Active Learning for Part-of-Speech Tagging: Accelerating Corpus Annotation

Eric Ringger*, Peter McClanahan*, Robbie Haertel*, George Busby*, Marc Carmen, James Carroll*, Kevin Seppi*, Deryle Lonsdale****

*Computer Science Department; **Linguistics Department
Brigham Young University
Provo, Utah, USA 84602

Abstract

In the construction of a part-of-speech annotated corpus, we are constrained by a fixed budget. A fully annotated corpus is required, but we can afford to label only a subset. We train a Maximum Entropy Markov Model tagger from a labeled subset and automatically tag the remainder. This paper addresses the question of where to focus our manual tagging efforts in order to deliver an annotation of highest quality. In this context, we find that active learning is always helpful. We focus on Query by Uncertainty (QBU) and Query by Committee (QBC) and report on experiments with several baselines and new variations of QBC and QBU, inspired by weaknesses particular to their use in this application. Experiments on English prose and poetry test these approaches and evaluate their robustness. The results allow us to make recommendations for both types of text and raise questions that will lead to further inquiry.

1 Introduction

We are operating (as many do) on a fixed budget and need annotated text in the context of a larger project. We need a fully annotated corpus but can afford to annotate only a subset. To address our budgetary constraint, we train a model from a manually annotated subset of the corpus and automatically annotate the remainder. At issue is where to focus manual annotation efforts in order to produce a complete annotation of highest possible quality. A follow-up question is whether these techniques work equally well on different types of text.

In particular, we require part-of-speech (POS) annotations. In this paper we employ a state-of-the-art tagger on both prose and poetry, and we examine multiple known and novel active learning (or sampling) techniques in order to determine which work best in this context. We show that the results obtained by a state-of-the-art tagger trained on a small portion of the data selected through active learning can approach the accuracy attained by human annotators and are on par with results from exhaustively trained automatic taggers.

In a study based on English language data presented here, we identify several active learning techniques and make several recommendations that we hope will be portable for application to other text types and to other languages. In section 2 we briefly review the state of the art approach to POS tagging. In section 3, we survey the approaches to active learning employed in this study, including variations on commonly known techniques. Section 4 introduces the experimental regime and presents results and their implications. Section 5 draws conclusions and identifies opportunities for follow-up research.

2 Part of Speech Tagging

Labeling natural language data with part-of-speech tags can be a complicated task, requiring much effort and expense, even for trained annotators. Several efforts, notably the Alembic workbench (Day et al., 1997) and similar tools, have provided interfaces to aid annotators in the process.

Automatic POS tagging of text using probabilistic models is mostly a solved problem but requires supervised learning from substantial amounts of training data. Previous work demonstrates the suitability of Hidden Markov Models for POS tagging (Kupiec, 1992; Brants, 2000). More recent work has achieved state-of-the-art results with Maxi-

Proceedings of the Linguistic Annotation Workshop, pages 101–108,
Prague, June 2007.

mum entropy conditional Markov models (MaxEnt CMMs, or MEMMs for short) (Ratnaparkhi, 1996; Toutanova & Manning, 2000; Toutanova et al., 2003). Part of the success of MEMMs can be attributed to the absence of independence assumptions among predictive features and the resulting ease of feature engineering. To the best of our knowledge, the present work is the first to present results using MEMMs in an active learning framework.

An MEMM is a probabilistic model for sequence labeling. It is a Conditional Markov Model (CMM as illustrated in Figure 1) in which a Maximum Entropy (MaxEnt) classifier is employed to estimate the probability distribution $p(t_i \mid \underline{w}, \underline{t}_{1..i-1}) \approx p_{ME}(t_i \mid w_i, \underline{f_i}, t_{i-1}, t_{i-2})$ over possible labels t_i for each element in the sequence—in our case, for each word w_i in a sentence $\underline{w}$. The MaxEnt model is trained from labeled data and has access to any predefined attributes (represented here by the collection $\underline{f_i}$) of the entire word sequence and to the labels of previous words ($\underline{t}_{1..i-1}$). Our implementation employs an order-two Markov assumption so the classifier has access only to the two previous tags t_{i-1}, t_{i-2}. We refer to the features $(w_i, \underline{f_i}, t_{i-1}, t_{i-2})$ from which the classifier predicts the distribution over tags as "the local trigram context".

