Selective Sampling In Natural Language Learning

Ido Dagan **Sean P. Engelson**
Department of Mathematics and Computer Science
Bar-Ilan University
52900 Ramat Gan, Israel
{dagan,engelson}@bimacs.cs.biu.ac.il

Abstract

Many corpus-based methods for natural language processing are based on supervised training, requiring expensive manual annotation of training corpora. This paper investigates reducing annotation cost by *selective sampling*. In this approach, the learner examines many unlabeled examples and selects for labeling only those that are most informative at each stage of training. In this way it is possible to avoid redundantly annotating examples that contribute little new information. The paper first analyzes the issues that need to be addressed when constructing a selective sampling algorithm, arguing for the attractiveness of committee-based sampling methods. We then focus on selective sampling for training probabilistic classifiers, which are commonly applied to problems in statistical natural language processing. We report experimental results of applying a specific type of committee-based sampling during training of a stochastic part-of-speech tagger, and demonstrate substantially improved learning rates over sequential training using all of the text. We are currently implementing and evaluating other variants of committee-based sampling, as discussed in the paper, in order to obtain further insight on optimal design of selective sampling methods.

1 Introduction

Many corpus-based methods for natural language processing are based on supervised training, acquiring information from a manually annotated corpus. Manual annotation, however, is typically very expensive. As a consequence, only a few large annotated corpora exist, mainly for the English language and not covering many genres of text. This situation makes it difficult to apply supervised learning methods to languages other than English, or to adapt systems to different genres of text. Furthermore, it is infeasible in many cases to develop new supervised methods that require annotations different from those which are currently available.

In some cases, manual annotation can be avoided altogether, using self-organized methods, such as was shown for part-of-speech tagging of English by Kupiec (1992). Even in Kupiec's tagger, though, manual (and somewhat unprincipled) biasing of the initial model was necessary to achieve satisfactory convergence. Recently, Elworthy (1994) investigated the effect of self-converging re-estimation for part-of-speech tagging and found that some initial manual training is needed. More generally, it is clear that self-organized methods are not applicable for all natural language processing tasks, and perhaps not even for part-of-speech tagging in some languages.

In this paper we investigate the *active learning* paradigm for reducing annotation cost. There are two types of active learning, in both of which the learner has some control over the choice of the examples which are labeled and used for training. The first type uses *membership queries*, in which the learner constructs examples and asks a teacher to label them (Angluin, 1988; MacKay, 1992b; Plutowski and White, 1993). While this approach provides proven computational advantages (Angluin, 1987), it is usually not applicable to natural language problems, for which it is very difficult to construct synthetically meaningful and informative examples. This difficulty can be overcome when a large corpus of unlabeled training data is available. In this case the second type of active learning, *selective sampling*, can be applied: The learner examines many unlabeled examples, and selects only those that are most informative for the learner at each stage of training. This way, it is possible to avoid the redundancy of annotating many examples that contribute roughly the same information to the learner.

The machine learning literature suggests several different approaches for selective sampling (Seung, Opper, and Sompolinsky, 1992; Freund et al., 1993; Cohn, Atlas, and Ladner, 1994; Lewis and Catlett, 1994; Lewis and Gale, 1994). In the first part of the paper, we analyze the different issues that need to be addressed when constructing a selective sampling algorithm. These include measuring the utility of an example for the learner, the number of models that will be used for the selection process, the method for selecting examples, and, for the case of the committee-based paradigm, alternative methods for generating committee members.

In the second part of the paper we focus on selective sampling for training probabilistic classifiers. In statistical natural language processing probabilistic classifiers are often used to select a preferred analysis of the linguistic structure of a text (for example, its syntactic structure (Black et al., 1993), word categories (Church, 1988), or word senses (Gale, Church, and Yarowsky, 1993)). Classification in this framework is performed by a probabilistic model which, given an input example, assigns a probability to each possible classification and selects the most probable one. The parameters of the model are estimated from the statistics of a training corpus.

