Proceedings of HLT-NAACL 2003
Main Papers , pp. 126-133
Edmonton, May-June 2003

Supervised and unsupervised PCFG adaptation to novel domains

Brian Roark and **Michiel Bacchiani**
AT&T Labs - Research
{roark,michiel}@research.att.com

Abstract

This paper investigates adapting a lexicalized probabilistic context-free grammar (PCFG) to a novel domain, using maximum *a posteriori* (MAP) estimation. The MAP framework is general enough to include some previous model adaptation approaches, such as corpus mixing in Gildea (2001), for example. Other approaches falling within this framework are more effective. In contrast to the results in Gildea (2001), we show F-measure parsing accuracy gains of as much as 2.5% for high accuracy lexicalized parsing through the use of out-of-domain treebanks, with the largest gains when the amount of in-domain data is small. MAP adaptation can also be based on either supervised or unsupervised adaptation data. Even when no in-domain treebank is available, unsupervised techniques provide a substantial accuracy gain over unadapted grammars, as much as nearly 5% F-measure improvement.

1 Introduction

A fundamental concern for nearly all data-driven approaches to language processing is the sparsity of labeled training data. The sparsity of syntactically annotated corpora is widely remarked upon, and some recent papers present approaches to improving performance in the absence of large amounts of annotated training data. Johnson and Riezler (2000) looked at adding features to a maximum entropy model for stochastic unification-based grammars (SUBG), from corpora that are not annotated with the SUBG, but rather with simpler treebank annotations for which there are much larger treebanks. Hwa (2001) demonstrated how active learning techniques can reduce the amount of annotated data required to converge on the best performance, by selecting from among the candidate strings to be annotated in ways which promote more informative examples for earlier annotation. Hwa (1999) and Gildea (2001) looked at adapting parsing models trained on large amounts of annotated data from outside of the domain of interest (out-of-domain), through the use of a relatively small amount of in-domain annotated data. Hwa (1999) used a variant of the inside-outside algorithm presented in Pereira and Schabes (1992) to exploit a partially labeled out-of-domain treebank, and found an advantage to adaptation over direct grammar induction. Gildea (2001) simply added the out-of-domain treebank to his in-domain training data, and derived a very small benefit for his high accuracy, lexicalized parser, concluding that even a large amount of out-of-domain data is of little use for lexicalized parsing.

Statistical model adaptation based on sparse in-domain data, however, is neither a new problem nor unique to parsing. It has been studied extensively by researchers working on acoustic modeling for automatic speech recognition (ASR) (Legetter and Woodland, 1995; Gauvain and Lee, 1994; Gales, 1998; Lamel et al., 2002). One of the methods that has received much attention in the ASR literature is maximum *a posteriori* (MAP) estimation (Gauvain and Lee, 1994). In MAP estimation, the parameters of the model are considered to be random variables themselves with a known distribution (the prior). The prior distribution and the maximum likelihood distribution based on the in-domain observations then give a posterior distribution over the parameters, from which the mode is selected. If the amount of in-domain (adaptation) data is large, the mode of the posterior distribution is mostly defined by the adaptation sample; if the amount of adaptation data is small, the mode will nearly coincide with the mode of the prior distribution. The intuition behind MAP estimation is that once there are sufficient observations, the prior model need no longer be relied upon.

Bacchiani and Roark (2003) investigated MAP adaptation of n-gram language models, in a way that is straightforwardly applicable to probabilistic context-free grammars (PCFGs). Indeed, this approach can be used for any generative probabilistic model, such as part-of-speech taggers. In their language modeling approach, in-domain counts are mixed with the out-of-domain model, so that, if the number of observations within the domain is small, the out-of-domain model is relied upon, whereas if the number of observations in the domain is high, the model will move toward a Maximum Likelihood (ML) estimate on the in-domain data alone. The case of a parsing model trained via relative frequency estimation is identical: in-domain counts can be combined with the out-of-domain model in just such a way. We will show below that weighted count merging is a special case of MAP adaptation; hence the approach of Gildea (2001) cited above is also a special case of MAP

adaptation, with a particular parameterization of the prior. This parameterization is not necessarily the one that optimizes performance.

In the next section, MAP estimation for PCFGs is presented. This is followed by a brief presentation of the PCFG model that is being learned, and the parser that is used for the empirical trials. We will present empirical results for multiple MAP adaptation schema, both starting from the Penn Wall St. Journal treebank and adapting to the Brown corpus, and vice versa. We will compare our supervised adaptation performance with the results presented in Gildea (2001). In addition to supervised adaptation, i.e. with a manually annotated treebank, we will present results for unsupervised adaptation, i.e. with an automatically annotated treebank. We investigate a number of unsupervised approaches, including multiple iterations, increased sample sizes, and self-adaptation.

