A Backoff Model for Bootstrapping Resources for Non-English Languages*

Chenhai Xi and Rebecca Hwa
Department of Computer Science
University of Pittsburgh
{chenhai,hwa}@cs.pitt.edu

Abstract

The lack of annotated data is an obstacle to the development of many natural language processing applications; the problem is especially severe when the data is non-English. Previous studies suggested the possibility of acquiring resources for non-English languages by bootstrapping from high quality English NLP tools and parallel corpora; however, the success of these approaches seems limited for dissimilar language pairs. In this paper, we propose a novel approach of combining a bootstrapped resource with a small amount of manually annotated data. We compare the proposed approach with other bootstrapping methods in the context of training a Chinese Part-of-Speech tagger. Experimental results show that our proposed approach achieves a significant improvement over EM and self-training and systems that are only trained on manual annotations.

1 Introduction

Natural language applications that use supervised learning methods require annotated training data, but annotated data is scarce for many non-English languages. It has been suggested that annotated data for these languages might be automatically created by leveraging parallel corpora and high-accuracy English systems (Yarowsky and Ngai, 2001; Diab and Resnik, 2002). The studies are centered around the assumption that linguistic analyses for English (e.g., Part-of-Speech tags, Word sense disambiguation, grammatical dependency relationships) are also valid analyses in the translation of the English. For example, in the English noun phrase *the red apples*, *red* modifies *apples*; the same modifier relationship also exists in its French translations *les pommes rouges*, even though the word orders differ. To the extent that the assumption is true, annotated data in the non-English language can be created by projecting English analyses across a word aligned parallel corpus. The resulting projected data can then serve as (albeit noisy) training examples to develop applications in the non-English language.

The projection approach faces both a theoretical and a practical challenge. Theoretically, it is well-known that two languages often do not express the same meaning in the same way (Dorr, 1994). Practically, the projection framework is sensitive to component errors. In particular, poor word alignments significantly degrade the accuracy of the projected annotations. Previous research on resource projection attempts to address these problems by redistributing the parameter values (Yarowsky and Ngai, 2001) or by applying transformation rules (Hwa et al.,

*We thank Stephen Clark, Roger Levy, Carol Nichols, and the three anonymous reviewers for their helpful comments.

2002). Their experimental results suggest that while these techniques can overcome some errors, they are not sufficient for projected data that are very noisy.

In this work, we tackle the same problems by relaxing the *zero manual annotation* constraint. The main question we address is: how can we make the most out of a small set of manually labeled data (on the non-English side). Following the work of Yarowsky and Ngai (2001) we focus on the task of training a Part-of-Speech (POS) tagger, but we conduct our experiments with the more dissimilar language pair of English-Chinese instead of English-French. Through empirical studies, we show that when the word alignment quality is sufficiently poor, the error correction techniques proposed by Yarowsky and Ngai are unable to remove enough mistakes in the projected data. We propose an alternative approach that is inspired by backoff language modeling techniques in which the parameters of two tagging models (one trained on manually labeled data; the other trained on projected data) are combined to achieve a more accurate final model.

2 Background

The idea of trying to squeeze more out of annotated training examples has been explored in a number of ways in the past. Most popular is the family of bootstrapping algorithms, in which a model is seeded with a small amount of labeled data and iteratively improved as more unlabeled data are folded into the training set, typically, through unsupervised learning. Another approach is *active learning* (Cohn et al., 1996), in which the model is also iteratively improved but the training examples are chosen by the learning model, and the learning process is supervised. Finally, the work that is the closest to ours in spirit is the idea of joint estimation (Smith and Smith, 2004).

Of the bootstrapping methods, perhaps the most well-known is the Expectation Maximization (EM) algorithm. This approach has been explored in the context of many NLP applications; one example is text classification (Nigam et al., 1999). Another bootstrapping approach reminiscent of EM is *self-training*. Yarowsky (1995) used this method for word sense disambiguation. In self-training, annotated examples are used as seeds to train an initial classifier with any supervised learning method. This initial classifier is then used to automatically annotate data from a large pool of unlabeled examples. Of these newly labeled data, the ones labeled with the highest confidence are used as examples to train a new classifier. Yarowsky showed that repeated application of this process resulted in a series of word sense classifiers with improved accuracy and coverage. Also related is the co-training algorithm (Blum and Mitchell, 1998) in which the bootstrapping process requires multiple learners that have different *views* of the problem. The key to co-training is that the views should be *conditionally independent* given the label. The strong independence requirement on the views is difficult to satisfy. For practical applications, different features sets or models (that are not conditionally independent) have been used as an approximation for different views. Co-training has been applied to a number of NLP applications, including POS-tagging (Clark et al., 2003), parsing (Sarkar, 2001), word sense disambiguation (Mihalcea, 2004), and base noun phrase detection (Pierce and Cardie, 2001). Due to the relaxation of the view independence assumption, most empirical studies suggest a marginal improvement. The common thread between EM, self-training, and co-training is that they all bootstrap off of unannotated data. In this work, we explore an alternative to "pure" unannotated data; our data have been automatically annotated with projected labels from English. Although the projected labels are error-prone, they provide us with more information than automatically predicted labels used in bootstrapping methods.

