

Uncertainty-Based Active Learning with Instability Estimation for Text Classification

JINGBO ZHU, Northeastern University, China
MATTHEW MA, Scientific Works

This article deals with pool-based active learning with uncertainty sampling. While existing uncertainty sampling methods emphasize selection of instances near the decision boundary to increase the likelihood of selecting informative examples, our position is that this heuristic is a surrogate for selecting examples for which the current learning algorithm iteration is likely to misclassify. To more directly model this intuition, this article augments such uncertainty sampling methods and proposes a simple *instability*-based selective sampling approach to improving uncertainty-based active learning, in which the instability degree of each unlabeled example is estimated during the learning process. Experiments on seven evaluation datasets show that instability-based sampling methods can achieve significant improvements over the traditional uncertainty sampling method. In terms of the average percentage of actively selected examples required for the learner to achieve 99% of its performance when training on the entire dataset, instability sampling and sampling by instability and density methods achieve better effectiveness in annotation cost reduction than random sampling and traditional entropy-based uncertainty sampling. Our experimental results have also shown that instability-based methods yield no significant improvement for active learning with SVMs when a popular sigmoidal function is used to transform SVM outputs to posterior probabilities.

Categories and Subject Descriptors: I.2.7 [Computing Methodologies]: Artificial Intelligence—*Natural Language Processing*; I.2.6 [Computing Methodologies]: Artificial Intelligence—*Learning*

General Terms: Algorithms, Theory, Language

Additional Key Words and Phrases: Active learning, uncertainty sampling, instability estimation, text classification, data annotation

ACM Reference Format:

Zhu, J. and Ma, M. 2012. Uncertainty-Based active learning with instability estimation for text classification. *ACM Trans. Speech Lang. Process.* 8, 4, Article 5 (February 2012), 21 pages.
DOI = 10.1145/2093153.2093154 <http://doi.acm.org/10.1145/2093153.2093154>

1. INTRODUCTION

Supervised learning methods generally estimate their parameters using labeled training data. However, creating a large labeled training corpus is expensive and time consuming in some real-world Natural Language Processing (NLP) applications such as word sense disambiguation [Chen et al. 2006; Zhu and Hovy 2007; Chan and Ng 2007] and Text Classification (TC) [Lewis and Gale 1994; McCallum and Nigam 1998a; Tong and Koller 2001], and is often a bottleneck to build a supervised classifier in many

This work was supported in part by the National Science Foundation of China (60873091; 61073140), Specialized Research Fund for the Doctoral Program of Higher Education (20100042110031) and the Fundamental Research Funds for the Central Universities.

Authors' addresses: J. Zhu (corresponding author), Key Laboratory of Medical Image Computing (Ministry of Education), Natural Language Processing Laboratory, Northeastern University, China; email: zhujingbo@mail.neu.edu.cn; M. Ma, Scientific Works, Princeton, NJ; email: mattma@ieee.org.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2012 ACM 1550-4875/2012/02-ART5 \$10.00

DOI 10.1145/2093153.2093154 <http://doi.acm.org/10.1145/2093153.2093154>

real-world domains. Among techniques used to address this issue, active learning is a widely used framework in which a learner has the ability to automatically select the most informative unlabeled examples for human annotation [Cohn et al. 1994; Seung et al. 1992], also referred to as *selective sampling*. An active learner starts with a small initial labeled set, carefully selects a few additional unlabeled examples for human annotation, and then uses this newly gained knowledge to retrain itself and request the next example. Active learning techniques are invaluable in applications where unlabeled data is easy to collect, and labeling of data is expensive.

Active learning aims to minimize the amount of human labeling effort required for a supervised classifier to achieve a satisfactory performance [Cohn et al. 1996; Settles 2009]. In this article, we focus on *pool-based active learning* in which the learner chooses the most informative unlabeled instances from a pool of unlabeled instances for human annotation, more specifically *uncertainty sampling* [Lewis and Gale 1994]. In tasks such as text classification [Lewis and Gale 1994; McCallum and Nigam 1998a] and word sense disambiguation [Chen et al. 2006; Zhu and Hovy 2007], the size of the labeled training dataset required to reach a given classification accuracy can be significantly reduced by using uncertainty-based selective sampling techniques.

The motivation behind this work is that misclassified unlabeled examples in active learning may convey more information than correctly classified unlabeled examples. If we were able to determine misclassifications automatically, machine learning of classifiers would be trivial. However, in practice, it is almost impossible to exactly determine which unlabeled example is misclassified before asking for human labeling because the true label of each unlabeled example is unknown in advance. One line of thinking is that the switching of a classification decision at the decision boundary due to model updates during active learning may be attributable to the fact that the unlabeled example is misclassified. We consider the identification of misclassified unlabeled examples as a problem of finding the most unstable unlabeled examples, namely *instability estimation*. We hypothesize that the most unstable unlabeled example is likely to be misclassified during an active learning process. This article thus presents a new selective sampling approach that estimates the instability of each unlabeled example, referred to as *instability sampling*. A high instability value is assigned to an example if such an example's classification decision changes when the system is retrained after each round of active learning. The key technical question in this framework is how to estimate the instability of each unlabeled example during the active learning process. To do this, we propose to calculate the uncertainty degree of each example at the current learning iteration, and additionally consider the change of its labels and uncertainty degrees during recent consecutive learning iterations. Experiments on seven publicly available evaluation datasets have shown that our instability sampling methods can achieve significant improvements over competing methods. Our instability sampling methods are easy to implement, and can be applied to several different learners, such as Maximum Entropy (MaxEnt), Naïve Bayes (NB), and Support Vector Machines (SVMs).

2. GENERAL ACTIVE LEARNING PROCESS

Formally, pool-based Active Learning (AL) is a two-stage process in which first a small number of labeled examples and a large number of unlabeled examples are collected and used to train a learner. This is followed by the iterative process of querying unlabeled examples, receiving labels for the queried examples, and retraining the learner. During the querying stage of the AL procedure, the active learning is given access to each unlabeled example, and determines whether or not to query the oracle for the label. Initially, the learner is not capable of reliably estimating instance label informativeness, making the labeling of arbitrary unlabeled examples informative for the

ALGORITHM 1: Pool-based Active Learning

Input: Initial small training set L , and pool of unlabeled data set U

Use L to train the initial classifier C

Repeat

- Use the current classifier C to label all unlabeled examples in U
- Select the most informative examples¹ from the unlabeled pool U , and ask an oracle for the label
- Augment L with new examples, and remove them from U
- Use L to retrain the current classifier C

Until the predefined stopping criterion SC is met.

Fig. 1. General pool-based active learning algorithm.

active learner. As the learning process continues, the ability of the learner improves along with its ability to predict the informativeness of querying unlabeled examples. At this point, the learner will discard most unlabeled examples as noninformative, and select exclusively the most informative examples for human labeling. The general AL process can be summarized as follows (see Figure 1).

In this article, our discussion focuses on the problem of uncertainty-based AL with supervised classification, more precisely *uncertainty sampling* [Lewis and Gale 1994]. The well-known *entropy* is a popular uncertainty measurement widely used in previous studies on uncertainty sampling [Tang et al. 2002; Chen et al. 2006; Zhu and Hovy 2007]. The uncertainty measurement function based on the entropy criterion can be expressed as

$$H(x) = - \sum_{y \in Y} P(y|x) \log P(y|x), \quad (1)$$

where $H(\cdot)$ is the uncertainty measurement function based on the entropy estimation of the classifier's posterior distribution. In uncertainty sampling, the selection of the most informative example u can be formulated as

$$u = \arg \max_{x \in U} H(x). \quad (2)$$

3. INSTABILITY-BASED SAMPLING

3.1. Motivation Analysis

In pool-based AL settings, the goal is to learn the parameters of a high-performance classifier with minimal human labeling effort. This may be achieved by attempting to select what are believed to be the most informative examples for human labeling during each round of AL [Tong and Koller 2001; Seung et al. 1992]. The heuristic used by traditional uncertainty sampling is to select new unlabeled examples on which the learner has low confidence regarding its prediction. In other words, the motivation behind uncertainty sampling is to find these most uncertain unlabeled examples near decision boundaries and to use them to clarify the position of the decision boundaries [Chen et al. 2006; Zhu et al. 2008a].

In active learning, since correctly classified unlabeled examples are likely to be already sufficiently determined by the model, misclassified unlabeled examples may

¹To decrease the number of times a learner is retrained during the active learning process, we use a batch mode sample selection to label the m most informative unlabeled examples at each learning cycle.

