Machine Learning manuscript No.
(will be inserted by the editor)

Active Learning for Logistic Regression: An Evaluation

Andrew I. Schein, Lyle H. Ungar

Department of Computer and Information Science
The University of Pennsylvania
3330 Walnut Street
Philadelphia, PA 19104-6389
{ais,ungar}@cis.upenn.edu

Received: date / Revised version: date

Abstract Which active learning methods can we expect to yield good performance in learning binary and multi-category logistic regression classifiers? Addressing this question is a natural first step in providing robust solutions for active learning across a wide variety of exponential models including maximum entropy, generalized linear, log-linear, and conditional random field models. For the logistic regression model we re-derive the variance reduction method known in experimental design circles as 'A-optimality.' We then run comparisons against different variations of the most widely used heuristic schemes: query by committee and uncertainty sampling, to discover which methods work best for different classes of problems and why. We find that among the strategies tested, the experimental design methods are most likely to match or beat a random sample baseline. The heuristic alternatives produced mixed results, with an uncertainty sampling variant called margin sampling providing the most promising performance at very low computational cost. Computational running times of the experimental design methods were a bottleneck to the evaluations. Meanwhile, evaluation of the heuristic methods lead to an accumulation of negative results. Such results demonstrate a need for improved active learning methods that will provide reliable performance at a reasonable computational cost.

1 Introduction

Procurement of labeled training data is the seminal step of training a supervised machine learning algorithm. A recent trend in machine learning

Send offprint requests to: Andrew I. Schein

has focused on *pool-based* settings where unlabeled data is inexpensive and available in large supply, but the labeling task is expensive. Pool-based active learning methods attempt to reduce the "cost" of learning in a pool-based setting by using a learning algorithm trained on the existing data and selecting the portion of the remaining data with the greatest expected benefit. In classification settings benefit may be measured in terms of the generalization accuracy (or error) of the final model.

The last decade has also seen increased use of the logistic regression classifier in machine learning applications, though under different names: multinomial regression, multi-class logistic regression or the maximum entropy classifier. In this study we address the question of how to best perform pool-based active learning with the logistic regression model. We view treatment of this problem as a natural first step in developing active learning solutions to the expansive set of models derived from the exponential family of distributions, of which logistic regression is a member.

1.1 Active Learning: a Definition

Active learning is defined as a setting where a learning agent interacts with its environment in procuring a training set, rather than passively receiving an i.i.d. sample from some underlying distribution. The term pool-based active learning is used to distinguish sampling a pre-defined pool of examples from other forms of active learning including methods that construct examples from R^n or other sets from first principles. Henceforth we will often use the term active learning to refer to pool-based active learning; since the study does not treat the other forms, no confusion will arise. Furthermore, we focus almost entirely on the problem of training classifiers.

The purpose of developing active learning methods is to achieve the best possible generalization error at the least cost, where cost is usually measured as a function of the number of examples labeled. Frequently we plot the tradeoff between number of examples labeled and generalization error through learning curves. It is commonly believed that there should exist active learning methods that perform at least as well as random sampling from a pool at worst, and these methods should often outperform random sampling. This belief is given theoretical justification under very specific assumptions (Freund, Seung, Shamir, & Tishby, 1997; Seung, Opper, & Sompolinsky, 1992), but is also occasionally contradicted by empirical evaluations of existing methods.

1.2 Background and Related Work

The earliest research in active learning stressed counterexample requests (*e.g.* (Angluin, 1987)) or query construction (Cohn, 1996; MacKay, 1991).

Focus soon turned to methods applicable to pool-based active learning including the query by committee method (Seung et al., 1992) and experimental design methods based on *A*-optimality (Cohn, 1996). The above methods are motivated by theory and explicit objective functions. Empirical evaluation of such objective function approaches has been scant due to computational costs associated with these methods. Of late, there are some signs of renewed interest in objective function approaches (Gilad-Bachrach, Navot, & Tishby, 2003).

There has been growing interest in application of active learning to real-world data sets. A trend of the last ten years (Abe & Mamitsuka, 1998; Banko & Brill, 2001; Chen, Schein, Ungar, & Palmer, 2006; Dagan & Engelson, 1995; Hwa, 2004; Lewis & Gale, 1994; McCallum & Nigam, 1998; Melville & Mooney, 2004; Roy & McCallum, 2001; Tang, Luo, & Roukos, 2002) has been to employ heuristic methods of active learning with no explicitly defined objective function. Uncertainty sampling (Lewis & Gale, 1994), query by committee (Seung et al., 1992)[1], and variants have proven particularly attractive because of their portability across a wide spectrum of machine learning algorithms. A subtrend in the field has sought to improve performance of heuristics by combining them with secondary heuristics such as: similarity weighting (McCallum & Nigam, 1998), interleaving active learning with EM (McCallum & Nigam, 1998), interleaving active learning with co-training (Steedman et al., 2003), and sampling from clusters (Tang et al., 2002), among others.

1.3 Purpose and Contributions of Study

The goal of this study is to learn which of the methods for active learning work best with logistic regression, and why methods perform badly when they do. We are interested both in the binary classification and less often explored multiple category settings. The experiments necessary to make conclusions about active learning help establish a picture of the 'state of the art' that will be useful for practitioners of active learning in addition to researchers in the field.

There are two main categories of methods that we evaluate. First, we re-examine the theory of experimental design in the context of the logistic regression classifier. A technique for minimizing prediction variance known as *A*-optimality emerges as a promising technique for active learning. The variance reduction technique is generalized in this work from a squared loss to a log loss. Second, we use these two loss-motivated methods as a baseline in evaluating six heuristic methods of active learning. Ultimately, we use the

[1] Query by Committee is a method with strong theoretical properties under limited circumstances (Freund et al., 1997; Seung et al., 1992), but the overwhelming trend has been to apply the method in circumstances where the theory does not apply. Often the term Query by Bagging is used to describe such *ad hoc* applications. Section 3 contains further discussion.

evaluations to make conclusions about the performance of different active learning methods.

The empirical investigations within this work have several distinguishing features. The focus of this study is on logistic regression, and methods that perform well (or poorly) with alternative machine learning algorithms may behave differently with logistic regression. Our evaluations of the experimental design methods are the largest scale of any to date in a pool-based active learning setting. So these evaluations are an opportunity to test the hypothesis that the computational costs of principled methods come with performance gains. Noting that heuristic methods occasionally perform worse than random, we also explore the causes of these failures, and identify conditions that lead the uncertainty sampling heuristics to failure.

2 The Logistic Regression Classifier

2.1 Bernoulli Model

Logistic regression (Hosmer & Lemeshow, 1989) can be viewed as arising from a Bernoulli model. Given a set of predictors, $\mathbf{x}_n$, we wish to determine the probability of a binary outcome y_n. We define a probability model:

$$\mathsf{P}(Y_n = 1|\mathbf{x}_n) \doteq \sigma(\mathbf{w} \cdot \mathbf{x}_n) \tag{1}$$

with corresponding likelihood function:

$$\mathsf{P}(\mathbf{y}|\mathbf{x}_n, n = 1 \ldots N) = \prod_n \sigma(\mathbf{w} \cdot \mathbf{x}_n)^{y_n} (1 - \sigma(\mathbf{w} \cdot \mathbf{x}_n))^{(1-y_n)} \tag{2}$$

$$= \prod_n \sigma(\mathbf{w} \cdot \mathbf{x}_n)^{y_n} \sigma(-\mathbf{w} \cdot \mathbf{x}_n)^{(1-y_n)}. \tag{3}$$

where the logistic function

$$\sigma(\theta) = \frac{1}{1 + \exp[-\theta]}. \tag{4}$$

is a continuous increasing function mapping any real valued θ into the interval $(0, 1)$, and thus is suitable for representing the probability of a Bernoulli trial outcome.

A useful variant for scientific and sociology experiments employs a binomial (Bickel & Doksum, 2001) rather than Bernoulli formulation to facilitate repeated trials.

2.2 Multinomial Model

When there are more than two outcome categories, the situation is a little more complex; the outcome variables Y_n take on one of three or more discrete outcomes rather than a 0 or a 1. We define a probability model as follows:

$$\mathsf{P}(Y_n = c|x_n) \doteq \pi(c, \mathbf{x}_n, \mathbf{w}) = \frac{\exp(\mathbf{w}_c \cdot \mathbf{x}_n)}{\sum_{c'} \exp(\mathbf{w}_{c'} \cdot \mathbf{x}_n)}. \tag{5}$$

The parameter vector $\mathbf{w}$ of the binary logistic model subdivides into a set of vectors $\mathbf{w}_c$: one for each category. The resulting likelihood is:

$$\mathsf{P}(\mathbf{y}|\mathbf{x}_n, n = 1 \ldots N) = \prod_{nc} \pi(c, \mathbf{x}_n, \mathbf{w})^{y_{nc}}. \tag{6}$$

The multinomial model is a generalization of the binary case, as can be seen by defining $\mathbf{w}_0 = \mathbf{0}$ and $\mathbf{w}_1 = \mathbf{w}$ in which case:

$$\mathsf{P}(Y_n = 1|\mathbf{x}_n) = \frac{\exp(\mathbf{w} \cdot \mathbf{x}_n)}{\exp(\mathbf{0} \cdot \mathbf{x}_n) + \exp(\mathbf{w} \cdot \mathbf{x}_n)} \tag{7}$$

$$= \frac{\exp(\mathbf{w} \cdot \mathbf{x}_n)}{1 + \exp(\mathbf{w} \cdot \mathbf{x}_n)} \tag{8}$$

$$= \frac{1}{1 + \exp(-\mathbf{w} \cdot \mathbf{x}_n)} \tag{9}$$

$$= \sigma(\mathbf{w} \cdot \mathbf{x}_n). \tag{10}$$

The logistic regression model is closely related to a large collection of well-worn models such as the exponential family of distributions (Bickel & Doksum, 2001), generalized linear models (McCullagh & Nelder, 1989), maximum entropy classifiers (Berger, Della Pietra, & Della Pietra, 1996), and conditional Markov random field models (Lafferty, McCallum, & Pereira, 2001). (Schein, 2005) reviews these relationships in greater depth.