With a somewhat different goal in mind, active learning addresses the problem of choosing the most informative data for annotators to label so that the model would achieve the greatest improvement. Active learning also has been applied to many NLP applications, including POS tagging (Engelson and Dagan, 1996) and pars-

ing (Baldridge and Osborne, 2003). The drawback of an active learning approach is that it assumes that a staff of annotators is waiting on call, ready to label the examples chosen by the system at every iteration. In practice, it is more likely that one could only afford to staff annotators for a limited period of time. Although active learning is not a focus in this paper, we owe some ideas to active learning in choosing a small initial set of training examples; we discuss these ideas in section 3.2.

More recently, Smith and Smith (2004) proposed to merge an English parser, a word alignment model, and a Korean PCFG parser trained from a small number of Korean parse trees under a unified log linear model. Their results suggest that a joint model produces somewhat more accurate Korean parses than a PCFG Korean parser trained on a small amount of annotated Korean parse trees alone. Their motivation is similar to the starting point of our work: that a word aligned parallel corpus and a small amount of annotated data in the foreign language side offer information that might be exploited. Our approach differs from theirs in that we do not optimize the three models jointly. One concern is that joint optimization might not result in optimal parameter settings for the individual components. Because our focus is primarily on acquiring non-English language resources, we only use the parallel corpus as a means of projecting resources from English.

3 Our Approach

This work explores developing a Chinese POS tagger without a large manually annotated corpus. Our approach is to train two separate models from two different data sources: a large corpus of automatically tagged data (projected from English) and a small corpus of manually tagged data; the two models are then combined into one via the Whitten-Bell backoff language model.

3.1 Projected Data

One method of acquiring a large corpus of automatically POS tagged Chinese data is by *projection* (Yarowsky and Ngai, 2001). This approach requires a sentence-aligned English-Chinese corpus, a high-quality English tagger, and a method of aligning English and Chinese words that share the same meaning. Given the parallel corpus, we tagged the English words with a publicly available maximum entropy tagger (Ratnaparkhi, 1996), and we used an implementation of the IBM translation model (Al-Onaizan et al., 1999) to align the words. The Chinese words in the parallel corpus would then receive the same POS tags as the English words to which they are aligned. Next, the basic projection algorithm is modified to accommodate two complicating factors. First, word alignments are not always one-to-one. To compensate, we assign a default tag to unaligned Chinese words; in the case of one-Chinese-to-many-English, the Chinese word would receive the tag of the final English word. Second, English and Chinese do not share the same tag set. Following Yarowsky and Ngai (2001), we define 12 equivalence classes over the 47 Penn-English-Treebank POS tags. We refer to them as *Core Tags*. With the help of 15 hand-coded rules and a Naive Bayes model trained on a small amount of manually annotated data, the Core Tags can be expanded to the granularity of the 33 Penn-Chinese-Treebank POS tags (which we refer to as *Full Tags*).

