Leveraging a Semantically Annotated Corpus to Disambiguate Prepositional Phrase Attachment

Guy Emerson and Ann Copestake
Computer Laboratory, University of Cambridge
15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom
{gete2, aac10}@cam.ac.uk

Abstract

Accurate parse ranking requires semantic information, since a sentence may have many candidate parses involving common syntactic constructions. In this paper, we propose a probabilistic framework for incorporating distributional semantic information into a maximum entropy parser. Furthermore, to better deal with sparse data, we use a modified version of Latent Dirichlet Allocation to smooth the probability estimates. This LDA model generates pairs of lemmas, representing the two arguments of a semantic relation, and can be trained, in an unsupervised manner, on a corpus annotated with semantic dependencies. To evaluate our framework in isolation from the rest of a parser, we consider the special case of prepositional phrase attachment ambiguity. The results show that our semantically-motivated feature is effective in this case, and moreover, the LDA smoothing both produces semantically interpretable topics, and also improves performance over raw co-occurrence frequencies, demonstrating that it can successfully generalise patterns in the training data.

1 Introduction

Ambiguity is a ubiquitous feature of natural language, and presents a serious challenge for parsing. For people, however, it does not present a problem in most situations, because only one interpretation will be sensible. In examples (1) and (2), fluent speakers will not consciously consider a gun-wielding dog or a moustache used as a biting tool. Both of these examples demonstrate syntactic ambiguity (the final prepositional phrase (PP) could modify the preceding noun, or the main verb), rather than lexical ambiguity (homophony or polysemy).

(1) The sheriff shot a dog with a rifle.

(2) The dog bit a sheriff with a moustache.

In many cases, parse ranking can be achieved by comparing syntactic structures, since some constructions are more common. In the above examples, however, the same set of structures are available, but the best parse differs: the PP should modify the verb "shot" in (1), but the noun "sheriff" in (2). Dealing with such cases requires semantic information.

A promising approach to represent lexical semantics assumes the distributional hypothesis, which was succinctly stated by Turney and Pantel (2010): "words that occur in similar contexts tend to have similar meanings". Our method uses corpus data to estimate the plausibility of semantic relations, which could then be exploited as features in a maximum entropy parser. In section 3, we first describe the general framework, then explain how it can be specialised to tackle PP-attachment.

To overcome data sparsity, we introduce a generative model based on Ó Séaghdha (2010)'s modified version of Latent Dirichlet Allocation (LDA), where two lemmas are generated at a time, which we use to represent the two arguments of a binary semantic relation. The probabilities produced by the LDA model can then be incorporated into a discriminative parse selection model, using our general framework.

Proceedings of the 11th International Conference on Computational Semantics, pages 1–11,
London, UK, April 15-17 2015. ©2015 Association for Computational Linguistics

This LDA model can be trained unsupervised using a semantically annotated corpus. To clarify what this means, it is helpful to distinguish two notions of "labelled data": linguistic annotations, and desired outputs. Following Ghahramani (2004), supervised learning requires both a set of inputs and a set of desired outputs, while unsupervised learning requires only inputs. Although we use a corpus with linguistic annotations, these are not desired outputs, and learning is unsupervised in this sense. Since our training data was automatically produced using a parser (as explained in section 4.2), our method can also be seen as self-training, where a statistical parser can be improved using unlabelled corpus data.

Because of its central role in linguistic processing, parse ranking has been extensively studied, and we review other efforts to incorporate semantic information in section 2. To evaluate our framework, we consider the special case of PP-attachment ambiguity, comparing the model's predictions with hand-annotated data, as explained in section 4. Results are presented in section 5, which we discuss in section 6. Finally, we give suggestions for future work in section 7, and conclude in section 8.

2 Related Work

The mathematical framework described in section 3.3 follows the "Rooth-LDA" model described by Ó Séaghdha (2010). However, he uses it to model verbs' selectional preferences, not for parse ranking. The main difference in this work is to train multiple such models and compare their probabilities.

The use of lexical information in parse ranking has been explored for some time. Collins (1996) used bilexical dependencies derived from parse trees, estimating the probabiliity of a relation given a sentence. We consider instead the plausibility of relations, which can be included in a more general ranking model.

Rei and Briscoe (2013) consider re-ranking the output of a parser which includes bilexical grammatical relations. They use co-occurrence frequencies to produce confidence scores for each relation, and combine these to produce a score for the entire parse. To smooth the scores, they use a semantic vector space model to find similar lexical items, and average the scores for all such items. From this point of view, our LDA model is an alternative smoothing method. Additionally, both our approach and theirs can be seen as examples of self-training. However, their re-ranking approach must be applied on the output of a parser, while we explain how such scores can be directly integrated as features in parse ranking.

Hindle and Rooth (1993) motived the use of lexical information for disambiguating PP-attachment. More recently, Zhao and Lin (2004) gave a state-of-the-art supervised algorithm for this problem. Given a new construction, they use a semantic vector space to find the most similar examples in the training data, and the most common attachment site among these is then assigned to the new example.