As a representative case for probabilistic classifiers we have chosen to experiment with selective sampling for stochastic part-of-speech tagging. We focus on committee-based methods, which we find particularly attractive for natural language processing due to their generality and simplicity. So far, we have evaluated one type of committee-based method, and found that it achieves substantially better learning rates than sequential training using all of the text. Thus, the learner reaches a given level of accuracy using far fewer training examples. Currently, we are implementing and comparing other variants of committee-based selection, as specified in the next section, hoping to provide further insight on the optimal design of a selective sampling method.

2 Issues in Active Learning

In this section, we will discuss the issues that affect the design of active learning methods, and their implications for performance. Our focus is on selective sampling methods, though some of our discussion applies to membership query approaches as well.

2.1 Measuring information content

The objective of selective sampling is to select those examples which will be most informative in the future. How can we determine the informativeness of an example? One method is to derive an explicit measure of the expected information gained by using the example (Cohn, Ghahramani, and Jordan, 1995; MacKay, 1992b; MacKay, 1992a). For example, MacKay (1992b) assesses the informativeness of an example, for a neural network learning task, as the expected decrease in the overall variance of the model's prediction, after training on the example. Explicit measures can be appealing, since they attempt to give a precise characterization of the information content of an example. Also, for membership querying, an explicit formulation of information content sometimes enables finding the most informative examples analytically, saving the cost of searching the example space. The use of explicit methods is limited, however, since explicit measures are generally (a) model-specific, (b) quite complex, and may require various approximations to be made practical, and (c) depend on the accuracy of the current hypothesis at any given step.

The alternative to measuring the informativeness of an example explicitly is to measure it implicitly, by quantifying the amount of uncertainty in the classification of the example, given the current training data. The committee-based paradigm (Seung, Opper, and Sompolinsky, 1992; Freund et al., 1993) does this, for example, by measuring the disagreement among committee members on a classification. The main advantage of the implicit approach is its generality, as there is no need for complicated model-specific derivations of expected information gain.

The informativeness, or utility, of a given example E depends on the examples we expect to see in the future. Hence, selection must be sensitive to the probability of seeing an example whose correct classification relies on the information contained in E. The *sequential selection* scheme, in which examples are selected sequentially on an individual basis, achieves this goal implicitly, since the training examples examined are drawn from the same distribution as future test examples. Examples may be selected from the input stream if their information content exceeds a threshold (Seung, Opper, and Sompolinsky, 1992; Freund et al., 1993; Matan, 1995). Alternatively, they may be selected randomly, with some probability proportional to information content, as we do in this paper, inspired by Freund's (1990) method for boosting. Another selection scheme that has been used (Lewis and Catlett, 1994) is *batch selection*, in which the k most informative examples in a batch of n examples are chosen for training (the process is then repeated on either the same or a new batch). In order for batch selection to properly relate to the distribution of examples, however, an explicit model of the probability distribution must be incorporated into the informativeness measure (this is also true generally for membership querying). Otherwise, the algorithm may concentrate its effort on learning from informative, but highly atypical, examples.

2.2 How many models?

In the implicit approach, the informativeness of an example is evaluated with respect to models derived from the training data at each stage. The key question then is how many models to use to evaluate an example. One approach is to use a single, optimal[1], model based on the training data seen so far. This approach is taken by Lewis and Gale (1994), for training a binary classifier.

[1] By 'optimal', we mean that model which would actually be used for classification.

They select for training those examples whose classification probability is closest to 0.5, i.e, those examples for which the current model is most uncertain.

There are some difficulties with the single model approach, however (Cohn, Atlas, and Ladner, 1994). First is the fact that a single model cannot adequately measure an example's informativeness with respect to the entire set of models allowed by the training data. Instead, what is obtained is a local estimate of the example's informativeness with respect to the single model used for evaluation. Furthermore, the single model approach may conflate two different types of classification uncertainty: (a) uncertainty due to insufficiency of training data, and (b) uncertainty due to inherent classification ambiguity with respect to the model class. We only want to measure the former, since the latter is unavoidable (given a model class). If the best model in the class will be uncertain on the current example no matter how much training is supplied, then the example does not contain useful information, despite current classification uncertainty.