2 MAP estimation

In the maximum *a posteriori* estimation framework described in detail in Gauvain and Lee (1994), the model parameters θ are assumed to be a random vector in the space Θ. Given an observation sample $\mathbf{x}$, the MAP estimate is obtained as the mode of the posterior distribution of θ denoted as $g(. \mid \mathbf{x})$

$$\theta_{\text{MAP}} = \underset{\theta}{\operatorname{argmax}}\, g(\theta \mid \mathbf{x}) = \underset{\theta}{\operatorname{argmax}}\, f(\mathbf{x} \mid \theta) g(\theta) \quad (1)$$

In the case of n-gram model adaptation, as discussed in Bacchiani and Roark (2003), the objective is to estimate probabilities for a discrete distribution across words, entirely analogous to the distribution across mixture components within a mixture density, which is a common use for MAP estimation in ASR. A practical candidate for the prior distribution of the weights $\omega_1, \omega_2, \cdots, \omega_K$, is its conjugate prior, the Dirichlet density,

$$g(\omega_1, \omega_2, \cdots, \omega_K \mid \nu_1, \nu_2, \cdots, \nu_K) \propto \prod_{i=1}^{K} \omega_i^{\nu_i - 1} \quad (2)$$

where $\nu_i > 0$ are the parameters of the Dirichlet distribution. With such a prior, if the expected counts for the i-th component is denoted as c_i, the mode of the posterior distribution is obtained as

$$\hat{\omega}_i = \frac{(\nu_i - 1) + c_i}{\sum_{k=1}^{K} (\nu_k - 1) + \sum_{k=1}^{K} c_k} \quad 1 \leq i \leq K. \quad (3)$$

We can use this formulation to estimate the posterior, but we must still choose the parameters of the Dirichlet. First, let us introduce some notation. A context-free grammar (CFG) $G = (V, T, P, S^\dagger)$, consists of a set of non-terminal symbols V, a set of terminal symbols T, a start symbol $S^\dagger \in V$, and a set of rule productions P of the form: $A \rightarrow \gamma$, where $A \in V$ and $\gamma \in (V \cup T)^*$. A probabilistic context-free grammar (PCFG) is a CFG with a probability assigned to each rule, such that the probabilities of all rules expanding a given non-terminal sum to one; specifically, each right-hand side has a probability given the left-hand side of the rule[1].

Let A denote the left-hand side of a production, and γ_i the i-th possible expansion of A. Let the probability estimate for the production $A \rightarrow \gamma_i$ according to the out-of-domain model be denoted as $\widetilde{\mathrm{P}}(\gamma_i \mid A)$ and let the expected adaptation counts be denoted as $c(A \rightarrow \gamma_i)$. Then the parameters of the prior distribution for left-hand side A are chosen as

$$\nu_i^A = \tau_A \widetilde{\mathrm{P}}(\gamma_i \mid A) + 1 \qquad 1 \leq i \leq K. \quad (4)$$

where τ_A is the left-hand side dependent prior weighting parameter. This choice of prior parameters defines the MAP estimate of the probability of expansion γ_i from the left-hand side A as

$$\hat{\mathrm{P}}(\gamma_i \mid A) = \frac{\tau_A \widetilde{\mathrm{P}}(\gamma_i \mid A) + c(A \rightarrow \gamma_i)}{\tau_A + \sum_{k=1}^{K} c(A \rightarrow \gamma_k)} \quad 1 \leq i \leq K. \quad (5)$$

Note that the MAP estimates with this parameterization reduce to the out-of-domain model parameters in the absence of adaptation data.

Each left-hand side A has its own prior distribution, parameterized with τ_A. This presents an over-parameterization problem. We follow Gauvain and Lee (1994) in adopting a parameter tying approach. As pointed out in Bacchiani and Roark (2003), two methods of parameter tying, in fact, correspond to two well known model mixing approaches, namely count merging and model interpolation.

Let $\widetilde{\mathrm{P}}$ and $\widetilde{\mathrm{c}}$ denote the probabilities and counts from the out-of-domain model, and let $\overline{\mathrm{P}}$ and $\overline{\mathrm{c}}$ denote the probabilities and counts from the adaptation model (i.e. in-domain).