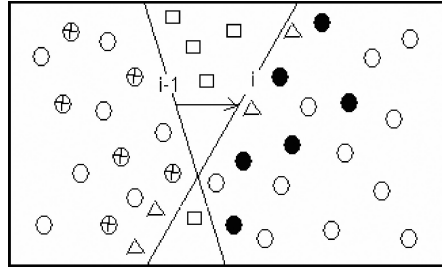


Fig. 2. Two thin lines represent decision boundaries generated at the $i - 1^{th}$ and i^{th} consecutive learning cycles. Blank circles, rectangles, and triangles denote three types of unlabeled examples for convenience of presentation. Solid circles (in the right area) and cross circles (in the left area) denote labeled samples with different labels in training data.

convey more information than correctly classified unlabeled examples, and consequently, learning misclassified examples has a higher chance to refine the position of current decision boundaries at the next learning iteration. In principle, some examples near current decision boundaries can be assumed to be most likely misclassified in traditional uncertainty sampling. However, in practice, it is an open issue how to identify which unlabeled example is misclassified before asking for human labeling, because the true label of each unlabeled example is unknown in advance. To address this challenge, we consider the problem of identifying misclassified unlabeled examples as the problem of finding the most *unstable* points (unlabeled examples) in the pool. An unstable point generally lies near the current decision boundary. The most unstable points within our framework are unlabeled examples with maximum uncertainty and different label predictions during recent consecutive learning cycles. We give an example to explain our motivation, as shown in Figure 2.

Figure 2 depicts two types of the most unstable examples marked as blank rectangles and blank triangles. Blank rectangles denote some unlabeled examples that are assigned to two different labels during the two recent consecutive learning cycles.² Blank triangles denote some unlabeled examples that become more uncertain at the i^{th} learning cycle compared to those at the $i - 1^{th}$ learning cycle. We think that both types of the most unstable examples are likely to be misclassified at the i^{th} learning cycle, and that labeling these unstable examples can help to clarify the decision boundaries, thus providing greater information gain to the active learner. In such a case, our methods tend to select these blank rectangles (for the label-sensitive case) and blank triangles (for the label-insensitive case) for active learning at the i^{th} learning cycle, as will be discussed in the next section.

3.2. Instability Estimation

Based on the preceding motivation analysis, this article introduces a new concept of *instability sampling* that selects the most unstable examples for labeling. It estimates the instability of each unlabeled example in terms of two factors of uncertainty and label prediction. In instability sampling, the selection of the most informative example u can be formulated as

$$u = \arg \max_{x \in U} IS(x), \quad (3)$$

where $IS(x)$ denotes the instability degree of an unlabeled example x .

²For example, an unlabeled example x was classified into class A at the $i - 1^{th}$ iteration, and class B at the i^{th} iteration.

To assess the instability of an unlabeled example, the simplest case is to consider the two recent consecutive learning cycles for instability estimation, as shown in Figure 2. The most unstable unlabeled examples can be determined based on the changes of their label predictions and uncertainty scores during recent consecutive learning cycles. The instability of an unlabeled example can be estimated using label prediction and uncertainty criteria. Our instability sampling techniques favor unlabeled examples with maximum instability.

This section presents two types of instability sampling techniques: *label-insensitive* and *label-sensitive* methods. Label-Insensitive Instability Sampling (LIIS) techniques select what is believed to be the most informative example from a set of unlabeled examples that become more uncertain during recent consecutive learning cycles, regardless of the changes of their label predictions, for example, the examples marked by triangles in Figure 2. Label-Sensitive Instability Sampling (LSIS) techniques select the most informative example from the set of unlabeled examples that have high uncertainty and different label predictions during recent consecutive learning cycles, for example, the examples marked by rectangles in Figure 2.

Without loss of generality, given an unlabeled example x at the i^{th} learning cycle, its instability value $IS_{LI}(x)$ in label-insensitive instability sampling can be estimated by³

$$IS_{LI}(x) = H^i(x) + \sum_{i-l < k \leq i} (H^k(x) - H^{k-1}(x)), \quad (4)$$

where $H^i(x)$ denotes the uncertainty values of x at the i^{th} learning iteration, l denotes the number of the preceding learning cycles considered for instability estimation, $l = 1$ represents the simplest case that considers two recent consecutive learning cycles for instability estimation, whereas the traditional uncertainty sampling is our instability sampling is when $l = 0$. The second term in the right hand of Eq. (4) indicates the change of uncertainty degree of x during recent consecutive learning cycles. A positive value of this term represents that x becomes more uncertain, and is likely to be an unstable example. In such a case, our instability sampling technique will give a *bonus* to it in selective sampling.

Similarly, given an unlabeled example x at the i^{th} learning cycle, by considering the factor of label predictions during recent consecutive learning cycles, its instability value $IS_{LS}(x)$ in label-sensitive instability sampling can be estimated by

$$IS_{LS}(x) = H^i(x) + \sum_{i-l < k \leq i} \delta(y^k, y^{k-1}) \times (H^k(x) - H^{k-1}(x)), \quad (5)$$

where $\delta(y^k, y^{k-1})$ is an indicator function whose value is 1 if two label predictions y^k and y^{k-1} are different, and 0 otherwise.

4. DENSITY-BASED SAMPLING

To improve uncertainty sampling, the concept of *density* has been adopted to determine whether an unlabeled example is highly representative [Tang et al. 2002; Shen et al. 2004; Zhu et al. 2008a; Nguyen and Smeulders 2004]. The motivation behind it is that unlabeled examples near the decision boundary and very close to other examples are more important than those that are isolated in the selective sampling. The density degree of an unlabeled example can be evaluated based on how many unlabeled examples

³In practice, the uncertainty value of an example can become smaller or bigger during the AL process. We also tried an absolute formulation the second right term in Eq. (4) for instability sampling. Experimental results show that the total absolute value of entropy changes (i.e., the second right term) will dominate the instability estimation on each example when increasing the k value, and results in negative performance.

are similar or close to it [Zhu et al. 2008a]. High density degree examples tend to be highly representative. That is, an example with high density degree is less likely to be an outlier [Roy and McCallum 2001]. In obtaining the density degree, the traditional cosine measure can be adopted to estimate the similarity between two examples, that is

$$\cos(w_i, w_j) = \frac{w_i \bullet w_j}{\|w_i\| \cdot \|w_j\|}, \quad (6)$$

where w_i and w_j are the feature vectors of examples i and j .

In practice, the unlabeled corpus is often very large (e.g., more than tens of thousands of unlabeled examples). It is unreasonable to exhaustively calculate similarities between any example and all the others in order to obtain the density degree. Tang et al. [2002] and Shen et al. [2004] applied a technique of evaluating the density of an example within a cluster. In their work, the unlabeled dataset is first grouped into a predefined number of clusters using K-means clustering. The density of an example can be defined as the average similarity between itself and the other examples within the same cluster [Shen et al. 2004]. Our first intuition in estimating density is to apply a clustering-based technique. However, there are some challenges. As in Tang et al. [2002], the average size of the resulting clusters is still very large. In this situation, the resulting density values of unlabeled examples are close to each other. Experimental results show that the size distribution of the resulting clusters is often skewed. It causes density estimation to be biased towards smaller clusters. An alternative approach is to use a similarity score threshold, for which a similarity score above a predefined threshold indicates similar examples. However, it is an open question how to predefine a proper similarity score threshold for different AL tasks.

To address these issues, Zhu et al. [2008a] proposed a *KNN-density* measure in which the density of an example is quantified by the average similarity between the example in question and its K most similar examples (i.e., K nearest neighbors). Given a set of K most similar examples $S(x) = \{s_1, s_2, \dots, s_K\}$ of the unlabeled example x , the average similarity $AS(x)$ between example x and its K most similar examples can be calculated by

$$AS(x) = \frac{\sum_{s_i \in S(x)} \cos(x, s_i)}{K}. \quad (7)$$

To select unlabeled examples with maximum uncertainty and highest density for human labeling, Zhu et al. [2008a] proposed a sampling technique named *Sampling by Uncertainty and Density* (SUD), in which a density-weighted entropy measure⁴ is adopted as follows.

$$u = \arg \max_{x \in U} (AS(x) \times H(x)) \quad (8)$$

Similarly, to combine the best of instability and density criteria, we prefer to select unlabeled examples with maximum instability and highest density for human annotation, which will be of high value to the learner. Motivated by the previous work [Zhu et al. 2008a], this article presents a technique of *Sampling by Instability and Density* (SID) in which a density-weighted instability measure (*density*instability*)⁵ is adopted

⁴Zhu et al. [2008a] reported that other ways like $\lambda^*AS(x) + (1 - \lambda)H(x)$ cannot outperform this simple multiplication way in active learning for text classification. Actually it is not easy to determine an appropriate value λ for a specific task.