From a practical perspective, it may be difficult to apply the single-model approach to complex or probabilistic classification tasks, such as those typical in natural language applications. Even for a binary classification task, Lewis and Gale (1994) found that reliable estimates of classification uncertainty were not directly available from their model, and so they had to approximate them using logistic regression. In other natural language tasks, obtaining reliable estimates of the uncertainty of a single model may be even more difficult. For example, in part-of-speech tagging, a 'classification' is a sequence of tags assigned to a sentence. Since there are many such tag sequences possible for a given sentence, the single-model approach would somehow have to compare the probability of the best classification with those of alternative classifications.

These difficulties are ameliorated by measuring informativeness as the level of disagreement between multiple models (a *committee*) constructed from the training data. Using several models allows greater coverage of the model space, while measuring disagreement in classification ensures that only uncertainty due to insufficient training (type (a) above) is considered. The number of committee members determines the precision with which the model space is covered. In other words, a larger number of committee members provides a better approximation of the model space.

2.3 Measuring disagreement

The committee-based approach requires measuring disagreement among the committee members. If two committee members are used, then either they agree, or they don't. If more are used some method must be devised to measure the amount of disagreement. One possibility is to consider the maximum number of votes for any one classification—if this number is low, then the committee members mostly disagree. However, in applications with more than two categories, this method does not adequately measure the spread of committee member classifications. Section 4.3 below describes how we use the entropy of the vote distribution to measure disagreement.

Given a measure of disagreement, we need a method which uses it to select examples for training. In the batch selection scheme, the examples with the highest committee disagreement in each batch would be selected for training (Lewis and Catlett, 1994)[2]. The selection decision is somewhat more complicated when sequential sampling is used. One option is to select just those examples whose

[2]Recall, though, the problem of modeling the distribution of input examples when using batch selection.

disagreement is above a threshold (Seung, Opper, and Sompolinsky, 1992). Another option, which may relate training better to the example distribution, is to use random selection (following the method used by Freund (1990) for boosting). In this method, the probability of an example is given by some function of the committee's disagreement on the example. In our work, we used a linear function of disagreement, as a heuristic for obtaining a selection probability (see Section 4.3 below). Finding an optimal function for this purpose remains an open problem.

2.4 Choosing committee members

There are two main approaches for generating committee-members: The *version space* approach and the *random sampling* approach. The version space approach, advocated by Cohn et al. (1994) seeks to choose committee members on the border of the space of models allowed by the training data (the *version space* (Mitchell, 1982)). Hence models are chosen for the committee which are as far from each other as possible, while being consistent with the training data. This ensures that any example on which the committee members disagree will restrict the version space.

The version space approach can be difficult to apply since finding models on the edge of the version space is a non-trivial problem in general. Furthermore, the approach is not directly applicable in the case of probabilistic classification models, where all models are possible, though not equally probable. The alternative is random sampling, in which models are sampled randomly from the set of possible models, given the training data. This approach was first advocated in theoretical work on the Query By Committee algorithm (Seung, Opper, and Sompolinsky, 1992; Freund et al., 1993), in which they assume a prior distribution on models and choose committee members randomly from the distribution restricted to the version space. In our work, we have applied the random sampling approach to probabilistic classifiers by computing an approximation to the posterior model distribution given the training data, and generating committee members from that distribution (see (Dagan and Engelson, 1995) and below for more detail). Matan (1995) presents two other methods for random sampling. In the first, he trains committee members on different subsets of the training data. It remains to be seen how this method compares with explicit generation from the posterior model distribution using the entire training set. For neural network models, Matan generates committee members by backpropagation training using different initial weights in the networks so that they reach different local minima.