3.2 Manually Annotated Data

Since the amount of manual annotation is limited, we must decide what type of data to annotate. In the spirit of active learning, we aim to select sentences that may bring about the greatest improvements in the accuracy of our model. Because it is well known that handling unknown words is a serious problem for POS taggers, our strategy for selecting sentences for manual annotation is to maximize the word coverage of the in ital model. That is, we wish to find a small set of sentences that would lead to the greatest reduction of currently unknown words Finding these sentences is a NP-hard problem because the 0/1 knapsack problem could be reduced to this problem in polynomial-time (Gurari, 1989). Thus, we developed an approximation algorithm for finding sentences with the maximum word

```
M : number of tokens will be annotated.
S={s_1, s_2, ..., s_n}: the unannotated corpus.
S_sel : set of selected sentences in S.
S_unsel : set of unselected sentences in S.
|S_sel| : number of tokens in S_sel.
TYPE(S_sel) : number of types in S_sel.
MWC:
    randomly choose S_sel ⊂ S
     such that|S_sel| ≤ M.
 For  each sentence s_i in S_sel
    find a sentence r_j  in S_unsel
      which maximizes swap_score(s_i, r_j).
    if swap_score(s_i, r_j) > 0
    {
        S_sel = (S_sel − s_i) ∪ r_j;
        S_unsel = (S_unsel − r_j) ∪ s_i;
    }

swap_score(s_i, r_j)
{
  S_sel_new = (S_sel − s_i) ∪ r_j;
  if ( |S_sel_new| > M ) return -1;
  else return TYPE(S_sel_new) − TYPE(S_sel);
}
```

Figure 1: The pseudo-code for MWC algorithm. The input is M and S and the output is S_{sel}

coverage of unknown words (MWC). This algorithm is described in Figure 1,

3.3 Basic POS Tagging Model

It is well known that a POS tagger can be trained with an HMM (Weischedel et al., 1993). Given a trained model, the most likely tag sequence $\hat{T} = \{t_1, t_2, \ldots t_n\}$ is computed for the input word sentence: $\hat{W} = \{w_1, w_2, \ldots w_n\}$:

$$\hat{T} = \arg\max_T P(T|W) = \arg\max_T P(W|T)P(T)$$

The transition probability $P(T)$ is approximated by a trigram model:

$$P(T) \approx p(t_1)p(t_2|t_1)\prod_{i=3}^{n} p(t_i|t_{i-1}, t_{i-2}),$$

and the observation probability $P(W|T)$ is computed by

$$P(W|T) \approx \prod_{i=1}^{n} p(w_i|t_i).$$

3.4 Combined Models

From the two data sources, two separate trigram taggers have been trained (T_{anno} from manually annotated data and T_{proj} from projected data). This section considers ways of combining them into a single tagger. The key insight that drives our approach is based on reducing the effect of unknown words. We see the two data sources as complementary in that the large projected data source has better word coverage while the manually labeled one is good at providing tag-to-tag transitions. Based on this principle, one way of merging these two taggers into a single HMM (denoted as T_{interp}) is to use interpolation:

$$\begin{aligned} p_{interp}(w|t) &= \lambda \times p_{anno}(w|t) \\ &\quad +(1-\lambda) \times p_{proj}(w|t) \\ p_{interp}(t_i|t_{i-1}, t_{i-2}) &= p_{anno}(t_i|t_{i-1}, t_{i-2}) \end{aligned}$$

where λ is a tunable weighting parameter[1] of the merged tagger. This approach may be problematic because it forces the model to always include some fraction of poor parameter values. Therefore, we propose to estimate the observation probabilities using backoff. The parameters of T_{back} are estimated as follows:

$$p_{back}(w|t) = \begin{cases} \alpha(t) \times p_{anno}(w|t) \text{ if } p_{anno}(w|t) > 0 \\ \beta(t) \times p_{proj}(w|t) \text{ if } p_{anno}(w|t) = 0 \end{cases}$$

$$p_{back}(t_i|t_{i-1}, t_{i-2}) = p_{anno}(t_i|t_{i-1}, t_{i-2})$$

where $\alpha(t)$ is a discounting coefficient and β is set to satisfy that $\sum_{\text{all words}} P(w|t) = 1$. The discounting coefficient is computed using the Witten-Bell discounting method:

$$\alpha(t) = \frac{C_{anno}(t)}{C_{anno}(t) + S_{anno}(t)},$$

where $C_{anno}(t)$ is the count of tokens whose tag is t in the manually annotated corpus and

[1] In our experiments, the value of λ is set to 0.8 based on held-out data.

$S_{anno}(t)$ is the seen types of words with tag t. In other words, we trust the parameter estimates from the model trained on manual annotation by default unless it is based on unreliable statistics.

4 Experiments

We conducted a suite of experiments to investigate the effect of allowing a small amount of manually annotated data in conjunction with using annotations projected from English. We first establish a baseline of training on projected data alone in Section 4.1. It is an adaptation of the approach described by Yarowsky and Ngai (2001). Next, we consider the case of using manually annotated data alone in Section 4.2. We show that there is an increase in accuracy when the MWC active learning strategy is used. In Section 4.3, we show that with an appropriate merging strategy, a tagger trained from both data sources achieves higher accuracy. Finally, in Section 4.4, we evaluate our approach against other semi-supervised methods to verify that the projected annotations, though noisy, contain useful information.

