Scaling to Very Very Large Corpora for Natural Language Disambiguation

Michele Banko and Eric Brill
Microsoft Research
1 Microsoft Way
Redmond, WA 98052 USA
{mbanko,brill}@microsoft.com

Abstract

The amount of readily available on-line text has reached hundreds of billions of words and continues to grow. Yet for most core natural language tasks, algorithms continue to be optimized, tested and compared after training on corpora consisting of only one million words or less. In this paper, we evaluate the performance of different learning methods on a prototypical natural language disambiguation task, confusion set disambiguation, when trained on orders of magnitude more labeled data than has previously been used. We are fortunate that for this particular application, correctly labeled training data is free. Since this will often not be the case, we examine methods for effectively exploiting very large corpora when labeled data comes at a cost.

1 Introduction

Machine learning techniques, which automatically learn linguistic information from online text corpora, have been applied to a number of natural language problems throughout the last decade. A large percentage of papers published in this area involve comparisons of different learning approaches trained and tested with commonly used corpora. While the amount of available online text has been increasing at a dramatic rate, the size of training corpora typically used for learning has not. In part, this is due to the standardization of data sets used within the field, as well as the potentially large cost of annotating data for those learning methods that rely on labeled text.

The empirical NLP community has put substantial effort into evaluating performance of a large number of machine learning methods over fixed, and relatively small, data sets. Yet since we now have access to significantly more data, one has to wonder what conclusions that have been drawn on small data sets may carry over when these learning methods are trained using much larger corpora.

In this paper, we present a study of the effects of data size on machine learning for natural language disambiguation. In particular, we study the problem of selection among confusable words, using orders of magnitude more training data than has ever been applied to this problem. First we show learning curves for four different machine learning algorithms. Next, we consider the efficacy of voting, sample selection and partially unsupervised learning with large training corpora, in hopes of being able to obtain the benefits that come from significantly larger training corpora without incurring too large a cost.

2 Confusion Set Disambiguation

Confusion set disambiguation is the problem of choosing the correct use of a word, given a set of words with which it is commonly confused. Example confusion sets include: {principle, principal}, {then, than}, {to,two,too}, and {weather,whether}.

Numerous methods have been presented for confusable disambiguation. The more recent set of techniques includes multiplicative weight-update algorithms (Golding and Roth, 1998), latent semantic analysis (Jones and Martin, 1997), transformation-based learning (Mangu and Brill, 1997), differential grammars (Powers,

1997), decision lists (Yarowsky, 1994), and a variety of Bayesian classifiers (Gale et al., 1993, Golding, 1995, Golding and Schabes, 1996). In all of these approaches, the problem is formulated as follows: Given a specific confusion set (e.g. {to,two,too}), all occurrences of confusion set members in the test set are replaced by a marker; everywhere the system sees this marker, it must decide which member of the confusion set to choose.

Confusion set disambiguation is one of a class of natural language problems involving disambiguation from a relatively small set of alternatives based upon the string context in which the ambiguity site appears. Other such problems include word sense disambiguation, part of speech tagging and some formulations of phrasal chunking. One advantageous aspect of confusion set disambiguation, which allows us to study the effects of large data sets on performance, is that labeled training data is essentially free, since the correct answer is surface apparent in any collection of reasonably well-edited text.

3 Learning Curve Experiments

This work was partially motivated by the desire to develop an improved grammar checker. Given a fixed amount of time, we considered what would be the most effective way to focus our efforts in order to attain the greatest performance improvement. Some possibilities included modifying standard learning algorithms, exploring new learning techniques, and using more sophisticated features. Before exploring these somewhat expensive paths, we decided to first see what happened if we simply trained an existing method with much more data. This led to the exploration of learning curves for various machine learning algorithms: winnow[1], perceptron, naïve Bayes, and a very simple memory-based learner. For the first three learners, we used the standard collection of features employed for this problem: the set of words within a window of the target word, and collocations containing words and/or parts of speech. The memory-based learner used only the word before and word after as features.