3 Information gain and probabilistic classifiers

In this section we focus on information gain in the context of probabilistic classifiers, and consider the desirable properties of examples that are selected for training. Generally speaking, a training example contributes data to several statistics, which in turn determine the estimates of several parameter values. An informative example is therefore one whose contribution to the statistics leads to a useful improvement of parameter estimates. We identify three properties of parameters for which acquiring additional statistics is most beneficial:

1. The current estimate of the parameter is uncertain due to insufficient statistics in the training set. An uncertain estimate is likely to be far from the true value of the parameter and can

cause incorrect classification. Additional statistics would bring the estimate closer to the true value.

2. Classification is sensitive to changes in the current estimate of the parameter. Otherwise, acquiring additional statistics is unlikely to affect classification and is therefore not beneficial.

3. The parameter takes part in calculating class probabilities for a large proportion of examples. Parameters that are only relevant for classifying few examples, as determined by the probability distribution of the input examples, have low utility for future estimation.

The committee-based sampling scheme, as we discussed above, tends to select examples that affect parameters with the above three properties. Property 1 is addressed by randomly picking parameter values for committee members from the posterior distribution of parameter estimates (given the current statistics). When the statistics for a parameter are insufficient the variance of the posterior distribution of the estimates is large, and hence there will be large differences in the values of the parameter picked for different committee members. Note that property 1 is not addressed when uncertainty in classification is only judged relative to a *single* model (Lewis and Gale, 1994). Such an approach can capture uncertainty with respect to given parameter values, in the sense of property 2, but it does not model uncertainty about the choice of these values in the first place (the use of a single model is also criticized by Cohn et al. (1994)).

Property 2 is addressed by selecting examples for which committee members highly disagree in classification. Thus, the algorithm tends to acquire statistics where uncertainty in parameter estimates entails uncertainty in actual classification. Finally, property 3 is addressed by independently examining input examples which are drawn from the input distribution. In this way, we implicitly model the expected utility of the statistics in classifying future examples. Such modeling is absent in 'batch' selection schemes, where examples with maximal classification uncertainty are selected from a large batch of examples (see also (Freund et al., 1993) for further discussion).

4 Implementation For Part-Of-Speech Tagging

We have currently begun to explore the space of selective sampling methods for the application of bigram part-of-speech tagging (see (Merialdo, 1991))[3]. A bigram model has three types of parameters: *transition probabilities* $P(t_i \rightarrow t_j)$ each giving the probability of a tag t_j occuring after the tag t_i, *lexical probabilities* $P(t|w)$ each giving the probability of a tag t occurring for a word w, and *tag probabilities* $P(t)$ each giving the marginal probability of a tag occurring. We have implemented and tested a committee-based sequential selection scheme, using random selection as described in Section 2.1. We generate committee members randomly, by approximating the posterior distributions of the transition and output probabilities, given the training data. Some details of our implementation are given below; the system is more fully described in (Dagan and Engelson, 1995).

[3] Bigram models are a subclass of Hidden Markov Models (HMM) (Rabiner, 1989).

4.1 Posterior distributions for bigram parameters

In this section, we consider how to approximate the posterior distributions of the parameters for a bigram tagging model.[4] (Our method can also be applied to Hidden Markov Models (HMMs) in general.) Let $\{\alpha_i\}$ be the set of all parameters of the model (i.e, all transition and lexical probabilities). First note that these define a number of multinomial probability distributions. Each multinomial corresponds to a conditioning event and its values are given by the corresponding set of conditioned events. For example, a transition probability parameter $P(t_i \rightarrow t_j)$ has conditioning event t_i and conditioned event t_j.

Let $\{u_i\}$ denote the set of possible values for given multinomial variable, and let $S = \{n_i\}$ denote a set of statistics extracted from the training set, where n_i is the number of times that the value u_i appears in the training set. We denote the total number of appearances of the multinomial variable as $N = \sum_i n_i$. The parameters whose distributions we wish to estimate are $\alpha_i = P(u_i)$.

The maximum likelihood estimate for each of the multinomial's distribution parameters, α_i, is $\hat{\alpha}_i = \frac{n_i}{N}$. In practice, this estimator is usually smoothed in some way to compensate for data sparseness. Such smoothing typically reduces the estimates for values with positive counts and gives small positive estimates for values with a zero count. For simplicity, we describe here the approximation of the posterior probabilities $P(\alpha_i = a_i|S)$ for the unsmoothed estimator[5].