2.3 Parameter Estimation

Analysis of the Hessian of the logistic regression log likelihood function reveals the model is convex in the parameters. Any number of standard convex optimization procedures including gradient, conjugate gradient, and Broyden, Fletcher, Goldfarb, and Shanno (BFGS) methods suffice (see (Nocedal & Wright, 1999) for a description of these algorithms). When the predictors are all positive ($x_{ni} \geq 0$), generalized iterative scaling and variants (Berger et al., 1996; Darroch & Ratcliff, 1972; Jin, Yan, Zhang, & Hauptmann, 2003) work as well. Iterative scaling procedures have the advantage that they are extremely simple to implement. Methods that take second order information into account such as conjugate gradient and BFGS are known

to converge quicker than generalized iterative scaling (GIS) and improved iterative scaling (IIS) in maximum entropy modeling (Malouf, 2002).

An important characteristic of the parameters of logistic regression are the existence and consistency of the maximum likelihood parameters. It can be shown for logistic regression parameters $\mathbf{w}$ and estimates $\hat{\mathbf{w}}$ that:

$$\mathcal{L}(\sqrt{n}(\hat{\mathbf{w}} - \mathbf{w})) \rightarrow \mathcal{N}(\mathbf{0}, F^{-1}(\mathbf{w})) \text{ and} \tag{11}$$

$$\hat{\mathbf{w}}_n = \overline{\mathbf{w}}_n + O_p(\frac{1}{n^{1/2}}). \tag{12}$$

F refers to the Fisher information matrix. The $\mathcal{L}$ in this notion refers to the distribution that its argument follows, $\hat{\mathbf{w}}_n$ and $\overline{\mathbf{w}}_n$ refer to estimate based on a sample and expected estimate of $\mathbf{w}$ respectively. $F(\mathbf{w})$ is the Fisher information matrix of the model, described in Section 4.2. The O_p notation refers to a rate of convergence in probability. The requisite theory for demonstrating Equations 11 and 12 is beyond the scope of this exposition, and we refer the reader to (Bickel & Doksum, 2001, Sections 6.2 and 6.5) for an account. We use Equations 11 and 12 in Section 4 in deriving an asymptotically correct estimate of variance.

3 Heuristic Active Learning for Logistic Regression

All of the pool-based active learning methods evaluated in this study fit into a common framework described by Algorithm 1. The key difference distinguishing methods of active learning is the method for ranking examples, amounting to different assessments the value of labeling individual examples. Usually, the ranking rule makes use of the model trained on the currently labeled data. This is the reason for the requirement of a partial training set when the algorithm begins.

Algorithm 1 A Generalized Active Learning Loop

Require: partial training set, pool of unlabeled examples
  repeat several times
    Select T random examples from pool
    Rank these T examples according to active learning rule
    Present the top-ranked example to oracle for labeling
    Augment the training set with the new observation
  until Training set reaches desirable size
end

Other variants of Algorithm 1 are used. For example, some researchers mix active learning with random labels. Others label the top n examples in addition to the top example in order to decrease the number times a learner is retrained. This evaluation will focus on labeling one example at a time. In principle this gives a rigorous method the opportunity to pick only the best examples.

The flexibility of the setting described by Algorithm 1 leaves open the possibility of using heuristics that are independent of the classification algorithm (in the present case, logistic regression). Research of the last fifteen years has produced many heuristics for ranking examples, and the most prominent methods are introduced in this section in anticipation of the evaluation. In the general classification setting that this study focuses on, little can be said that relates these approaches to explicit objective functions. Under a few assumptions, including at a minimum the assumption that classification is a noise free function of the predictors, it may be possible to establish a relationship between each of these methods and an objective function.

In our evaluations we look at three types of heuristics for active learning: uncertainty sampling, query by committee and classifier certainty. We describe these methods along with their computational complexities, and then briefly review variations of these methods in the remaining subsections.

3.1 Uncertainty Sampling

Uncertainty sampling is a term invented by Lewis and Gale (Lewis & Gale, 1994), though the ideas can be traced back to the query methods of Hwang *et al.* (Hwang, Choi, Oh, & Marks, 1991) and Baum (Baum, 1991). We discuss the Lewis and Gale variant since it is widely implemented and general to probabilistic classifiers such as logistic regression. The uncertainty sampling heuristic chooses for labeling the example for which the model's current predictions are least certain. The intuitive justification for this approach is that regions where the model is uncertain indicate a decision boundary, and clarifying the position of decision boundaries is the goal of learning classifiers.

A key question is how to measure uncertainty. Different methods of measuring uncertainty will lead to different variants of uncertainty sampling. We will look at two such measures. As a convenient notation we use $\mathbf{q}$ to represent the trained model's predictions, with q_c equal to the predicted probability of class c. One method is to pick the example whose prediction vector $\mathbf{q}$ displays the greatest Shannon entropy:

$$-\sum_c q_c \log q_c. \tag{13}$$

Such a rule means ranking candidate examples in Algorithm 1 by Equation 13.

An alternative method picks the example with the smallest margin: the difference between the largest two values in the vector $\mathbf{q}$. In other words, if c, c' are the two most likely categories for observation $\mathbf{x}_n$, the margin is measured as follows:

$$M_n = |\hat{\mathsf{P}}(c|\mathbf{x}_n) - \hat{\mathsf{P}}(c'|\mathbf{x}_n)|. \tag{14}$$

In this case, Algorithm 1 would rank examples by increasing values of margin, with the smallest value at the top of the ranking.

The original definition of uncertainty sampling (Lewis & Gale, 1994) describes the method in the binary classification setting, where the two definitions of uncertainty are equivalent. We are not aware of previous usages of minimum margin sampling active learning in multiple category settings except when motivated as a variant of query by committee (Section 3.2).

Using uncertainty sampling, the computational cost of picking an example from T candidates is: $O(TDK)$ where D is the number of predictors, K is the number of categories. In the evaluations we refer to the different uncertainty methods as entropy and margin sampling.

3.2 Query by Committee

Query by committee (QBC) was proposed by Seung, Opper and Sompolinksy (Seung et al., 1992), and then rejustified for the perceptron case by Freund *et al.* (Freund et al., 1997). The method assumes (a) A noise-free (*e.g.* separable) classification task and (b) A binary classifier with a Gibbs training procedure. Under these assumptions and a few others (Freund et al., 1997; Seung et al., 1992) a procedure can be found that guarantees exponential decay in the generalization error:

$$E_g \sim e^{-nI(\infty)} \tag{15}$$

where $I(\infty)$ denotes a limiting (in committee size) query information gain and n is the size of the training set.

A description of the query by committee algorithm follows. A committee of k models M_i are sampled from the version space over the existing training set using a Gibbs training procedure. The next training example is picked to minimize the entropy of the distribution over the model parameter posteriors. In the case of perceptron learning, this is achieved by selecting query points of prediction disagreement. The method is repeated until enough training examples are found to reduce error to an acceptable level.

Alas, the assumptions of the method are frequently broken, and in particular the noise-free assumption does not apply to logistic regression on the data sets we intend to use in the evaluations. The noise-free assumption is critical to QBC, since the method depends on an ability to permanently discard a portion of version space (the volume the parameters may occupy) with each query. Version space volume in the noisy case is analogous to the D-optimality score, since a determinant is essentially a volume measure. Generally the model variance, as measured through the D-optimality score of linear and non-linear models, does not decrease exponentially in the training set size even under optimal conditions.

The use of the query by committee method in situations where the assumptions do not apply is an increasing trend with the modifications of Abe

and Mamitsuka (Abe & Mamitsuka, 1998) and McCallum and Nigam (McCallum & Nigam, 1998) who substitute bagging for the Gibbs training procedure. The term "query by bagging" (QBB) is becoming a catchphrase for algorithms that take a bagging approach to implementing the query by committee procedure. Query by bagging is implemented as follows. An ensemble of models $\hat{f}_i$ is formed from the existing training set using the bagging procedure (Breiman, 1996). An observation is picked from the pool that maximizes disagreement among the ensemble members. The procedure is repeated until enough training examples are chosen.

As a modification to Algorithm 1, the following pseudocode replaces the original line that producing a ranking:

Use bagging (Breiman, 1996) to train B classifiers $\hat{f}_i$

Rank candidates by disagreement among the $\hat{f}_i$

The definition of disagreement is wide open and several methods have been proposed. A margin-based disagreement method (14) is to average the predictions of the $\hat{f}_i$ (normalizing to ensure a proper distribution), and using the margin computation of Equation 14. We refer to this method as QBB-AM (Abe & Mamitsuka, 1998) (query by bagging followed by author's initials).