Unlike Zhao and Lin, and many other authors tackling this problem using the Penn Treebank, our model is unsupervised and generative. The first fact makes more data available for training, since we can learn from unambiguous cases, and the second plays an important role in building a framework that can handle arbitrary types of ambiguity. This provides a significant advantage over many discriminative approaches to PP-attachment: despite Zhao and Lin's impressive results, it is unclear how their method could be extended to cope with arbitrary ambiguity in a full sentence.

Clark et al. (2009) use lexical similarity measures in resolving coordination ambiguities. They propose two similarity systems, one based on WordNet, and the other on distributional information extracted from Wikipedia using the C&C parser. Hogan (2007) also consider similarity, both of the head words and also in terms of syntactic structure. However, while similarity might be appropriate for handing coordination, since conjuncts are likely to be semantically similar, this does not generalise well to other relations, where the lexical items involved may be semantically related, but not similar.

Bergsma et al. (2011) approach coordination ambiguity using annotated text, aligned bilingual text, and plain monolingual text, building statistics of lexical association. However, this method works at the string level, without semantic annotations, and there is no clear generalisation to other semantic relations.

Agirre et al. (2008) use lexical semantics in parsing, both in general and considering PP-attachment in particular. They replace tokens with more general WordNet synsets, which reduces data sparsity for standard lexicalised parsing techniques. Our LDA approach essentially provides an alternative method to back-off to semantic classes, without having to deal with the problem of word sense disambiguation.

3 Generative Model

3.1 Modelling an Arbitrary Relation

Despite the vast variety of syntactic frameworks, many parsers will produce semantic or syntactic relations in some form. We might therefore rephrase parse ranking as follows: given a set of candidate parses, choose the one with the most plausible relations.

Given a binary relation $x \xrightarrow{r} y$ between lexical items x and y, we can consider the joint probability distribution $P(r, x, y)$, which is the chance that, if we are given a random instance of any binary relation, we observe it to be the relation r between items x and y. However, rare lexical items will have low probabilities, even if they are a close semantic fit, so we should normalise by the words' overall probability of occurrence, $P(x)$ and $P(y)$, as shown in (3). The denominator can be interpreted as co-occurrence of x and y under the null hypothesis that they are generated independently, according to their overall frequency. We do not normalise by $P(r)$, so that the frequency of the relation is still taken into account, which is important, as we will see in section 3.2.

$$score(r, x, y) = \frac{P(r, x, y)}{P(x) P(y)} \tag{3}$$

A Maximum Entropy parser (MaxEnt; Berger et al., 1996) relies on a set of features $f_1, \ldots, f_m$ with corresponding weights $\lambda_1, \ldots, \lambda_m$. The probability of a parse t for a sentence s is given in (4), where Z is a normalisation constant which can often be neglected. The values of the weights λ_i are chosen to maximise the likelihood of training data, sometimes including a Gaussian prior for regularisation.

$$P(t|s) = \frac{1}{Z} \exp \sum_{i=1}^{m} \lambda_i f_i(t) \tag{4}$$

To incorporate the above scores into a MaxEnt parser, we could define a feature which sums the scores of all relations in a parse. However, the scores in (3) are always positive, so this would bias us towards parses with many relations. Instead, we can take the logarithm of the score, so that plausible relations are rewarded, and implausible ones penalised.[1] For a parse t containing k relations $x_i \xrightarrow{r_i} y_i$, we define f to be the sum of the log-scores, as shown in (5). Given a grammar and decoder that can generate candidate parses, this feature allows us to exploit semantic information in parse ranking.

$$f(t) = \sum_{i=1}^{k} \log\left(score(r_i, x_i, y_i)\right) \tag{5}$$

3.2 Application to PP-attachment

The effect of such a model on a wide-coverage parser will be complicated by interactions with other components. To evaluate it independently, we restrict attention to PP-attachment in four-lemma sequences $w = (v, n_1, p, n_2)$, of the form (*verb, noun, preposition, noun*), where (p, n_2) forms a PP which could attach to either the verb v, or the verb's direct object n_1. Surrounding context is not considered. For example, we could have the sequence (*eat, pasta, with, fork*).

We consider two relations, both mediated by the preposition p: for nominal attachment, a relation $r_{p,N}$ between n_1 and n_2; and for verbal attachment, a relation $r_{p,V}$ between v and n_2.

Given a sequence w, we seek the probability of attachment to n_1 or v, which we denote as $P(N|w)$ and $P(V|w)$, respectively. Taking their ratio and applying Bayes rule yields (6). To use the scores defined in (3), we first make two independence assumptions: if the PP is attached to n_1, then v is independent, and if the PP is attached to v, then n_1 is independent. We then make the approximation that the probabilities $P(N|p)$ and $P(V|p)$ for this particular ambiguity are proportional to the probabilities of observing $r_{p,N}$ and $r_{p,V}$ in general.[2] This precisely gives us a ratio of plausibility scores, shown in (9).

