APPLYING ACTIVE LEARNING TO BIOMEDICAL TEXT PROCESSING

By

YUKUN CHEN

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

In partial fulfillment of the requirements

For the degree of

MASTER OF SCIENCE

In

Biomedical Informatics

August, 2013

Nashville, Tennessee

Approved:

Professor Hua Xu

Professor Joshua C. Denny

Professor Thomas Lasko

Professor Qiaozhu Mei

Table of Contents

ACKNOWLEDGEMENTS

I would like to thank my thesis committee for their guidance throughout this work. I would especially like to thank my advisor Dr. Hua Xu who has guided my work over the past two years, helped me to develop my research interests and skills, and taught me to become a better person in life. I would also like to thank my other committee members Dr. Josh Denny, Dr. Tom Lasko, and Dr. Qiaozhu Mei who advised me to develop better methods, broadened my view in biomedicine, and advanced my education in biomedical informatics. I am also grateful for many others in Hua's group who gave invaluable assistance: Min Jiang, Dr. Yonghui Wu, Dr. Buzhou Tang, and Dr. Mei Liu.

I would also like to express my gratitude to the Department of Biomedical Informatics. I appreciate the support of the students, faculty, and staff, without which I could not have completed the projects with two published journal papers in two years. I am grateful to the support of projects partially by NIH grants NLM R01LM010681 and NCI R01CA141307. I would also like to thank the organizers from 2010 i2b2/VA NLP challenge sharing the dataset for the concept assertion classification task.

Finally, I am very grateful for the support of my friends and family. I know I would not be where I am today without my father Dr. Jinde Chen and mother Misha Cao, and their unconditional love, support, and encouragement.

ABSTRACT

Objective: Supervised machine learning methods have shown good performance in text classification tasks in the biomedical domain, but they often require large annotated corpora, which are costly to develop. Our goal is to assess whether active learning strategies can be integrated with supervised machine learning methods, thus reducing the annotation cost while keeping or improving the quality of classification models for biomedical text.

Methods: We have applied active learning to two biomedical natural language processing (NLP) tasks: 1) the assertion classification task in the 2010 i2b2/VA Clinical NLP Challenge, which was to determine the assertion status of clinical concepts; and 2) a supervised word sense disambiguation (WSD) task that was to disambiguate 197 ambiguous words and abbreviations in MEDLINE abstracts. We developed Support Vector Machines (SVMs) based classifiers for both tasks. We then implemented several existing and newly developed active learning algorithms to integrate with SVM classifiers and evaluated their performance on both tasks.

Results: In assertion classification task, our results showed that to achieve the same classification performance, active learning strategies required much fewer samples than the random sampling method. For example, to achieve an AUC of 0.79, the random sampling method used 32 samples, while our best active learning algorithm required only 12 samples, a reduction of 62.5% in manual annotation effort. In the WSD task, our results also demonstrated that active learners significantly outperformed the passive learner, showing better performance for 177 out of 197 (89.8%) ambiguous terms.

Further analysis showed that to achieve an average accuracy of 90%, the passive learner needed 38 samples, while the active learners needed only 24 annotated samples, a 37% reduction of annotation effort. Moreover, we also analyzed cases where active learning algorithms did not achieve superior performance and summarized three causes: (1) poor model in early learning stage; (2) easy WSD cases; and (3) difficult WSD cases, which provide useful insight for future improvements.

Conclusion: Both studies demonstrated that integrating active learning strategies with supervised learning methods could effectively reduce annotation cost and improve the classification models in biomedical text processing.

CHAPTER I

INTRODUCTION

1.1 Active learning

Active learning is a technique under the subject of machine learning or, more generally, artificial intelligence. [1] The main hypothesis of active learning is that a learning machine could quickly improve its performance while using less training samples if it could actively select samples for learning. The active learner, which uses smart querying algorithms for sample selection, is highly suitable for supervised machine learning tasks where unlabeled data is plentifully available and easy to obtain, but labeling a sample is difficult, expensive, or time-consuming.