A Viterbi decoder is a dynamic programming algorithm that applies the MaxEnt classifier to score multiple competing tag-sequence hypotheses efficiently and to produce the best tag sequence, according to the model. We approximate Viterbi very closely using a fast beam search. Essentially, the decoding process involves sequential classification, conditioned on the (uncertain) decisions of the previous local trigram context classifications. The chosen tag sequence $\hat{\underline{t}}$ is the tag sequence maximizing the following quantity:

$$\begin{aligned}\hat{\underline{t}} &= \arg\max_{\underline{t}} P(\underline{t} \mid \underline{w}) \\ &= \arg\max_{\underline{t}} \prod_{i=1..n} p_{ME}(t_i \mid w_i, \underline{f_i}, t_{i-1}, t_{i-2})\end{aligned}$$

The features used in this work are reasonably typical for modern MEMM feature-based POS tagging and consist of a combination of lexical, orthographic, contextual, and frequency-based information. In particular, for each word the following features are defined: the textual form of the word itself, the POS tags of the preceding two words, and the textual form of the following word. Following Toutanova and Manning (2000) approximately, more information is defined for words that are considered rare (which we define here as words that occur fewer than fifteen times). We consider the tagger to be near-state-of-the-art in terms of tagging accuracy.

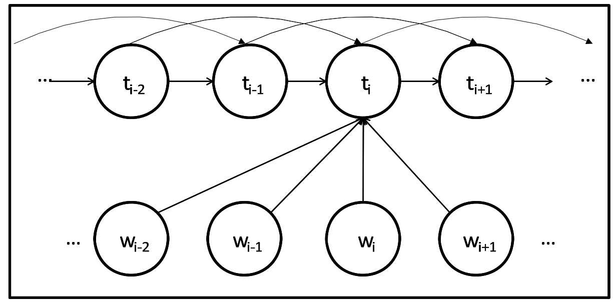

Figure 1. Simple Markov order 2 CMM, with focus on the i-th hidden label (or tag).

3 Active Learning

The objective of this research is to produce more high quality annotated data with less human annotator time and effort. Active learning is an approach to machine learning in which a model is trained with the selective help of an oracle. The oracle provides labels on a sufficient number of "tough" cases, as identified by the model. Easy cases are assumed to be understood by the model and to require no additional annotation by the oracle. Many variations have been proposed in the broader active learning and decision theory literature under many names, including "active sampling" and "optimal sampling."

In active learning for POS tagging, as in other applications, the oracle can be a human. For experimental purposes, a human oracle is simulated using pre-labeled data, where the labels are hidden until queried. To begin, the active learning process requires some small amount of training data to seed the model. The process proceeds by identifying the data in the given corpus that should be tagged first for maximal impact.

3.1 Active Learning in the Language Context

When considering the role of active learning, we were initially drawn to the work in active learning for classification. In a simple configuration, each instance (document, image, etc.) to be labeled can be considered to be independent. However, for active learning for the POS tagging problem we considered the nature of human input as an oracle for the task. As an approximation, people read sentences as propositional atoms, gathering contextual cues from the sentence in order to assemble the meaning of the whole. Consequently, we thought it unreasonable to choose the word as the granularity for active learning. Instead, we begin with the assumption that a human will usually require much of the sentence or at least local context from the sentence in order to label a single word with its POS label. While focusing on a single word, the human may as well label the entire sentence or at least correct the labels assigned by the tagger for the sentence. Consequently, the sentence is the granularity of annotation for this work. (Future work will question this assumption and investigate tagging a word or a subsequence of words at a time.) This distinguishes our work from active learning for classification since labels are not drawn from a fixed set of labels. Rather, every sentence of length n can be labeled with a tag sequence drawn from a set of size T^n, where T is the size of the per-word tag set. Granted, many of the options have very low probability.