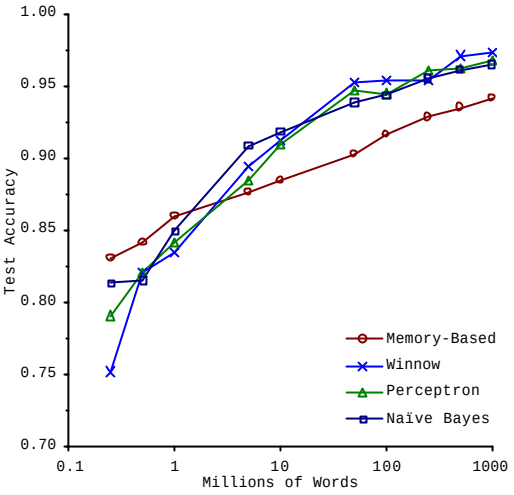

Figure 1. Learning Curves for Confusion Set Disambiguation

We collected a 1-billion-word training corpus from a variety of English texts, including news articles, scientific abstracts, government transcripts, literature and other varied forms of prose. This training corpus is three orders of magnitude greater than the largest training corpus previously used for this problem. We used 1 million words of Wall Street Journal text as our test set, and no data from the Wall Street Journal was used when constructing the training corpus. Each learner was trained at several cutoff points in the training corpus, i.e. the first one million words, the first five million words, and so on, until all one billion words were used for training. In order to avoid training biases that may result from merely concatenating the different data sources to form a larger training corpus, we constructed each consecutive training corpus by probabilistically sampling sentences from the different sources weighted by the size of each source.

In Figure 1, we show learning curves for each learner, up to one billion words of training data. Each point in the graph is the average performance over ten confusion sets for that size training corpus. Note that the curves appear to be log-linear even out to one billion words.

Of course for many problems, additional training data has a non-zero cost. However,

[1] Thanks to Dan Roth for making both Winnow and Perceptron available.

these results suggest that we may want to reconsider the trade-off between spending time and money on algorithm development versus spending it on corpus development. At least for the problem of confusable disambiguation, none of the learners tested is close to asymptoting in performance at the training corpus size commonly employed by the field.

Such gains in accuracy, however, do not come for free. Figure 2 shows the size of learned representations as a function of training data size. For some applications, this is not necessarily a concern. But for others, where space comes at a premium, obtaining the gains that come with a billion words of training data may not be viable without an effort made to compress information. In such cases, one could look at numerous methods for compressing data (e.g. Dagan and Engleson, 1995, Weng, et al, 1998).

4 The Efficacy of Voting

Voting has proven to be an effective technique for improving classifier accuracy for many applications, including part-of-speech tagging (van Halteren, et al, 1998), parsing (Henderson and Brill, 1999), and word sense disambiguation (Pederson, 2000). By training a set of classifiers on a single training corpus and then combining their outputs in classification, it is often possible to achieve a target accuracy with less labeled training data than would be needed if only one classifier was being used. Voting can be effective in reducing both the bias of a particular training corpus and the bias of a specific learner. When a training corpus is very small, there is much more room for these biases to surface and therefore for voting to be effective. But does voting still offer performance gains when classifiers are trained on much larger corpora?

The complementarity between two learners was defined by Brill and Wu (1998) in order to quantify the percentage of time when one system is wrong, that another system is correct, and therefore providing an upper bound on combination accuracy. As training size increases significantly, we would expect complementarity between classifiers to decrease. This is due in part to the fact that a larger training corpus will reduce the data set variance and any bias arising from this. Also, some of the differences between classifiers might be due to how they handle a sparse training set.

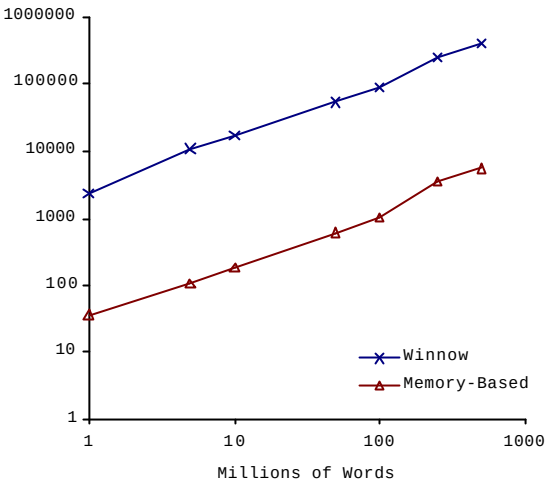

Figure 2. Representation Size vs. Training Corpus Size

As a result of comparing a sample of two learners as a function of increasingly large training sets, we see in Table 1 that complementarity does indeed decrease as training size increases.