⁵We also tried other combination ways such as $\lambda^*AS(x) + (1 - \lambda)IS(x)$. Interestingly, experimental results show that the simple multiplication way of Eq. (9) outperforms the linear combination, which is similar to that reported by Zhu et al. [2008a].

as

$$u = \arg \max_{x \in U} (AS(x) \times IS(x)), \quad (9)$$

where $IS(\cdot)$ is $IS_{LS}(\cdot)$ or $IS_{LI}(\cdot)$, depending on which instability sampling technique is used. The motivation of the density*instability measure is to use the density factor to adjust the instability $IS(x)$ of an unlabeled example x . A more unstable example with high density should be assigned a higher instability value. Our SID method favors the unlabeled example with high instability and high density at each learning cycle.

In the SID method, a general approach would re-estimate the density of each unlabeled example at each learning cycle. When the scale of the unlabeled pool is very large, such a re-estimation becomes cost prohibitive. Optimization has to be used in order to make our SID method more practical. One approximation method is to first calculate the similarity of each example pair at the beginning of AL.⁶ For each unlabeled example, all other examples in the unlabeled pool can be first ranked in decreasing order of similarity to the current example. The similarity of each example pair is estimated only once during the whole AL process. If an example is chosen at the i^{th} learning cycle, we remove it immediately from the ranked list of each of the rest of the unlabeled examples for the next learning cycle. It incurs little computational cost to calculate the density of an unlabeled example x by looking up top- K examples in its ranked list at each learning cycle. This implementation of density estimation is very efficient.

5. EVALUATION

5.1. Settings

We utilized a state-of-the-art Maximum Entropy (MaxEnt) model [Berger et al. 1996] to design the basic classifier for each classification task. The advantage of the MaxEnt model is its ability to freely incorporate features from diverse sources into a single, well-grounded statistical model. A publicly available MaxEnt toolkit⁷ was used in our comparison experiments. To evaluate the effectiveness of various instability sampling techniques, we followed previous studies on active learning for NLP applications [Lewis and Gale 1994; McCallum and Nigam 1998a; Tong and Koller 2001; Chen et al. 2006; Zhu and Hovy 2007; Chan and Ng 2007], and selected three types of NLP classification applications on seven publicly available real-world datasets. These NLP classification applications are listed as follows.

—*Text classification task.* Four publicly available datasets are used in this active learning comparison experiment: WebKB, Comp2a, Comp2b, and 20-NewsGroups datasets. The WebKB and 20-NewsGroups datasets have been widely used in text classification research. Following previous studies [McCallum and Nigam 1998b], we used the four most populous categories of the WebKB set: *student*, *faculty*, *course*, and *project*. The Comp2a dataset consists of *comp.os.ms-windows.misc* and *comp.sys.ibm.pc.hardware* subset of 20-NewsGroups. The Comp2b dataset consists of *comp.graphics* and *comp.windows.x* categories from 20-NewsGroups. Both datasets have been previously used in active learning for text classification [Roy and McCallum 2001; Schein and Ungar 2007]. We first processed all datasets by removing corpus words contained in a stop-word list. A MaxEnt model has been trained for the text classifier. No feature selection technique is used for the text classifier, because experimental results show that using feature selection seems to cause a negative effect on the performance of active learning for text classification [Zhu et al. 2008a].

⁶Thanks to the reviewer for suggesting that nearest neighbors of each example can be computed more efficiently using an inverted index for optimization.

⁷<http://homepages.inf.ed.ac.uk/s0450736/maxent-toolkit.html>.

Table I. Descriptions of These Datasets Used in the Active Learning Simulations, Including the Number of Classes and Class Distribution

Data Set	Classes	Class Distribution
Comp2a	2	983/1000
Comp2b	2	999/1000
WebKB	4	504/930/1641/1124
20-NewsGroups	20	983/999/1000/.../1000
MPQA	2	4958/6081
Interest	6	500/1252/178/66/361/11
Line	6	404/2218/373/374/349/429

- Opinion analysis task.* To analyze an opinioned text, the first step is to build a classifier that identifies opinion-bearing sentences in the text under a two-way classification framework [Kim 2006]. These opinion-bearing sentences express an opinion, namely, subjective sentences. The Multi-Perspective Question Answering (MPQA) opinion corpus [Wiebe et al. 2003] contains news articles manually annotated using an annotation scheme for subjectivity. According to the opinion annotation scheme, all sentences in the MPQA can be divided into two categories: *subjective* and *objective*. As shown in Table I, the MPQA corpus contains 4958 objective sentences and 6081 subjective sentences. In this work, opinion analysis is viewed as a binary classification task in which a MaxEnt model is trained for the classifier for active learning, and only four types of tokens, *noun*, *verb*, *adjective*, and *adverb*, are considered as features.
- Word sense disambiguation task.* WSD can be treated as a text classification problem. Two publicly available real-world datasets are used in this evaluation task: Interest and Line datasets. The Interest dataset was developed by Bruce and Wiebe [1994]. It consists of 2369 sentences of the noun “interest” with its correct sense manually labeled. The noun “interest” has six different senses in this dataset. The Interest dataset has been previously used for WSD study [Ng and Lee 1996]. In the Line dataset, each instance of “line” has been tagged with one of six WordNet senses. The Line dataset has been used in some previous studies on WSD [Leacock 1993]. To build the MaxEnt-based classifier for the WSD task, three knowledge sources are used to capture contextual information: *unordered single words in topical context*, *POS of neighboring words with position information*, and *local collocations*. These are the same knowledge sources used in some other word sense disambiguation studies [Lee and Ng 2002].

In our experimental evaluations, we utilized accuracy as the performance evaluation metric for each classification task. To globally compare two different AL methods, we adopted the deficiency metric [Baram et al. 2004] that has been widely used in previous studies on AL [Schein and Ungar 2007; Zhu et al. 2008a]. The deficiency metric between two active learning methods REF and AL is defined by

$$Def_n(AL, REF) = \frac{\sum_{t=1}^n (\varphi_n(REF) - \varphi_t(AL))}{\sum_{t=1}^n (\varphi_n(REF) - \varphi_t(REF))}, \quad (10)$$

where REF is the baseline method (i.e., traditional uncertainty sampling method in this work), and AL is one of our active learning methods such as instability sampling techniques. $\varphi_t(REF)$ and $\varphi_t(AL)$ denote the evaluation performance (i.e., accuracy value) at the t^{th} learning iteration of active learning methods REF and AL, respectively. n refers to the number of annotated examples at the stopping point; a smaller deficiency value (<1.0) indicates the proposed active learning method (AL) is performing better than the reference (REF) method.

In the following comparison experiments, we used 10% randomly chosen data for held-out evaluation and the other 90% as the pool of unlabeled data for each round

Table II. Average Deficiency (see Eq. (10)) Achieved by Various Methods, Compared to Traditional Uncertainty Sampling (Entropy)

Data Set	LSIS	LIIS	LSIS*Density	LIIS*Density
Comp2a	0.504 ⁺	0.506 ⁺	0.481⁺	0.548 ⁺
Comp2b	0.314 ⁺	0.299 ⁺	0.124 ⁺	0.048⁺
WebKB	0.792 ⁺	0.713⁺	0.845	0.880
MPQA	-0.628 ⁺	-0.862⁺	0.371 ⁺	-0.167 ⁺
Interest	0.895	1.031	0.366 ⁺	0.361⁺
Line	0.930	0.941	0.804⁺	0.866
20-NewsGroups	1.101	1.083	-0.245 ⁺	-0.408⁺

The stop point is 200 examples. For each dataset, the winner appears in bold-face. The symbol “+” indicates a significant difference based on the paired t-test with p-value > 0.05.

of the active learning. An active learning algorithm starts with an initial training set in which five labeled samples are randomly chosen from the pool, and selects five unlabeled examples for human labeling at each cycle. Following the previous study [Zhu et al. 2008a], the parameter K in Eq. (7) is set to be 20 for density estimation. Since it is difficult from a practical perspective to preemptively define an appropriate K value for density estimation, we also constructed some sensitivity analysis experiments to investigate the effects of parameters K on density estimation and the size parameter B on batch selection (i.e., the number of chosen examples per iteration). A tenfold cross-validation was performed. All results reported are the average of ten trials.