[1] The expected value of the log-score is equal to the mutual information of x and y, minus the conditional entropy of r given x and y. A smaller bias would therefore remain, depending on which of these two quantities is larger.

[2] Technically, as we move from (7) to (8), we shift from considering a probability space over four-lemma sequences to a probability space over binary relations. We abuse notation in using the same P to denote probabilities in both spaces.

$$\frac{P(N|w)}{P(V|w)} = \frac{P(N|p)\,P(v,n_1,n_2|p,N)}{P(V|p)\,P(v,n_1,n_2|p,V)} \tag{6}$$

$$\approx \frac{P(N|p)\,P(n_1,n_2|p,N)\,P(v)}{P(V|p)\,P(v,n_2|p,V)\,P(n_1)} \tag{7}$$

$$\approx \frac{P(r_{p,N})\,P(n_1,n_2|r_{p,N})\,P(v)\,P(n_2)}{P(r_{p,V})\,P(v,n_2|r_{p,V})\,P(n_1)\,P(n_2)} \tag{8}$$

$$= \frac{score(r_{p,N},n_1,n_2)}{score(r_{p,V},v,n_2)} \tag{9}$$

In the context of a MaxEnt parser, suppose we have defined f, as in (5), with weight λ. For parses t_N and t_V representing nominal and verbal attachment, whose features are identical except for f, the ratio in their probabilities is shown in (10). This depends precisely on the ratio of plausbility scores, hence using f is equivalent to making the above independence assumptions and approximations.

$$\frac{P(t_N)}{P(t_V)} = \left(\frac{score(r_{p,N},n_1,n_2)}{score(r_{p,V},v,n_2)}\right)^{\lambda} \tag{10}$$

In the following section, we describe a generative model to produce better estimates of the probabilities $P(n_1,n_2|r_{p,N})$ and $P(v,n_2|r_{p,V})$. Note that a discriminative model would have to consider all three lemmas v, n_1, and n_2, which would both reduce the amount of training data (since unambiguous cases only using two lemmas must be discarded), and increase the number of model parameters (since we must account for three lemmas, not two). These two facts combined could strongly encourage overfitting.

3.3 Latent Dirichlet Allocation

In its original formulation, Latent Dirichlet Allocation (LDA; Blei et al., 2003) models the topics present in a collection of documents. Ó Séaghdha (2010) adapted this framework to model verb-object collocations. Instead of considering a document and the words it contains, we consider a relation (such as the verb-object relation) and all instances of that relation in some corpus (verbs paired with their objects). The aim is to overcome data sparsity, generalising from specific corpus examples to unseen collocations. This is achieved using latent variables, or "topics".

Intuitively, each topic should correspond to two sets of lemmas, whose members have a strong semantic connection via the given relation. For example, the sets {*run, walk, stroll, gallop*} and {*road, street, path, boulevard*} are semantically related via a preposition like *down*. A rare combination such as *gallop* and *boulevard* might not be observed in training, but should still be considered plausible.

Although LDA was first introduced as a clustering algorithm, we are interested in the probability of generation, and the topic assignments themselves can be discarded.

3.3.1 Formal Description

A pair (v,n) is generated from a relation r in two stages. First, we generate a topic z from the relation, and then independently generate v and n from the topic. To do this, we associate with each relation a distribution $\theta^{(r)}$ over topics, and with each topic a pair of distributions $\varphi^{(z)}$ and $\psi^{(z)}$ over words. Symbolically, we can write this as in (12), where Cat denotes a categorical[3] distribution, i.e. one where each probability is defined separately.

To prevent overfitting, we define Bayesian priors, to specify the kinds of distribution for θ, φ and ψ that we should expect. The most natural choice is a Dirichlet distribution, as it is the conjugate prior of a categorical distribution, which simplifies calculations. We have three priors, as shown in (11), with hyperparameters α, β and γ. The entire generative process is shown using plate notation in figure 1, where R relations are generated, each with M instances, using T topics.

[3]Sometimes known as *multinomial* or *unigram*.

$$\theta \sim \text{Dir}(\alpha), \quad \varphi \sim \text{Dir}(\beta), \quad \psi \sim \text{Dir}(\gamma) \tag{11}$$

$$z \sim \text{Cat}\left(\theta^{(r)}\right), \quad v \sim \text{Cat}\left(\varphi^{(z)}\right), \quad n \sim \text{Cat}\left(\psi^{(z)}\right) \tag{12}$$

We apply this framework to PP-attachment by replacing the pair (v, n) with either: (n_1, n_2) for nominal attachment, or (v, n_2) for verbal attachment. Each preposition is therefore associated with two LDA models, which yield probabilities $P(v, n_2|r_{p,V})$ and $P(n_1, n_2|r_{p,N})$ for use in equation (8).