Training Size (words)	Complementarity(L1,L2)
10^6	0.2612
10^7	0.2410
10^8	0.1759
10^9	0.1612

Table 1. Complementarity

Next we tested whether this decrease in complementarity meant that voting loses its effectiveness as the training set increases. To examine the impact of voting when using a significantly larger training corpus, we ran 3 out of the 4 learners on our set of 10 confusable pairs, excluding the memory-based learner. Voting was done by combining the normalized score each learner assigned to a classification choice. In Figure 3, we show the accuracy obtained from voting, along with the single best learner accuracy at each training set size. We see that for very small corpora, voting is beneficial, resulting in better performance than any single classifier. Beyond 1 million words, little is gained by voting, and indeed on the

largest training sets voting actually hurts accuracy.

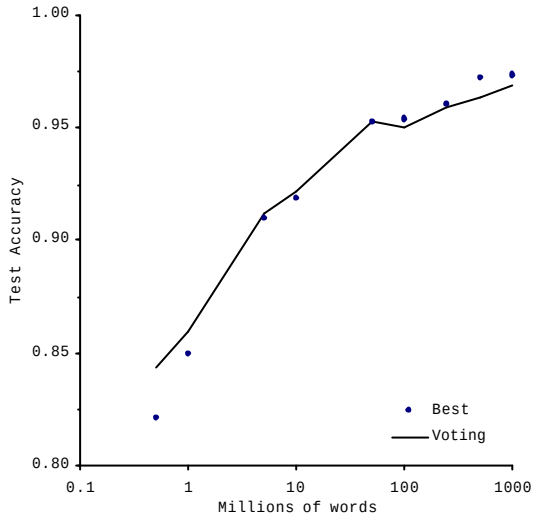

Figure 3. Voting Among Classifiers

5 When Annotated Data Is Not Free

While the observation that learning curves are not asymptoting even with orders of magnitude more training data than is currently used is very exciting, this result may have somewhat limited ramifications. Very few problems exist for which annotated data of this size is available for free. Surely we cannot reasonably expect that the manual annotation of one billion words along with corresponding parse trees will occur any time soon (but see (Banko and Brill 2001) for a discussion that this might not be completely infeasible). Despite this pitfall, there are techniques one can use to try to obtain the benefits of considerably larger training corpora without incurring significant additional costs. In the sections that follow, we study two such solutions: active learning and unsupervised learning.

5.1 Active Learning

Active learning involves intelligently selecting a portion of samples for annotation from a pool of as-yet unannotated training samples. Not all samples in a training set are equally useful. By concentrating human annotation efforts on the samples of greatest utility to the machine learning algorithm, it may be possible to attain better performance for a fixed annotation cost than if samples were chosen randomly for human annotation.

Most active learning approaches work by first training a seed learner (or family of learners) and then running the learner(s) over a set of unlabeled samples. A sample is presumed to be more useful for training the more uncertain its classification label is. Uncertainty can be judged by the relative weights assigned to different labels by a single classifier (Lewis and Catlett, 1994). Another approach, committee-based sampling, first creates a committee of classifiers and then judges classification uncertainty according to how much the learners differ among label assignments. For example, Dagan and Engelson (1995) describe a committee-based sampling technique where a part of speech tagger is trained using an annotated seed corpus. A family of taggers is then generated by randomly permuting the tagger probabilities, and the disparity among tags output by the committee members is used as a measure of classification uncertainty. Sentences for human annotation are drawn, biased to prefer those containing high uncertainty instances.

While active learning has been shown to work for a number of tasks, the majority of active learning experiments in natural language processing have been conducted using very small seed corpora and sets of unlabeled examples. Therefore, we wish to explore situations where we have, or can afford, a non-negligible sized training corpus (such as for part-of-speech tagging) and have access to very large amounts of unlabeled data.