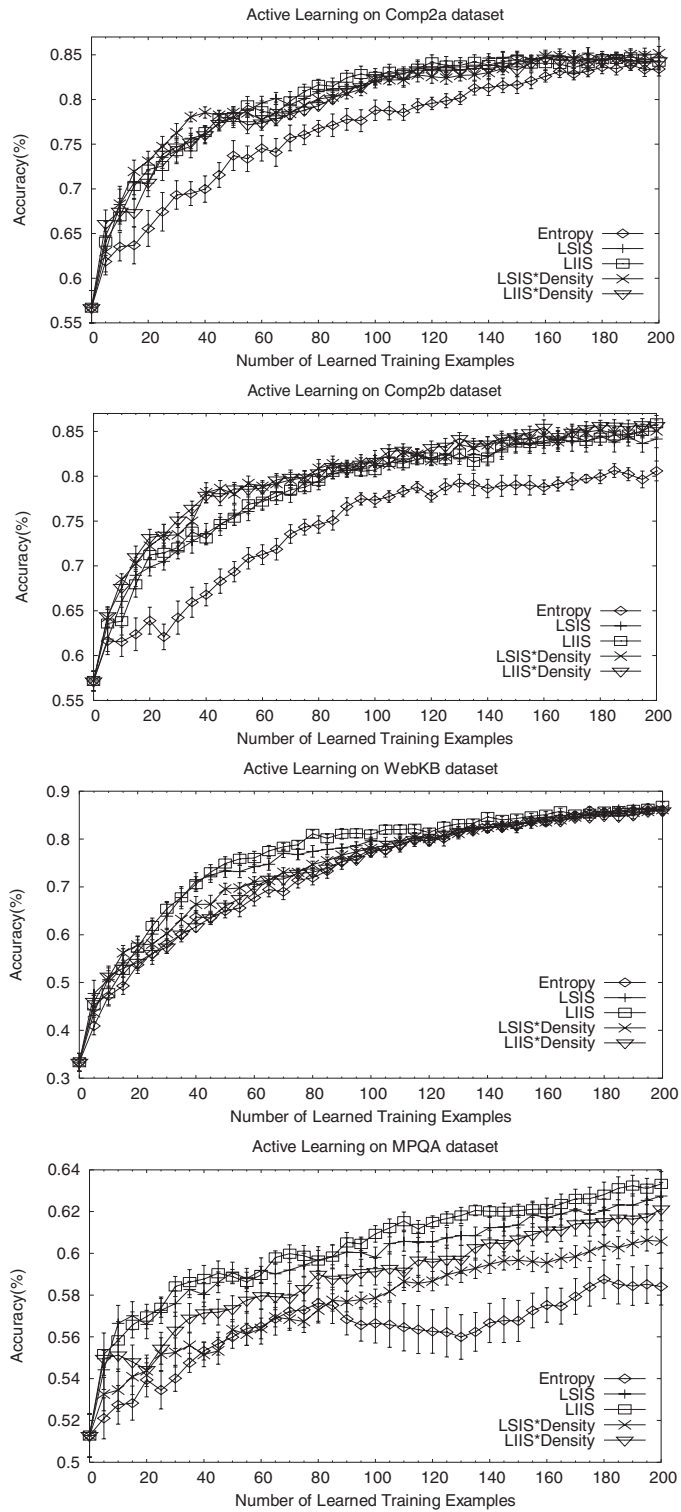
5.2. Results of Instability Estimation with $l = 1$

We first evaluate the effectiveness of various AL methods. In these experiments,⁸ for each instability sampling method, $l = 1$ denotes that instability estimation is implemented during the two recent consecutive learning cycles. In Figure 3, confidence bars with confidence at 95% level indicate the variability of each method, which is estimated using the paired t-test method.

Figure 3 depicts the effectiveness of various instability sampling (LSIS and LIIS), sampling by instability and density (LSIS*Density and LIIS*Density), and traditional uncertainty sampling (Entropy) methods on seven evaluation datasets. Table II shows the average deficiency value of each method in comparison to the baseline method (Entropy). Note that a smaller deficiency value (<1.0) indicates better performance. Compared to Entropy, LSIS and LIIS can achieve significant improvements on four datasets including the Comp2a, Comp2b, WebKB, and MPQA. Among these four sets, Table II shows that LIIS and LSIS obtain similar performances on the Comp2a set while LIIS outperforms LSIS on the other three sets. LSIS favors the most uncertain examples whose label predictions are different during recent consecutive learning cycles, for example, blank rectangle examples in Figure 2. However, blank triangle examples in Figure 2 could be the most informative cases that are not favored by LSIS, but are favored by LIIS. It is a possible reason that explains why LIIS can achieve better performance than LSIS on the Comp2b, WebKB, and MPQA datasets.

Table II shows that LSIS achieves slightly better performance than Entropy on two WSD evaluation sets while LIIS achieves slightly better performance than Entropy on the Line set. As reported by some studies on AL [Schein and Ungar 2007], the 20-NewsGroups is a very hard evaluation set for multi-category active learning, and random sampling can often defeat Entropy on this set during the early learning stages. In other words, many uncertain examples learned during the early learning stages are

⁸The X-axis label “number of learned training examples” shown in all figures indicates the number of newly labeled training examples.



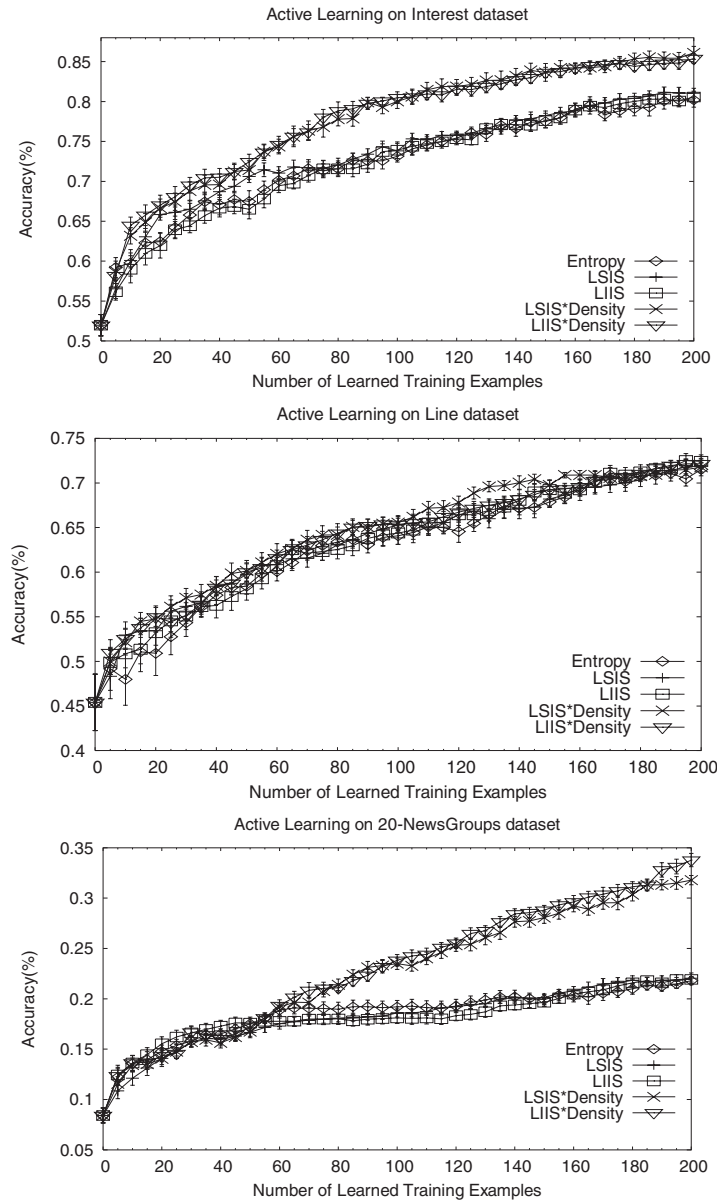


Fig. 3. Results of various AL methods on seven evaluation sets.

likely to be outliers or noisy data that cannot provide much help to the learner. Since instability estimation is also implemented based on uncertainty, LSIS and LIIS would yield no significant improvements for multi-category active learning on data with noise during the early learning stages, as compared to Entropy.