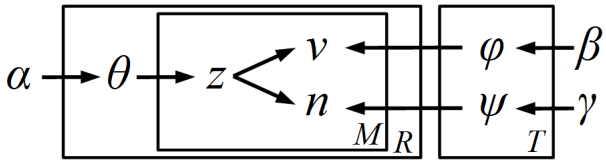

Figure 1: The modified LDA model

3.3.2 Inference

Defining the LDA model requires fixing four hyperparameters: the number of latent topics T, and the three Dirichlet priors α, β, and γ. Given these, and some training data, we can infer latent topic assignments z, and categorical distributions θ, φ, and ψ. However, for new instances, the distributions are semantically informative, while the topic assignments are not. Hence, we would like to sum over all topic assignments to obtain the marginal posterior distributions for θ, φ, and ψ. Calculating this is intractable, but we can approximate it using Gibbs sampling, applied to LDA by Griffiths and Steyvers (2004). This assigns a topic to each token (each pair of lemmas), and iteratively changes one topic assignment, conditioning on all others. Given a sample set of topic assignments, we can estimate the distributions θ, φ, and ψ, as shown in (13). Finally, we estimate the marginal probability of generating a pair (x, y), as shown in (14). The formulae also make clear the effect of the Dirichlet priors - compared to a maximum likelihood estimate, they smooth the probabilities by adding virtual samples to each $f_{..}$ term.

$$\hat{\theta}_z = \frac{f_{zr} + \alpha_z}{f_{\cdot r} + \sum_{z'} \alpha_{z'}} \quad , \quad \hat{\varphi}_x^{(z)} = \frac{f_{zx} + \beta}{f_{z\cdot} + V\beta} \quad , \quad \hat{\psi}_y^{(z)} = \frac{f_{zy} + \gamma}{f_{z\cdot} + V\gamma} \tag{13}$$

$$\hat{P}(x, y) = \sum_z \hat{\theta}_z \hat{\varphi}_x^{(z)} \hat{\psi}_y^{(z)} \tag{14}$$

A single Gibbs sample will not be representative of the overall distribution, so we must average the probabilities from several samples. However, the topics themselves are labelled arbitrarily, so we cannot average the statistics $\hat{\theta}$, $\hat{\varphi}$, and $\hat{\psi}$. Nonetheless, the statistic $\hat{P}$ is invariant under re-ordering of topics and can therefore be meaningfully averaged. This gives us a better approximation of the true value, and the standard deviation provides an error estimate, which we explore in section 5.3.

3.3.3 Model Selection

Training requires fixing the hyperparameters T, α, β and γ in advance. Griffiths and Steyvers (2004) recommend setting parameters to maximise the training data's log-likelihood L. However, this could result in overfitting, if more parameters are used than necessary; intuitively, some topics may end up matching random noise. One alternative is the Akaike Information Criterion (AIC; Akaike, 1974), which penalises the dimensionality k of the parameter space, and is defined as $2k - 2\log(L)$. Bruce and Wiebe (1999) demonstrate that such a criterion in natural language processing can avoid overfitting.

We have $T - 1$ independent parameters from θ, and $T(V - 1)$ from each of φ and ψ, where V is the vocabulary size.[4] Neglecting lower order terms, this gives $k = 2TV$. However, rare lemmas appear in few topics, giving sparse frequency counts, so k is effectively much lower. We are not aware of a method to deal with such sparse values. However, a simple work-around is to pretend V is smaller, for example $V = 1000$, effectively ignoring parameters for rare lexical items.

[4]Reordering topics only represents a finite number $T!$ of symmetries, and therefore does not reduce the dimensionality.

4 Experimental Setup

4.1 Choice of Prepositions

We trained models for the following prepositions: *as, at, by, for, from, in, on, to, with*. They were chosen for their high frequency of both attachment sites. Rare prepositions (such as *betwixt*) were discarded because of limited data. Prepositions with a strong preference of attachment site (such as *of*) were discarded because choosing the more common site already provides high performance.

	Instances	Proportion N
as	1,119,000	20.3 %
at	1,238,000	37.4 %
by	612,000	29.3 %
for	2,236,000	55.7 %
from	1,056,000	43.6 %

	Instances	Proportion N
in	5,288,000	37.6 %
on	1,628,000	49.7 %
to	1,411,000	46.1 %
with	1,638,000	37.4 %

Table 1: Number of training instances, with proportion of nominal attachment

4.2 Training Data

We trained the model using the WikiWoods corpus (Flickinger et al., 2010), which is both large, and also has rich syntactic and semantic annotations. It was produced from the full English Wikipedia using the PET parser (Callmeier, 2000; Toutanova et al., 2005) trained on the gold-standard subcorpus WeScience (Ytrestøl et al, 2009), and using the English Resource Grammar (ERG; Flickinger, 2000). Of particular note is that the ERG incorporates Minimal Recursion Semantics (MRS; Copestake et al., 2005), which can be expressed using dependency graphs (Copestake, 2009).