An alternative approach to measuring disagreement is to take the average prediction (as above) and measure the average KL divergence from the average:

$$\frac{1}{B}\sum_{b=1}^{B}\mathsf{KL}(\hat{f}_b||\hat{f}_{\text{avg}}) \tag{16}$$

Larger values of average divergence indicate more disagreement, and so ranking occurs from larger to smaller values in Algorithm 1. Following the convention of using the author's initials, we refer to this method as QBB-MN (McCallum & Nigam, 1998). Under these two disagreement measures, query by bagging methods take only slightly more computational time than certainty sampling methods: $O(BTDK)$; the cause of the difference is inclusion of the bag size B in the formula.

3.3 Classifier Certainty

For logistic regression and other probabilistic classifiers, several researchers have proposed minimizing the entropy of the algorithm's predictions (MacKay, 1991, 1992; Roy & McCallum, 2001)[2]:

$$\text{CC} = -\sum_{p\in\text{Pool}}\sum_{c}\hat{\mathsf{P}}(c|\mathbf{x}_p)\log\hat{\mathsf{P}}(c|\mathbf{x}_p) \tag{17}$$

[2] Some readers familiar with the language modeling literature will be used to "prediction entropy" as a measure of performance. However, in language modeling, it is actually a cross-entropy that is measured, not prediction entropy for the reasons outlined below.

as a criteria for picking a training set. The sum is over the pool of unlabeled data and the set of categories. In intuitive terms Equation 17 measures degree of certainty of the individual classifications over the pool, and so we call the method the Classifier Certainty (CC) method. In order to rank examples in Algorithm 1, an expected value of CC is computed with respect to the current model $\hat{\mathsf{P}}$ for each candidate. The expectation is over possible labelings of the candidate. A more detailed explanation of the expectation procedure is given in Section 4.3.

Note however, that CC is not a proper loss function and minimization need not lead to good accuracy; Equation 17 does not depend on the true probabilities P but only the estimates $\hat{\mathsf{P}}$. For example, we often find ourselves certain of facts or beliefs that are later found not be true. Restricting the search for examples to those that makes us more certain of previously held beliefs can be a bad choice when learning.

Excluding the cost of model fitting, implementation of CC is at worse: $O(TNKD)$, where N is the number of observations from the pool used to compute the benefit of adding an observation, D is the number of predictors, T is the number of candidates evaluated for labeling, and K is the number of categories. An approximation that saves computational time is Monte Carlo sampling from the pool to assess the benefit of labeling. For example, in our evaluations, we sample 300 examples from the pool to assess model improvement.

3.4 Heuristic Generalizations and Variations

Uncertainty sampling and query by committee methods appear so general in their implementation that it is tempting to port the methods to more complex problems than the classification setting. Such has happened in the case of part of speech tagging, where the query by committee methods are generalized to apply to hidden Markov models (Dagan & Engelson, 1995). In parsing, uncertainty sampling (Hwa, 2004) and other heuristic approaches have been applied (Tang et al., 2002).

A recent trend in the pool-based active learning literature has been to take various approaches, usually uncertainty sampling or query by committee and try to improve performance through additional heuristics. Such schemes include: observation similarity weighting (McCallum & Nigam, 1998), sampling from clusters (Tang et al., 2002), interleaving labeling with EM (McCallum & Nigam, 1998), interleaving labeling with co-training (Steedman et al., 2003), increasing diversity of ensembles (Melville & Mooney, 2004), among others. These sorts of variations are so numerous that we are unable to evaluate them here.

4 Loss Function Active Learning for Logistic Regression

In this section we explore a methodology for active learning of the logistic regression classifier using explicit loss functions. The techniques are moti-

Table 1 Notation used in the decomposition of squared error.

E	Expectation with respect to actual distribution governing $(\mathbf{x}, y)$.
$\mathsf{E}_{\mathcal{D}_s}$	Expectation with respect to training sets of size s. The s variable is often left implicit.
$\pi(c, x, \hat{w}; \mathcal{D})$	Model's probability of c given x. Parameter vector $\hat{\mathbf{w}}$ is determined by a training set $\mathcal{D}$. The variables $\hat{\mathbf{w}}$ or $\mathcal{D}$ are frequently dropped in the notation for this reason.
$\pi(c, x, w)$	Model's probability of c given x using arbitrary weight vector $\mathbf{w}$.

vated by experimental design, but have not been used in active learning of the logistic regression classifier. What makes these loss functions appealing is that they define an explicit criterion for labeling examples, and can use a large class of loss functions. For that reason, we detail their derivation in depth. Our derivations are for arbitrary numbers of categories. In the binary classification setting, many of the formula simplify (Schein, 2005).

4.1 A Squared Error Decomposition for Probabilistic Classification

Squared error is a loss function more often associated with regression rather than classifier settings. However, the loss is still applicable to classifiers and so we exploit its analytical properties in this setting. Geman (Geman, Bienenstock, & Doursat, 1992) provides a detailed account of the bias/variance decomposition for squared loss; We will use some details of the decomposition to understand what a variance reduction approach to active learning accomplishes.

Analysis of squared error begins with the decomposition into training set independent and dependent terms:

$$\sum_c \mathsf{E}[(1_c - \pi(c, \mathbf{x}; \mathcal{D}))^2 | \mathbf{x}, \mathcal{D}] = \sum_c \mathsf{E}[(1_c - \mathsf{E}[c|\mathbf{x}])^2 | \mathbf{x}, \mathcal{D}] \quad \textit{“noise”} \quad (18)$$
$$+ \sum_c (\pi(c, \mathbf{x}; \mathcal{D}) - \mathsf{E}[c|\mathbf{x}])^2$$

The left hand side is the squared error for a single observation $(\mathbf{x}, y)$; the variable 1_c is an indicator function taking on the value 1 when the observation has label c, and 0 otherwise. The expectation E is with respect to the true distribution producing $(y, \mathbf{x})$.

A further expectation with respect to the distribution generating $(\mathbf{x}, y)$ gives the expected loss over a test set. However we hold $\mathbf{x}$ constant to simplify the notation for the time being. The variable $\mathcal{D}$ represents a training set distribution, for our purposes a multiset of s observations (x, y) sampled from the underlying distribution governing $(\mathbf{x}, y)$. The first term of the decomposition (18) named “noise” represents error that is training set independent: the expectation is conditioned on the training set $\mathcal{D}$. Another interpretation of the first term is the portion of error induced when the

actual distribution of categories (conditioned on $\mathbf{x}$) is used in making predictions.

In contrast, the second term of the decomposition depends on the particular training set since no conditioning on $\mathcal{D}$ occurs. A sensible analysis on the second term is to consider the expectation with respect to alternative training sets $\mathcal{D}$. Taking such an expectation produces the mean squared error (MSE) of the model:

$$\text{MSE} \doteq \sum_c \mathsf{E}_{\mathcal{D}}[(\pi(c, \mathbf{x}; \mathcal{D}) - \mathsf{E}[c|\mathbf{x}])^2]. \tag{19}$$

The MSE decomposes as follows:

$$\begin{aligned} \text{MSE} = &\sum_c (\mathsf{E}_{\mathcal{D}}[\pi(c, \mathbf{x}; \mathcal{D})] - \mathsf{E}[c|\mathbf{x}])^2 \quad \textit{“squared bias”} \\ &+ \sum_c \mathsf{E}_{\mathcal{D}}[(\pi(c, \mathbf{x}; \mathcal{D}) - \mathsf{E}_{\mathcal{D}}[\pi(c, \mathbf{x}; \mathcal{D})])^2]. \quad \textit{“variance”} \end{aligned} \tag{20}$$

The bias term captures the difference between $\mathsf{E}_{\mathcal{D}}\pi(c, \mathbf{x}; \mathcal{D})$ (the expected model from a fixed size sample) and the distribution that actually generates y from x. The variance term captures the variability of the model under resampling data sets of fixed size, represented by $\mathsf{E}_{\mathcal{D}}$.

The notation can capture training sets of differing size using the variable s thusly: $\mathcal{D}_s$, in which case it is useful to consider the limiting behavior of variance and squared bias as the training set size grows. Variance is then:

$$\sum_c \lim_{s\to\infty} \mathsf{E}_{\mathcal{D}_s}\left[(\pi(c, \mathbf{x}; \mathcal{D}_s) - \lim_{s\to\infty} \mathsf{E}_{\mathcal{D}_s}[\pi(c, \mathbf{x}; \mathcal{D}_s)])^2\right] = 0. \tag{21}$$

The variance of the model disappears as the training set grows. This is a consequence of the consistency of the parameter estimates of the model (Bickel & Doksum, 2001).

For the squared bias term we have:

$$\sum_c \left[\lim_{s\to\infty} \mathsf{E}_{\mathcal{D}_s}[\pi(c, \mathbf{x}; \mathcal{D})] - \mathsf{E}[c|\mathbf{x}]\right]^2 \geq 0. \tag{22}$$

When equality holds for the limiting bias term, we say the model is *consistent*. In general, when modeling problems involving real world data, logistic regression is not consistent. This is true, for instance, when the appropriate predictors are missing. In other situations, all necessary predictors are available, but the probability model governing y given x is not in the class of distributions that logistic regression can encode.