To underscore our choice of annotating at the granularity of a sentence, we also note that a maximum entropy classifier for isolated word tagging that leverages attributes of neighboring words—but is blind to all tags—will underperform an MEMM that includes the tags of neighboring words (usually on the left) among its features. Previous experiments demonstrate the usefulness of tags in context on the standard Wall Street Journal data from the Penn Treebank (Marcus et al., 1999). A MaxEnt isolated word tagger achieves 93.7% on words observed in the training set and 82.6% on words unseen in the training set. Toutanova and Manning (2000) achieves 96.9% (on seen) and 86.9% (on unseen) with an MEMM. They surpassed their earlier work in 2003 with a "cyclic dependency network tagger", achieving 97.2%/89.05% (seen/unseen) (Toutanova et al., 2003). The generally agreed upon upper bound is around 98%, due to label inconsistencies in the Treebank. The main point is that effective use of contextual features is necessary to achieve state of the art performance in POS tagging.

In active learning, we employ several sets of data that we refer to by the following names:

- <u>Initial Training</u>: the small set of data used to train the original model before active learning starts
- <u>Training</u>: data that has already been labeled by the oracle as of step i in the learning cycle
- <u>Unannotated</u>: data not yet labeled by the oracle as of step i
- <u>Test</u> (specifically Development Test): labeled data used to measure the accuracy of the model at each stage of the active learning process. Labels on this set are held in reserve for comparison with the labels chosen by the model. It is the accuracy on this set that we report in our experimental results in Section 4.

Note that the Training set grows at the expense of the Unannotated set as active learning progresses.

Active Learning for POS Tagging consists of the following steps:

1. Train a model with Initial Training data
2. Apply model to Unannotated data
3. Compute potential informativeness of each sentence
4. Remove top n sentences with most potential informativeness from Unannotated data and give to oracle
5. Add n sentences annotated (or corrected) by the oracle to Training data
6. Retrain model with Training data
7. Return to step 2 until stopping condition is met.

There are several possible stopping conditions, including reaching a quality bar based on accuracy on the Test set, the rate of oracle error corrections in the given cycle, or even the cumulative number of oracle error corrections. In practice, the exhaustion of resources, such as time or money, may completely dominate all other desirable stopping conditions.

Several methods are available for determining which sentences will provide the most information. Expected Value of Sample Information (EVSI) (Raiffa & Schlaiffer, 1967) would be the optimal approach from a decision theoretic point of view, but it is computationally prohibitive and is not considered here. We also do not consider the related notion of query-by-model-improvement or other methods (Anderson & Moore, 2005; Roy & McCallum, 2001a, 2001b). While worth exploring, they do not fit in the context of this current work and should be considered in future work. We focus here on the more widely used Query by Committee (QBC) and Query by Uncertainty (QBU), including our new adaptations of these.

Our implementation of maximum entropy training employs a convex optimization procedure known as LBFGS. Although this procedure is relatively fast, training a model (or models in the case of QBC) from scratch on the training data during every round of the active learning loop would prolong our experiments unnecessarily. Instead we start each optimization search with a parameter set consisting of the model parameters from the previous iteration of active learning (we call this "Fast MaxEnt"). In practice, this converges quickly and produces equivalent results.

3.2 Query by Committee

Query by Committee (QBC) was introduced by Seung, Opper, and Sompolinsky (1992). Freund, Seung, Shamir, and Tishby (1997) provided a careful analysis of the approach. Engelson and Dagan (1996) experimented with QBC using HMMs for POS tagging and found that selective sampling of sentences can significantly reduce the number of samples required to achieve desirable tag accuracies. Unlike the present work, Engelson & Dagan were restricted by computational resources to selection from small windows of the Unannotated set, not from the entire Unannotated set. Related work includes learning ensembles of POS taggers, as in the work of Brill and Wu (1998), where an ensemble consisting of a unigram model, an N-gram model, a transformation-based model, and an MEMM for POS tagging achieves substantial results beyond the individual taggers. Their conclusion relevant to this paper is that different taggers commit complementary errors, a useful fact to exploit in active learning. QBC employs a committee of *N* models, in which each model votes on the correct tagging of a sentence. The potential informativeness of a sentence is measured by the total number of tag sequence disagreements (compared pair-wise) among the committee members. Possible variants of QBC involve the number of committee members, how the training data is split among the committee members, and whether the training data is sampled with or without replacement.

A potential problem with QBC in this application is that words occur with different frequencies in the corpus. Because of the potential for greater impact across the corpus, querying for the tag of a more frequent word may be more desirable than querying for the tag of a word that occurs less frequently, even if there is greater disagreement on the tags for the less frequent word. We attempted to compensate for this by weighting the number of disagreements by the corpus frequency of the word in the full data set (Training and Unannotated). Unfortunately, this resulted in worse performance; solving this problem is an interesting avenue for future work.

