0.371544
Bottom 10 Inference Rules Ranked by LDA-SP		
antecedent	consequent	KL-div
lose to	shall take	10.011848
should play	could do	10.028904
could play	get in	10.048857
will start at	move to	10.060994
shall keep	will spend	10.105493
should play	get in	10.131299
shall pay to	leave for	10.131364
shall keep	return to	10.149797
shall keep	could do	10.178032
shall maintain	have spent	10.221618

Table 3: Top 10 and Bottom 10 ranked inference rules ranked by LDA-SPafter automatically filtering out negations and antonyms (using WordNet).

date).

For this task we need to map the inferred topics to a set of classes, such as those defined by the WordNet hypernym hierarchy. We investigated using Resnik's (1996) class-based approach to computing selectional preferences using the multinomial topic-argument distributions (as opposed to distributions condtioned on the relation), however this suffers from the same problems as directly applying class-based approaches (poor lexical coverage, and ambiguity issues). In addition some topics are incoherent due to noise in our data, and relations which have no strong selectional preference. Recent work in automatic labeling and coherence evaluation of inferred topics (Mei et al., 2007; Newman et al., 2010) may improve on these results, however as we have a relatively small number of topics (600 total - 300 for each argument) we simply chose to label them manually. Note that although some labeling effort is needed *per topic*, there are many fewer topics than relations. By associating a label with each topic we can infer labels over arbitrary predicates.

The manual mapping from topics to WordNet proceeded as as follows. We first applied Resnik's approach, using the YAGO ontology(Suchanek et al., 2008). For each topic we produced a list of candidate WordNet classes ranked by Selectional Association, from which we manually picked the best class which (intuitively) described the topic (from inspecting the 20 arguments assigned highest probability). If the topic seemed incoherent or didn't fit anywhere in the WordNet noun hypernym taxonomy, we simply labeled it with a special symbol $\emptyset$.

To evaluate how well our manual labeling of topics carries over to producing selectional restrictions for unseen relations we used the same random sample of 100 relations used in the pseudodisambiguation experiment[4]. For each relation we counted the frequency of each arg1-arg2 topic pair (in the 5 gibbs samples), and picked the 5 most frequent pairs. We then removed any pairs for which one (or both) topics were labeled $\emptyset$. This resulted in a set of 189 class pairs. We labeled each according to human intuition, and found that 146 or 76% seemed reasonable. A few examples are displayed in table 4.

We feel that these preliminary results seem

[4] Note that these 100 were not part of the original 3,000 used to infer the topics, and are therefore representative of new "unseen" relations.

arg1 class	relation	arg2 class
politician#1	was running for	leader#1
people#1	will love	show#3
organization#1	has responded to	accusation#2
administrative_unit#1	has appointed	administrator#3

Table 4: Human-Interpretable Selectional Preferences.

promising, and we hope to release a large repository of human-interpretable selectional preferences in the near future.

5 Conclusions

We have presented an application of Topic Modeling to the problem of Selectional Preferences. A Topic Modeling approach is a natural fit to this problem, as it simultaneously infer groups of related words (topics) and distributions over these topics. This approach is capable of producing human interpretable classes, however avoids the drawbacks of class-based approaches (poor lexical coverage and ambiguity). LDA-SPalso achieves state-of-the art performance on predictive tasks such as pseudodisambiguation, and filtering incorrect inferences.

References

Michele Banko and Oren Etzioni. 2008. The tradeoffs between open and traditional relation extraction. In *Proceedings of ACL-08: HLT*, pages 28–36, Columbus, Ohio, June. Association for Computational Linguistics.

Shane Bergsma, Dekang Lin, and Randy Goebel. 2008. Discriminative learning of selectional preference from unlabeled text. In *EMNLP '08: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 59–68, Morristown, NJ, USA. Association for Computational Linguistics.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Andrew Carlson, Justin Betteridge, Richard C. Wang, Estevam R. Hruschka Jr., and Tom M. Mitchell. 2010. Coupled semi-supervised learning for information extraction. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM 2010)*.