We define several terms to denote the limiting error of the model:

$$\text{Residual Bias} = \sum_c \left[\lim_{s\to\infty} \mathsf{E}_{\mathcal{D}_s}[\pi(c, \mathbf{x}; \mathcal{D})] - \mathsf{E}[c|\mathbf{x}]\right]^2 . \tag{23}$$

and

$$\text{Residual Error} = \sum_c \mathsf{E}[(1_c - \mathsf{E}[c|\mathbf{x}])^2|\mathbf{x}, \mathcal{D}] + \text{Residual Bias} \quad (24)$$

$$= \text{Noise} + \text{Residual Bias}. \quad (25)$$

This last term consists of the training set-independent error of Equation 18 and the portion of bias that is training set size independent. For now, we define our goal in learning as minimizing squared error. From the various decompositions we see that this is equivalent to minimizing MSE, and thus both bias and variance. To achieve our goals, we may focus on decreasing bias, variance or both simultaneously. While estimation of bias may be possible, for instance following (Cohn, 1997), we leave this subject for future work, and focus on estimation of variance and its consequences for active learning. When using flexible models with large numbers of features, variance is often more of a problem then bias.

4.2 A Variance Estimating Technique

The decomposition (20) suggests that minimization of the variance will decrease MSE. Fortunately, statistical theory governing prediction variance provides a convenient mechanism for estimating variance over a pool of unlabeled data points. Without this theory, a bootstrap approach to variance estimation (Saar-Tsechansky & Provost, 2001) would be the only recourse. Minimization of this variance is known in the field of optimal design of experiments as A-optimality, *c.f.* (Chaloner & Larntz, 1989). We derive the requisite theory for multinomial logistic regression below.

Taking two terms of a Taylor expansion of $\pi(c, \mathbf{x}, \mathbf{w}; \mathcal{D})$:

$$\pi(c, \mathbf{x}, \hat{\mathbf{w}}; \mathcal{D}) = \pi(c, \mathbf{x}, \mathbf{w}) + \mathbf{g}(c)(\hat{\mathbf{w}} - \mathbf{w}) + O(\frac{1}{\sqrt{s}}), \quad (26)$$

where $\mathbf{w}$ and $\hat{\mathbf{w}}$ are the expected (with respect to $\mathcal{D}$ of fixed size) and current estimates of the parameters, and s is once again the size of the training set. The $\mathcal{D}$ parameter disappears from the first term since $\mathbf{w}$ is a free parameter in this setting rather than something determined by a training set $\mathcal{D}$, in contrast to $\hat{\mathbf{w}}$ in previous equations.

The gradient vector $\mathbf{g}(c)$ indexed by category/predictor pairs (c', i) is defined as follows:

$$g_{c'i}(c) = \frac{\partial}{\partial w_{c'i}} \pi(c, \mathbf{x}, \mathbf{w}) \quad (27)$$

$$= \begin{cases} \pi(c, \mathbf{x}, \mathbf{w})(1 - \pi(c, \mathbf{x}, \mathbf{w}))x_i & \text{c = c'} \\ -\pi(c, \mathbf{x}, \mathbf{w})\pi(c', \mathbf{x}, \mathbf{w})x_i & \text{otherwise.} \end{cases} \quad (28)$$

Computing the variance of the Taylor approximation (26) produces:

$$\begin{aligned} \mathrm{Var}[\pi(c, \mathbf{x}, \hat{\mathbf{w}})] &\simeq \mathrm{Var}[\mathbf{g}_n(c)(\hat{\mathbf{w}}_c - \mathbf{w}_c)] && (29) \\ &= \mathbf{g}(c)' F^{-1} \mathbf{g}(c) && (30) \end{aligned}$$

The asymptotics in (26) and the variance calculation of Equation 29 follow from normality of the maximum likelihood estimate:

$$\hat{\mathbf{w}} \sim \mathcal{N}(\mathbf{w}, F^{-1}). \tag{31}$$

F is the Fisher information matrix with dimensions $(k \cdot d) \times (k \cdot d)$ defined as follows:

$$F_{(ci)(c'j)} = \mathsf{E}_{(\mathbf{x},y)\sim\mathcal{D}} \begin{cases} x_i^2 \pi(c, \mathbf{x}, \mathbf{w})\pi(\neg c, \mathbf{x}, \mathbf{w}) + \frac{1}{\sigma_p^2} & c = c' \text{ and } i = j \\ x_i x_j \pi(c, \mathbf{x}, \mathbf{w})\pi(\neg c, \mathbf{x}, \mathbf{w}) & c = c' \text{ and } i \neq j \\ x_i x_j \pi(c, \mathbf{x}, \mathbf{w})\pi(c', \mathbf{x}, \mathbf{w}) & c \neq c'. \end{cases} \tag{32}$$

One final bit of algebra allows more efficient computation of the variance. Define $A_n(c) = \mathbf{g}_n(c)\mathbf{g}_n(c)'$, $A_n = \sum_c A_n(c)$ and $A = \sum_n A_n$, where n indexes individual observations in the pool. With these few tricks, a compact representation of the variance computation follows:

$$\begin{aligned} \sum_{n \in \mathrm{Pool}} \sum_c \mathrm{Var}\left[\hat{\pi}(c|\mathbf{x}_n)\right] &\simeq \sum_{nc} \mathbf{g}_n(c)' F^{-1} \mathbf{g}_n(c) && (33) \\ &= \sum_{nc} \mathrm{tr}\left\{\mathbf{g}_n(c)\mathbf{g}_n(c)' F^{-1}\right\} && (34) \\ &= \sum_{nc} \mathrm{tr}\left\{A_n(c) F^{-1}\right\} && (35) \\ &= \mathrm{tr}\left\{A F^{-1}\right\} && (36) \\ &\doteq \phi(\mathcal{D}, A). && (37) \end{aligned}$$

Using the variance estimated over the pool is intended to give an estimate of variance over the actual distribution of observations. As the pool size increases this is a reasonable assumption.

Equation 36 is the A-optimality objective function for multinomial regression with the A matrix that gives the method its name. Some choose to denote the A matrix $A(\mathbf{w})$ in order to make explicit the dependence of the matrix on the parameters. However, the $\phi(\mathcal{D}, A)$ notation for variance illustrates the dependence on the training set ($\mathcal{D}$) and validation sets (A), and will be useful in Section 4.3. We refer to the method as *variance reduction active learning*, noting that the greedy method we will employ in picking examples will not lead to optimal solutions.

The technique of A-optimality for logistic regression has been developed previously (Chaloner & Larntz, 1989; Davis & Prieditis, 1999) in the context of designing location/scale two parameter logistic regression experiments. Such two-parameter experiments are useful for determining the dosage of a compound that leads to an outcome (*e.g.* death in an animal subject)

at some probability, for instance 50% of the time. We are not aware of any previous use of the method in logistic regression models with more than two parameters or more than two categories. Nor are we aware of evaluations of the method in pool-based active learning of logistic regression.

4.3 How to Pick the Next Example

Equation (36) shows how to compute the expected variance of a fitted model using a fixed training set. We now need to derive a quantity that describes the expected benefit of labeling a new observation. The training set $\mathcal{D}$ consists of a sequence of observations: $\{(x_n, y_n)\}_1^N$. Using the current estimated model $\pi(y, \mathbf{x}, \hat{\mathbf{w}})$, the expected benefit of labeling observation $\mathbf{x}$ is:

$$\begin{aligned} \mathsf{E}[\text{Loss}] &= \pi(c_0, \mathbf{x}, \hat{\mathbf{w}})\phi(\mathcal{D} \cup (\mathbf{x}, c_0), A) \\ &+ \quad \vdots \\ &+ \pi(c_k, \mathbf{x}, \hat{\mathbf{w}})\phi(\mathcal{D} \cup (\mathbf{x}, c_k), A). \end{aligned} \tag{38}$$

Informally, equation 39 represents the possible changes in ϕ weighted by current estimates of the scenario's likelihood.

Ignoring model-fitting, the worst-case computational cost associated with picking a new example is:[3] $O(TNK^2(K + D^2) + TK^3D^3)$, where N is the number of pool examples used to create the A matrix, T is the number of candidates evaluated for inclusion in the training set, K is the number of categories and D are the number of predictors in the model. The N term may be reduced using Monte Carlo sampling from the pool. The term $TNK^2(K + D^2)$ corresponds to creation of the A matrix, while the term TK^3D^3 corresponds to inversion and multiplication by the F matrix. Model training can be safely ignored from such analysis when the training set size is small relative to pool size, as is the case in the evaluations of this study.