3.3 Query by Uncertainty

The idea behind active sampling based on uncertainty appears to originate with Thrun and Moeller (1992). QBU has received significant attention in general. Early experiments involving QBU were conducted by Lewis and Gale (1994) on text classification, where they demonstrated significant benefits of the approach. Lewis and Catlett (1994) examined its application for non-probabilistic learners in conjunction with other probabilistic learners under the name "uncertainty sampling." Brigham Anderson (2005) explored QBU using HMMs and concluded that it is sometimes advantageous. We are not aware of any published work on the application of QBU to POS tagging. In our implementation, QBU employs a single MEMM tagger. The MaxEnt model comprising the tagger can assess the probability distribution over tags for any word in its local trigram context, as illustrated in the example in Figure 2.

			NN	*0 .85*
			VB	*0.13*
			...	
RB	**DT**	**JJS**	*CD*	*2.0E-7*
Perhaps	**the**	**biggest**	**hurdle**	…

Figure 2. Distribution over tags for the word "hurdle" in italics. The local trigram context is in boldface.

In Query by Uncertainty (QBU), the informativeness of a sample is assumed to be the uncertainty in the predicted distribution over tags for that sample, that is the entropy of $p_{ME}(t_i \mid w_i, \underline{f}_i, t_{i-1}, t_{i-2})$. To determine the potential informativeness of a word, we can measure the entropy in that distribution. Since we are selecting sentences, we must extend our measure of uncertainty beyond the word.

3.4 Adaptations of QBU

There are several problems with the use of QBU in this context:

- Some words are more important; i.e., they contain more information perhaps because they occur more frequently.

- MaxEnt estimates per-word distributions over tags, not per-sentence distributions over tag sequences.
- Entropy computations are relatively costly.

We address the first issue in a new version of QBU which we call "Weighted Query by Uncertainty" (WQBU). In WQBU, per-word uncertainty is weighted by the word's corpus frequency.

To address the issue of estimating per-sentence uncertainty from distributions over tag *sequences*, we have considered several different approaches. The per-word (conditional) entropy is defined as follows:

$$H(T_i \mid w_i, \underline{f_i}, t_{i-1}, t_{i-2}) = -\sum_{t_i \in Tagset} p_{ME}(t_i \mid w_i, \underline{f_i}, t_{i-1}, t_{i-2}) \cdot \log p_{ME}(t_i \mid w_i, \underline{f_i}, t_{i-1}, t_{i-2})$$

where T_i is the random variable for the tag t_i on word w_i, and the features of the context in which w_i occurs are denoted, as before, by the collection $\underline{f_i}$ and the prior tags t_{i-1}, t_{i-2}. It is straightforward to calculate this entropy for each word in a sentence from the Unannotated set, if we assume that previous tags t_{i-1}, t_{i-2} are from the Viterbi (best) tag sequence (for the entire sentence) according to the model.

For an entire sentence, we estimate the tag-sequence entropy by summing over all possible tag sequences. However, computing this estimate exactly on a 25-word sentence, where each word can be labeled with one of 35 tags, would require $35^{25} = 3.99 \cdot 10^{38}$ steps. Instead, we approximate the per-sentence tag sequence distribution entropy by summing per-word entropy:

$$\hat{H}(\underline{T} \mid \underline{w}) \approx -\sum_{w_i \in \underline{w}} H(T_i \mid w_i, \underline{f_i}, t_{i-1}, t_{i-2})$$

This is the approach we refer to as QBU in the experimental results section. We have experimented with a second approach that estimates the per-sentence entropy of the tag-sequence distribution by Monte Carlo decoding. Unfortunately, current active learning results involving this MC POS tagging decoder are negative on small Training set sizes, so we do not present them here. Another alternative approximation worth pursuing is computing the per-sentence entropy using the n-best POS tag sequences. Very recent work by Mann and McCallum (2007) proposes an approach in which exact sequence entropy can be calculated efficiently. Further experimentation is required to compare our approximation to these alternatives.