2.1 Count Merging

If the left-hand side dependent prior weighting parameter is chosen as

$$\tau_A = \widetilde{\mathrm{c}}(A) \frac{\alpha}{\beta}, \quad (6)$$

the MAP adaptation reduces to count merging, scaling the out-of-domain counts with a factor α and the in-domain counts with a factor β:

$$\begin{aligned} \hat{\mathrm{P}}(\gamma_i \mid A) &= \frac{\widetilde{\mathrm{c}}(A) \frac{\alpha}{\beta} \widetilde{\mathrm{P}}(\gamma_i \mid A) + \overline{\mathrm{c}}(A \rightarrow \gamma_i)}{\widetilde{\mathrm{c}}(A) \frac{\alpha}{\beta} + \overline{\mathrm{c}}(A)} \\ &= \frac{\alpha \widetilde{\mathrm{c}}(A \rightarrow \gamma_i) + \beta \overline{\mathrm{c}}(A \rightarrow \gamma_i)}{\alpha \widetilde{\mathrm{c}}(A) + \beta \overline{\mathrm{c}}(A)} \end{aligned} \quad (7)$$

[1] An additional condition for well-formedness is that the PCFG is consistent or tight, i.e. there is no probability mass lost to infinitely large trees. Chi and Geman (1998) proved that this condition is met if the rule probabilities are estimated using relative frequency estimation from a corpus.

2.2 Model Interpolation

If the left-hand side dependent prior weighting parameter is chosen as

$$\tau_A = \begin{cases} \overline{\mathrm{c}}(A)\frac{\lambda}{1-\lambda}, 0 < \lambda < 1 & \text{if } \overline{\mathrm{c}}(A) > 0 \\ 1 & \text{otherwise} \end{cases} \quad (8)$$

the MAP adaptation reduces to model interpolation using interpolation parameter λ:

$$\begin{aligned} \widehat{\mathrm{P}}(\gamma_i \mid A) &= \frac{\overline{\mathrm{c}}(A)\frac{\lambda}{1-\lambda}\widetilde{\mathrm{P}}(\gamma_i \mid A) + \overline{\mathrm{c}}(A \rightarrow \gamma_i)}{\overline{\mathrm{c}}(A)\frac{\lambda}{1-\lambda} + \overline{\mathrm{c}}(A)} \\ &= \frac{\frac{\lambda}{1-\lambda}\widetilde{\mathrm{P}}(\gamma_i \mid A) + \overline{\mathrm{P}}(\gamma_i \mid A)}{\frac{\lambda}{1-\lambda} + 1} \\ &= \lambda\widetilde{\mathrm{P}}(\gamma_i \mid A) + (1-\lambda)\overline{\mathrm{P}}(\gamma_i \mid A) \quad (9) \end{aligned}$$

2.3 Other Tying Candidates

While we will not be presenting empirical results for other parameter tying approaches in this paper, we should point out that the MAP framework is general enough to allow for other schema, which could potentially improve performance over simple count merging and model interpolation approaches. For example, one may choose a more complicated left-hand side dependent prior weighting parameter such as

$$\tau_A = \begin{cases} \overline{\mathrm{c}}(A)\frac{\lambda}{1-\lambda}, 0 < \lambda < 1 & \text{if } \widetilde{\mathrm{c}}(A) \gg \overline{\mathrm{c}}(A) > \theta \\ \widetilde{\mathrm{c}}(A)\frac{\alpha}{\beta} & \text{otherwise} \end{cases} \quad (10)$$

for some threshold θ. Such a schema may do a better job of managing how quickly the model moves away from the prior, particularly if there is a large difference in the respective sizes of the in-domain and out-of domain corpora. We leave the investigation of such approaches to future research.

Before providing empirical results on the count merging and model interpolation approaches, we will introduce the parser and parsing models that were used.

3 Grammar and parser

For the empirical trials, we used a top-down, left-to-right (incremental) statistical beam-search parser (Roark, 2001a; Roark, 2003). We refer readers to the cited papers for details on this parsing algorithm. Briefly, the parser maintains a set of candidate analyses, each of which is extended to attempt to incorporate the next word into a fully connected partial parse. As soon as "enough" candidate parses have been extended to the next word, all parses that have not yet attached the word are discarded, and the parser moves on to the next word. This beam search is parameterized with a base beam parameter γ, which controls how many or how few parses constitute "enough". Candidate parses are ranked by a figure-of-merit, which promotes better candidates, so that they are worked on earlier. The figure-of-merit consists of the probability of the parse to that point times a look-ahead statistic, which is an estimate of how much probability mass it will take to connect the parse with the next word. It is a generative parser that does not require any pre-processing, such as POS tagging or chunking. It has been demonstrated in the above papers to perform competitively on standard statistical parsing tasks with full coverage. Baseline results below will provide a comparison with other well known statistical parsers.