The active learning process evolves the learning model when new samples are actively chosen from the large unlabeled pool and annotated in each iteration. The overall goal is to optimize the learning process by maximizing the quality of supervised learning model and minimizing human annotation effort. Active learning is often compared with passive learning that randomly selects samples for annotation when building classification models. Many studies have demonstrated that active learning could outperform passive learning in supervised machine learning tasks.

Researchers have applied active learning to many areas such as image classification and retrieval, [2] gene expression analysis, [3] and drug discovery. [4] For these tasks, labeled samples are expensive to obtain or otherwise limited; however, unlabeled samples are largely available and inexpensive to access. In an image recognition task,

researchers showed that near-optimal performance could be reached using 25% less annotated images, when compared with traditional passive learning. [2] In another study, Liu demonstrated that a support vector machine classifier could achieve desired performance in the cancer classification task based on expression data from DNA microarray hybridization experiments, with a reduction of 82% in annotation cost. [3] In machine learning based drug discovery experiments, active learning was also successfully used to reduce biochemistry lab costs while improve the yield. [4] With the growth of available textual data such as web pages, active learning has also been applied to statistical natural language processing (NLP) tasks such as text classification [5,6] and information extraction, [7] and have shown promising results.

1.2 Overview of active learning algorithms

The pool-based active learning approach to classification [5] is practical for many real-world learning problems. It is often used in scenarios where a learner can access a large pool of unlabeled data with low cost and can then request true labels for selected samples. An active learning system mainly consists of a classification model and an active sample selection (also called querying) algorithm. The classification model can be built by using classical supervised machine learning algorithms. The querying step is to select the instances that are most promising in improving the predictive performance of the model. Many querying algorithms exist, and can be categorized into six types according to an active learning literature survey: [1] uncertainty sampling, [8] query-by-committee (QBC), [9] expected gradient length (EGL), [10] fisher information, [11] estimated error reduction, [12] and information density. [7]

Uncertainty sampling is the simplest and most commonly used query algorithm, which tends to query samples that are least certain about their labels. The uncertainty could be measured by different metrics, such as the confidence about the most possible label, the margin between two most possible labels, [13] and entropy. [14] For binary classification problem, uncertainty sampling algorithms based on different measurements are equivalent because they would query the samples with a class posterior probability closest to 0.5.

The QBC algorithm tends to select samples that generate the most disagreement from a committee of models. The models in the committee are all trained on the same labeled set but represent different hypothesis. The level of disagreement could be computed based on different voting strategies, such as vote entropy [15] and Kullback-Leibler (KL) divergence. [16] The QBC algorithm is, however, sensitive to the type of classification models selected.

The EGL algorithm tends to select the samples that would have produced the greatest change to the current model if we knew their labels. It is not a very practical solution because the computational cost is huge if we find the gradient change by testing all possible labels for each unlabeled sample. However, this approach has been shown to work well in empirical studies. [10] A similar algorithm to EGL is called expected error reduction, which tends to query samples that reduce the generalization error of a model. But it is also the most computationally expensive querying algorithm.

The fisher information algorithm selects the samples that could indirectly reduce the generalization error by minimizing output variance. It is equivalent to selecting samples that could maximize its fisher information. This is also an algorithm with high

computational complexity because to estimate output variance, it needs to compute a K by K matrix for each new sample, where K is the number of parameters in the model.

The information density algorithm does not rely on classification models but only the distribution of the data. It tends to query the most representative samples based on similarity among samples. The information density method could sometimes be combined with uncertainty sampling algorithm in order to select the most informative samples that are not only uncertain, but also the representatives of a dataset (e.g. centers of dense regions of data).

For a given dataset and a querying algorithm, a typical active learning protocol includes following steps:

(1) Initialize the labeled training set $L = L_0$, the pool of unlabeled set $U = U_0$, and a test set T.

(2) Train the classification model based on L and predict the probability of class label for each instance in U and T.

(3) Rank the instances in U based on the querying algorithm and assign labels (from human experts) for the top $b(i)$ samples in U, where $b(i)$, the **batch size** of active learning, is the number of querying samples at iteration i.

(4) Add the $b(i)$ instance(s) with label(s) to L and remove from U.

(5) Compute the classification performance in AUC score or accuracy (ACC) on the test set T and store in AUC(i) or ACC(i).