Table II shows that LSIS*Density and LIIS*Density can achieve better performance than Entropy on all evaluation sets in terms of average deficiency. Traditional uncertainty sampling would fail by selecting outliers, particularly on data with noise [Schein and Ungar 2007]. Figure 3 and Table II show that our techniques of sampling by

Table III. Average percentage (%) of Examples Regained to Achieve 99% of the Full Performance Using All Examples for Each Method

Data Set	Random Sampling	Entropy	LSIS	LIIS	LSIS* Density	LIIS* Density
Comp2a	69.29	35.83	32.11	28.45	34.64	26.47
Comp2b	58.22	30.92	29.80	30.36	23.95	24.79
WebKB	23.97	18.54	15.87	13.62	10.59	10.86
MPQA	75.84	27.68	21.99	25.31	25.11	26.78
Interest	66.19	21.83	20.95	21.83	20.64	20.23
Line	72.52	37.53	35.65	32.84	37.13	36.35
20-NewsGroups	59.44	30.26	30.74	31.47	28.75	28.98

The winner for each dataset appears in boldface.

instability and density (LSIS*Density and LIIS*Density) are better able to avoid selecting outliers or noisy examples for active learning than traditional uncertainty sampling because an uncertain example with high instability and density is less likely to be an outlier or a noisy example. Table II shows that LSIS*Density and LIIS*Density can achieve the best performances on four out of seven datasets, and LSIS and LIIS achieve the best performances on the other three evaluation datasets. It seems that incorporating the density factor into instability sampling methods cannot guarantee achieving performance improvement in some cases. However, it seems that LSIS*Density and LIIS*Density can outperform LSIS and LIIS on imbalanced data such as the Interest set, or on a dataset with a large class set such as the 20-NewsGroups set.

It is worth investigating some behaviors of various AL methods under different data domains and class distributions. Notice that sentiment analysis on MPQA can be viewed as a binary text classification task. Figure 3 shows that LSIS and LIIS can achieve significant improvement on four classification datasets except the 20-NewsGroups, as compared to traditional uncertainty sampling (Entropy). These four datasets have a small number of categories and a much balanced class distribution. Experiments also showed that pure instability sampling methods (LSIS and LIIS) would yield unsatisfactory performance on a dataset which has a large number of categories such as the 20-NewsGroups and a skewed class distribution such as Interest and Line. When incorporating the density factor, instability sampling methods (LSIS*Density and LIIS*Density) can perform better on all datasets in comparison to traditional uncertainty sampling (Entropy). Therefore, incorporating instability and density factors is a good choice for AL on imbalanced data or a dataset that contains a large number of categories.

To further investigate how much annotation cost can be reduced by various AL methods, Table III reports the average percentage of examples learned by each method to achieve 99% of the full performance using all examples for training. A smaller average percentage value indicates that more human annotation efforts can be reduced.

As seen from Table III, random sampling would select more than 50% of examples to achieve 99% full performance (using all examples for training) except for the WebKB set. Table III shows that traditional uncertainty sampling (Entropy) can achieve lower average percentage values than random sampling on all evaluation sets, and instability sampling (LSIS and LIIS) and sampling by instability and density (LSIS*Density and LIIS*Density) methods can reduce annotation cost more effectively than random sampling and Entropy. Among all the methods, LSIS and LIIS achieve the best performance on two sets while LSIS*Density and LIIS*Density work the best on five out of seven datasets. Besides, compared to random sampling, the largest reduction (-53.85%) in annotation cost is achieved by LSIS on the MPQA set. The LIIS*Density results in the largest reduction (-9.36%) in annotation cost on the Comp2a, compared to traditional uncertainty sampling (Entropy).

Table IV. Average Deficiency Values Achieved by LSIS and LSIS*Density with Different l Values on Four Evaluation Sets, as Compared to Traditional Uncertainty Sampling (Entropy)

AL Method	Comp2a	Comp2b	WebKB	MPQA
LSIS($l = 1$)	0.504	0.314	0.792	-0.628
LSIS($l = 2$)	0.442	0.272	0.742	-0.553
LSIS($l = 3$)	0.520	0.173	0.696	-0.747
LSIS($l = 4$)	0.520	0.171	0.706	-0.595
LSIS($l = 5$)	0.692	0.145	0.678	-0.592
LSIS($l = 6$)	0.705	0.143	0.661	-0.446
LSIS($l = 1$)*Density	0.481	0.124	0.845	0.371
LSIS($l = 2$)*Density	0.424	0.187	0.842	-0.098
LSIS($l = 3$)*Density	0.468	0.149	0.838	0.116
LSIS($l = 4$)*Density	0.472	0.083	0.898	0.132
LSIS($l = 5$)*Density	0.469	0.049	0.879	-0.016
LSIS($l = 6$)*Density	0.424	0.147	0.874	0.063

The stop point is 200 examples. For each dataset, the bold number indicates the best performance.

Table V. Average Deficiency Values Achieved by LIIS and LIIS*Density with Different l Values on Four Evaluation Sets, as Compared to Traditional Uncertainty Sampling (Entropy)

AL Method	Comp2a	Comp2b	WebKB	MPQA
LIIS($l = 1$)	0.506	0.299	0.713	-0.862
LIIS($l = 2$)	0.502	0.223	0.762	-1.017
LIIS($l = 3$)	0.427	0.009	0.693	-0.991
LIIS($l = 4$)	0.472	0.232	0.681	-0.649
LIIS($l = 5$)	0.567	0.199	0.714	-0.876
LIIS($l = 6$)	0.492	0.179	0.810	-0.895
LIIS($l = 1$)*Density	0.548	0.048	0.880	-0.167
LIIS($l = 2$)*Density	0.509	0.103	0.893	-0.306
LIIS($l = 3$)*Density	0.512	-0.007	0.920	-0.139
LIIS($l = 4$)*Density	0.477	0.047	0.863	-0.708
LIIS($l = 5$)*Density	0.469	-0.048	0.912	-0.531
LIIS($l = 6$)*Density	0.533	-0.017	0.869	-0.280

The stop point is 200 examples. For each dataset, the bold number indicates the best performance.

5.3. Sensitivity Analysis

In this section, we designed some experiments to investigate the effectiveness of various instability sampling methods by varying l values, as shown in Tables IV and V. These comparison experiments aim to investigate the effectiveness of instability sampling methods (LSIS and LIIS) and sampling by instability and density (LSIS*Density and LIIS*Density) with different l values, that is, more than one preceding learning cycle used for instability estimation.

When comparing to the simplest case of instability sampling methods ($l = 1$), Tables IV and V show that considering more preceding learning cycles for instability estimation can possibly improve the performance of instability sampling methods. For example, LSIS achieves the best performance on the Comp2a, Comp2b, WebKB, and MPQA sets when l values are set to 2, 6, 6, and 3. In terms of average deficiency, the best performance of 0.424 on the Comp2a, -0.048 on the Comp2b, 0.661 on the WebKB, and -1.017 on the MPQA are achieved by LSIS*Density($l = 2$ or 6), LIIS($l = 5$)*Density, LSIS($l = 6$), and LIIS($l = 2$), respectively.

As mentioned in Section 4, there is a parameter K used for density estimation. We designed some experiments of sampling by instability and density (LSIS*Density and LIIS*Density) with different K values for density estimation on four evaluation sets.

Table VI. Average Deficiency Values Achieved by Sampling by Instability and Density with Different K Values for Density Estimation, as Compared to Traditional Uncertainty Sampling (Entropy)

AL Method	Comp2a	Comp2b	WebKB	MPQA
LSIS*Density($K = 10$)	0.483	0.030	0.841	0.057
LSIS*Density($K = 20$)	0.481	0.124	0.845	0.371
LSIS*Density($K = 30$)	0.459	0.220	0.876	0.525
LSIS*Density($K = 40$)	0.485	0.184	0.811	0.294
LSIS*Density($K = 50$)	0.532	0.217	0.808	0.402
LSIS*Density($K = 100$)	0.588	0.149	0.689	0.355
LIIS*Density($K = 10$)	0.507	0.036	0.952	-0.309
LIIS*Density($K = 20$)	0.548	0.048	0.880	-0.167
LIIS*Density($K = 30$)	0.495	0.118	0.926	-0.014
LIIS*Density($K = 40$)	0.471	0.181	0.866	-0.197
LIIS*Density($K = 50$)	0.503	0.105	0.823	-0.098
LIIS*Density($K = 100$)	0.569	0.146	0.833	0.130

The parameter l is set to 1 for all instability sampling methods. The stop point is 200 examples. For each dataset, the bold number indicates the best performance.

Table VII. Average Deficiency Values Achieved by Various AL Methods (AL) with $B = 1$ for Batch Selection, Compared to the Corresponding Methods (REF) with $B = 5$ (the default setting) on the Comp2a Dataset

Entropy ($B = 5$ vs $B = 1$)	LSIS ($B = 5$ vs $B = 1$)	LIIS ($B = 5$ vs $B = 1$)
0.91	0.93	0.96

The parameter l is set to 1 for all instability sampling methods. The stop point is 200 examples.