We can use bagging (Breiman, 1996), a technique for generating a committee of classifiers, to assess the label uncertainty of a potential training instance. With bagging, a variant of the original training set is constructed by randomly sampling sentences with replacement from the source training set in order to produce N new training sets of size equal to the original. After the N models have been trained and run on the same test set, their classifications for each test sentence can be compared for classification agreement. The higher the disagreement between classifiers, the more useful it would be to have an instance

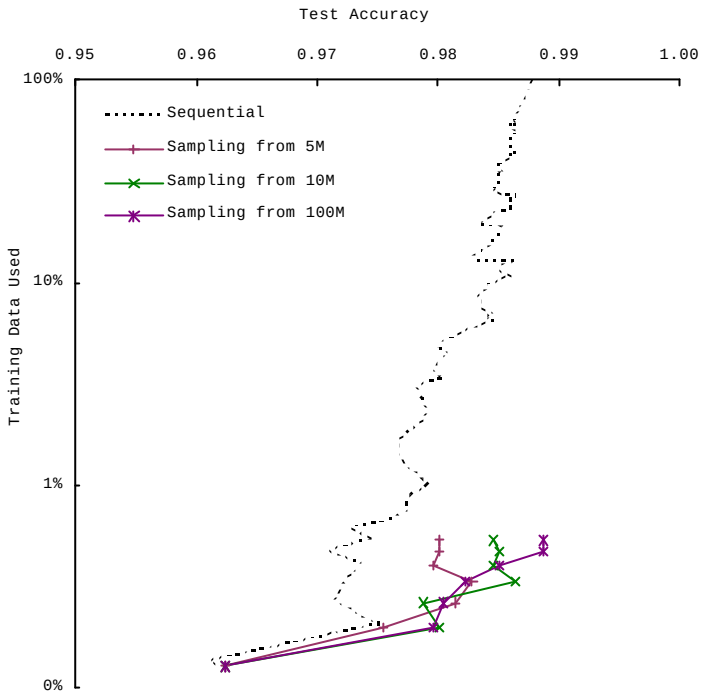

Figure 4. Active Learning with Large Corpora

manually labeled.

We used the naïve Bayes classifier, creating 10 classifiers each trained on bags generated from an initial one million words of labeled training data. We present the active learning algorithm we used below.

Initialize: Training data consists of X words correctly labeled

Iterate:

1) Generate a committee of classifiers using bagging on the training set

2) Run the committee on unlabeled portion of the training set
3) Choose M instances from the unlabeled set for labeling - pick the M/2 with the greatest vote entropy and then pick another M/2 randomly – and add to training set

---

We initially tried selecting the M most uncertain examples, but this resulted in a sample too biased toward the difficult instances. Instead we pick half of our samples for annotation randomly and the other half from those whose labels we are most uncertain of, as judged by the entropy of the votes assigned to the instance by the committee. This is, in effect, biasing our sample toward instances the classifiers are most uncertain of.

We show the results from sample selection for confusion set disambiguation in Figure 4. The line labeled "sequential" shows test set accuracy achieved for different percentages of the one billion word training set, where training instances are taken at random. We ran three active learning experiments, increasing the size of the total unlabeled training corpus from which we can pick samples to be annotated. In all three cases, sample selection outperforms sequential sampling. At the endpoint of each training run in the graph, the same number of samples has been annotated for training. However, we see that the larger the pool of candidate instances for annotation is, the better the resulting accuracy. By increasing the pool of unlabeled training instances for active learning, we can improve accuracy with only a fixed additional annotation cost. Thus it is possible to benefit from the availability of extremely large corpora without incurring the full costs of annotation, training time, and representation size.

5.2 Weakly Supervised Learning

While the previous section shows that we can benefit from substantially larger training corpora without needing significant additional manual annotation, it would be ideal if we could improve classification accuracy using only our seed annotated corpus and the large unlabeled corpus, without requiring any additional hand labeling. In this section we turn to unsupervised learning in an attempt to achieve this goal. Numerous approaches have been explored for exploiting situations where some amount of annotated data is available and a much larger amount of data exists unannotated, e.g. Marialdo's HMM part-of-speech tagger training (1994), Charniak's parser retraining experiment (1996), Yarowsky's seeds for word sense disambiguation (1995) and Nigam et al's (1998) topic classifier learned in part from unlabelled documents. A nice discussion of this general problem can be found in Mitchell (1999).