(6) Iterate steps (2) to (5) until the stop criterion (e.g. unlabeled samples in the pool are used up) is met.

(7) Evaluate this learning process by using the global learning score based on the learning curve that plots ACC(i) as a function of the batch size b(i).

1.3 Active learning for NLP tasks in the general English domain

Using the active learning protocol described in previous section, researchers have successfully applied it to various NLP tasks in the general English domain, such as named entity recognition, [7] part-of-speech tagging, [17] parsing, [18] word sense disambiguation, [19] automatic translation, [20] and sequence segmentation. [21]

Settles et al [7] proposed the density-weighted active learning algorithm, which combined diversity with uncertainty information, and outperformed uncertainty sampling, QBC, fisher information, and EGL algorithms, in eight named entity recognition tasks, on average. Ringger et al [17] applied Query-by-Uncertainty and Query-by-Committee active learning algorithms to accelerating the construction of a part-of-speech annotated corpus. Becker and Osborne [18] reported a two-stage model for learning grammars actively and showed that their method performed better than original form of uncertainty sampling and similar to a standard Query-by-Committee method. Chen et al [19] successfully used active learning to reduce the annotation effort while maintaining good performance for a word sense disambiguation task of five English verbs with coarse-grained senses. Kuo et al [20] effectively built an adaptive learning framework for automatic construction of transliteration lexicons and it minimized human supervision for data labeling. Sassano [21] explored how active learning with support vector machine could be applied to Japanese word segmentation and showed that their technique outperformed the method in previous

research and could significantly reduce required labeled examples to achieve a given level of accuracy.

1.4 Active learning for NLP tasks in the biomedical domain

1.4.1 Clinical NLP

In the past decade, increasing adoption of Electronic Medical Records (EMRs) in the healthcare industry has made practice-based clinical data available electronically. These detailed longitudinal clinical data are not only useful for clinical care but have also been increasingly used for clinical, genomic, and translational studies. [22-25] Because EMRs contain large amounts of textual data, studies of clinical NLP technologies have received great attention. [26-28] A number of clinical NLP systems have been developed, such as MedLEE (Medical Language Extraction and Encoding System), [29-31] cTAKES, [32] MedEx, [33] MetaMap, [34] and KnowledgeMap. [35]

More recently, statistical NLP methods have been applied to clinical text and they often involve building classification models based on annotated corpora. For example, in the 2010 i2b2 clinical NLP challenge, researchers have developed various supervised machine learning methods to recognize clinical entities in discharge summaries. [36] Thus annotation for clinical textual data, such as discharge summary and progress report, is essential for the development and the evaluation of machine learning-based clinical NLP approaches. However, annotating clinical text often requires domain expert manual review, which can be very expensive and time-consuming. Therefore, we believe that active learning framework would be very valuable for clinical NLP research.

1.4.2 Biomedical literature text processing

The phenomenal growth of biomedical literature has made it difficult for biomedical scientists in assimilating the high rate of new publications. [37] For example, MEDLINE, a medical citation database, currently contains over 20-million citations, with a growth rate of 4% over the past 20 years. [38] Text processing techniques that can automatically find relevant articles (information retrieval) and extract specific information (information extraction) are highly desirable. Therefore, many informatics researchers have focused on developing NLP methods and tools for biomedical literature (also called bioNLP). In many bioNLP tasks, supervised machine learning methods have shown great performance and more and more annotated biomedical corpora are being developed. [39] To improve the efficiency in building such annotated biomedical corpora, researchers have started investigating active learning methods, hoping to reduce annotation cost and improve machine learning model quality. [40-42]

1.4.3 Active learning in clinical and biomedical NLP tasks

As mentioned previously, supervised machine learning for clinical and biomedical NLP tasks requires a large number of annotated samples, which are even more expensive to build than the ones in general English NLP tasks because of the involvement of clinical domain experts. Active learning is well motivated in this domain as an alternative solution. Although researchers have shown that active learning is beneficial in many domains, few studies have investigated active learning techniques in clinical and biomedical NLP tasks.