4.4 A Generalization to Other Loss Functions

Minimizing variance (20) is equivalent to minimizing squared loss:

$$L(\mathbf{p}, \mathbf{q}) = \sum_c (p_c - q_c)^2, \tag{39}$$

with vectors $\mathbf{p}$ and $\mathbf{q}$ defined with components $p_c = \mathsf{E}_{\mathcal{D}}[\pi(c, \mathbf{x}_n, \hat{\mathbf{w}}; \mathcal{D})]$ and $q_c = \pi(c, \mathbf{x}_n, \hat{\mathbf{w}}; \mathcal{D})$. The natural next step is to develop a technique applicable to other loss functions for these values of $\mathbf{p}$ and $\mathbf{q}$. Many common loss functions, including both squared and log loss, have the convenient property that they are twice differentiable and the second term of their Taylor

[3] We assume naive implementations for the matrix calculations in this analysis.

approximation disappears. The first three terms of a Taylor expansion of this class of loss functions produces an approximation:

$$L(\mathbf{p},\mathbf{q}) \simeq L(\mathbf{p},\mathbf{p}) + 0 + \ (\mathbf{p}-\mathbf{q})' \ \left\{ \frac{1}{2}\frac{\partial^2}{\partial \mathbf{q}^2} L(\mathbf{p},\mathbf{q})|_{\mathbf{q}=\mathbf{p}} \right\} \ (\mathbf{p}-\mathbf{q}). \tag{40}$$

Now, taking the expectation with respect to the training sets of size $\mathcal{D}$ ($\mathsf{E}_{\mathcal{D}}$) we have:

$$\mathsf{E}_{\mathcal{D}}[L(\mathbf{p},\mathbf{q})] \simeq L(\mathbf{p},\mathbf{p}) + \frac{1}{2}\mathsf{E}_{\mathcal{D}}[(\mathbf{p}-\mathbf{q})' \ \left\{ \frac{\partial^2}{\partial \mathbf{q}^2} L(\mathbf{p},\mathbf{q})|_{\mathbf{q}=\mathbf{p}} \right\} \ (\mathbf{p}-\mathbf{q})]. \tag{41}$$

In the special case of squared loss $L(\mathbf{p},\mathbf{q}) = \sum_c (p_c - q_c)^2$, the approximation is exact, and the variance minimization criteria (36) emerges:

$$\mathsf{E}_{\mathcal{D}}[L(\mathbf{p},\mathbf{q})] = \sum_c \mathsf{Var}[q_c], \quad \text{where} \tag{42}$$

$$\mathsf{Var}[q_c] = \mathsf{E}_{\mathcal{D}}[(q_c - \mathsf{E}_{\mathcal{D}}[q_c])^2]. \tag{43}$$

Unfortunately, not all loss functions are amenable to this analysis. For example, 0/1 loss is not differentiable. Further discussion of this technique can be found in (Buja, Stuetzle, & Shen, 2005).

4.5 A Log Loss Method of Active Learning

Applying the Taylor expansion method to log loss we find:

$$L(\mathbf{p},\mathbf{q}) \simeq -\sum_c p_c \log p_c + 0 + \sum_c \frac{1}{2p_c}\mathsf{Var}[q_c]. \tag{44}$$

The first term is a constant with respect to training set inputs. The third term is identical to the variance reduction criteria 36, but with the A matrix reweighted by a factor of $\frac{1}{2p_c}$. Furthermore, the computational complexity of implementing the log loss procedure remains identical to that of variance reduction.

As a reminder, the procedure estimates a log loss based on the expected value over training sets of fixed size $\mathsf{E}_{\mathcal{D}}$:

$$L(\mathbf{p},\mathbf{q}) = \sum_c \mathsf{E}_{\mathcal{D}}[\pi(c,\mathbf{x}_n,\hat{\mathbf{w}};\mathcal{D})] \log\left(\pi(c,\mathbf{x}_n,\hat{\mathbf{w}};\mathcal{D})\right) \tag{45}$$

rather than the correct probability distribution generating categories c given predictors $\mathbf{x}$:

$$L(\mathbf{p},\mathbf{q}) = \sum_c \mathsf{E}[y_c|x] \log\left(\pi(c,\mathbf{x}_n,\hat{\mathbf{w}};\mathcal{D})\right). \tag{46}$$

4.6 Applicability of the Approach to Conditional Exponential Models

The method of estimating variance relied on the ability to perform an approximation by means of Taylor series, compute the variance of the second term, and showing that the higher order terms vanish. What of the maximum entropy classifier (Section (Berger et al., 1996)) and conditional random field models (Section (Lafferty et al., 2001))? We expect that the variance estimation technique will carry over to these more general forms of conditional exponential models. Demonstrating this result is beyond the scope of the present work.

5 Evaluation

The evaluations in this study have specific goals: to discover which methods work in addition to why methods perform badly when they do. Towards this end, we assembled a suite of machine learning data sets consisting of a diverse number of predictors, categories and domains. In this section, we describe our evaluation methodology, present the most salient of our results and interpret their meaning. Necessarily, evaluation of the loss function methods require setting the parameters of evaluation in a way to make loss function strategies computationally tractable. It follows that the heuristics should be evaluated with the same parameter settings when/if applicable. Surprisingly, the evaluation of heuristic methods of this section revealed active learning is often worse than random sampling. There is a possibility that thesenegative results reflect a particularly unfortunate evaluation design decision rather than reflecting an underlying systemic problems with the heuristic methods. We systematically modify the evaluation design for evaluating the heuristics in order to rule out this possibility in Section 5.5.

5.1 Active Learning Methods and Method-Specific Parameter Settings

The evaluations consist of seven different methods of pool-based active learning in addition to two "straw men:" random sampling from the pool as well as random sampling combined with the bagging procedure. The active learning methods tested include: variance reduction (Equation 36), log loss reduction (Equation 44), minimum margin sampling and maximum entropy sampling (Section 3.1), QBB-MN and QBB-AM (Section 3.2), and classifier certainty (CC) (Section 3.3).

Several of the active learning methods require method-specific parameter settings. For example, the variance reduction, log loss reduction and CC methods require a random sample from the pool of some predetermined size to assess expected benefit of example labeling. In the case of variance reduction and log loss reduction the random sample composes the A matrix. All evaluations employ a sample size of 300 for assessing benefit of labeling.

Table 2 Descriptions of the data sets used in the evaluation. Included are counts of: the number of categories (Classes), the number of observations (Obs), the test set size after splitting the data set into pool/test sets (Test), the number of predictors (Pred), the number of observations in the majority category (Maj), and the training set stopping point for the evaluation (Stop).

Data Set	Classes	Obs	Test	Pred	Maj	Stop
Art	20	20,000	10,000	5	3635	300
ArtNoisy	20	20,000	10,000	5	3047	300
ArtConf	20	20,000	10,000	5	3161	120
Comp2a	2	1989	1000	6191	997	150
Comp2b	2	2000	1000	8617	1000	150
LetterDB	26	20,000	5000	16	813	300
NewsGroups	20	18,808	5000	16,400	997	300
OptDigits	10	5620	1000	64	1611	300
TIMIT	20	10,080	2000	12	1239	300
WebKB	4	4199	1000	7543	1641	300

The QBB methods, QBB-MN and QBB-AM rely on bagging, and so the evaluation requires a bag size setting. Following (McCallum & Nigam, 1998), the bag size is 3. Section 5.5 explores sensitivity of the results to the choice of 3.

5.2 Evaluation Data Sets and Data Set-Specific Evaluation Parameters

We tested these seven active learning methods on ten data sets (see Table 2 for summary of data sets). From the UCI machine learning repository of data sets (Blake & Merz, 1998) we used LetterDB (Frey & Slate, 1991) and OptDigits (Kaynak, 1995). We used the TIMIT database (Garofolo et al., 1993) to make predictions in a voice recognition domain. Web pages from the WebKB database (Craven et al., 2000) provided a document classification task. For additional document classification tasks we took the 20 NewsGroups topic disambiguation task (Mitchell, 1997; Nigam, Lafferty, & McCallum, 1999), along with two data sets made from different subsets of the NewsGroups categories. We used three artificial data sets to explore the effects of adding different types of noise to data.

5.2.1 Data Set Evaluation Parameters Several parameters of the evaluation are intrinsic to the data sets. For instance, how many random examples should serve as a "seed" set before active learning begins? This section presents results for seed size 20. Other starting seed sizes are shown in Section 5.5 to have minimal effect.

Another choice is the stopping point for the evaluation. The evaluation uses 300 as a stopping point except when there is good reason not to. Smaller stopping points are used for three (of ten) data sets: ArtConf, Comp2a, and Comp2b, and the sections on processing of the individual data sets present

the reasons for these decisions. A summary of the actual stopping points is included in Table 2.

The test set size for each data set is another tunable parameter. The data set is split into a pool and test set as part of a 10 fold cross validation. In other words this splitting occurs 10 times with ten results averaged into a final accuracy. Table 2 shows test set sizes used for different data sets. As described below, the results are not sensitive to these exact values; What is important to the qualitative results of this and subsequent sections is that both the pool and test set are quite large, facilitating hypothesis testing on the averaged results.

5.2.2 Natural Data Sets Seven of the evaluation data sets are "natural," that is they come from some real world domain rather than an artificial stochastic generation engine. The data sets are: LetterDB, OptDigits, TIMIT, NewsGroups, Comp2a, Comp2b, and WebKB. The paragraphs below describe the sources and pre-processing steps for each of these natural data sets.

The LetterDB database consists of 20,000 instances of uppercase capital letters in a variety of fonts and distortions. The predictors are 16 numerical attributes computed from statistical moments and edge counts. LetterDB was the most computationally intensive data set we attempted loss-based active learning on, and evaluations employing seed size 20 took approximately three weeks to run to completion using ten machines (each machine ran one tenth of the ten-fold cross-validation). The OptDigits data set consists of 5,620 examples of handwritten digits from 43 people. The predictors consist of counts of the number of "on bits" in each of 64 regions.

We processed the WebKB and NewsGroups data set by running a stop word list and using a count cutoff of 5 or fewer documents. Numbers were converted to a generic N token. The Comp2a data set consists of the `comp.os.ms-windows.misc` and `comp.sys.ibm.pc.hardware` subset of NewsGroups used previously in an active learning evaluation (Roy & McCallum, 2001). The Comp2b data set consists of `comp.graphics` and `comp.windows.x` categories from the same study. We employed a count cut-off of 2 or fewer documents to trim down the vocabulary for these two binary-category data sets.

Of the four document classification problems only the two binary classification problems proved feasible to test the objective function approaches due to computational limitations. Implementation tricks included elimination of non-occurring token counts from the matrix computations of the loss function methods in addition to application of the Sherman-Morrison formula. Due to computational time costs of the loss function methods, we stopped training after 150 examples for these two document data sets.