We approximate $P(\alpha_i = a_i|S)$ by first assuming that the multinomial is a collection of independent binomials, each corresponding to a single value u_i of the multinomial; we then separately apply the constraint that the parameters of all these binomials should sum to 1. For each such binomial, we approximate $P(\alpha_i = a_i|S)$ as a truncated normal distribution (restricted to [0,1]), with estimated mean $\mu = \frac{n_i}{N}$ and variance $\sigma^2 = \frac{\mu(1-\mu)}{N}$.[6] We found in practice, however, very small differences between parameter values drawn from this distribution, and consequently too few disagreements between committee members to be useful for sampling. We therefore also incorporate a 'temperature' parameter, t, which is used as a multiplier for the variance estimate σ^2. In other words, we actually approximate $P(\alpha_i = a_i|S)$ as a truncated normal distribution with mean μ and variance $\sigma^2 t$.

To generate a particular multinomial distribution, we randomly choose values for its parameters α_i from their binomial distributions, and renormalize them so that they sum to 1.

To generate a random model given statistics S, we note that all of its parameters $P(t_i \rightarrow t_j)$ and $P(t|w)$ are independent of each other. We thus independently choose values for the model's parameters from their multinomial distributions.

[4] We do not randomize over tag probability parameters, since the amount of data for tag frequencies is large enough to make their MLEs quite definite.

[5] In the implementation we smooth the MLE by interpolation with a uniform probability distribution, following Merialdo (1991). Adaptation of $P(\alpha_i = a_i|S)$ to the smoothed version of the estimator is simple.

[6] The normal approximation, while convenient, can be avoided. The posterior probability $P(\alpha_i = a_i|S)$ for the multinomial is given exactly by the Dirichlet distribution (Johnson, 1972) (which reduces to the Beta distribution in the binomial case).

4.2 Examples in bigram training

Typically, concept learning problems are formulated such that there is a set of training examples that are independent of each other. When training a bigram model (indeed, any HMM), however, this is not true, as each word is dependent on that before it. This problem may simply be solved by considering each sentence as an individual example. More generally, we can break the text at any point where tagging is unambiguous. In particular, suppose we have a lexicon which specifies which parts-of-speech are possible for each word (i.e, which of the parameters $P(t|w)$ are positive). In bigram tagging, we can use unambiguous words (those with only one possible part of speech) as example boundaries. This allows us to train on smaller examples, focusing training more on the truly informative parts of the corpus.

4.3 Measuring disagreement

We now consider how to measure disagreement among a set of committee members which each assign a tag sequence to a given word sequence. Since we want to quantify the spread of classification across committee member, we suggest using the entropy of the distribution of classes assigned by committee members to an example. (Other methods might also be useful, eg, variance for real-valued classification.)

Let $V(t, w)$ be the number of committee members (out of k members) 'voting' for tag t for the word w. Then w's *vote entropy* is

$$VE(w) = -\sum_t \frac{V(t,w)}{k} \log \frac{V(t,w)}{k}$$

To measure disagreement over an entire word sequence W, we use the average, $\overline{VE}(W)$, of the voting entropy over all ambiguous words in the sequence.

As a function for translating from average vote entropy to probability, we use a simple linear function of the normalized average vote entropy:

$$P_{\text{label}}(W) = \frac{e}{\log k} \overline{VE}(W)$$

where e is an *entropy gain* system parameter, which controls the overall frequency with which examples are selected, and the $\log k$ term normalizes for the number of committee members. Thus examples with higher average entropy are more likely to be selected for training.

5 Experimental Results

In this section, we describe the results of applying the committee-based sampling method to bigram part-of-speech tagging, as compared with standard sequential sampling. Evaluation was performed using the University of Pennsylvania tagged corpus from the ACL/DCI CD-ROM I. For ease of implementation, we used a complete (closed) lexicon which contains all the words in the corpus[7].