We use an English-Chinese Federal Broadcast Information Service (FBIS) corpus as the data source for the projected annotation. We simulated the manual annotation process by using the POS tags provided by the Chinese Treebank version 4 (CHTB). We used about a thousand sentences from CHTB as held-out data. The remaining sentences are split into ten-fold cross validation sets. Each test set contains 1400 sentences. Training data are selected (using MWC) from the remaining 12600 sentences. The reported results are the average of the ten trials. One tagger is considered to be better than another if, according to the paired t-test, we are at least 95% confident that their difference in accuracy is non-zero. Performance is measured in terms of the percentage of correctly tagged tokens in the test data. For comparability with T_{proj} (which assumes no availability of manually annotated data), most experimental results are reported with respect to the reduced Core Tag gold standard; evaluation against the full 33 CHTB tag gold standard is reported in Section 4.4.

4.1 Tagger Trained from Projected Data

To determine the quality of T_{proj} for Chinese, we replicate the POS-tagging experiment in Yarowsky and Ngai (2001). Trained on all projected data, the tagger has an accuracy of 58.2% on test sentences. The low accuracy rate suggests that the projected data is indeed very noisy. To reduce the noise in the projected data, Yarowsky and Ngai developed a re-estimation technique based on the observation that words in French, English and Czech have a strong tendency to exhibit only a single core POS tag and very rarely have more than two. Applying the same re-estimation technique that favors this bias to the projected Chinese data raises the final tagger accuracy to 59.1%. That re-estimation did not help English-Chinese projection suggests that the dissimilarity between the two languages is an important factor. A related reason for the lower accuracy rate is due to poor word alignments in the English-Chinese corpus. As a further noise reduction step, we automatically filter out sentence pairs that were poorly aligned (i.e., the sentence pairs had too many unaligned words or too many one-to-many alignments). This results in a corpus of about 9000 FBIS sentences. A tagger trained on the filtered data has an improved accuracy of 64.5%. We take this to be T_{proj} used in later experiments.

4.2 Taggers Trained from Manually Labeled Data

This experiment verifies that the Maximum Word Coverage (MWC) selection scheme presented in Section 3.2 is helpful in selecting data for training T_{anno}. We compare it against random selection. Figure 2 plots the taggers' performances on test sentences as the number of manually annotated tokens increase from 100 to 30,000. We see that the taggers trained on data selected by MWC outperform those trained on randomly selected data. Thus, in the main experiments, we always use MWC to select the set of manually tagged data for training T_{anno}.

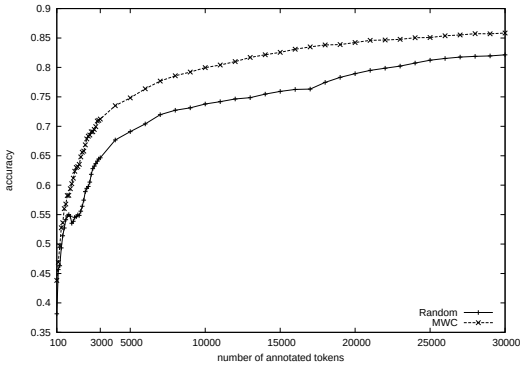


Figure 2: A comparison between MWC and random selection.

4.3 Evaluation of the Combined Taggers

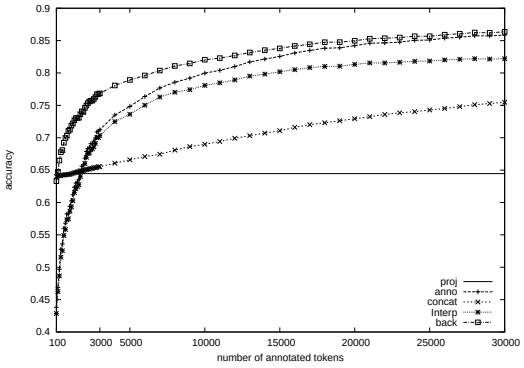


Figure 3: A comparison of the proposed backoff approach against alternative methods of combining T_{proj} and T_{anno}