The TIMIT database was formatted into $10,080$ points consisting of the first 12 Bark-scale PLP coefficients (excluding coefficient 0, which usually hurts performance). The points represent the male speakers from dialect

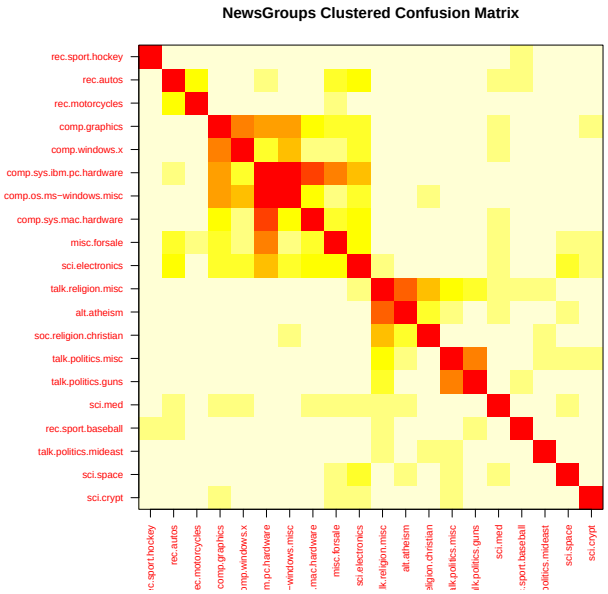

Fig. 1 Clusters of topics based on distance measured on confusion matrix rows. The confusion matrix was computed in this case after training on the entire pool and averaging over 10 pool/test splits.

regions 1 through 3. The goal is to predict which of 20 different vowel sounds are uttered.

5.2.3 Artificial Data Sets We constructed three artificial data sets to explore the effects of two different types of noise on the modeling performance. The first type of noise is the prediction residual error (Equation 25). As a reminder, this is the portion of squared error that is independent of training set size. The residual error may be estimated when the training set is sufficiently large that the mean squared error (Equation 19) becomes negligible.

We explore the effects of increased residual error using two similar artificial data sets. The first, named Art, consists of 20 categories and 5 predictors with observations generated according to: $\mathbf{x}_n \sim \mathcal{N}(0, I)$ and $\mathbf{w}_c \sim \mathcal{N}(0, 5I)$. Art serves as a noise-free baseline data set. The second data set, ArtNoisy, is generated similarly except the probabilities are formed by adding a noise term to the dot product calculation of Equation 5: $\mathbf{w}_c \cdot \mathbf{x}_n + G_{nc}$, where $G_{nc} \sim \mathcal{N}(0, 10)$. Thus, ArtConf models the presence of unknown features that influence the true probabilities of an outcome: a form of noise that will increase residual error.

A second type of noise involves different levels of confusion among the categories. For instance, when categories are related by clusters, we would

expect members of the same cluster to be more difficult to disambiguate than two categories in different clusters. The NewsGroups data set is an example of a data set with intrinsic category clusters as can be seen in the list of topics or by clustering the rows of a confusion matrix (see Figure 1 for list of topics and result of clustering).

One hypothesis we would like to explore is that heuristics that sample uncertain regions should fall prey to intrinsically uncertain regions that have little teaching value. We generate a third data set, ArtConf, consisting of two regions of predictor space and 20 categories in order to test our ability to construct intrinsically confusable regions. In the first region, predictor no. 1 is set to 1, all remaining 5 predictors are set to 0 and categories 0 or 1 are assigned with equal probability. Region 1 is the intrinsically uncertain region, and 33% of the observations inhabit this space. In region 2, predictor no. 1 is set to 0, and the remaining 18 categories generate the remaining 5 predictors according to a multinomial naive Bayes model (McCallum & Nigam, 1998). In other words, categories 1 and 2 are intrinsically hard to disambiguate, but the remaining categories are relatively easy to disambiguate.

The ArtConf data set has the property that learning the generation function takes relatively few examples. This is a byproduct of the simplistic generation process. As a result, tangible learning improvement disappears by 150 examples. Hypothesis testing results, box plots and means are reported at a stopping point of 120 observations for this reason.

5.3 Evaluation Design

An average of results over 10 random pool/test set splits formed the core of our evaluation technique. Table 2 indicates the pool and test set sizes; to compute the pool set size, subtract the test set size from the number of observations in the entire data set. On each of the 10 runs, the same random seed examples of size 20, 50, 100 or 200 were given to the learners which proceeded to use their example selecting function to select new examples. Only results for the seed size 20 are reported; results from alternative starting points look more and more like random observation sampling as the seed size increases. Results for the alternative starting points are available on request.

Results are reported once the learner has reached the data set stopping points given in Table 4. At each iteration of observation selection, 10 candidates were chosen at random from the pool and the tested method chose the next example from those 10. The number 10 was used because larger numbers cause variance, log loss and CC methods to slow proportionately (see discussions of asymptotics, Section 4.3). On the other hand, fixing the sample size at 10 allows for fair comparison across all methods. Section 5.5 examines the sensitivity of the heuristic methods' performance to this parameter.

Table 3 Average accuracy and squared error (Equation 18, left hand side) results for the tested data sets when the entire pool is used as the training set. The data sets are sorted by squared error as detailed in Section 5.4.

Data Set	Accuracy	Squared Error
TIMIT	0.525	0.616
ArtNoisy	0.602	0.52
LetterDB	0.764	0.352
NewsGroups	0.820	0.296
ArtConf	0.844	0.155
WebKB	0.907	0.143
Art	0.919	0.130
Comp2a	0.885	0.086
Comp2b	0.889	0.083
OptDigits	0.964	0.059

All evaluations employed a logistic regression using the regularization $\sigma_p^2 = 1$ for 100 iterations or convergence for the seed set. Once additional data was added, the model parameters were updated 20 iterations or until convergence. In generating results for straw men bagging and random sampling, the same seed examples are used, and then followed by additional random sampling to form training sets of appropriate size.

5.4 Evaluation Results

This section presents several different views of the evaluation results incorporating various tables and figures. A guiding principle to keep in mind is that each of these devices present the same evaluation, but explores different components. For instance, Figures 2-5 present learning curves for each of the data set in the right column, while the left column shows Box plots of the distribution of accuracies at the stopping point (300 observations in most cases). Table 3 shows the accuracies attainable by training on the entire pool of unlabeled data. This information gives an understanding of how much continued labeling of training data can help. The learning curves in Figures 2-5 convey the same information as a horizontal line on top of the y-axis. Table 4 contains the result of a hypothesis test on the mean stopping point accuracy: comparing different alternatives to random sampling.

Variance and log loss reduction gave the best results; they provided above-random performance on four of the data sets while never giving less than random performance. The results do not support any definitive reason to draw favorites between variance or log loss. Though not statistically significant, the weak performance on the TIMIT data set by variance reduction suggests favoring log loss.

Maximum entropy sampling results are the worst of all methods tested. In order to assess what properties of the data sets cause entropy sampling to

Table 4 Results of hypothesis tests comparing bagging and seven active learning method accuracies to random sampling at the final training set size. '+' indicates statistically significant improvement and '-' indicates statistically significant deterioration. 'NA' indicates 'not applicable.' Figures 2-5 display the actual results used for hypothesis testing as a box plot.

Data Set	random	bagging	variance	log loss
Art	NA	-	+	+
ArtNoisy	NA	-	+	+
ArtConf	NA			
Comp2a	NA	-		
Comp2b	NA			
LetterDB	NA	-	+	+
NewsGroups	NA	-	NA	NA
OptDigits	NA		+	+
TIMIT	NA	-		
WebKB	NA	-	NA	NA

	CC	QBB-MN	QBB-AM	entropy	margin
Art	+	+	+	+	+
ArtNoisy		+		-	+
ArtConf				-	-
Comp2a	-				
Comp2b					
LetterDB	+	-	+	-	+
NewsGroups	NA	-	-	-	-
OptDigits	+	+	+	+	+
TIMIT	-		+	-	+
WebKB	NA	+	+	+	+

fail we report the residual error (Equation 25) of each data set after training on the entire pool in Table 3. The data sets sort neatly by noise, with entropy sampling failing on more noisy data such as TIMIT and performing at least as well as random for all data sets less noisy than WebKB.

Margin sampling results are quite good except for two notable failures on the ArtConf and NewsGroups data set. The artconf data set was constructed in such a way as to sucker uncertainty sampling methods into sampling regions with low utility. One hypothesis we had was that in the NewsGroups data set we would see similar behavior with increased sampling among more confusible categories, for instance over sampling of computer-related topics. This did not occur in practice on any of the data sets (Schein, 2005).