Table VI shows the effectiveness of sampling by instability and density methods with different K values varying between 10, and 100. For LSIS*Density, the settings of $K = 30, 10, 100$, and 10 achieve the best performance on the Comp2a, Comp2b, WebKB, and MPQA sets, respectively. For LIIS*Density, the settings of $K = 40, 10, 50$, and 10 work the best on the Comp2a, Comp2b, WebKB, and MPQA sets, respectively. Table VI also shows that a larger K value would cause negative effects on the performance, for example, $K = 100$ for LSIS*Density on the Comp2a and Comp2b sets. It is apparent that the default setting of $K (=20)$ value used in our preceding experiments is not the best choice. And in practice, there is much room to improve the performance by choosing an appropriate K value for density estimation.

Table VII shows the results of instability sampling methods with different sizes of batch selection. In the default setting ($B = 5$), AL algorithms choose five examples for oracle labeling at each learning iteration, referred to as batch mode active learning. In such a case, it is possible that some of the chosen examples are similar to each other, namely the *redundancy* problem. Table VII shows that an AL method with $B = 1$ slightly outperforms that with $B = 5$. This can be explained by the fact that there is not a redundancy problem existing in an AL method with $B = 1$. In practice, a larger B value can decrease the number of times a learner is retrained during the AL process, and improve the efficiency of the AL method, but would cause little sacrifice to the AL performance.

5.4. Agreement Analysis

It is worth investigating how much agreement there is between the examples selected by various AL methods. In the following experiments, the agreement percentage (AP)⁹

⁹Since the probability of querying a particular example label is not random for each AL round, agreement percentage is more appropriate than the kappa measure.

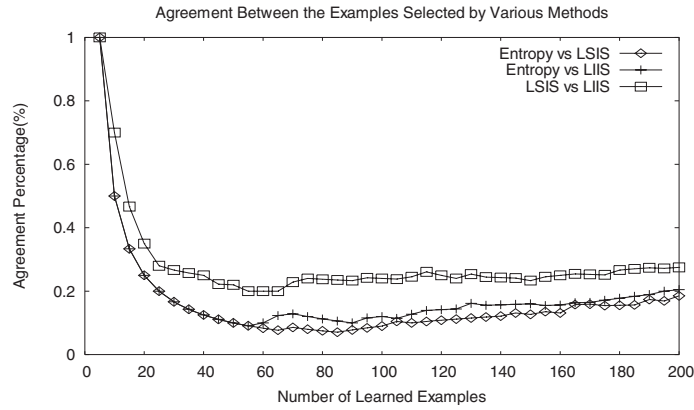


Fig. 4. Agreement analysis between the examples selected by uncertainty sampling (Entropy) and instability sampling (LSIS and LIIS) methods with $l = 1$ on the Comp2a set.

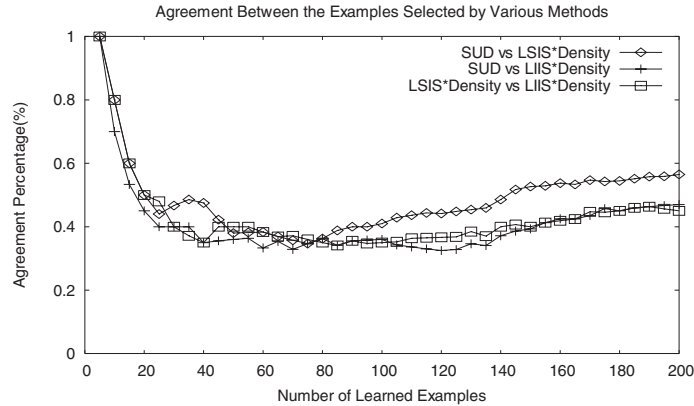


Fig. 5. Agreement analysis between the examples selected by Sampling by Uncertainty and Density (SUD) and Sampling by Instability and Density (LSIS*Density and LIIS*Density) with $l = 1$ and $K = 20$ on the Comp2a set.

between two methods M1 and M2 is estimated by

$$AP(M1, M2) = \frac{N_{same}(M1, M2)}{N}, \quad (11)$$

where N is the total number of used training examples until the current iteration. $N_{same}(M1, M2)$ is the total number of examples in common used by M1 and M2 methods until the current iteration.

Figures 4 through 6 show the agreement between various AL methods on the Comp2a dataset. Since LSIS and LIIS are instability sampling techniques, it is reasonable to see that both methods can obtain the highest agreement, in comparison to the other two combinations of uncertainty sampling, as shown in Figure 4. The agreements between the two combinations of uncertainty sampling are almost the same. After learning sixty examples, uncertainty sampling and LIIS only achieve slightly higher agreement percentage than that between uncertainty sampling and LSIS.

Figure 5 shows that various AL methods with density achieves high agreement at the early learning stages. As the number of learning iterations is increasing (after the 80th point), the combination of SUD and LSIS*Density achieves higher agreement

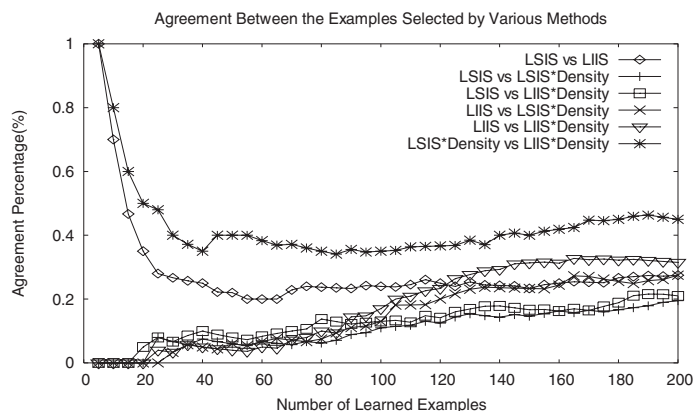


Fig. 6. Agreement analysis between the examples selected by instability sampling and sampling by instability and density methods with $l = 1$ and $K = 20$ on the Comp2a set.

than the other two combinations. Figure 6 shows agreement analysis on six combinations of four instability-based sampling methods. The combination of LSIS*Density and LIIS*Density achieves the best agreement, followed by that of LSIS and LIIS. It is interesting to see that four out of six combinations do not select the same examples until having learned the 15th example. Their agreement percentages increase as the number of learning iterations increases. The highest agreement percentage of 30% is achieved at the 200th point. The agreement of LIIS and LIIS*Density is higher than that of LSIS and LIIS after learning 120 examples. It seems that with an increasing number of learning iterations, LIIS (LSIS) and LIIS*Density (LSIS*Density) tend to make the same or similar decisions on selection of the most informative examples.

The incorporation of density factor is shown to achieve a higher agreement than other combinations in our AL methods. As AL methods with the density factor tend to prefer examples with high density score for selective sampling, the density factor can possibly make more contribution to selective sampling than the uncertainty factor. We will further study the effectiveness of uncertainty and density factors in selective sampling in the context of uncertainty-based AL in future work.

5.5. Extension to Active Learning with SVMs

This section provides some experimental results to demonstrate the generality of our methods by applying nonprobabilistic classifiers to active learning. We applied instability sampling techniques (LSIS and LIIS) for active learning with SVMs on three datasets, including the MPQA (for binary AL), WebKB (for multiclass AL) and Interest (imbalanced data) sets, as shown in Figures 7 through 9. We adopted the one-against-all to build a multiclass SVM classifier which is an ensemble of binary classifiers. In our baseline method, referred to as the margin-based method (Margin for short), the uncertainty of an unlabeled example can be defined as the absolute value of the difference between the two largest outputs of the decision functions [Schapire et al. 1997; Shen et al. 2004; Vlachos 2008]. Since standard SVMs only produce an uncalibrated value as a margin that is not a probability, it is problematic to simply use margin as the uncertainty value to estimate the $IS(x)$ value of each unlabeled example x during the active learning process. To resolve this issue, we utilized the technique proposed by Platt [1999], in which a sigmoid function is used to map the SVM outputs into probabilities for instability estimation. The softmax method [Milgram et al. 2006] is used to generalize the sigmoid function for the multiclass case.

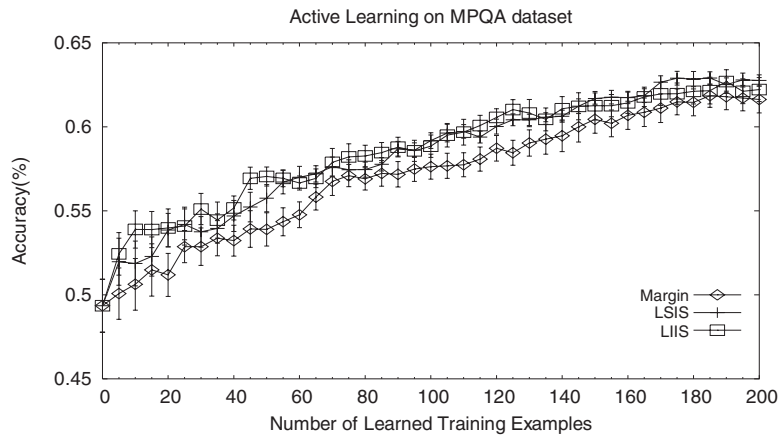


Fig. 7. Results of various AL methods with SVM classifier on the MPQA set.