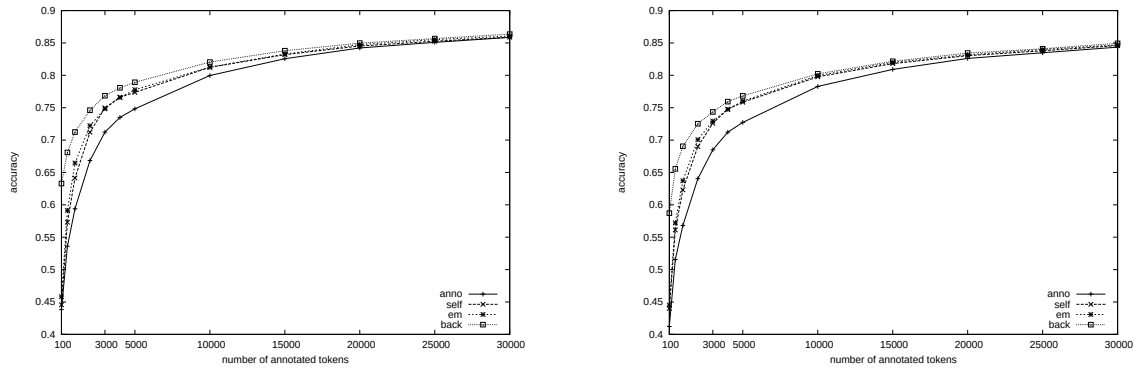
To investigate how T_{anno} and T_{proj} might be merged to form a higher quality tagger, we conduct an experiment to evaluate the different alternatives described in section 3.4: T_{interp}, and T_{back}. They are compared against three baselines: T_{anno}, T_{proj}, and T_{concat}, a tagger trained from the concatenation of the two data sources. To determine the effect of manual annotation, we vary the size of the training set for T_{anno} from 100 tokens (fewer than 10 sentences) to 30,000 tokens (about 1000 sentences). The learning curves are plotted in Figure 3. The result suggests that T_{back} successfully incorporates information from both the manually annotated data and the projected data. The improvement over training on manually annotated data alone (T_{anno}) is especially high when fewer than 10,000 manually annotated tokens are available. As expected, T_{interp}, and T_{concat} perform worse than T_{anno} because they are not as effective at discounting the erroneous projected annotations.

4.4 Comparisons with Other Semi-Supervised Approaches

This experiment evaluates the proposed backoff approach against two other semi-supervised approaches: self-training (denoted as T_{self}) and EM (denoted as T_{em}). Both start with a fully supervised model (T_{anno}) and iteratively improve it by seeing more unannotated data.[2] As discussed earlier, a major difference between our proposed approach and the bootstrapping methods is that our approach makes use of annotations projected from English while the bootstrapping methods rely on unannotated data alone. To investigate the effect of leveraging from English resources, we use the Chinese portion of the FBIS parallel corpus (the same 9000 sentences as the training corpus of T_{proj} but without the projected tags) as the unannotated data source for the bootstrapping methods.

Figure 4 compares the four learning curves. We have evaluated them both in terms of the Core Tag gold standard and in terms of Full Tag gold standard. Although all three approaches produce taggers with higher accuracies than that of T_{anno}, our backoff approach outperforms both self-training and EM. The difference is especially prominent when manual annotation is severely limited. When more manual annotations are made available, the gap narrows; however, the differences are still statistically significant at 30,000 manually annotated tokens. These results suggest that projected data have more useful information than unannotated data.

[2] In our implementation of self-training, the top 10% of the unannoated sentences with the highest confidence scores is selected. The confidence score is computed as: $\frac{\log P(T|W)}{\text{length of the sentence}}$.



(a) (b)

Figure 4: A comparison of Backoff against self-training and EM. (a) Evaluation against the Core Tag gold standard. (b) Evaluation against the Full Tag gold standard.

5 Discussion

While the experimental results support our intuition that T_{back} is effective in making use of both data sources, there are still two questions worth addressing. First, there may be other ways of estimating the parameters of a merged HMM from the parameters of T_{anno} and T_{proj}. For example, a natural way of merging the two taggers into a single HMM (denoted as T_{merge}) is to use the values of the observation probabilities from T_{proj} and the values of the transition probabilities from T_{anno}:

$$\begin{aligned} p_{merge}(w|t) &= p_{proj}(w|t), \\ p_{merge}(t_i|t_{i-1}, t_{i-2}) &= p_{anno}(t_i|t_{i-1}, t_{i-2}). \end{aligned}$$