The PCFG is a *Markov* grammar (Collins, 1997; Charniak, 2000), i.e. the production probabilities are estimated by decomposing the joint probability of the categories on the right-hand side into a product of conditionals via the chain rule, and making a Markov assumption. Thus, for example, a first order Markov grammar conditions the probability of the category of the i-th child of the left-hand side on the category of the left-hand side and the category of the (i-1)-th child of the left-hand side. The benefits of Markov grammars for a top-down parser of the sort we are using is detailed in Roark (2003). Further, as in Roark (2001a; 2003), the production probabilities are conditioned on the label of the left-hand side of the production, as well as on features from the left-context. The model is smoothed using standard deleted interpolation, wherein a mixing parameter λ is estimated using EM on a held out corpus, such that probability of a production $A \rightarrow \gamma$, conditioned on j features from the left context, $X_1^j = X_1 \ldots X_j$, is defined recursively as

$$\begin{aligned} \mathrm{P}(A \rightarrow \gamma \mid X_1^j) &= \mathrm{P}(\gamma \mid A, X_1^j) \quad (11) \\ &= (1-\lambda)\widehat{\mathrm{P}}(\gamma \mid A, X_1^j) + \lambda\mathrm{P}(\gamma \mid A, X_1^{j-1}) \end{aligned}$$

where $\widehat{\mathrm{P}}$ is the maximum likelihood estimate of the conditional probability. These conditional probabilities decompose via the chain rule as mentioned above, and a Markov assumption limits the number of previous children already emitted from the left-hand side that are conditioned upon. These previous children are treated exactly as other conditioning features from the left context. Table 1 gives the conditioning features that were used for all empirical trials in this paper. There are different conditioning features for parts-of-speech (POS) and non-POS non-terminals. Deleted interpolation leaves out one feature at a time, in the reverse order as they are presented in the table 1.

The grammar that is used for these trials is a PCFG that is induced using relative frequency estimation from a transformed treebank. The trees are transformed with a selective left-corner transformation (Johnson and Roark, 2000) that has been flattened as presented in Roark (2001b). This transform is only applied to left-recursive productions, i.e. productions of the form $A \rightarrow A\gamma$. The transformed trees look as in figure 1. The transform has the benefit for a top-down incremental parser of this sort of delaying many of the parsing decisions until later in the string, without unduly disrupting the immediate dominance relationships that provide conditioning features for the probabilistic model.

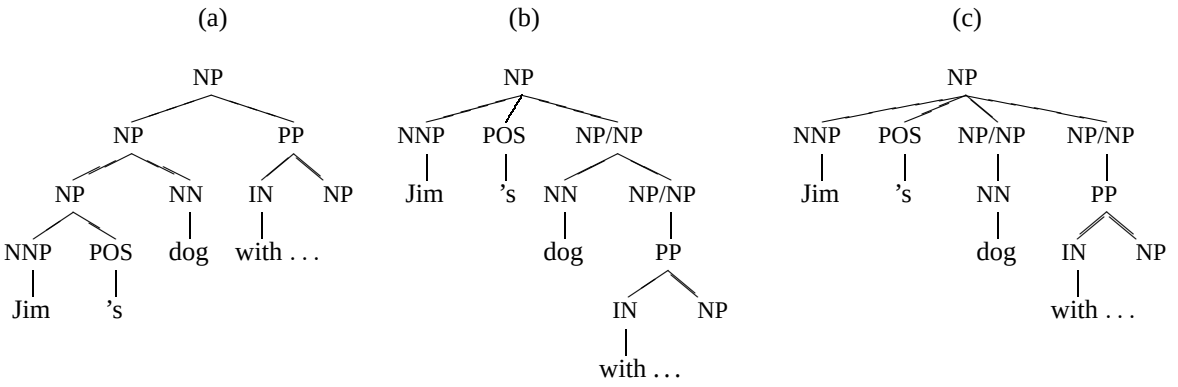

Figure 1: Three representations of NP modifications: (a) the original treebank representation; (b) Selective left-corner representation; and (c) a flat structure that is unambiguously equivalent to (b)

	Features for non-POS left-hand sides
0	Left-hand side (LHS)
1	Last child of LHS
2	2nd last child of LHS
3	3rd last child of LHS
4	Parent of LHS (PAR)
5	Last child of PAR
6	Parent of PAR (GPAR)
7	Last child of GPAR
8	First child of conjoined category
9	Lexical head of current constituent
	Features for POS left-hand sides
0	Left-hand side (LHS)
1	Parent of LHS (PAR)
2	Last child of PAR
3	Parent of PAR (GPAR)
4	POS of C-Commanding head
5	C-Commanding lexical head
6	Next C-Commanding lexical head

Table 1: Conditioning features for the probabilistic CFG used in the reported empirical trials

The parse trees that are returned by the parser are then detransformed to the original form of the grammar for evaluation[2].