[7]We used the lexicon provided with Brill's part-of-speech tagger (Brill, 1992). While in an actual application the lexicon would not be complete, our results using a complete lexicon are still valid, since evaluation is comparative.

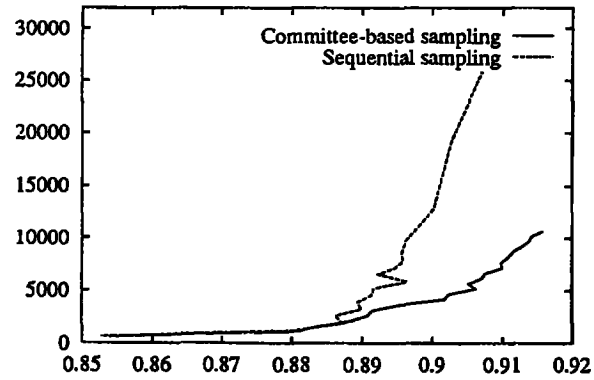

Figure 1: Amount of training (number of ambiguous words in the training sentences) plotted (y-axis) versus classification accuracy (x-axis). The committee-based sampling used $k = 10$ committee members, entropy gain $e = 0.3$, and temperature $t = 50$.

Approximately 63% of the tokens (word occurences) in the corpus were ambiguous in the lexicon.

For evaluation, we compared the learning efficiency of our committee-based selection algorithm to that of sequential selection. In sequential selection, training examples are selected sequentially from the corpus. We make the common assumption that the corpus is a random stream of example sentences drawn from the distribution of the language.

The committee-based sampling algorithm was initialized using the first 1,000 words from the corpus (624 of which were unambiguous), and then sequentially examined the following examples in the corpus for possible labeling. Testing was performed on a separate portion of the corpus consisting of 20,000 words. We compare the amount of training required by the different methods to achieve a given tagging accuracy on the test set, where both the amount of training and tagging accuracy are measured only over ambiguous words.

In Figure 1, we present a plot of training effort versus accuracy achieved, for both sequential sampling and committee-based sampling ($k = 10$, $e = 0.3$, and $t = 50$). The curves start together, but the efficiency of committee-based selection begins to be evident when we seek 89% accuracy. Committee-based selection requires less than one-fourth the amount of training that sequential selection does to reach 90.5% accuracy. Furthermore, the efficiency improvement resulting from using committee-based sampling greatly increases with the desired accuracy. This is in line with the results of Freund et al. (1993) on Query By Committee sampling, in which they prove exponential speedup under certain theoretical assumptions.

Figures 2 and 3 demonstrate that our results are qualitatively the same for different values of the system parameters. Figure 2 shows a plot comparable to Figure 1, for sampling using 5 committee members. Results are substantially the same, though the speedup is slightly less and there is more oscillation in accuracy at low training levels, due to the greater coarseness of the evaluation of information content. In Figure 3 we show a similar plot on a different test set of 20,000 words, for sampling using 10 committee members and an entropy gain of 0.5. Again, we see a similar efficiency gain for committee-based sampling as compared with sequential sampling.

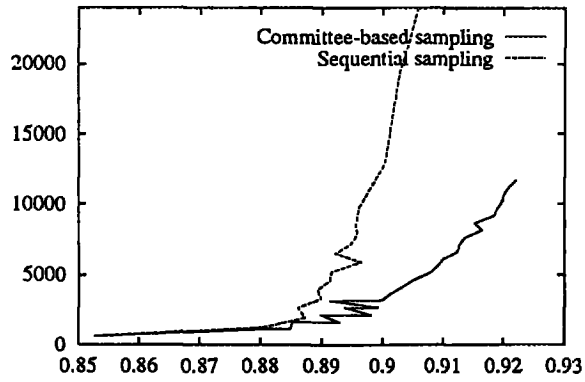

Figure 2: Further results comparing committee-based and sequential selection. Amount of training (number of ambiguous words in the training sentences) plotted against desired accuracy. (a) Training vs. accuracy for $k = 5$, $e = 0.3$, and $t = 50$.