Another is the reverse of T_{merge}:

$$\begin{aligned} p_{rev\_merge}(w|t) &= p_{anno}(w|t) \\ p_{rev\_merge}(t_i|t_{i-1}, t_{i-2}) &= p_{proj}(t_i|t_{i-1}, t_{i-2}) \end{aligned}$$

T_{merge} is problematic because it ignores all manual word-tag annotations; however, T_{rev_merge}'s learning curve is nearly identical to that of T_{anno} (graph not shown). Its models do not take advantage of the broader word coverage of the projected data, so it does not perform as well as T_{back}. T_{rev_merge} outperforms T_{merge} when trained from more than 2000 manually annotated tokens. We make two observations from this finding. One is that the differences between $p_{proj}(t_i|t_{i-1}, t_{i-2})$ and $p_{anno}(t_i|t_{i-1}, t_{i-2})$ are not large. Another is that the success of the merged HMM tagger hinges on the goodness of the observation probabilities, $p(w|t)$. This is in accord with our motivation in improving the reliability of $p(w|t)$ through backoff.

Second, while our experimental results suggest that T_{back} outperforms self-training and EM, these approaches are not incompatible with one another. Because T_{back} is partially estimated from the noisy corpus of projected annotations, it might be further improved by applying a bootstrapping algorithm over the noisy corpus (with the projected tags removed). To test our hypothesis, we initialized the self-training algorithm with a backoff tagger that used 3000 manually annotated tokens. This led to a slight but statistically significant improvement, from 74.3% to 74.9%.

6 Conclusion and Future Work

In summary, we have shown that backoff is an effective technique for combining manually annotated data with a large but noisy set of automatically annotated data (from projection). Our ap-

proach is the most useful when a small amount of annotated tokens is available. In our experiments, the best results were achieved when we used 3000 manually annotated tokens (approximately 100 sentences).

The current study points us to several directions for future work. One is to explore ways of applying the proposed approach to other learning models. Another is to compare against other methods of combining evidences from multiple learners. Finally, we will investigate whether the proposed approach can be adapted to more complex tasks in which the output is not a class label but a structure (e.g. parsing).

References

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, I. Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation. Technical report, JHU. citeseer.nj.nec.com/al-onaizan99statistical.html.

Jason Baldridge and Miles Osborne. 2003. Active learning for HPSG parse selection. In *Proceedings of the 7th Conference on Natural Language Learning*, Edmonton, Canada, June.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 1998 Conference on Computational Learning Theory*, pages 92–100, Madison, WI.

Stephen Clark, James Curran, and Miles Osborne. 2003. Bootstrapping pos-taggers using unlabelled data. In *Proc. of the Computational Natural Language Learning Conference*, pages 164–167, Edmonton, Canada, June.

David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. 1996. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145.

Mona Diab and Philip Resnik. 2002. An unsupervised method for word sense tagging using parallel corpora. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA.

Bonnie J. Dorr. 1994. Machine translation divergences: A formal description and proposed solution. *Computational Linguistics*, 20(4):597–635.

Sean P. Engelson and Ido Dagan. 1996. Minimizing manual annotation cost in supervised training from copora. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 319–326, Santa Cruz, CA.

Eitan Gurari. 1989. *An Introduction to the Theory of Computation*. Ohio State University Computer Science Press.

Rebecca Hwa, Philip S. Resnik, Amy Weinberg, and Okan Kolak. 2002. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Rada Mihalcea. 2004. Co-training and self-training for word sense disambiguation. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL-2004)*.

Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom Mitchell. 1999. Text Classification from Labeled and Unlabeled Documents using EM. *Machine Learning*, 1(34).

David Pierce and Claire Cardie. 2001. Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP-01)*, pages 1–9, Pittsburgh, PA.

Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In Eric Brill and Kenneth Church, editors, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 133–142. Association for Computational Linguistics, Somerset, New Jersey.

Anoop Sarkar. 2001. Applying co-training methods to statistical parsing. In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics*, pages 175–182, Pittsburgh, PA, June.

David A. Smith and Noah A. Smith. 2004. Bilingual parsing with factored estimation: Using English to parse Korean. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*.

Ralph Weischedel, Richard Schwartz, Jeff Palmucci, Marie Meteer, and Lance Ramshaw. 1993. Coping with ambiguity and unknown words through probabilistic models. *Comput. Linguist.*, 19(2):361–382.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics*, pages 200–207.

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, MA.