Figueroa et al. [43] applied active learning to two clinical text classification tasks including smoking status and depression status extraction, and one non-clinical classification task using SVM. They implemented distance-based (DIST), diversity-based (DIV), and the combination of both active learning algorithms (CMB), and compared the performance with passive learning. Their results showed that DIST and CMB algorithms significantly performed better than passive learning. They also suggested that DIV performed better on data with higher diversity and DIST on data with lower uncertainty.

Kim et al. [44] presented an active learning strategy in the biological named entity recognition task based on the data from MEDLINE abstracts and GENIA corpus. [45] Their method considered both entropy-based uncertainty from classifiers and the diversity of a corpus. To achieve 67.17% in F-score, the proposed strategy used 11000 sentences, which reduced 35.43% of the training examples comparing with random sampling (passive learning).

Wallace et al. [41] studied an application of active learning to the problem of biomedical citation screening for systematic reviews at the Tufts Evidence-based Practice Center. They proposed a novel active learning strategy that exploited a priori domain knowledge provided by the expert (specially, labeled features) and extended this model via a Linear Programming algorithm for situations where the expert can provide ranked labeled features. Uncertainty sampling with SVM performed better than random sampling when using accuracy as model evaluation metric; however, recall, which is important for citation screening, was not improved. This was probably due to the imbalanced class and the hasty generalization problem. But their results demonstrated that using the prior knowledge could positively guide active learning.

Miller et al. [42] explored various active learning methods for clinical coreference annotation workflows. Their paper indicated that traditional active learning approach might not be feasible for this task because coreference annotations required context information between entity mentions referring to the same entity. They finally proposed a hybrid sample selection approach that was primarily based on instance selection algorithms.

This thesis presents two of our recent studies of applying active learning to clinical text and biomedical literature in chapter 2 and 3, respectively. The first study investigated the application of active learning to the assertion classification of concepts in clinical text. [46] The second study explored the use of active learning in supervised word sense disambiguation (WSD) in biomedical literature. [47] Both tasks are required to construct supervised machine learning models to accurately identify either binary or multiple classes. It was not known whether active learning could be helpful for clinical assertion classification task or biomedical WSD task before we conducted these two studies. Both studies in this thesis mainly focused on uncertainty sampling, the most widely used querying method in active learning. For the task of active learning in assertion classification, we also assessed our newly developed querying algorithms such as "model change" and "uncertainty sampling with bias". [48] For biomedical WSD task, we applied three existing uncertainty sampling algorithms which could deal with multiple-class classification problems. Both studies demonstrated that integrating active learning strategies with supervised learning methods could effectively reduce annotation cost and improve the classification models in biomedical text processing.

CHAPTER II

STUDY I: APPLYING ACTIVE LEARNING TO ASSERTION CLASSIFICATION OF CONCEPTS IN CLINICAL TEXT

2.1 Introduction

In this chapter, we describe an application of active learning to a clinical text classification task: to determine assertions of clinical concepts, using an annotated corpus from the 2010 i2b2 Clinical NLP Challenge. We implemented and evaluated several active learning algorithms, including some that are newly developed, and our results showed that some active learning strategies outperformed random sampling methods significantly. This chapter is organized as the following: Section 2 presents datasets and methods that we used in this study, such as cross validation experiments, active learning strategies including classification models and querying algorithms, and evaluation; Section 3 displays the experiment results; Section 4 discusses the significance of our results.

2.2 Methods

2.2.1 Datasets

We used the manually annotated training set for concept assertion classification in the 2010 i2b2/VA NLP challenge, [36] which was organized by i2b2 (the Center of Informatics for Integrating Biology and the Bedside) at Partners Health Care System and Veterans Affairs (VA), Salt Lake City Health Care System. The assertion classification

task is to assign one of six labels ("absent", "associated with someone else", "conditional", "hypothetical", "possible", and "present") to medical problems identified from clinical text (discharge summaries and some progress notes collected from three institutions). We participated in the challenge and developed an SVM-based system for the assertion classification task, and we ranked fourth among over 20 participating teams (no statistically significant difference from the top three systems). [49]

For this study, we used the same set of features as described in our previous work and we wanted to assess whether active learning algorithms could reduce sample size while retaining good performance. The feature set includes: (1) window of context, the size of which is optimized; (2) direction with distance in the window of context (e.g., third word on the left); (3) bi-grams identified within the context window; (4) part of speech tags of context words; (5) normalized concepts and semantic types identified by an NLP system (MedLEE), [30] such as certainty, UMLS CUIs, and semantic types; (6) source and section of its clinical note.