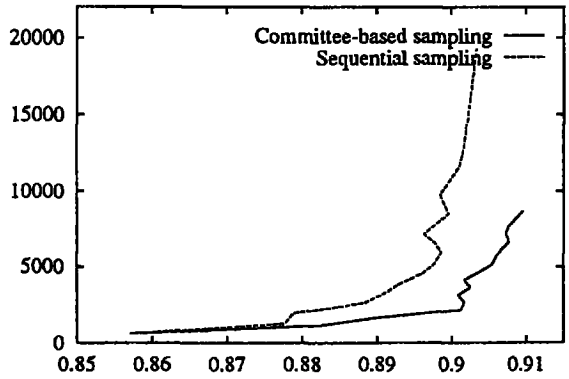

Figure 3: Training vs. accuracy for $k = 10$, $e = 0.5$ and $t = 50$, for a different test set of 20,000 words.

Figure 4 compares our random selection scheme with a batch selection scheme (similar to that of Lewis and Catlett (1994)). In the batch selection scheme a large number of examples are examined (the *batch size*) and the best n examples in the batch are used for training (we call n the *selection size*). As discussed above in Section 2.1, we expect random selection to outperform batch selection, since random selection also takes into account the frequency with which examples occur. As can be seen in figure 4, this hypothesis is borne out, although batch selection still improves significantly over sequential sampling.

6 Conclusions

Annotating large textual corpora for training natural language models is a costly process. Selective sampling methods can be applied to reduce annotation cost, by avoiding redundantly annotating ex-

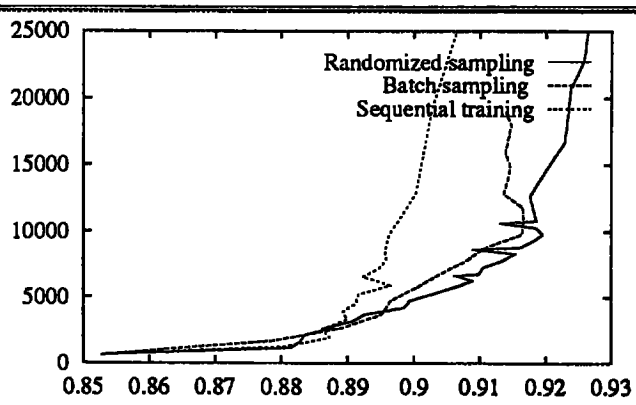

Figure 4: Comparison of random vs. batch committee-based sampling. Amount of training (number of ambiguous words in the training sentences) plotted (y-axis) versus classification accuracy (x-axis). Both methods used $k = 5$ committee members and a temperature of $t = 50$; random sampling used an entropy gain $e = 0.3$; batch sampling used a batch size of 1000 and selection size of 10.

amples that contribute little new information to the learner. Information content may be measured either explicitly, by means of an analytically derived formula, or implicitly, by estimating model classification uncertainty. Implicit methods for measuring information gain are generally simpler and more general than explicit methods. Sequential selection, unlike batch selection, implicitly measures the expected utility of an example relative to the example distribution.

The number of models used to evaluate informativeness in the implicit approach is crucial. If only one model is used, information gain is only measured relative to a small part of the model space. When multiple models are used, there are different ways to generate them from the training data. The version space approach is applicable to problems where a version space exists and can be efficiently represented. The method is most useful when the probability distribution of models in the version space is uniform. Random selection is more broadly applicable, however, especially to learning probabilistic classifiers. It is as yet unclear how different random sampling methods compare. In this paper, we have empirically examined two types of committee-based selective sampling methods, and our results suggest the utility of selective sampling for natural language learning tasks. We are currently experimenting further with different methods for committee-based sampling.

The generality obtained from implicitly modeling information gain suggests using committee-based sampling also in non-probabilistic contexts, where explicit modeling of information gain may be impossible. In such contexts, committee members might be generated by randomly varying some of the decisions made in the learning algorithm. This approach might, for example, be profitably applied to learning Hidden Markov Model (HMM) structure (Stolcke and Omohundro, 1992), in addition to estimating HMM parameters.