For the trials reported in the next section, the base beam parameter is set at $\gamma = 10$. In order to avoid being pruned, a parse must be within a probability range of the best scoring parse that has incorporated the next word. Let k be the number of parses that have incorporated the next word, and let $\tilde{p}$ be the best probability from among that set. Then the probability of a parse must be above $\frac{\tilde{p}k^3}{10^\gamma}$ to avoid being pruned.

[2]See Johnson (1998) for a presentation of the transform/detransform paradigm in parsing.

4 Empirical trials

The parsing models were trained and tested on treebanks from the Penn Treebank II. For the Wall St. Journal portion, we used the standard breakdown: sections 2-21 were kept training data; section 24 was held-out development data; and section 23 was for evaluation. For the Brown corpus portion, we obtained the training and evaluation sections used in Gildea (2001). In that paper, no held-out section was used for parameter tuning[3], so we further partitioned the training data into kept and held-out data. The sizes of the corpora are given in table 2, as well as labels that are used to refer to the corpora in subsequent tables.

4.1 Baseline performance

The first results are for parsing the Brown corpus. Table 3 presents our baseline performance, compared with the Gildea (2001) results. Our system is labeled as 'MAP'. All parsing results are presented as labeled precision and recall. Whereas Gildea (2001) reported parsing results just for sentences of length less than or equal to 40, our results are for all sentences. The goal is not to improve upon Gildea's parsing performance, but rather to try to get more benefit from the out-of-domain data. While our performance is 0.5-1.5 percent better than Gildea's, the same trends hold – low eighties in accuracy when using the Wall St. Journal (out-of-domain) training; mid eighties when using the Brown corpus training. Notice that using the Brown held out data with the Wall St. Journal training improved precision substantially. Tuning the parameters on in-domain data can make a big difference in parser performance. Choosing the smoothing parameters as Gildea did, based on the distribution within the corpus itself, may be effective when parsing within the same distribution, but appears less so when using the treebank for parsing outside of the domain.

[3]According to the author, smoothing parameters for his parser were based on the formula from Collins (1999).

Corpus;Sect	Used for	Sentences	Words
WSJ;2-21	Training	39,832	950,028
WSJ;24	Held out	1,346	32,853
WSJ;23	Eval	2,416	56,684
Brown;T	Training	19,740	373,152
Brown;H	Held out	2,078	40,046
Brown;E	Eval	2,425	45,950

Table 2: Corpus sizes

System	Training	Heldout	LR	LP
Gildea	WSJ;2-21		80.3	81.0
MAP	WSJ;2-21	WSJ;24	81.3	80.9
MAP	WSJ;2-21	Brown;H	81.6	82.3
Gildea	Brown;T,H		83.6	84.6
MAP	Brown;T	Brown;H	84.4	85.0

Table 3: Parser performance on Brown;E, baselines. Note that the Gildea results are for sentences $\leq$ 40 words in length.

Table 4 gives the baseline performance on section 23 of the WSJ Treebank. Note, again, that the Gildea results are for sentences $\leq$ 40 words in length, while all others are for all sentences in the test set. Also, Gildea did not report performance of a Brown corpus trained parser on the WSJ. Our performance under that condition is not particularly good, but again using an in-domain held out set for parameter tuning provided a substantial increase in accuracy, somewhat more in terms of precision than recall. Our baseline results for a WSJ section 2-21 trained parser are slightly better than the Gildea parser, at more-or-less the same level of performance as Charniak (1997) and Ratnaparkhi (1999), but several points below the best reported results on this task.