The relations mentioned in section 3.2 are not explicit in the ERG, since prepositions are represented as nodes, with edges to mark their arguments. To produce a set of training data, we searched for all preposition nodes[5] in the corpus, which either had both arguments ARG1 and ARG2 saturated, or, if no ARG1 was present, was the ARG1 of another node. We split the data based on nominal or verbal attachment, discarding PPs attached to other parts of speech. Each training instance was then a tuple of the form (v, p, n) or (n_1, p, n_2), for verbal or nominal attachment, respectively. We used lemmas rather than wordforms, to reduce data sparsity. The WeScience subcorpus was withheld from training, since it was used for evaluation (see section 4.3). In total, 16m instances were used, with a breakdown in table 1.

4.3 Evaluation Data

Two datasets were used in evaluation. We produced the first from WeScience, the manually treebanked portion of the Wikipedia data used to produce WikiWoods. This dataset allows evaluation in the same domain and with the same annotation conventions as the training data. We extracted all potentially ambiguous PPs from the DMRS structures: for PPs attached to a noun, the noun must be the object of a verb, and for PPs attached to a verb, the verb must have an object. Duplicates were removed, since this would unfairly weight those examples: some repeated cases, such as (*store metadata in format*), are limited in their domain. If the same tuple occurred with different attachment sites, the most common site was used, which happened twice, or if neither was more common, it was discarded, which happened four times. This produced 3485 unique sequences, of which 2157 contained one of the nine prepositions under consideration. The data is available on `https://github.com/guyemerson/WeSciencePP`.

The second data set was extracted from the Penn Treebank by Ratnaparkhi et al. (1994). This dataset has been widely used, allowing a comparison with other approaches. We extracted tuples with one of

[5]The ERG includes some prepositions in the "sense" field of a verb, rather than as a separate node. This is done for semantically opaque constructions, such as *rely on a friend*, where the meaning cannot be described in terms of *rely* and *on a friend*. We may wish to ignore such cases for two reasons: firstly, the preposition often appears either immediately following the verb or sentence-finally, which makes ambiguous sentences less common; secondly, the semantics is often idiosyncratic and hence less amenable to generalisations across lemmas. We discuss these cases further in section 6.1.

the relevant prepositions, lemmatised all words, and removed out of vocabulary items. This gave 1240 instances from the evaluation section of the corpus. We note that the data is noisy: it contains 'nouns' such as *the* (98 times), *all* (10 times), and *'s* (10 times), which are impossible under the annotation conventions of WikiWoods. We discuss limitations of evaluating against this dataset in section 6.1.

4.4 Baselines

We give results compared to two baselines. The low baseline chooses the most common attachment site for each preposition, as seen in the training data, regardless of the other lexical items. The high baseline is the maximum likelihood estimate, using Laplace smoothing with parameter 0.01. Comparing to the low baseline shows the effect of our framework using the feature defined in (5), while comparing to the high baseline shows the effect of the LDA smoothing. Additionally, we can consider an LDA model with a single topic, which is equivalent to the simpler smoothing method of backing off to bigram frequencies.

5 Results

5.1 Model Selection

We varied T to find the effect on the log-likelihood and the AIC (taking $V = 1000$), either fixing $\alpha = 50/T$, and $\beta = \gamma = 0.01$, which follows the recommendations of Steyvers and Griffiths (2007), or using hyperparameter optimisation, which allows asymmetric α. The results are shown in figure 2. For unoptimised models, using the log-likelihood suggests $T \approx 70$, and the AIC suggests $T \approx 35$. For the optimised model, the AIC suggests $T \approx 40$; however, the log-likelihood has not yet found its maximum, suggesting a much larger value, exactly what AIC is designed to avoid.

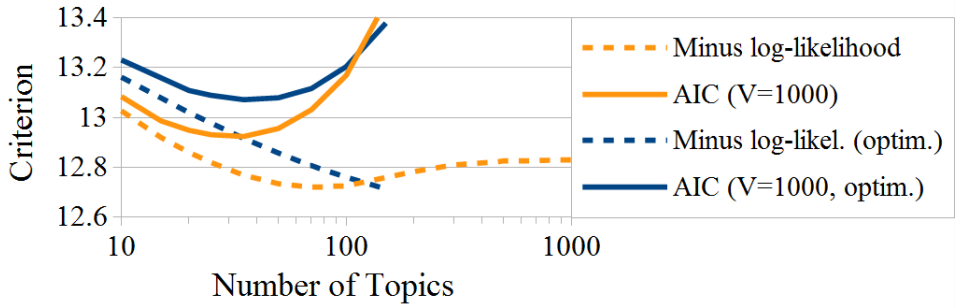

Figure 2: Model selection for LDA

5.2 Evaluation

Overall accuracy in choosing the correct attachment site is given in table 2. The large gap between the high and low baselines shows the importance of lexical information. The high baseline and the 1-topic model (i.e. backing off to bigrams) show similar performance. The best performing LDA models achieve 3 and 7 percentage point increases for WeScience and the Penn Treebank, demonstrating the effectiveness of this smoothing method. The higher gain for the Penn Treebank suggests that smoothing is more important when evaluating across domains.