Harr Chen, S. R. K. Branavan, Regina Barzilay, and David R. Karger. 2009. Global models of document structure using latent permutations. In *NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 371–379, Morristown, NJ, USA. Association for Computational Linguistics.

Stephen Clark and David Weir. 2002. Class-based probability estimation using a semantic hierarchy. *Comput. Linguist.*, 28(2):187–206.

Ido Dagan, Lillian Lee, and Fernando C. N. Pereira. 1999. Similarity-based models of word cooccurrence probabilities. In *Machine Learning*, pages 34–1.

Hal Daumé III and Daniel Marcu. 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 305–312, Sydney, Australia, July. Association for Computational Linguistics.

Hal Daume III. 2007. hbc: Hierarchical bayes compiler..

Katrin Erk. 2007. A simple, similarity-based model for selectional preferences. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 216–223, Prague, Czech Republic, June. Association for Computational Linguistics.

Elena Erosheva, Stephen Fienberg, and John Lafferty. 2004. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5220–5227, April.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana maria Popescu, Tal Shaked, Stephen Soderl, Daniel S. Weld, and Er Yates. 2005. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165:91–134.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Comput. Linguist.*, 28(3):245–288, September.

T. L. Griffiths and M. Steyvers. 2004. Finding scientific topics. *Proc Natl Acad Sci U S A*, 101 Suppl 1:5228–5235, April.

Zornitsa Kozareva, Ellen Riloff, and Eduard Hovy. 2008. Semantic class learning from the web with hyponym pattern linkage graphs. In *Proceedings of ACL-08: HLT*, pages 1048–1056, Columbus, Ohio, June. Association for Computational Linguistics.

Hang Li and Naoki Abe. 1998. Generalizing case frames using a thesaurus and the mdl principle. *Computational Linguistics*, 24:239–248.

Dekang Lin and Patrick Pantel. 2001. Dirt-discovery of inference rules from text. In *Proceedings of ACM Conference on Knowledge Discovery and Data Mining(KDD-01)*, pages pp. 323–328.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics*, pages 768–774, Morristown, NJ, USA. Association for Computational Linguistics.

Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. 2007. Automatic labeling of multinomial topic models. In *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 490–499, New York, NY, USA. ACM.

David Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. 2009. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 880–889, Singapore, August. Association for Computational Linguistics.

David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *NAACL-HLT*.

Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins. 2008. Pig latin: a not-so-foreign language for data processing. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1099–1110, New York, NY, USA. ACM.

Patrick Pantel, Rahul Bhagat, Bonaventura Coppola, Timothy Chklovski, and Eduard H. Hovy. 2007. Isp: Learning inferential selectional preferences. In *HLT-NAACL*, pages 564–571.

Patrick Andre Pantel. 2003. *Clustering by committee*. Ph.D. thesis, Edmonton, Alta., Canada.

Joseph Reisinger and Marius Pasca. 2009. Latent variable models of concept-attribute attachment. 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*, pages 620–628, Suntec, Singapore, August. Association for Computational Linguistics.

P. Resnik. 1996. Selectional constraints: an information-theoretic model and its computational realization. *Cognition*, 61(1-2):127–159, November.

Philip Resnik. 1997. Selectional preference and sense disambiguation. In *Proc. of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, Washington, DC.

Mats Rooth, Stefan Riezler, Detlef Prescher, Glenn Carroll, and Franz Beil. 1999. Inducing a semantically annotated lexicon via em-based clustering. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 104–111, Morristown, NJ, USA. Association for Computational Linguistics.

Fabian Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. YAGO - a large ontology from Wikipedia and WordNet. *Elsevier Journal of Web Semantics*, 6(3):203–217, September.

Tae Yano, William W. Cohen, and Noah A. Smith. 2009. Predicting response to political blog posts with topic models. In *NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 477–485, Morristown, NJ, USA. Association for Computational Linguistics.

L. Yao, D. Mimno, and A. Mccallum. 2009. Efficient methods for topic model inference on streaming document collections. In *The 15th ACM SIGKDD Conference On Knowledge Discovery and Data Mining (KDD 2009)*.