The training set from the challenge contained 349 notes, with 11,967 medical problems annotated with one of the six assertion statuses. Given the availability of large annotated data, active learning may not be needed for this specific assertion classification task. However, we utilized this available large data set to evaluate the performance of different active learning algorithms, which should be useful for many other tasks where large annotated data are not available. Moreover, active learning on multi-class classification tasks is more complicated than that on binary classification tasks. Therefore, as an initial study, we focused on the investigation of active learning algorithms for binary classification problems. We converted the multi-class assertion

classification task into a binary classification problem, by considering "present" to be the positive class and all others as the negative class. We refer to this dataset as ASSERTION in this study and investigated active learning algorithms for the binary classification of assertion ("present" vs. "non-present").

In addition, we used NOVA, a dataset of English text from the 2010 active learning challenge, [50] as the benchmark for this study. NOVA comes from the 20-Newsgroup dataset[51], which is a popular benchmark dataset for experiments in text applications of machine learning techniques, such as text classification and text clustering. Each text to be classified comes from an email that was posted to one or several newsgroups. The NOVA data are selected from both politics and religion, topics considered as positive and negative class, respectively. The feature set of data is in binary representation using a bag-of-words with a vocabulary of approximately 17,000 words.

Table 1 shows the comparison of the properties of the two datasets. They were both annotated with binary labels. All features for both datasets were binary only. Both datasets were very sparse (sparsity is equal to the ratio between the number of cells with value zero and the total number cells in the data matrix), but the class distribution also was different for two datasets. Additionally, the ASSERTION dataset contained information at the sentence level, while the NOVA dataset was at the document level. The ASSERTION dataset is probably more difficult to classify because it has much higher number of features than NOVA.

Table 1. Experimental datasets for Active Learning.

Dataset Name	Number of samples	Number of Positive samples	Positive Rate	Number of Features	Feature Type	Sparsity	Class Type
ASSERTION	11,967	8,051	0.6728	71,986	Binary	0.9994	Binary
NOVA	19,466	2,769	0.2845	16,969	Binary	0.9967	Binary

2.2.2 Cross validation on active learning

To set up a pool-based active learning framework, a pool of unlabeled samples and an independent test set were initialized. The variability in performance could have been high if many different partitions in the data were created for generating the unlabeled pool and test set. To fully use both datasets and generate reliable results, 3-fold stratified cross validation was performed on active learning. On each of the cross validation iterations, the pool of unlabeled samples was from two folds and the evaluation of performance was based on the remaining fold. The validation results were averaged over three iterations.

2.2.3 Classification algorithm

To mainly focus on improving the querying algorithm, the same classifier with the same parameter was used on each run of classification (training and testing). In our preliminary experiments for selecting the best classifier and parameter, the linear Logistic Regression classifier outperformed linear SVM and Naïve Bayesian classifiers in 3-fold cross validation for all samples in both the ASSERTION and NOVA datasets. Therefore,

the Logistic Regression model implemented in the package "Liblinear" [52] was used. It can output the posterior probability as the prediction value. This output would be used as the input for most querying algorithms.

2.2.4 Active learning strategy

Based on the protocol of active learning described in Section 2, the global performance (learning curve) is influenced by many factors during the active learning process, such as initial performance (the classification performance based on the initial training set), the batch size, the stop criteria, the querying algorithm, etc. However, we designed the active learning experiment so that the querying algorithm would be the most influential factor. We fixed the initial and the final performance points in the learning curve as well as the batch size for each querying algorithm as follows.

We randomly selected three positive samples and three negative samples as the initial training set. In each iteration of the cross validation, all experiments with different querying algorithms would use the same initial training set and, therefore, have the same initial point in the learning curve.

According to the stop criteria, the active learning process stopped when the entire pool of unlabeled samples was queried or U was empty. In each iteration of the cross validation, all experiments with different querying algorithms would have the same final point in the learning curve.