An alternative approach that eliminates the overhead of entropy computations entirely is to estimate per-sentence uncertainty with $1 - P(\hat{t})$, where $\hat{t}$ is the Viterbi (best) tag sequence. We call this scheme QBUV. In essence, it selects a sample consisting of the sentences having the highest probability that the Viterbi sequence is wrong. To our knowledge, this is a novel approach to active learning.

4 Experimental Results

In this section, we examine the experimental setup, the prose and poetry data sets, and the results from using the various active learning algorithms on these corpora.

4.1 Setup

The experiments focus on the annotation scenario posed earlier, in which budgetary constraints afford only some number *x* of sentences to be annotated. The *x*-axis in each graph captures the number of sentences. For most of the experiments, the graphs present accuracies on the (Development) Test set. Later in this section, we present results for an alternate metric, namely number of words corrected by the oracle.

In order to ascertain the usefulness of the active learning approaches explored here, the results are presented against a baseline in which sentences are selected randomly from the Unannotated set. We consider this baseline to represent the use of a state-of-the-art tagger trained on the same amount of data as the active learner. Due to randomization, the random baseline is actually distinct from experiment to experiment without any surprising deviations. Also, each result curve in each graph represents the average of three distinct runs.

Worth noting is that most of the graphs include active learning curves that are run to completion; namely, the rightmost extent of all curves represents the exhaustion of the Unannotated data. At this extreme point, active learning and random sample selection all have the same Training set. In the scenarios we are targeting, this far right side is not of interest. Points representing smaller amounts of annotated data are our primary interest.

In the experiments that follow, we address several natural questions that arise in the course of applying active learning. We also compare the variants of QBU and QBC. For QBC, committee members divide the training set (at each stage of the active learning process) evenly. All committee members and final models are MEMMs. Likewise, all variants of QBU employ MEMMs.

4.2 Data Sets

The experiments involve two data sets in search of conclusions that generalize over two very different kinds of English text. The first data set consists of English prose from the POS-tagged one-million-word Wall Street Journal text in the Penn Treebank (PTB) version 3. We use a random sample of the corpus constituting 25% of the traditional training set (sections 2–21). Initial Training data consists of 1% of this set. We employ section 24 as the Development Test set. Average sentence length is approximately 25 words.

Our second experimental set consists of English poetry from the British National Corpus (BNC) (Godbert & Ramsay, 1991; Hughes, 1982; Raine, 1984). The text is also fully tagged with 91 parts of speech from a different tag set than the one used for the PTB. The BNC XML data was taken from the files B1C.xml, CBO.xml, and H8R.xml. This results in a set of 60,056 words and 8,917 sentences.

4.3 General Results

To begin, each step in the active learning process adds a batch of 100 sentences from the Unannotated set at a time. Figure 3 demonstrates (using QBU) that the size of a query batch is not significant in these experiments.

The primary question to address is whether active learning helps or not. Figure 4 demonstrates that QBU, QBUV, and QBC all outperform the random baseline in terms of total, per-word accuracy on the Test set, given the same amount of Training data. Figure 5 is a close-up version of Figure 4, placing emphasis on points up to 1000 annotated sentences. In these figures, QBU and QBUV vie for the best performing active learning algorithm. These results appear to give some useful advice captured in Table 1. The first column in the table contains the starting conditions. The remaining columns indicate that for between 800-1600 sentences of annotation, QBUV takes over from QBU as the best selection algorithm.

The next question to address is how much initial training data should be used; i.e., when should we start using active learning? The experiment in Figure 6 demonstrates (using QBU) that one should use as little data as possible for Initial Training Data. There is always a significant advantage to starting early. In the experiment documented in

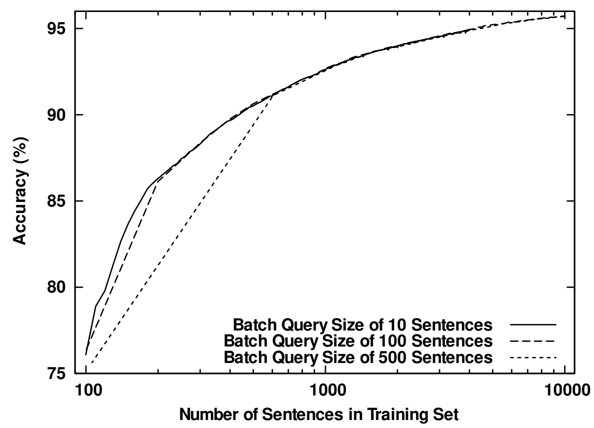

Figure 3. Varying the size of the query batch in active learning yields identical results after the first query batch.