4.2 Supervised adaptation

Table 5 presents parsing results on the Brown;E test set for models using both in-domain and out-of-domain training data. The table gives the adaptation (in-domain) treebank that was used, and the τ_A that was used to combine the adaptation counts with the model built from the out-of-domain treebank. Recall that $\alpha\widetilde{\mathrm{c}}(A)$ times the out-of-domain model yields count merging, with α the ratio of out-of-domain to in-domain counts; and $\alpha\overline{\mathrm{c}}(A)$ times the out-of-domain model yields model interpolation, with α the ratio of out-of-domain to in-domain probabilities. Gildea (2001) merged the two corpora, which just adds the counts from the out-of-domain treebank to the in-domain treebank, i.e. $\alpha = 1$. This resulted in a 0.25 improvement in the F-measure. In our case, combining the counts in this way yielded a half a point, perhaps because of the in-domain tuning of the smoothing parameters. However, when we optimize α empirically on the held-out corpus, we can get nearly a full point improvement. Model interpolation in this case performs nearly identically to count merging.

System	Training	Heldout	LR	LP
MAP	Brown;T	Brown;H	76.0	75.4
MAP	Brown;T	WSJ;24	76.9	77.1
Gildea	WSJ;2-21		86.1	86.6
MAP	WSJ;2-21	WSJ;24	86.9	87.1
Charniak (1997)	WSJ;2-21	WSJ;24	86.7	86.6
Ratnaparkhi (1999)	WSJ;2-21		86.3	87.5
Collins (1999)	WSJ;2-21		88.1	88.3
Charniak (2000)	WSJ;2-21	WSJ;24	89.6	89.5
Collins (2000)	WSJ;2-21		89.6	89.9

Table 4: Parser performance on WSJ;23, baselines. Note that the Gildea results are for sentences $\leq$ 40 words in length. All others include all sentences.

Adaptation to the Brown corpus, however, does not adequately represent what is likely to be the most common adaptation scenario, i.e. adaptation to a consistent domain with limited in-domain training data. The Brown corpus is not really a domain; it was built as a balanced corpus, and hence is the aggregation of multiple domains. The reverse scenario – Brown corpus as out-of-domain parsing model and Wall St. Journal as novel domain – is perhaps a more natural one. In this direction, Gildea (2001) also reported very small improvements when adding in the out-of-domain treebank. This may be because of the same issue as with the Brown corpus, namely that the optimal ratio of in-domain to out-of-domain is not 1 and the smoothing parameters need to be tuned to the new domain; or it may be because the new domain has a million words of training data, and hence has less use for out-of-domain data. To tease these apart, we partitioned the WSJ training data (sections 2-21) into smaller treebanks, and looked at the gain provided by adaptation as the in-domain observations grow. These smaller treebanks provide a more realistic scenario: rapid adaptation to a novel domain will likely occur with far less manual annotation of trees within the new domain than can be had in the full Penn Treebank.

Table 6 gives the baseline performance on WSJ;23, with models trained on fractions of the entire 2-21 test set. Sections 2-21 contain approximately 40,000 sentences, and we partitioned them by percentage of total sentences. From table 6 we can see that parser performance degrades quite dramatically when there is less than 20,000 sentences in the training set, but that even with just 2000 sentences, the system outperforms one trained on the Brown corpus.

Table 7 presents parsing accuracy when a model trained on the Brown corpus is adapted with part or all of the WSJ training corpus. From this point forward, we only present results for count merging, since model interpolation consistently performed 0.2-0.5 points below the count merging

System	Training	Heldout	Adapt	τ_A	Baseline			Adapted			ΔF
					LR	LP	F	LR	LP	F	
Gildea	WSJ;2-21		Brown;T,H	$\widetilde{c}(A)$	83.6	84.6	84.1	83.9	84.8	84.35	0.25
MAP	WSJ;2-21	Brown;H	Brown;T	$\widetilde{c}(A)$	84.4	85.0	84.7	84.9	85.6	85.25	0.55
MAP	WSJ;2-21	Brown;H	Brown;T	$0.25\widetilde{c}(A)$	84.4	85.0	84.7	85.4	85.9	85.65	0.95
MAP	WSJ;2-21	Brown;H	Brown;T	$0.20\widetilde{c}(A)$	84.4	85.0	84.7	85.3	85.9	85.60	0.90

Table 5: Parser performance on Brown;E, supervised adaptation

System	Training	%	Heldout	LR	LP
MAP	WSJ;2-21	100	WSJ;24	86.9	87.1
MAP	WSJ;2-21	75	WSJ;24	86.6	86.8
MAP	WSJ;2-21	50	WSJ;24	86.3	86.4
MAP	WSJ;2-21	25	WSJ;24	84.8	85.0
MAP	WSJ;2-21	10	WSJ;24	82.6	82.6
MAP	WSJ;2-21	5	WSJ;24	80.4	80.6

Table 6: Parser performance on WSJ;23, baselines

approach[4]. The τ_A mixing parameter was empirically optimized on the held out set when the in-domain training was just 10% of the total; this optimization makes over a point difference in accuracy. Like Gildea, with large amounts of in-domain data, adaptation improved our performance by half a point or less. When the amount of in-domain data is small, however, the impact of adaptation is much greater.