Another important area for future work is in developing selective sampling methods which are independent of the eventual learning method to be applied. This would be of considerable advantage in developing selectively annotated corpora for general research use. Recent work on heterogeneous uncertainty sampling (Lewis and Catlett, 1994) supports this idea, as they show positive results

for using one type of model for example selection and a different type for classification.

Acknowledgements

Discussions with Yoav Freund and Yishai Mansour greatly enhanced this work. The second author gratefully acknowledges the support of the Fulbright Foundation.

References

Angluin, Dana. 1987. Learning regular sets from queries and counterexamples. *Information and Computation*, 75(2):87–106, November.

Angluin, Dana. 1988. Queries and concept learning. *Machine Learning*, 2:319–342.

Black, Ezra, Fred Jelinek, John Lafferty, David Magerman, Robert Mercer, and Salim Roukos. 1993. Towards history-based grammars: using richer models for probabilistic parsing. In *Proc. of the Annual Meeting of the ACL.*

Brill, Eric. 1992. A simple rule-based part of speech tagger. In *Proc. of ACL Conference on Applied Natural Language Processing.*

Church, Kenneth W. 1988. A stochastic parts program and noun phrase parser for unrestricted text. In *Proc. of ACL Conference on Applied Natural Language Processing.*

Cohn, David, Les Atlas, and Richard Ladner. 1994. Improving generalization with active learning. *Machine Learning*, 15.

Cohn, David A., Zoubin Ghahramani, and Michael I. Jordan. 1995. Active learning with statistical models. In G. Tesauro, D. Touretzky, and J. Alspector, editors, *Advances in Neural Information Processing*, volume 7. Morgan Kaufmann.

Dagan, Ido and Sean Engelson. 1995. Committee-based sampling for training probabilistic classifiers. In *Proc. Int'l Conference on Machine Learning.* (to appear).

Elworthy, David. 1994. Does Baum-Welch re-estimation improve taggers? In *ANLP*, pages 53–58.

Freund, Y., H. S. Seung, E. Shamir, and N. Tishby. 1993. Information, prediction, and query by committee. In S. Hanson et al., editor, *Advances in Neural Information Processing*, volume 5. Morgan Kaufmann.

Freund, Yoav. 1990. An improved boosting algorithm and its implications on learning complexity. In *Proc. Fifth Workshop on Computational Learning Theory.*

Gale, William, Kenneth Church, and David Yarowsky. 1993. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26:415–439.

Johnson, Norman L. 1972. *Continuous Multivariate Distributions.* John Wiley & Sons, New York.

Kupiec, Julian. 1992. Robust part-of-speech tagging using a hidden makov model. *Computer Speech and Language*, 6:225–242.

Lewis, D. and J. Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *Machine Learning Proceedings of the 11th International Conference.*

Lewis, D. and W. Gale. 1994. Training text classifiers by uncertainty sampling. In *Proceedings of ACM-SIGIR Conference on Information Retrieval.*

MacKay, David J. C. 1992a. The evidence framework applied to classification networks. *Neural Computation*, 4.

MacKay, David J. C. 1992b. Information-based objective functions for active data selection. *Neural Computation*, 4.

Matan, Ofer. 1995. On-site learning. Submitted for publication.

Merialdo, Bernard. 1991. Tagging text with a probabilistic model. In *Proc. Int'l Conf. on Acoustics, Speech, and Signal Processing.*

Mitchell, T. 1982. Generalization as search. *Artificial Intelligence*, 18.

Plutowski, Mark and Halbert White. 1993. Selecting concise training sets from clean data. *IEEE Trans. on Neural Networks*, 4(2).

Rabiner, Lawrence R. 1989. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2).

Seung, H. S., M. Opper, and H. Sompolinsky. 1992. Query by committee. In *Proc. ACM Workshop on Computational Learning Theory.*

Stolcke, A. and S. Omohundro. 1992. Hidden Markov Model induction by Bayesian model merging. In *Advances in Neural Information Processing*, volume 5. Morgan Kaufmann.