The question we want to answer is whether there is something to be gained by combining unsupervised and supervised learning when we scale up both the seed corpus and the unlabeled corpus significantly. We can again use a committee of bagged classifiers, this time for unsupervised learning. Whereas with active learning we want to choose the most uncertain instances for human annotation, with unsupervised learning we want to choose the instances that have the highest probability of being correct for automatic labeling and inclusion in our labeled training data.

In Table 2, we show the test set accuracy (averaged over the four most frequently occurring confusion pairs) as a function of the number of classifiers that agree upon the label of an instance. For this experiment, we trained a collection of 10 naïve Bayes classifiers, using bagging on a 1-million-word seed corpus. As can be seen, the greater the classifier agreement, the more likely it is that a test sample has been correctly labeled.

Classifiers In Agreement	Test Accuracy
10	0.8734
9	0.6892
8	0.6286
7	0.6027
6	0.5497
5	0.5000

Table 2. Committee Agreement vs. Accuracy

Since the instances in which all bags agree have the highest probability of being correct, we attempted to automatically grow our labeled training set using the 1-million-word labeled seed corpus along with the collection of naïve Bayes classifiers described above. All instances from the remainder of the corpus on which all 10 classifiers agreed were selected, trusting the agreed-upon label. The classifiers were then retrained using the labeled seed corpus plus the new training material collected automatically during the previous step.

In Table 3 we show the results from these unsupervised learning experiments for two confusion sets. In both cases we gain from unsupervised training compared to using only the seed corpus, but only up to a point. At this point, test set accuracy begins to decline as additional training instances are automatically harvested. We are able to attain improvements in accuracy for free using unsupervised learning, but unlike our learning curve experiments using correctly labeled data, accuracy does not continue to improve with additional data.

	{then, than}		{among, between}	
	Test Accuracy	% Total Training Data	Test Accuracy	% Total Training Data
10^6-wd labeled seed corpus	0.9624	0.1	0.8183	0.1
seed+$5x10^6$ wds, unsupervised	0.9588	0.6	0.8313	0.5
seed+10^7 wds, unsupervised	0.9620	1.2	0.8335	1.0
seed+10^8 wds, unsupervised	0.9715	12.2	0.8270	9.2
seed+$5x10^8$ wds, unsupervised	0.9588	61.1	0.8248	42.9
10^9 wds, supervised	0.9878	100	0.9021	100

Table 3. Committee-Based Unsupervised Learning

Charniak (1996) ran an experiment in which he trained a parser on one million words of parsed data, ran the parser over an additional 30 million words, and used the resulting parses to reestimate model probabilities. Doing so gave a small improvement over just using the manually parsed data. We repeated this experiment with our data, and show the outcome in Table 4. Choosing only the labeled instances most likely to be correct as judged by a committee of classifiers results in higher accuracy than using all instances classified by a model trained with the labeled seed corpus.

	Unsupervised: All Labels	Unsupervised: Most Certain Labels
	{then, than}	
10^7 words	0.9524	0.9620
10^8 words	0.9588	0.9715
$5x10^8$ words	0.7604	0.9588
	{among, between}	
10^7 words	0.8259	0.8335
10^8 words	0.8259	0.8270
$5x10^8$ words	0.5321	0.8248

Table 4. Comparison of Unsupervised Learning Methods

In applying unsupervised learning to improve upon a seed-trained method, we consistently saw an improvement in performance followed by a decline. This is likely due to eventually having reached a point where the gains from additional training data are offset by the sample bias in mining these instances. It may be possible to combine active learning with unsupervised learning as a way to reduce this sample bias and gain the benefits of both approaches.

6 Conclusions

In this paper, we have looked into what happens when we begin to take advantage of the large amounts of text that are now readily available. We have shown that for a prototypical natural language classification task, the performance of learners can benefit significantly from much larger training sets. We have also shown that both active learning and unsupervised learning can be used to attain at least some of the advantage that comes with additional training data, while minimizing the cost of additional human annotation. We propose that a logical next step for the research community would be to direct efforts towards increasing the size of annotated training collections, while deemphasizing the focus on comparing different learning techniques trained only on small training corpora. While it is encouraging that there is a vast amount of on-line text, much work remains to be done if we are to learn how best to exploit this resource to improve natural language processing.