For batch size selection, we used 2^{i+2} training samples with labels where i is the index of iteration in the active learning process up to the total number of training

samples. For example, the size of labeled training set L on each iteration would be 8, 16, 32, 64, 128,…, 4096, …, and the maximum number.

The querying algorithm is the function to assess how informative each instance x is in unlabeled pool U. x^* is selected as the most informative sample according to the function $x^* = \text{argmax} Q(\mathbf{x})$, where $Q(\mathbf{x})$ is the querying function that outputs the informativeness or querying value (**Q value**) for data matrix **x** in U.

2.2.4.1 Uncertainty sampling based algorithms

Uncertainty sampling queries the sample with the least certainty or on the decision boundary. The simplest uncertainty sampling algorithm is called Least Confidence (**LC**), which is straightforward for the probabilistic models:

$$Q^{\text{LC}}(\mathbf{x}) = 1 - P(\mathbf{y}^*|\mathbf{x};\theta)$$

where $\mathbf{y}^*$ is the most likely label sequence for **x**. θ is the model that generates the posterior probability P of label **y** given data matrix **x**. In the binary classification case, LC is equivalent to querying the instance with the highest Q value (or uncertainty value) that is nearest the 0.5 posterior probability of being in the positive or negative class. In the case of the ASSERTION dataset, if the concept term was classified as "present" with the probability closer to 0.5 versus "non-present," the term was more likely to be selected for annotation in the next iteration of active learning.

During the active learning process, the class distribution of the training set could become imbalanced (with more positive/negative than negative/positive samples). At this point, we assume that the sample in the minority class is more informative. Moreover, we

would like to balance the training set as much as possible in the early iteration of active learning because the classifier would tend to ignore the minority class, resulting in a poor prediction model, especially with a small number of labeled training samples. Therefore, we implemented another uncertainty sampling algorithm called Least Confidence with Bias (**LCB**) [48] , which considers both the uncertainty value from the current prediction model and the proportion of class labels in the training set. LCB is more likely to query the instances around the decision boundary and compensates for class imbalance.

Let *pp* be the percentage of positive labels in the current training set. We defined $P_{\max}$ as the posterior probability that gives the highest Q value in LCB function $Q^{\mathrm{LCB}}(\mathbf{x})$:

$$Q^{LCB}(x) = \begin{cases} \dfrac{P(y=1 \mid x;\theta)}{P_{\max}}; \text{ if } P(y=1 \mid x;\theta) < P_{\max} \\ \dfrac{1-P(y=1 \mid x;\theta)}{1-P_{\max}}; \text{ otherwise} \end{cases}$$

where $P_{\max} = mean(0.5, 1-pp)$. When $P_{\max} = 0.5$ or $pp = 0.5$, it is equivalent to LC.

Both LC and LCB methods depend on the quality of the prediction model because both algorithms control the sample selection based on the posterior probability output from the model. When the model is poor, it propagates the negative effect to the querying algorithm. LCB could bias the Q value so that the model can converge more quickly to a good one by balancing the training set in the early stage of active learning. However, when the model improves, the bias could increase too much. So we also proposed another modified version of uncertainty sampling called Least Confidence with Dynamic Bias (**LCB2**), which also considers the size of the current training set. Note that the model is likely to be more reliable when the classification model is trained by a larger set of samples. For the binary classification problem, we have more confidence that the highest

Q value is at the point closer to the posterior probability of 0.5 when more labeled training samples are used. $Q^{LCB2}(\mathbf{x})$ is the same as $Q^{LCB}(\mathbf{x})$ except for $P_{\max}$:

$$P_{\max} = w_b*(1-pp) + w_u*0.5$$

where w_b is the weight of bias and w_u is the weight of uncertainty, and $w_b = 1 - w_u$, where w_u is the ratio of $|L|$, the size of the current labeled set, and $|U_0|$, the size of initial unlabeled pool: $w_u = |L|/|U_0|$. When $w_u = 0$, it is equivalent to LCB; when $w_u = 1$, it is equivalent to LC.