4.3 Unsupervised adaptation

Bacchiani and Roark (2003) presented unsupervised MAP adaptation results for n-gram models, which use the same methods outlined above, but rather than using a manually annotated corpus as input to adaptation, instead use an automatically annotated corpus. Their automatically annotated corpus was the output of a speech recognizer which used the out-of-domain n-gram model. In our case, we use the parsing model trained on out-of-domain data, and output a set of candidate parse trees for the strings in the in-domain corpus, with their normalized scores. These normalized scores (posterior probabilities) are then used to give weights to the features extracted from each candidate parse, in just the way that they provide expected counts for an expectation maximization algorithm.

For the unsupervised trials that we report, we collected up to 20 candidate parses per string[5]. We were interested in investigating the effects of adaptation, not in optimizing performance, hence we did not empirically optimize the mixing parameter τ_A for the new trials, so as to avoid obscuring the effects due to adaptation alone. Rather, we used the best performing parameter from the supervised trials, namely $0.20\widetilde{c}(A)$. Since we are no longer limited to manually annotated data, the amount of in-domain WSJ data that we can include is essentially unlimited. Hence the trials reported go beyond the 40,000 sentences in the Penn WSJ Treebank, to include up to 5 times that number of sentences from other years of the WSJ.

Table 8 shows the results of unsupervised adaptation as we have described it. Note that these improvements are had without seeing any manually annotated Wall St. Journal treebank data. Using the approximately 40,000 sentences in f2-21, we derived a 3.8 percent F-measure improvement over using just the out of domain data. Going beyond the size of the Penn Treebank, we continued to gain in accuracy, reaching a total F-measure improvement of 4.2 percent with 200 thousand sentences, approximately 5 million words. A second iteration with this best model, i.e. re-parsing the 200 thousand sentences with the adapted model and re-training, yielded an additional 0.65 percent F-measure improvement, for a total F-measure improvement of 4.85 percent over the baseline model.

A final unsupervised adaptation scenario that we investigated is self-adaptation, i.e. adaptation on the test set itself. Because this adaptation is completely unsupervised, thus does not involve looking at the manual annotations at all, it can be equally well applied using the test set as the unsupervised adaptation set. Using the same adaptation procedure presented above on the test set itself, i.e. producing the top 20 candidates from WSJ;23 with normalized posterior probabilities and re-estimating, we produced a self-adapted parsing model. This yielded an F-measure accuracy of 76.8, which is a 1.1 percent improvement over the baseline.

5 Conclusion

What we have demonstrated in this paper is that maximum *a posteriori* (MAP) estimation can make out-of-domain training data beneficial for statistical parsing. In the most likely scenario – porting a parser to a novel domain for which there is little or no annotated data – the improvements can be quite large. Like active learning, model adaptation can reduce the amount of annotation required to converge to a best level of performance. In fact, MAP coupled with active learning may reduce the required amount of annotation further.

There are a couple of interesting future directions for this

[4]This is consistent with the results presented in Bacchiani and Roark (2003), which found a small but consistent improvement in performance with count merging versus model interpolation for n-gram modeling.

[5]Because of the left-to-right, heuristic beam-search, the parser does not produce a chart, rather a set of completed parses.

System	% of WSJ;2-21	τ_A	Baseline			Adapted			ΔF
			LR	LP	F	LR	LP	F	
Gildea	100	$\widetilde{c}(A)$	86.1	86.6	86.35	86.3	86.9	86.60	0.25
MAP	100	$0.20\widetilde{c}(A)$	86.9	87.1	87.00	87.2	87.5	87.35	0.35
MAP	75	$0.20\widetilde{c}(A)$	86.6	86.8	86.70	87.1	87.3	87.20	0.50
MAP	50	$0.20\widetilde{c}(A)$	86.3	86.4	86.35	86.7	86.9	86.80	0.45
MAP	25	$0.20\widetilde{c}(A)$	84.8	85.0	84.90	85.3	85.5	85.40	0.50
MAP	10	$0.20\widetilde{c}(A)$	82.6	82.6	82.60	84.3	84.4	84.35	1.75
MAP	10	$\widetilde{c}(A)$	82.6	82.6	82.60	83.2	83.4	83.30	0.70
MAP	5	$0.20\widetilde{c}(A)$	80.4	80.6	80.50	83.0	83.1	83.05	2.55

Table 7: Parser performance on WSJ;23, supervised adaptation. All models use Brown;T,H as the out-of-domain treebank. Baseline models are built from the fractions of WSJ;2-21, with no out-of-domain treebank.