The choices of hyperparameters suggested by the log-likelihood and AIC closely agree with the best performing model. The results also suggest that the LDA smoothing is robust to choosing too high a value for T. As we can see in table 2, there is only a small drop in performance with larger values of T. This result agrees with Wallach et al. (2009), who show that LDA, as applied to topic modelling, is reasonably robust to large choices of T, and that it is generally better to set T too high than too low.

Surprisingly, hyperparameter optimisation (allowing α to be asymmetric) did not provide a significant change in performance, even though we might expect some topics to be more common.

			Accuracy	
T	Samp.	Optim.?	WeSci	PTB
1	-	-	0.708	0.659
35	10	no	0.744	**0.701**
50	10	no	0.745	0.697
50	30	no	**0.747**	0.698
50	10	yes	0.741	0.695
70	10	no	0.736	0.694
70	30	no	0.738	0.696
70	10	yes	0.741	0.700
100	10	no	0.735	0.700
300	10	no	0.738	0.680
High baseline			0.718	0.629
Low baseline			0.609	0.571

Table 2: Performance of our model, varying number of topics T, number of Gibbs samples, and hyper-parameter optimisation. The highest scores for each dataset are shown in bold.

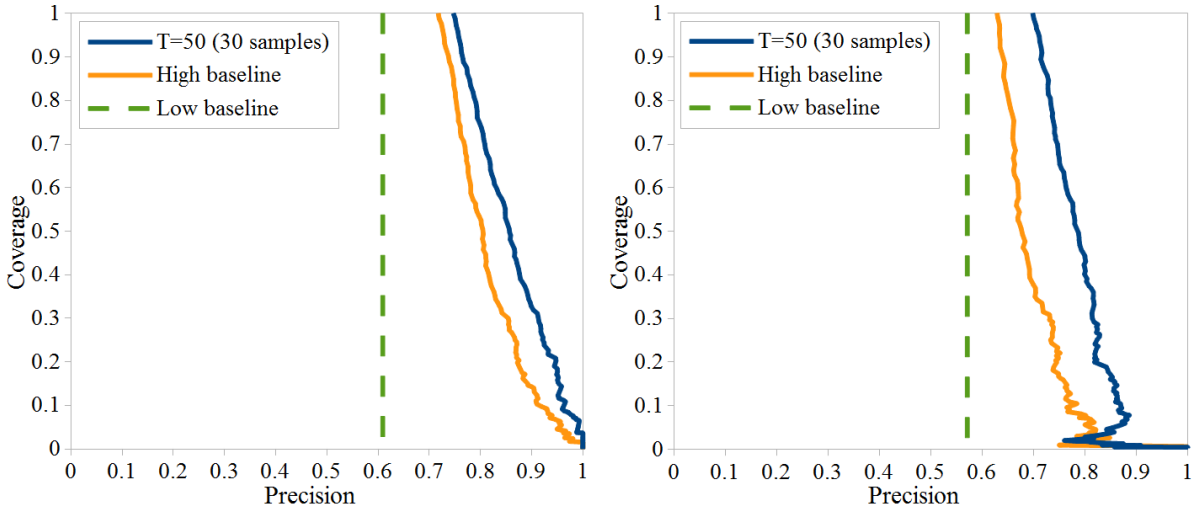

Figure 3: Coverage against precision (left WeScience, right Penn Treebank)

Since the model is probabilistic, we can interpret it conservatively, and only predict attachment if the log-odds are above a threshold. This reduces coverage, but could increase precision. We can be more confident when Gibbs samples produce similar probabilities, so we make the threshold a function of the estimated error, as in (15).[6] Here, ε_N and ε_V denote the standard error in log-probability for the nominal and verbal models - for k samples with standard deviation s, the standard error in the mean is $\varepsilon = \frac{1}{\sqrt{k}}s$. When summing independent errors, the total error is the square root of the sum of their squares.

$$|\log P(N|w) - \log P(V|w)| > \lambda \left(1 + \sqrt{\varepsilon_N^2 + \varepsilon_V^2}\right) \quad (15)$$

Graphs of coverage against precision are given in figure 3, for both datasets. As the threshold increases, the curve moves down (lower coverage) and to the right (higher precision). The increase in precision shows that the estimated probability does indeed correlate with the probability of being correct. The difference between the two solid curves shows the effect of the LDA smoothing.

5.3 Variability of Gibbs Samples

To explore how stable the probability estimates are, we evaluated the individual Gibbs samples of the $T = 50$ model. If the standard deviation s is larger than the difference in log-probability Δ, then the

[6] More complicated functions did not appear to offer any advantages; we omit results for brevity.

attachment site predicted by a single sample is not reliable - this was true for 18% of the WeScience data. If the standard error $\varepsilon = \frac{1}{\sqrt{30}}s$ is larger than Δ, then the attachment site predicted by the averaged model is not reliable - this was true for 3% of the WeScience data. Furthermore, the average accuracy of a single $T = 50$ sample on the WeScience dataset was 0.734 (standard deviation 0.0035) Hence, averaging over 30 samples reduces the number of unreliable cases by a factor of six, and increases accuracy by 1.3 percentage points.