2.2.4.2 Model change sampling based algorithms

Model change sampling algorithm (**MC**) is a heuristic method to improve the querying method that relies on the classification model. For example, uncertainty sampling might fail to find the most uncertain samples when given a poor probabilistic model for classification. It is as difficult as finding the true decision boundary by classification model. We implemented the idea of model change for querying on top of model dependent querying methods such as uncertainty sampling. The MC algorithm considers the Q value from not only the current model but also the previous one. It controls the sample selection based on the change of Q values from different models during the active learning process.

We derived the heuristic function based on the following assumption. When the classification model is improving during the active learning process, the posterior predictions for each sample will be closer to either zero or one. In other words, the Q value for each sample, which is the uncertainty value based on LC, LCB or LCB2,

becomes smaller and smaller. The heuristic function takes into account the change of Q values over different models. The model change sampling algorithm ranks the unlabeled instances based on the following rule: the instance with the most increasing Q values is the most informative one. If the Q values for all instances are decreasing, the instance with the least decreasing Q value is also considered as the most informative one in the dataset. It also needs to consider the improvement of the model during the active learning process. The Q value for the previous model is discounted because the current model is intuitively better than the previous one.

$$Q^{MC}(\mathbf{x}) = Q(\mathbf{x}, i) - w_o * Q(\mathbf{x}, i\text{-}1)$$

where i represents the current iteration in the active learning process, i-1 is the index of the previous iteration; w_o is the weight of the old model, which is equal to $1/|L|$ ($|L|$ is the size of the current training set). We applied this formula to uncertainty sampling based querying methods (LC, LCB, and LCB2) so that we had three MC querying algorithms in our study: Least Confidence with Model Change (**LCMC**), Least Confidence with Bias and Model Change, (**LCBMC**), and Least Confidence with Dynamic Bias and Model Change (**LCB2MC**).

2.2.4.3 Information density based algorithms

The information density (**ID**) framework proposed by Settles and Craven[7] considers not only the uncertainty of instances but also the data distribution. The most uncertain instance lies on the decision boundary, but it is not necessarily representative of

other instances in the distribution. Thus knowing its label is not likely to improve the prediction model. Here is the ID-based querying function $Q^{ID}(\mathbf{x})$:

$$Q^{ID}(\mathbf{x}) = Q^{US}(\mathbf{x}) * Q^{D}(\mathbf{x})^{\beta}$$

where $Q^{US}(\mathbf{x})$ is the Q value by any uncertainty sampling based method (like LC, LCB, or LCB2); $Q^{D}(\mathbf{x})$ is the density function to compute how representative it is for any given instance in the unlabeled set; β is the control factor for the density term. In this study, we implemented an information density approach based on the Euclidean distance to the centers of labeled set *L*. These centers can represent the dense regions in the input space[16]. In our preliminary study, we only considered one center because it is difficult to determine the appropriate numbers of centers for selecting the most representative sample:

$$Q^{D}(\mathbf{x}) = \frac{1}{1 + dist(\mathbf{x}, \hat{x})}$$

where $\hat{x}$ is the mean vector for each variable over all samples in the labeled set *L*; *dist*(.) is the function for computing the Euclidean distance to this mean vector for each sample in **x**. We called this method Information Density Based on Distance to Center (**IDD**). In our experiment, we used method LCB2 in the first term $Q^{US}(\mathbf{x})$ of IDD.

2.2.5 Evaluation

We applied the same evaluation measures used for the active learning challenge 2010[53]. The prediction for the performance of active learning was evaluated according to the Area under the Learning Curve (**ALC**). The learning curve plotted the Area Under the ROC curve score (**AUC**) computed on all the samples in the test set as a function of the

number of labels queried. The global score or ALC score was normalized based on the following function:

$$\text{ALC score} = \frac{\text{ALC} - Arand}{A\text{max} - Arand}$$

where *A*max is the area under the best achievable learning curve (1.00 AUC on all points of the learning curve) and *A*rand is the area under the learning curve obtained by random prediction (0.50 AUC on all points of the learning curve). The learning curve of two neighbor points was interpolated linearly.