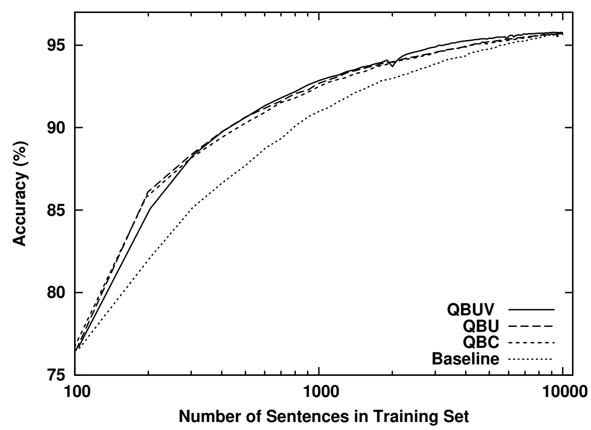

Figure 4. The best representatives of each type of active learner beat the baseline. QBU and QBUV trade off the top position over QBC and the Baseline.

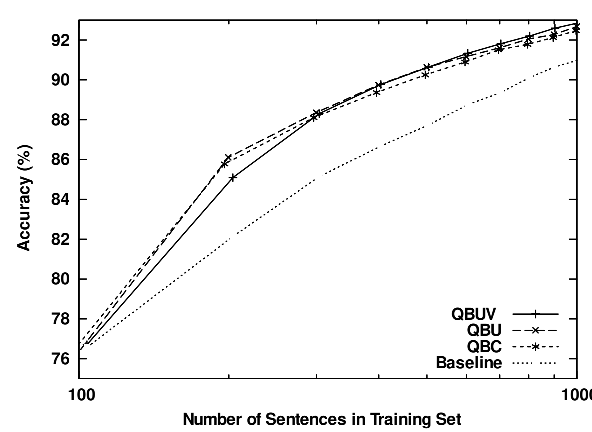

Figure 5. Close-up of the low end of the graph from Figure 4. QBUV and QBU are nearly tied for best performance.

this figure, a batch query size of one was employed in order to make the point as clearly as possible. Larger batch query sizes produce a graph with similar trends as do experiments involving larger Unannotated sets and other active learners.

	100	200	400	800	1600	3200	6400
QBU	76.26	**86.11**	**90.63**	**92.27**	93.67	94.65	95.42
QBUV	**76.65**	85.09	89.75	92.24	**93.72**	**94.96**	**95.60**
QBC	76.19	85.77	89.37	91.78	93.49	94.62	95.36
Base	76.57	82.13	86.68	90.12	92.49	94.02	95.19

Table 1. The best models (on PTB WSJ data) with various amounts of annotation (columns).

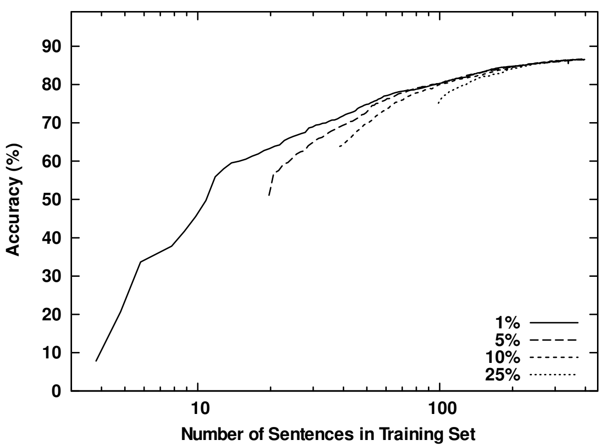

Figure 6. Start active learning as early as possible for a head start.

4.4 QBC Results

An important question to address for QBC is what number of committee members produces the best results? There was no significant difference in results from the QBC experiments when using between 3 and 7 committee members. For brevity we omit the graph.

4.5 QBU Results

For Query by Uncertainty, the experiment in Figure 7 demonstrates that QBU is superior to QBUV for low counts, but that QBUV slightly overtakes QBU beyond approximately 300 sentences. In fact, all QBU variants, including the weighted version, surpassed the baseline. WQBU has been omitted from the graph, as it was inferior to straightforward QBU.