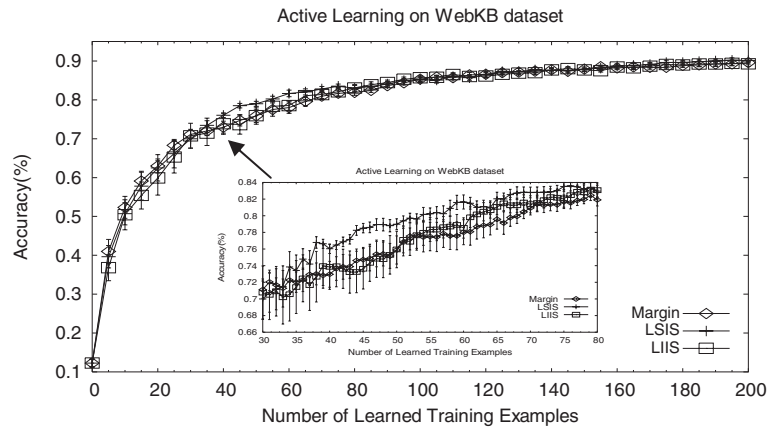


Fig. 8. Results of various AL methods with SVM classifier on the WebKB set.

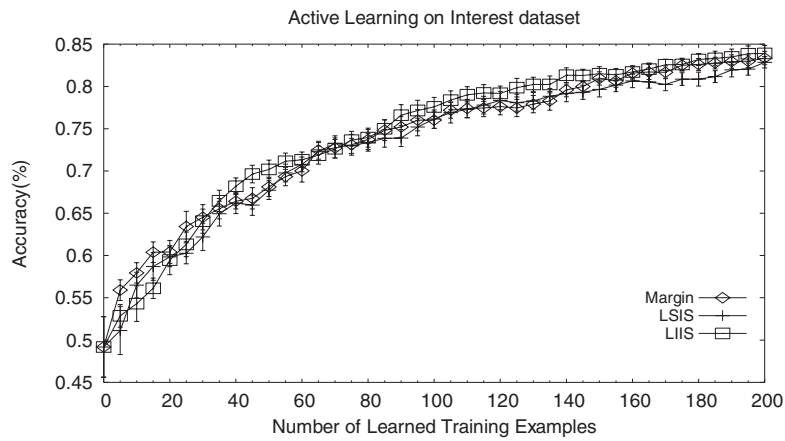


Fig. 9. Results of various AL methods with SVM classifier on the Interest set.

Figures 7 through 9 depict the effectiveness of the margin-based method and our instability sampling methods (LSIS and LIIS) for active learning with SVMs on three types of datasets. Figure 7 shows that LSIS and LIIS achieve significant improvements over the margin-based method on the MPQA set. LSIS only achieves significant improvements on multiclass active learning on the WebKB set during early learning stages, for example, from 30th to 80th iterations as shown in Figure 8. LIIS and the margin-based method obtain similar performance on WebKB. Figure 9 shows that LIIS achieves better performance than the margin-based method during the 90th to 150th iterations, and LSIS yields unsatisfactory performance on the imbalanced data Interest. As discussed before, in principle, our instability sampling methods can be applied to any classifier with calibrated posterior probabilities as outputs. The crucial issue of applying instability sampling techniques to active learning with SVMs is how to map standard SVM outputs into calibrated posterior probabilities for instability estimation. This work adopts a sigmoidal function to transform SVM outputs to posterior probabilities. However, there is no theoretical proof that a posterior probability has a sigmoidal shape and most likely it does not. We will further consider this probability transformation issue in the context of AL in our future work.

6. RELATED WORK

In recent years Active Learning (AL) has been widely studied in various Natural Language Processing (NLP) tasks, such as Word Sense Disambiguation (WSD) [Chen et al. 2006; Zhu and Hovy 2007; Chan and Ng 2007], Part-Of-Speech tagging (POS) [Ringger et al. 2007; Haertel et al. 2008b], Text Classification (TC) [Lewis and Gale 1994; McCallum and Nigam 1998a; Tong and Koller 2001], Named Entity Recognition (NER) [Shen et al. 2004; Jones 2005; Tomanek et al. 2007], chunking [Ngai and Yarowsky 2000], Information Extraction (IE) [Thompson et al. 1999; Culotta and McCallum 2005], statistical machine translation [Haffari and Sarkar 2009] and statistical parsing [Hwa 2000; Tang et al. 2002; Becker and Osborne 2005]. These AL techniques generally take a sampling method of estimating the uncertainty of each unlabeled example, namely *uncertainty-based sampling* techniques.

Uncertainty-based sampling techniques would face the outlier problem in which an unlabeled example with maximum uncertainty can be an outlier [Roy and McCallum 2001; Zhu et al. 2008a]. Previous studies have attempted to solve this problem. Cohn et al. [1996] and Roy and McCallum [2001] proposed a method that directly optimizes expected future error on future test examples. In practice, their methods are generally intractable due to high computational cost for selecting the most informative example from a large unlabeled pool. Roth and Small [2006] presented a mistake-driven active learning technique in which two metrics including average Hamming error per query and average global error per query are adopted for selective sampling. Settles and Craven [2008] considered a gradient-based method by querying the instance that would impart the greatest change to the current model. Yu et al. [2010] proposed a selective sampling technique using a global entropy reduction maximization criterion over all examples instead of only one example we wish to select.

Tang et al. [2002] adopted a sampling scheme of “most uncertain per cluster” for syntactic parsing, in which the learner selects the sentence with the highest uncertain score from each cluster and uses density to weigh the selected examples. In fact, the scheme of using the most uncertain example per cluster still cannot solve the outlier problem [Zhu et al. 2008a].

Shen et al. [2004] and Zhu et al. [2008a] proposed to select examples based on the density criterion. Their methods adopt an assumption that an example with high density degree is less likely to be an outlier. In Shen et al. [2004], the density of an unlabeled example is evaluated within a cluster, and multiple criteria are linearly

combined with different coefficients. However, as different values of the coefficients are associated with various applications, it is difficult to determine those coefficients automatically. Zhu et al. [2009] proposed a density-based reranking technique in which a KNN-density measure is used to rank top- N uncertain examples. Their experimental results demonstrate that density-based reranking methods sometimes could cause negative effects.

However, there is not an effective solution to utilize the useful information conveyed by misclassified examples to improve selective sampling, because in practice, it is an open issue to know exactly which unlabeled example is misclassified before asking for human labeling at each learning cycle. Our instability sampling techniques aim to address this challenge.

In our instability-based sampling techniques, instability estimation not only utilizes the uncertainty value and label prediction of an unlabeled example x at the current learning cycle, but also incorporates its uncertainty values and label predictions at the preceding learning cycles. Such preceding knowledge (i.e., uncertainty values at the preceding learning cycles) was utilized to learn a stopping criterion for active learning [Zhu et al. 2008b, 2010; Bloodgood and Vijay-Shanker 2009]. To our knowledge, incorporating the preceding knowledge for selecting the most informative examples is seldom mentioned in previous studies on uncertainty-based active learning.

7. CONCLUSION AND DISCUSSION

This article proposes a new selective sampling technique with instability estimation to improve uncertainty-based active learning. Experiments show that instability-based sampling techniques achieve promising results on various evaluation datasets. It is noteworthy that our instability sampling methods are easy to implement and applicable to other selective sampling schemes. For example, to apply instability-based sampling techniques to a committee-based sampling paradigm, we can adopt one of uncertainty measurements such as *vote entropy* [Seung et al. 1992] to measure the uncertainty of each unlabeled example. Since committee-based sampling does not directly assign an appropriate label prediction to each unlabeled example, we can utilize an additional supervised classifier to produce the class prediction of each unlabeled example by inducing from the current labeled data for the label-sensitive instability sampling technique.

Traditional active learning techniques assume that the annotation of each example costs the same. However, in some real-world NLP applications such syntactic parsing, the annotation costs of two different examples (e.g., sentences) are not equal [Haertel et al. 2008a, 2008b]. In this case, it is worthwhile incorporating some effective cost-sensitive machine learning techniques in our instability sampling methods to improve AL performance, and this is one of the practical and challenging directions [Baldrige and Palmer 2009] of our future work. We will further study other effective techniques to determine misclassified unlabeled examples for selective sampling and apply our methods in cost-sensitive active learning.