Adaptation Sentences	Iteration	LR	LP	F-measure	ΔF
0	0	76.0	75.4	75.70	
4000	1	78.6	77.9	78.25	2.55
10000	1	78.9	78.0	78.45	2.75
20000	1	79.3	78.5	78.90	3.20
30000	1	79.7	78.9	79.30	3.60
39832	1	79.9	79.1	79.50	3.80
100000	1	79.7	79.2	79.45	3.75
200000	1	80.2	79.6	79.90	4.20
200000	2	80.6	80.5	80.55	4.85

Table 8: Parser performance on WSJ;23, unsupervised adaptation. For all trials, the base training is Brown;T, the held out is Brown;H plus the parser output for WSJ;24, and the mixing parameter τ_A is $0.20\widetilde{c}(A)$.

research. First, a question that is not addressed in this paper is how to best combine both supervised and unsupervised adaptation data. Since each in-domain resource is likely to have a different optimal mixing parameter, since the supervised data is more reliable than the unsupervised data, this becomes a more difficult, multi-dimensional parameter optimization problem. Hence, we would like to investigate automatic methods for choosing mixing parameters, such as EM. Also, an interesting question has to do with choosing which treebank to use for out-of-domain data. For a new domain, is it better to choose as prior the balanced Brown corpus, or rather the more robust Wall St. Journal treebank? Perhaps one could use several out-of-domain treebanks as priors. Most generally, one can imagine using k treebanks, some in-domain, some out-of-domain, and trying to find the best mixture to suit the particular task.

The conclusion in Gildea (2001), that out-of-domain treebanks are not particularly useful in novel domains, was premature. Instead, we can conclude that, just as in other statistical estimation problems, there are generalizations to be had from these out-of-domain trees, providing more robust estimates, especially in the face of sparse training data.

References

Michiel Bacchiani and Brian Roark. 2003. Unsupervised language model adaptation. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, pages 598–603.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, pages 132–139.

Zhiyi Chi and Stuart Geman. 1998. Estimation of probabilistic context-free grammars. *Computational Linguistics*, 24(2):299–305.

Michael J. Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 16–23.

Michael J. Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Michael J. Collins. 2000. Discriminative reranking for natural language parsing. In *The Proceedings of the 17th International Conference on Machine Learning*.

M. J. F. Gales. 1998. Maximum likelihood linear transformations for hmm-based speech recognition. *Computer Speech and Language*, pages 75–98.

Jean-Luc Gauvain and Chin-Hui Lee. 1994. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE Transactions on Speech and Audio Processing*, 2(2):291–298.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proceedings of the Sixth Conference on Empirical Methods in Natural Language Processing (EMNLP-01)*.

Rebecca Hwa. 1999. Supervised grammar induction using training data with limited constituent information. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*.

Rebecca Hwa. 2001. On minimizing training corpus for parser acquisition. In *Proceedings of the Fifth Computational Natural Language Learning Workshop*.

Mark Johnson and Stefan Riezler. 2000. Exploiting auxiliary distributions in stochastic unification-based grammars. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*.

Mark Johnson and Brian Roark. 2000. Compact non-left-recursive grammars using the selective left-corner transform and factoring. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING)*, pages 355–361.

Mark Johnson. 1998. PCFG models of linguistic tree representations. *Computational Linguistics*, 24(4):617–636.

L. Lamel, J.-L. Gauvain, and G. Adda. 2002. Unsupervised acoustic model training. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 877–880.

C. J. Legetter and P.C. Woodland. 1995. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer Speech and Language*, pages 171–185.

Fernando C.N. Pereira and Yves Schabes. 1992. Inside-outside reestimation from partially bracketed corpora. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, pages 128–135.

Adwait Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy models. *Machine Learning*, 34:151–175.

Brian Roark. 2001a. Probabilistic top-down parsing and language modeling. *Computational Linguistics*, 27(2):249–276.

Brian Roark. 2001b. *Robust Probabilistic Predictive Syntactic Processing*. Ph.D. thesis, Brown University. http://arXiv.org/abs/cs/0105019.

Brian Roark. 2003. Robust garden path parsing. *Natural Language Engineering*, 9(2):1–24.