4.6 Results on the BNC

Next we introduce results on poetry from the British National Corpus. Recall that the feature set employed by the MEMM tagger was optimized for performance on the Wall Street Journal. For the experiment presented in Figure 8, all data in the Training and Unannotated sets is from the BNC, but we employ the same feature set from the WSJ experiments. This result on the BNC data shows first of all that tagging poetry with this tagger leaves a final shortfall of approximately 8% from the WSJ results. Nonetheless and more importantly, the active learning trends observed on the WSJ still hold. QBC is better than the baseline, and QBU and QBUV trade off for first place. Furthermore, for low numbers of sentences, it is overwhelmingly to one's advantage to employ active learning for annotation.

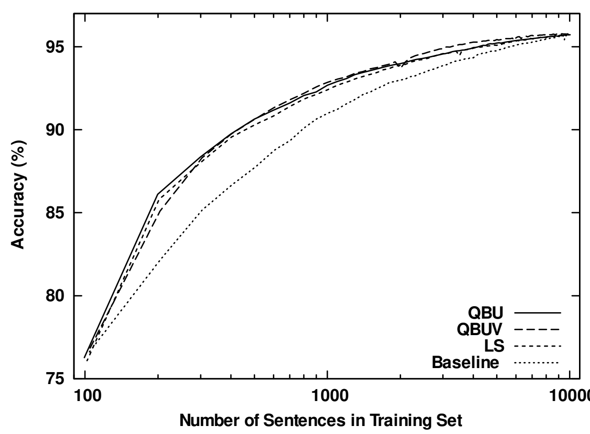

Figure 7. QBUV is superior to QBU overall, but QBU is better for very low counts. Both are superior to the random baseline and the Longest Sentence (LS) baseline.

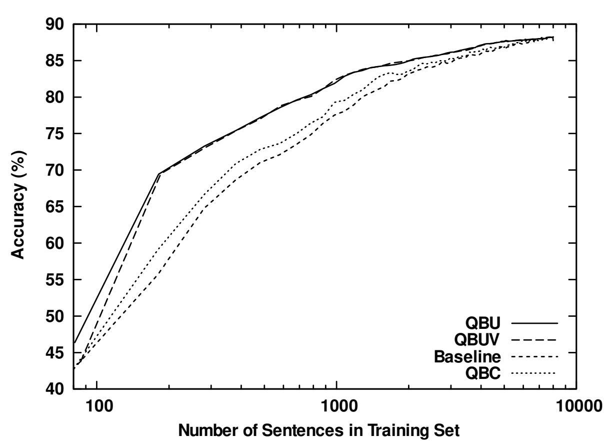

Figure 8. Active learning results on the BNC poetry data. Accuracy of QBUV, QBU, and QBC against the random baseline. QBU and QBUV are nearly indistinguishable.

4.7 Another Perspective

Next, briefly consider a different metric on the vertical axis. In Figure 9, the metric is the total number of words changed (corrected) by the oracle. This quantity reflects the cumulative number of differences between the tagger's hypothesis on a sentence (at the point in time when the oracle is queried) and the oracle's answer (over the training set). It corresponds roughly to the amount of time that would be required for a human annotator to correct the tags suggested by the model. This figure reveals that QBUV makes significantly more changes than QBU, QBC, or LS (the Longest Sentence baseline). Hence, the superiority of QBU over QBUV, as measured by this metric, appears to outweigh the small wins provided by QBUV when measured by accuracy alone. That said, the random baseline makes the fewest changes of all. If this metric (and not some combination with accuracy) were our only consideration, then active learning would appear not to serve our needs.

This metric is also a measure of how well a particular query algorithm selects sentences that especially require assistance from the oracle. In this sense, QBUV appears most effective.