5.4 Semantic Content of Topics

Figure 4 shows that genuine semantic information is inferred. We could characterise the first topic as describing a BUILDING in an AREA. However, the second topic reminds us that, since the topics are unsupervised, there may not always be a neat characterisation: the n_2 lemmas are all war-related, except for *election*. There is still a plausible connection between *election* and most of the n_1 lemmas, but we leave the reader to decide if elections are indeed like wars.

For large T, many topics are completely unused (with no tokens assigned to the topic), agreeing with the above conclusions that the optimal value of T is around 50.

n_1	*school, building, station, house, church, home, street, center, office, college*
n_2	*area, city, town, district, country, village, state, neighborhood, center, county*

n_1	*preparation, plan, time, way, force, date, support, responsibility, point, base*
n_2	*invasion, war, attack, operation, battle, campaign, deployment, election, landing, assault*

Figure 4: Most likely lemmas in two inferred topics (from $T = 50$ samples). Top: *in*. Bottom: *for*.

6 Discussion

6.1 Comparison with Other Approaches to PP-attachment

Our reported accuracy on the Penn Treebank data appears lower than state-of-the-art approaches, such as Zhao and Lin (2004)'s nearest-neighbour algorithm (described in section 2), which achieves 86.5% accuracy. However, the figures cannot be directly compared, for three main reasons.

Firstly, there will be a performance drop due to the change of domain - for instance, the PTB has more financial content. To quantify the domain difference, we can find the probability of generating the test data. For the $T = 50$ model, the average probability of a tuple is 8.9 times lower for the PTB than for WeScience, indicating it would be unlikely to find the PTB instances in the WikiWoods domain.

Secondly, we considered only nine prepositions, which cover just 40% of the test data. Many other prepositions are easier to deal with; for example, *of* constitutes nearly a third of all instances (926 out of 3097), but 99.1% are attached to the noun. If we simply choose the most frequent attachment site for prepositions not in our model, we achieve 79.0% accuracy, which is 7.5% lower than state-of-the-art, but this difference is well within the cross-domain drops in performance reported by McClosky et al. (2010), which vary from 5.2% to 32.0%, and by MacKinlay et al. (2011), which vary from 5.4% to 15.8%.

Thirdly, there are annotation differences between WikiWoods and the PTB, which would cause a drop in performance even if the domain were the same. As a striking example, *to* is the best performing preposition in WeScience (94% accuracy, over a baseline of 74%), but has mediocre performance on the PTB (70% accuracy, over a baseline of 61%). Much of this drop can be explained by the fact that *to* is often subcategorised for, both by verbs (*give to, pay to, provide to*), and by nouns (*exception to, damage to*). For such cases, the ERG includes *to* in the verb or noun's lexical entry, and there is no preposition in the semantics, so they do not appear in the WikiWoods training data. As a result, these cases in the PTB are often misclassified.

Finally, it may appear that performance on WeScience is also lower than state-of-the-art, but this dataset may in fact be more difficult than the PTB dataset. To quantify how useful each slot of the 4-tuple is for predicting the attachment site, we can use the conditional entropy of the attachment site given

a slot.[7] A value of 0 would imply it is perfectly predictive. For the verb slot, and both of the noun slots, the WeScience data has higher conditional entropy than the PTB[8] (1% higher for v, 17% higher for n_1, and 11% higher for n_2), suggesting that predicting attachment in the PTB data is an easier task.

6.2 Quality of Training Data

Flickinger et al. (2010) estimate the quality of the automatic WikiWoods annotations by sampling 1000 sentences and inspecting them manually to find errors. They judge "misattachment of a modifying prepositional phrase" to be a minor error, which is particularly of note considering such errors provide us with inaccurate training data. In their sample, 65.7% of sentences contained no minor errors. They do not give a breakdown of error types, so it is not possible to determine the accuracy for PP-attachment, but it is clear that a significant number of such errors were present. The results therefore indicate that our model enjoys some robustness to errors in its training data.

7 Future Work

PP-attachment ambiguities represent a fraction of all syntactic ambiguities. The most important future step is therefore to confirm the effectiveness of our framework in a wide-coverage parser, as explained in section 3.1. Additionally, the LDA smoothing could be integrated with other approaches, such as Rei and Briscoe (2013)'s reranking method, described in section 2.

The LDA model could be trained on multiple relations simultaneously, to account for cases where more than one preposition is possible, as shown in (16). This could reduce data sparsity and hence improve performance, particularly for rare prepositions. This requires no change to the mathematical formalism, simply involving multiple samples from the same Dirichlet distribution α.

(16) They walked {along, across, down} the road.

To simplify model selection, we could use a Hierarchical Dirichlet Process (Teh et al., 2006), which modifies LDA to allow an arbitrary number of topics.