Table 5 gives an alternative explanation for margin's samplings lackluster performance on the Newsgroups data set; the ability to identify the two categories forming the margin in the NewsGroups data set is much harder than any of the data sets we tested. This is a problem specific to margin sampling in the multi-category active learning, and has not been reported

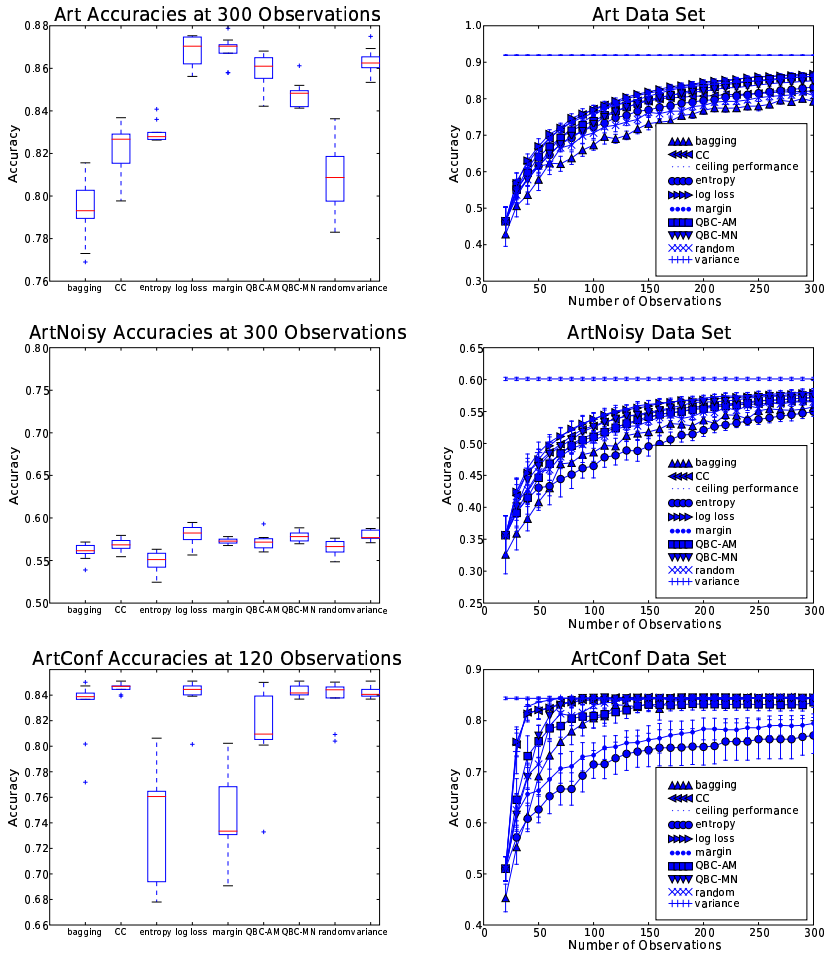

Fig. 2 Box plots and learning curves for Art, ArtNoisy and ArtConf data sets. Box plots show the distribution of the accuracy at the training set stopping point. Confidence bars indicate the variability of competing active learning schemes.

before. Still, margin sampling provides performance competitive to the alternative heursistics at the best computational cost.

Before examining the QBB method results it is useful to analyze bagging since it is a key ingredient. The results for bagging are almost entirely negative, a possibility anticipated in the bagging literature (Breiman, 1996). Our own results in measuring variance (Schein, 2005) indicate that variance is usually small in comparison to squared error. In contrast, bagging is known to work well with highly unstable methods such as decision trees, which are associated with large amounts of variance. We speculate that it would take very many bag members to improve the variance of the logistic

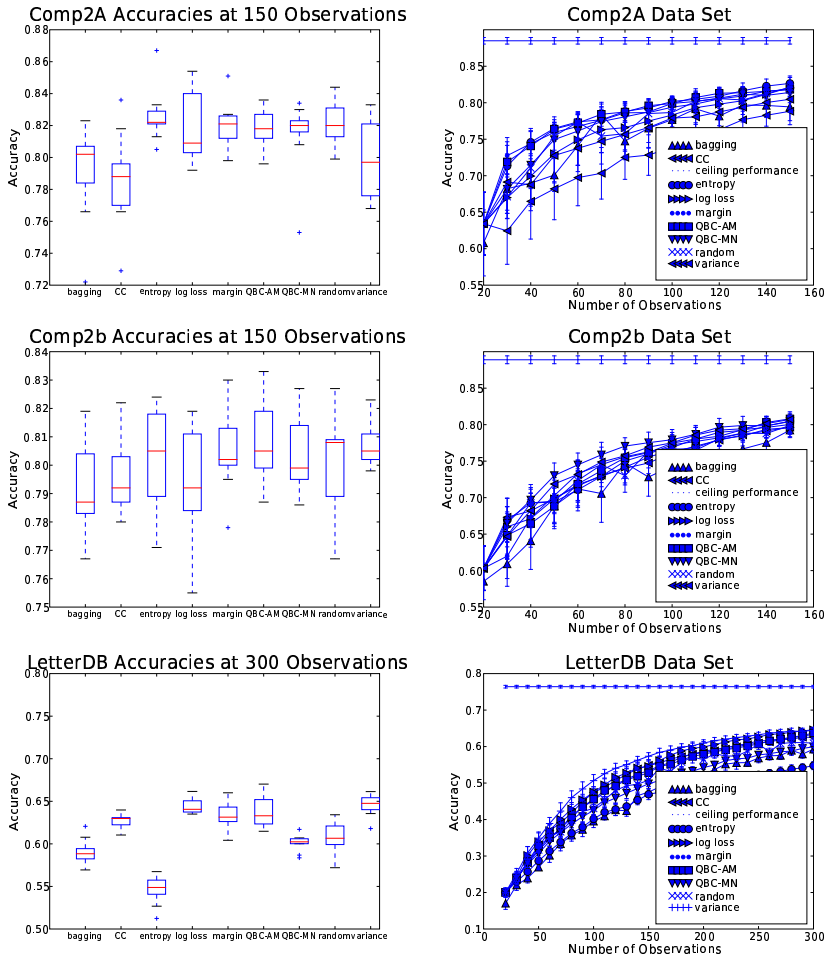

Fig. 3 Box plots and learning curves for Comp2a, Comp2b and LetterDB data sets. Box plots show the distribution of the accuracy at the training set stopping point. Confidence bars indicate the variability of competing active learning schemes.

regression model. MacKay (MacKay, 1992) gives a parametric solution to the problem of variance reduction of logistic regression that may prove more expedient.

5.5 Alternative Evaluation Design Decisions

The results in Section 5.4 suggest that the experimental design methods more reliably match or beat random performance than the heuristic methods. There is a possibility that the heuristic methods are handicapped under

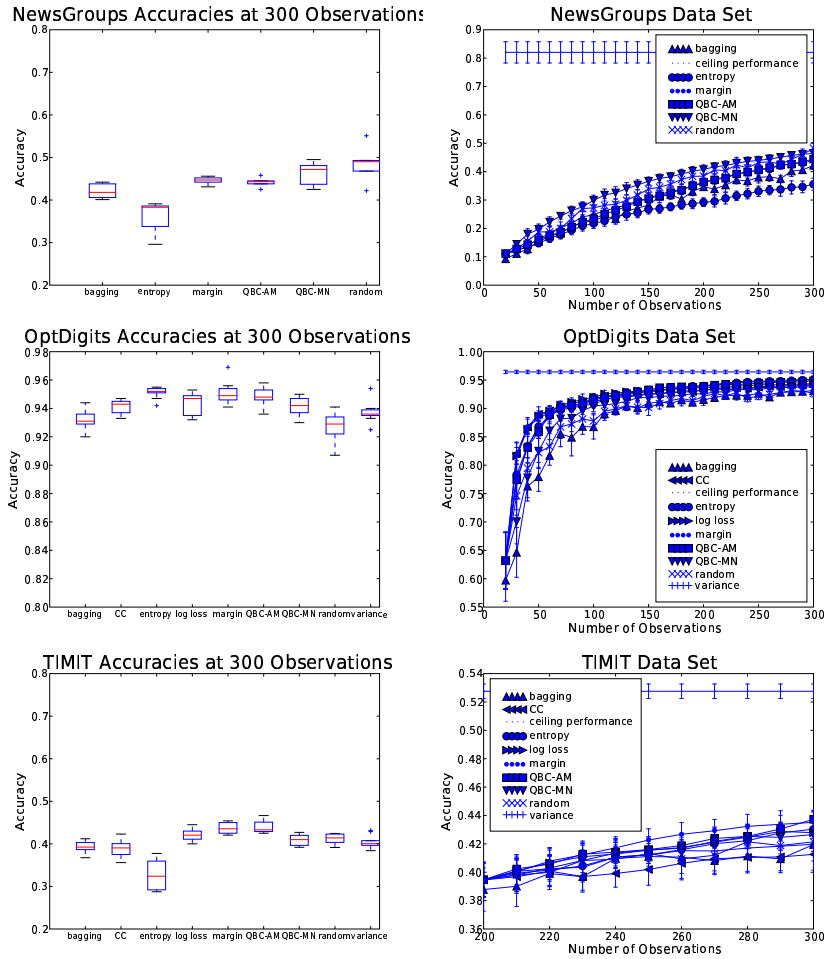

Fig. 4 Box plots and learning curves for NewsGroups, OptDigits and TIMIT data sets. Box plots show the distribution of the accuracy at the training set stopping point. Confidence bars indicate the variability of competing active learning schemes.

certain experimental parameter settings, and so we examine the alternative settings. Space restrictions prevent displaying the full set of tables and plots as shown in Section 5.4. Instead, the alternative evaluation design choices are described briefly below along with the experimental outcomes. (Schein, 2005) provides greater details of these experimental results.

1. Evaluation Starting Points. The evaluations reported in the previous sections begin with 20 random examples. Experiments employing a starting point of 300 and stopping point of 600 produced no substantial changes in the outcomes.