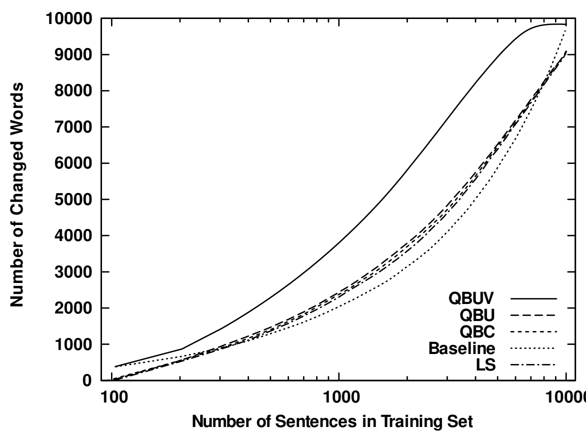

Figure 9. Cumulative number of corrections made by the oracle for several competitive active learning algorithms. QBU requires fewer corrections than QBUV.

5 Conclusions

Active learning is a viable way to accelerate the efficiency of a human annotator and is most effective when done as early as possible. We have presented state-of-the-art tagging results using a fraction of the labeled data. QBUV is a cheap approach to performing active learning, only to be surpassed by QBU when labeling small numbers of sentences.

We are in the midst of conducting a user study to assess the true costs of annotating a sentence at a time or a word at a time. We plan to incorporate these specific costs into a model of cost measured in time (or money) that will supplant the metrics reported here, namely accuracy and number of words corrected. As noted earlier, future work will also evaluate active learning at the granularity of a word or a subsequence of words, to be evaluated by the cost metric.

References

Anderson, B., and Moore, A. (2005). "Active Learning for HMM: Objective Functions and Algorithms." ICML, Germany.

Brants, T., (2000). "TnT -- a statistical part-of-speech tagger." ANLP, Seattle, WA.

Brill, E., and Wu, J. (1998). "Classifier combination for improved lexical disambiguation." Coling/ACL, Montreal, Quebec, Canada. Pp. 191-195.

Day, D., et al. (1997). "Mixed-Initiative Development of Language Processing Systems." ANLP, Washington, D.C.

Engelson, S. and Dagan, I. (1996). "Minimizing manual annotation cost in supervised training from corpora." ACL, Santa Cruz, California. Pp. 319-326.

Freund, Y., Seung, H., Shamir, E., and Tishby, N. (1997). "Selective sampling using the query by committee algorithm." Machine Learning, 28(2-3):133-168.

Godbert, G. and Ramsay, J. (1991). "For now." In the British National Corpus file B1C.xml. London: The Diamond Press (pp. 1-108).

Hughes, T. (1982). "Selected Poems." In the British National Corpus file H8R.xml. London: Faber & Faber Ltd. (pp. 35-235).

Kupiec, J. (1992). "Robust part-of-speech tagging using a hidden Markov model." Computer Speech and Language 6, pp. 225-242.

Lewis, D., and Catlett, J. (1994). "Heterogeneous uncertainty sampling for supervised learning." ICML.

Lewis, D., and Gale, W. (1995). "A sequential algorithm for training text classifiers: Corrigendum and additional data." SIGIR Forum, 29 (2), 13--19.

Mann, G., and McCallum, A. (2007). "Efficient Computation of Entropy Gradient for Semi-Supervised Conditional Random Fields". NAACL-HLT.

Marcus, M. et al. (1999). "Treebank-3." Linguistic Data Consortium, Philadelphia, PA.

Raiffa, H. and Schlaiffer, R. (1967). *Applied Statistical Decision Theory*. New York: Wiley Interscience.

Raine, C. (1984). "Rich." In the British National Corpus file CB0.xml. London: Faber & Faber Ltd. (pp. 13-101).

Ratnaparkhi, A. (1996). "A Maximum Entropy Model for Part-Of-Speech Tagging." EMNLP.

Roy, N., and McCallum, A. (2001a). "Toward optimal active learning through sampling estimation of error reduction." ICML.

Roy, N. and McCallum, A. (2001b). "Toward Optimal Active Learning through Monte Carlo Estimation of Error Reduction." ICML, Williamstown.

Seung, H., Opper, M., and Sompolinsky, H. (1992). "Query by committee". COLT. Pp. 287-294.

Thrun S., and Moeller, K. (1992). "Active exploration in dynamic environments." NIPS.

Toutanova, K., Klein, D., Manning, C., and Singer, Y. (2003). "Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network." HLT-NAACL. Pp. 252-259.

Toutanova, K. and Manning, C. (2000). "Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger." EMNLP, Hong Kong. Pp. 63-70.