References

Banko, M. and Brill, E. (2001). *Mitigating the Paucity of Data Problem.* Human Language Technology.

Breiman L., (1996). *Bagging Predictors*, Machine Learning 24 123-140.

Brill, E. and Wu, J. (1998). *Classifier combination for improved lexical disambiguation*. In Proceedings of the 17th International Conference on Computational Linguistics.

Charniak, E. (1996). *Treebank Grammars*, Proceedings AAAI-96 , Menlo Park, Ca.

Dagan, I. and Engelson, S. (1995). *Committee-based sampling for training probabilistic classifiers*. In Proc. ML-95, the 12th Int. Conf. on Machine Learning.

Gale, W. A., Church, K. W., and Yarowsky, D. (1993). *A method for disambiguating word senses in a large corpus.* Computers and the Humanities, 26:415--439.

Golding, A. R. (1995). *A Bayesian hybrid method for context-sensitive spelling correction*. In Proc. 3rd Workshop on Very Large Corpora, Boston, MA.

Golding, A. R. and Roth, D.(1999), *A Winnow-Based Approach to Context-Sensitive Spelling Correction*. Machine Learning, 34:107--130.

Golding, A. R. and Schabes, Y. (1996). *Combining trigram-based and feature-based methods for context-sensitive spelling correction*. In Proc. 34th Annual Meeting of the Association for Computational Linguistics, Santa Cruz, CA.

Henderson, J. C. and Brill, E. (1999). *Exploiting diversity in natural language processing: combining parsers*. In 1999 Joint Sigdat Conference on Empirical Methods in Natural Language Processing and Very Large Corpora. ACL, New Brunswick NJ. 187-194.

Jones, M. P. and Martin, J. H. (1997). *Contextual spelling correction using latent semantic analysis*.

In Proc. 5th Conference on Applied Natural Language Processing, Washington, DC.

Lewis, D. D., & Catlett, J. (1994). *Heterogeneous uncertainty sampling*. Proceedings of the Eleventh International Conference on Machine Learning (pp. 148--156). New Brunswick, NJ: Morgan Kaufmann.

Mangu, L. and Brill, E. (1997). *Automatic rule acquisition for spelling correction*. In Proc. 14th International Conference on Machine Learning. Morgan Kaufmann.

Merialdo, B. (1994). *Tagging English text with a probabilistic model*. Computational Linguistics, 20(2):155--172.

Mitchell, T. M. (1999), *The role of unlabeled data in supervised learning*, in Proceedings of the Sixth International Colloquium on Cognitive Science, San Sebastian, Spain.

Nigam, N., McCallum, A., Thrun, S., and Mitchell, T. (1998). *Learning to classify text from labeled and unlabeled documents*. In Proceedings of the Fifteenth National Conference on Artificial Intelligence. AAAI Press..

Pedersen, T. (2000). *A simple approach to building ensembles of naive bayesian classifiers for word sense disambiguation*. In Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics May 1-3, 2000, Seattle, WA

Powers, D. (1997). *Learning and application of differential grammars*. In Proc. Meeting of the ACL Special Interest Group in Natural Language Learning, Madrid.

van Halteren, H. Zavrel, J. and Daelemans, W. (1998). *Improving data driven wordclass tagging by system combination*. In COLING-ACL'98, pages 491497, Montreal, Canada.

Weng, F., Stolcke, A, & Sankar, A (1998). *Efficient lattice representation and generation*. Proc. Intl. Conf. on Spoken Language Processing, vol. 6, pp. 2531-2534. Sydney, Australia.

Yarowsky, D. (1994). *Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French*. In Proc. 32nd Annual Meeting of the Association for Computational Linguistics, Las Cruces, NM.

Yarowsky, D. (1995) *Unsupervised word sense disambiguation rivaling supervised methods*. In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics. Cambridge, MA, pp. 189-196, 1995.