8 Conclusion

We have described a novel framework for incorporating distributional semantic information in a maximum entropy parser. Within this framework, we used a generative model based on Latent Dirichlet Allocation, in order to overcome data sparsity. We evaluated this approach on the specific task of resolving PP-attachment ambiguity, explaining how this problem relates to the general case. The LDA model successfully extracted semantic information from corpus data, and outperformed a maximum likelihood baseline. Furthermore, we demonstrated that training the model is robust to various hyperparameter settings, which suggests that this method should be easy to apply to new settings. These results indicate that this is a promising approach to integrating distributional semantics with parse ranking.

References

Agirre, E., T. Baldwin, and D. Martinez (2008). Improving parsing and PP attachment performance with sense information. In *Proc. ACL*.

Akaike, H. (1974). A new look at the statistical model identification. *Automatic Control, IEEE*.

Berger, A. L., V. J. D. Pietra, and S. A. D. Pietra (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*.

[7] Ideally, we would calculate conditional entropy given the entire 4-tuple. However, almost all tuples are unique in the dataset, making estimates of entropy very error prone. Hence, we report conditional entropy given only one slot.

[8] No unbiased estimator of entropy exists for discrete distributions (Paninski, 2003). To mitigate against the effect of sample size, we averaged entropy estimates for subsamples of the WeScience dataset, to match the size of the PTB dataset.

Bergsma, S., D. Yarowsky, and K. Church (2011). Using large monolingual and bilingual corpora to improve coordination disambiguation. In *Proc. ACL*.

Blei, D. M., A. Y. Ng, and M. I. Jordan (2003). Latent Dirichlet allocation. *The Journal of Machine Learning Research*.

Bruce, R. F. and J. M. Wiebe (1999). Decomposable modeling in natural language processing. *Computational Linguistics*.

Callmeier, U. (2001). Efficient parsing with large-scale unification grammars. Master's thesis, Universität des Saarlandes, Saarbrücken, Germany.

Clark, S., A. Copestake, J. R. Curran, Y. Zhang, A. Herbelot, J. Haggerty, B.-G. Ahn, C. Van Wyk, J. Roesner, J. Kummerfeld, et al. (2009). Large-scale syntactic processing: Parsing the web final report of the 2009 JHU CLSP workshop.

Collins, M. J. (1996). A new statistical parser based on bigram lexical dependencies. In *Proc. ACL*.

Copestake, A. (2009). Slacker semantics: Why superficiality, dependency and avoidance of commitment can be the right way to go. In *Proc. EACL*.

Copestake, A., D. Flickinger, C. Pollard, and I. A. Sag (2005). Minimal recursion semantics: An introduction. *Research on Language and Computation*.

Flickinger, D. (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering*.

Flickinger, D., S. Oepen, and G. Ytrestøl (2010). WikiWoods: Syntacto-semantic annotation for English Wikipedia. In *Proc. LREC*.

Ghahramani, Z. (2004). Unsupervised learning. In *Advanced Lectures on Machine Learning*. Springer.

Griffiths, T. L. and M. Steyvers (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*.

Hindle, D. and M. Rooth (1993). Structural ambiguity and lexical relations. *Computational linguistics*.

Hogan, D. (2007). Coordinate noun phrase disambiguation in a generative parsing model. In *Proc. ACL*.

MacKinlay, A., R. Dridan, D. Flickinger, and T. Baldwin (2011). Cross-domain effects on parse selection for precision grammars. *Research on Language and Computation*.

McClosky, D., E. Charniak, and M. Johnson (2010). Automatic domain adaptation for parsing. In *Proc. NAACL*.

Ó Séaghdha, D. (2010). Latent variable models of selectional preference. In *Proc. ACL*.

Paninski, L. (2003). Estimation of entropy and mutual information. *Neural computation*.

Ratnaparkhi, A., J. Reynar, and S. Roukos (1994). A maximum entropy model for prepositional phrase attachment. In *Proc. Workshop on Human Language Technology*. ACL.

Rei, M. and T. Briscoe (2013). Parser lexicalisation through self-learning. In *Proc. NAACL-HLT*.

Steyvers, M. and T. Griffiths (2007). Probabilistic topic models. *Handbook of latent semantic analysis*.

Teh, Y. W., M. I. Jordan, M. J. Beal, and D. M. Blei (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association*.

Toutanova, K., C. D. Manning, D. Flickinger, and S. Oepen (2005). Stochastic HPSG parse selection using the Redwoods corpus. *Journal of Research on Language and Computation*.

Turney, P. D. and P. Pantel (2010). From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*.

Wallach, H., D. Mimno, and A. McCallum (2009). Rethinking LDA: Why priors matter. *Advances in Neural Information Processing Systems*.

Ytrestøl, G., S. Oepen, and D. Flickinger (2009). Extracting and annotating Wikipedia sub-domains. In *Proceedings of the 7th International Workshop on Treebanks and Linguistic Theories*.

Zhao, S. and D. Lin (2004). A nearest-neighbor method for resolving PP-attachment ambiguity. In *Natural Language Processing–IJCNLP 2004*. Springer.