ACKNOWLEDGMENTS

Thanks to Muhua Zhu for implementing some comparison experiments of active learning with SVMs.

REFERENCES

- YORAM, B., EL-YANIV, R., AND LUZ, K. 2004. Online choice of active learning algorithms. *J. Mach. Learn. Res.*, 5, 255–291.
- BALDRIDGE, J. AND PALMER, A. 2009. How well does active learning actually work? Time-Based evaluation of cost-reduction strategies for language documentation. In *Proceedings of the ACL09 Conference*.

- BECKER, M. AND OSBORNE, M. 2005. A two-stage method for active learning of statistical grammars. In *Proceedings of the IJCAI'05 Conference*. 991–996
- BERGER, A. L., DELLA PIETRA, V. J., AND DELLA PIETRA, S. A. 1996. A maximum entropy approach to natural language processing. *Comput. Linguistics*. 22, 1, 39–71.
- BLOODGOOD, M. AND VIJAY-SHANKER, K. 2009. A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL)*, 39–47.
- BRUCE, R. AND WIEBE, J. 1994. Word-Sense disambiguation using decomposable models. In *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*. 139–146.
- CHAN, Y. S. AND NG, H. T. 2007. Domain adaptation with active learning for word sense disambiguation. In *Proceedings of the 45th Annual Meeting on Association for Computational Linguistics*. 49–56.
- CHEN, J., SCHEIN, A., UNGAR, L., AND PALMER, M. 2006. An empirical study of the behavior of active learning for word sense disambiguation. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. 120–127.
- COHN, D., ATLAS, L., AND LADNER, R. 1994. Improving generalization with active learning. *Mach. Learn.* 15, 2, 201–221.
- COHN, D. A., GHARAMANI, Z., AND JORDAN, M. I. 1996. Active learning with statistical models. *J. Artif. Intell. Res.* 4, 129–145.
- CULOTTA, A. AND MCCALLUM, A. 2005. Reducing labeling effort for structured prediction tasks. In *Proceedings of AAAI'05 Conference*. 746–751
- HAERTEL, R., RINGGER, E., SEPPI, K., CARROLL, J., AND MCCLANAHAN, P. 2008a. Assessing the costs of sampling methods in active learning for annotation. In *Proceedings of the ACL08 Conference*. 65–68
- HAERTEL, R. A., SEPPI, K. D., RINGGER, E. K., AND CARROLL, J. L. 2008b. Return on investment for active learning. In *Proceedings of NIPS'08 Workshop on Cost-Sensitive Learning*.
- HAFFARI, G. AND SARKAR, A. 2009. Active learning for multilingual statistical machine translation. In *Proceedings of the ACL'09 Conference*. 181–189.
- HWA, R. 2000. Sample selection for statistical grammar induction. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*. 45–52.
- JONES, R. 2005. Learning to extract entities from labeled and unlabeled text. PhD thesis, Carnegie Mellon University.
- KIM, S. M. 2006. Identification, classification, and analysis of opinions on the web. PhD thesis, University of Southern California.
- LEACOCK, C., TOWELL, G., AND VOORHEES, E. 1993. Corpus-Based statistical sense resolution. In *Proceedings of the ARPA Workshop on Human Language Technology*. 260–265.
- LEE, Y. AND NG, H. 2002. An empirical evaluation of knowledge sources and learning algorithm for word sense disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 41–48.
- LEWIS, D. D. AND GALE, W. A. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3–12.
- MCCALLUM, A. AND NIGAM, K. 1998a. Employing EM in pool-based active learning for text classification. In *Proceedings of 15th International Conference on Machine Learning*. 350–358.
- MCCALLUM, A. AND NIGAM, K. 1998b. A comparison of event models for naïve bayes text classification. In *Proceedings of the AAAI'98 Workshop on Learning for Text Categorization*.
- MILGRAM, J., CHERIET, M., AND SABOURIN, R. 2006. One against one or one against all: Which one is better for handwriting recognition with SVMs? In *Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition*.
- NG, H. AND LEE, H. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th Annual Meeting of the Association of Computational Linguistics*. 40–47.
- NGAI, G. AND YAROWSKY, D. 2000. Rule writing or annotation: Cost-Efficient resource usage for based noun phrase chunking. In *Proceedings of the 38th Annual Meeting of the Association of Computational Linguistics*.
- NGUYEN, H. AND SMEULDERS, A. 2004. Active learning with pre-clustering. In *Proceedings of the International Conference on Machine Learning (ICML'04)*. 623–630.
- PLATT, J. C. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*. MIT Press.

- RINGGER, E., McCLANAHAN, P., HAERTEL, R., BUSBY, G., CARMEN, M., CARROLL, J., SEPPI, K., AND LONSDALE, D. 2007. Active learning for part-of-speech tagging: Accelerating corpus annotation. In *Proceedings of the ACL'07 Linguistic Annotation Workshop (LAW)*.
- ROTH, D. AND SMALL, K. 2006. Margin-Based active learning for structured output spaces. In *Proceedings of the ECML'06 Conference*.
- NICHOLAS, R. AND MCCALLUM, A. 2001. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the 18th International Conference on Machine Learning*. 441–448.
- SCHAPIRE, R. E., FREUND, Y., BARTLETT, P., AND LEE, W. S. 1997. Boosting the margin: A new explanation for the effectiveness of voting methods. In *Proceedings of the 14th International Conference Machine Learning*. 322–330.
- SCHEIN, A. I. AND UNGAR, L. H. 2007. Active learning for logistic regression: An evaluation. *Mach. Learn.* 68, 3, 235–265.
- SETTLES, B. 2009. Active learning literature survey. Computer Science Tech. rep. 1648, University of Wisconsin-Madison.
- SETTLES, B. AND CRAVEN, M. 2008. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1070–1079.
- SEUNG, H. S., OPPER, M., AND SOMPOLINSKY, H. 1992. Query by committee. In *Proceedings of the 5th Annual ACM Conference on Computational Learning Theory*. 287–294.
- SHEN, D., ZHANG, J., SU, J., ZHOU, G., AND TAN, C.-L. 2004. Multi-Criteria-Based active learning for named entity recognition. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*. 589–596.
- TANG, M., LUO, X., AND ROUKOS, S. 2002. Active learning for statistical natural language parsing. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. 120–127.
- THOMPSON, C. A., CALIFF, M. E., AND MOONEY, R. J. 1999. Active learning for natural language parsing and information extraction. In *Proceedings of the 16th International Conference on Machine Learning*. 406–414.
- TOMANEK, K., WERMTER, J., AND HAHN, U. 2007. An approach to text corpus construction which cuts annotation costs and maintains reusability of annotated data. In *Proceedings of the Joint Meeting of the Conference on Empirical Methods on Natural Language Processing and the Conference on Natural Language Learning*. 486–495.
- TONG S. AND KOLLER, D. 2001. Support vector machine active learning with applications to text classification. *J. Mach. Learn. Res.* 45–66.
- VLACHOS, A. 2008. A stopping criterion for active learning. *Comput. Speech Lang.* 22, 3, 295–312.
- WIEBE, J., BRECK, E., AND BUCKLEY, C. 2003. Recognizing and organizing opinions expressed in the world press. In *Proceedings of the AAAI Spring Symposium on New Directions in Question Answering*.
- YU, D., VARADARAJAN, B., DENG, L., AND ACERO, A. 2010. Active learning and semi-supervised learning for speech recognition: A unified framework using the global entropy reduction maximization criterion. *Comput. Speech Lang.*, 24, 3, 433–444.
- ZHU, J. AND HOVY, E. 2007. Active learning for word sense disambiguation with methods for addressing the class imbalance problem. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. 783–790.
- ZHU, J., WANG, H., YAO, T., AND TSOU, B. 2008a. Active learning with sampling by uncertainty and density for word sense disambiguation and text classification. In *Proceedings of the 22nd International Conference on Computational Linguistics*. 1137–1144.
- ZHU, J., WANG, H., AND HOVY, E. 2008b. Multi-Criteria-Based strategy to stop active learning for data annotation. In *Proceedings of the 22nd International Conference on Computational Linguistics*. 1129–1136.
- ZHU, J., WANG, H., HOVY, E., AND MA, M. 2010. Confidence-Based stopping criteria for active learning for data annotation. *ACM Trans. Speech Lang. Process.* 6, 3, 1–24.
- ZHU, J., WANG, H., AND TSOU, B. K. 2009. A density-based re-ranking technique for active learning for data annotation. In *Proceedings of the 22nd International Conference on the Computer Processing of Oriental Languages (ICCPOL)*. 1–10.

Received March 2011; revised June 2011, September 2011; accepted November 2011