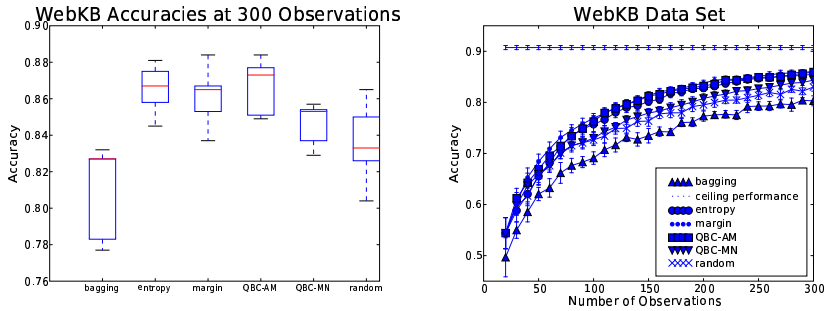

Fig. 5 Box plot and learning curves for the WebKB data set. The Box plot shows the distribution of the accuracy at the training set stopping point. Confidence bars indicate the variability of competing active learning schemes.

Table 5 The average percentage of matching test set margins when comparing models trained on data sets of size 300 to a model trained on the pool. Margins match if they are formed from the same pair of categories. Ten repetitions of the experiment produce the averages below.

Data Set	Correct Margin Percentage
Art	64.1
ArtNoisy	58.6
ArtConf	51.1
LetterDB	36.8
NewsGroups	15.1
OptDigits	57.8
TIMIT	34.4

2. Larger Candidate Sample Size. The evaluations reported in the previous sections allow the algorithms to select a single observation from 10 random candidates at each iteration of Algorithm 1. Changing the number of candidates to 300 has no substantial effects on the outcomes.
3. Larger Bag Sizes for QBB Methods. The evaluations reported in the previous sections use a bag size of 3. Increasing the bag size to 15 has no subtantial effects on the outcomes.

In summary, the experiments under alternative evaluation choices confirm the negative results of the previous section, and suggest that the heuristics are prone to failure in a wide range of settings.

6 Conclusions

The evaluations establish that loss function active learning of logistic regression is the most robust strategy available, providing attractive results

yet never performing worse than random sampling. Future work in active learning using logistic regression will benefit from evaluating against these gold standard methods. Furthermore, we have dismissed a complaint that the method is computationally intractable by evaluating these methods on a wide variety of domains.

The results also expose the weaknesses of many of the active learning algorithms. The loss function methods have the disadvantage of memory and computational complexity, and we were unable to evaluate them on two of the larger document classification tasks. All of the heuristic methods fail to beat random sampling on some portion of the evaluation. The result is so surprising that a separate section (5.5) is included to verify that negative heuristic performance is not an artifact of an "unlucky" evaluation design.

We find that most heuristics perform roughly equally well in comparison to each other, but it is easier to analyse the cause of failure among the simplest heuristics. In the case of uncertainty sampling using the Shannon entropy measure of uncertainty, bad performance goes hand in hand with noise, as defined by the portion of squared error that is training set size independent. For margin sampling, inability to identify the pairs of categories forming the margin on multi-category problems is the biggest danger, as seen on the NewsGroups data set. In spite of this observation, margin sampling competes favorably with the alternative heurstics and is the most computationally efficient method examined. Improving the performance of this method in the multi-category setting remains a promising direction for future research.

References

Abe, N., & Mamitsuka, H. (1998). Query learning strategies using boosting and bagging. In *Proceedings of the 15th international conference on machine learning (icml1998)* (p. 1-10).

Angluin, D. (1987). Learning regular sets from queries and counterexamples. *Information and Computation*, *75*, 87-106.

Banko, M., & Brill, E. (2001). Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39'th annual acl metting (acl2001).*

Baum, E. B. (1991). Neural net algorithms that learn in polynomial time from examples and queries. *IEEE Transactions on Neural Networks*, *2*(1).

Berger, A. L., Della Pietra, S. A., & Della Pietra, V. J. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, *22*(1), 39–71.

Bickel, P. J., & Doksum, K. A. (2001). *Mathematical statistics* (2nd ed., Vol. 1). Prentice Hall, New Jersey.

Blake, C., & Merz, C. (1998). *UCI repository of machine learning databases.*

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123-140.

Buja, A., Stuetzle, W., & Shen, Y. (2005). Degrees of boosting: A study of loss functions for classification and class probability estimation. *working paper*.

Chaloner, K., & Larntz, K. (1989). Optimal bayesian design applied to logistic regression experiments. *Journal of Statistical Planning and Inference*, *21*, 191-208.

Chen, J., Schein, A. I., Ungar, L. H., & Palmer, M. S. (2006). An empirical study of the behavior of active learning for word sense disambiguation. In *Proceedings of the 2006 human language technology conference - north american chapter of the association for computational linguistics annual meeting HLT-NAACL 2006.*

Cohn, D. A. (1996). Neural network exploration using optimal experimental design. *Neural Networks*, *9*(6), 1071-1083.

Cohn, D. A. (1997). Minimizing statistical bias with queries. In *Advances in neural information processing systems 9.* MIT Press.

Craven, M., DiPasquo, D., Freitag, D., McCallum, A. K., Mitchell, T. M., Nigam, K., et al. (2000). Learning to construct knowledge bases from the World Wide Web. *Artificial Intelligence*, *118*(1/2), 69–113.

Dagan, I., & Engelson, S. P. (1995). Committee-based sampling for training probabilistic classifiers. In *International conference on machine learning* (p. 150-157).

Darroch, J. N., & Ratcliff, D. (1972). Generalized iterative scaling for log-linear models. *Annals of Mathematical Statistics*, *43*, 1470-1480.

Davis, R., & Prieditis, A. (1999). Designing optimal sequential experiments for a bayesian classifier. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *21*(3).

Freund, Y., Seung, H. S., Shamir, E., & Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine Learning*, *28*, 133-168.

Frey, P. W., & Slate, D. J. (1991). Letter recognition using holland-style adaptive classifiers. *Machine Learning*, *6*(2).

Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., & Dahlgren, N. (1993). *Darpa timit acoustic-phonetic continuous speech corpus cd-rom.* NIST.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, *4*, 1-58.

Gilad-Bachrach, R., Navot, A., & Tishby, N. (2003). *Kernel query by committee (KQBC)* (Tech. Rep. No. 2003-88). Leibniz Center, the Hebrew University.

Hosmer, D. E., & Lemeshow, S. (1989). *Applied logistic regression.* John Wiley and Sons, Inc.

Hwa, R. (2004). Sample selection for statistical parsing. *Computational Linguistics.* (to appear)

Hwang, J.-N., Choi, J. J., Oh, S., & Marks, R. J. (1991). Query-based learning applied to partially trained multilayer perceptrons. *IEEE Transactions on Neural Networks*, *2*(1).

Jin, R., Yan, R., Zhang, J., & Hauptmann, A. G. (2003). A faster iterative scaling algorithm for conditional exponential model. In *Proceedings of the twentieth international conference on machine learning (icml-2001), washington, d.c.*

Kaynak, C. (1995). *Methods of combining multiple classifiers and their applications to handwritten digit recognition.* Unpublished master's thesis, Bogazici University.

Lafferty, J. D., McCallum, A., & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning* (pp. 282–289). Morgan Kaufmann Publishers Inc.

Lewis, D. D., & Gale, W. A. (1994). A sequential algorithm for training text classifiers. In W. B. Croft & C. J. van Rijsbergen (Eds.), *Proceedings of SIGIR-94, 17th ACM international conference on research and development in information retrieval* (pp. 3–12). Dublin, IE: Springer Verlag, Heidelberg, DE.

MacKay, D. J. C. (1991). *Bayesian methods for adaptive models.* Unpublished doctoral dissertation, California Institute of Technology.

MacKay, D. J. C. (1992). The evidence framework applied to classification networks. *Neural Computation*, *4*(5), 698-714.

Malouf, R. (2002). *A comparison of algorithms for maximum entropy parameter estimation.*

McCallum, A., & Nigam, K. (1998). Employing em in pool-based active learning for text classification. In *Proceedings of the 15th international conference on machine learning (icml1998).*

McCullagh, P., & Nelder, J. A. (1989). *Generalized linear models* (2 ed.). CRC Press.

Melville, P., & Mooney, R. (2004). Diverse ensembles for active learning. In *Proceedings of the 21st international conference on machine learning (icml-2004)* (p. 584-591).

Mitchell, T. M. (1997). *Machine learning.* McGraw-Hill.

Nigam, K., Lafferty, J., & McCallum, A. (1999). Using maximum entropy for text classification. In *Ijcai-99 workshop on machine learning for information filtering.*

Nocedal, J., & Wright, S. J. (1999). *Numerical optimization.* Springer-Verlag.

Roy, N., & McCallum, A. (2001). Toward optimal active learning through sampling estimation of error reduction. In *Proc. 18th international conf. on machine learning* (pp. 441–448). Morgan Kaufmann, San Francisco, CA.

Saar-Tsechansky, M., & Provost, F. (2001). Active learning for class probability estimation and ranking. In *Proc. of the International Joint Conference on Artificial Intelligence* (pp. 911–920).

Schein, A. I. (2005). *Active learning for logistic regression.* Dissertation in Computer and Information Science, The University of Pennsylvania.

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. In *Computational learning theory* (p. 287-294).

Steedman, M., Hwa, R., Clark, S., Osborne, M., Sarkar, A., Hockenmaier, J., et al. (2003). Example selection for bootstrapping statistical parsers. In *the proceedings of the annual meeting of the north american chapter of the acl, edmonton, canada.*

Tang, M., Luo, X., & Roukos, S. (2002). Active learning for statistical natural language parsing. In *Acl 2002.*