In the x-axis of the learning curve, we used log2 scaling. It is consistent with the batch size (2^{i+2}) of active learning, and this scaling actually increases the difficulty of getting a high global score because each additional labeled sample in the early stage of active learning is much more important than the one in the late stage. The performance in the early stages is more significant for the global score, so our target was also to improve the prediction model given a small number of training samples with labels.

Three learning curves were generated in the 3-fold cross validation of active learning for the experiment of each querying algorithm. Then the average learning curve was determined by averaging the AUC scores on each corresponding point from the three learning curves. The final global score of each querying algorithm was the ALC score from the average learning curve.

We ran the active learning experiments for eight querying algorithms and two datasets. The passive learner used the random querying method, while the active learner used other querying approaches. Since the passive learner generated results with high variance from the random factor for sampling, we averaged the learning curves of the

random querying method over 50 runs using the same start point, end point, and batch size.

2.3 Results

Results for the ASSERTION dataset showed that the ALC scores of all active learning methods except IDD outperformed the baseline using the random sampling method. In terms of the global performance, the active learner LCBMC had the best performance on both the ASSERTION and NOVA datasets. Most of the other active learners also performed better than passive leaner. LCB improved the performance by the basic uncertainty sampling method LC, while LCB2 could generate a better learning curve than LCB. The performances of LC, LCB, and LCB2 were consistent for both datasets. The model change-based method improved the uncertainty sampling methods LC and LCB in both datasets, but the performance of LCB2MC was poorer than LCB2. The active learners LC and IDD did not perform well in our experiments on both datasets.

Table 2 shows the cross validation results of ALC scores for both datasets and the different querying algorithms. ALC scores from individual folds, as well as the average of the three folds, were reported.

Table 2. ALC results for Threefold Cross Validation of Active Learning for Two Datasets and Eight Querying Methods.

Dataset	Querying Method			Fold1	Fold2	Fold3	Average	Standard Deviation
	Category	New/Existing	Name					
ASSERTION Dataset	Uncertainty Sampling	Existing	LC	0.7160	0.7524	0.7586	**0.7423**	0.0230
		Existing	LCB	0.7423	0.7836	0.7560	**0.7606**	0.0210
		New	LCB2	0.7536	0.7773	0.7597	**0.7635**	0.0123
	Model Change	New	LCMC	0.7171	0.7656	0.7644	**0.7490**	0.0277
		New	LCBMC	0.7503	0.7839	0.7803	**0.7715**	0.0184
		New	LCB2MC	0.7182	0.7624	0.7615	**0.7474**	0.0253
	Information Density	Existing	IDD	0.7144	0.7268	0.6947	**0.7120**	0.0162
	Baseline	Existing	Random (50 runs)	0.7151	0.7647	0.7434	**0.7411**	0.0249
NOVA Dataset	Uncertainty Sampling	Existing	LC	0.7643	0.6805	0.7251	**0.7233**	0.0419
		Existing	LCB	0.8524	0.8163	0.8603	**0.8430**	0.0235
		New	LCB2	0.8722	0.8344	0.8546	**0.8537**	0.0189
	Model Change	New	LCMC	0.8771	0.8144	0.8472	**0.8462**	0.0314
		New	LCBMC	0.8702	0.8289	0.8719	**0.8570**	0.0244
		New	LCB2MC	0.8768	0.8323	0.8295	**0.8462**	0.0265
	Information Density	Existing	IDD	0.7297	0.6970	0.7161	**0.7143**	0.0164
	Baseline	Existing	Random (50 runs)	0.8151	0.7847	0.8001	**0.8000**	0.0152

Note: LC: Least Confidence; LCB: Least Confidence with Bias; LCB2: Least Confidence with Dynamic Bias; LCMC: Least Confidence with Model Change; LCBMC: Least Confidence with Bias and Model Change; LCB2MC: Least Confidence with Dynamic Bias and Model Change; IDD: Information Density based on Distance to Center.

Figure 1 and Figure 2 show the average learning curves for datasets ASSERTION and NOVA, respectively, for all eight querying methods. In general, LCBMC, which had the highest global score, showed stability with small training sample sizes. On the other hand, the querying methods with low global scores performed poorly or were unstable in the early stage of the active learning process.

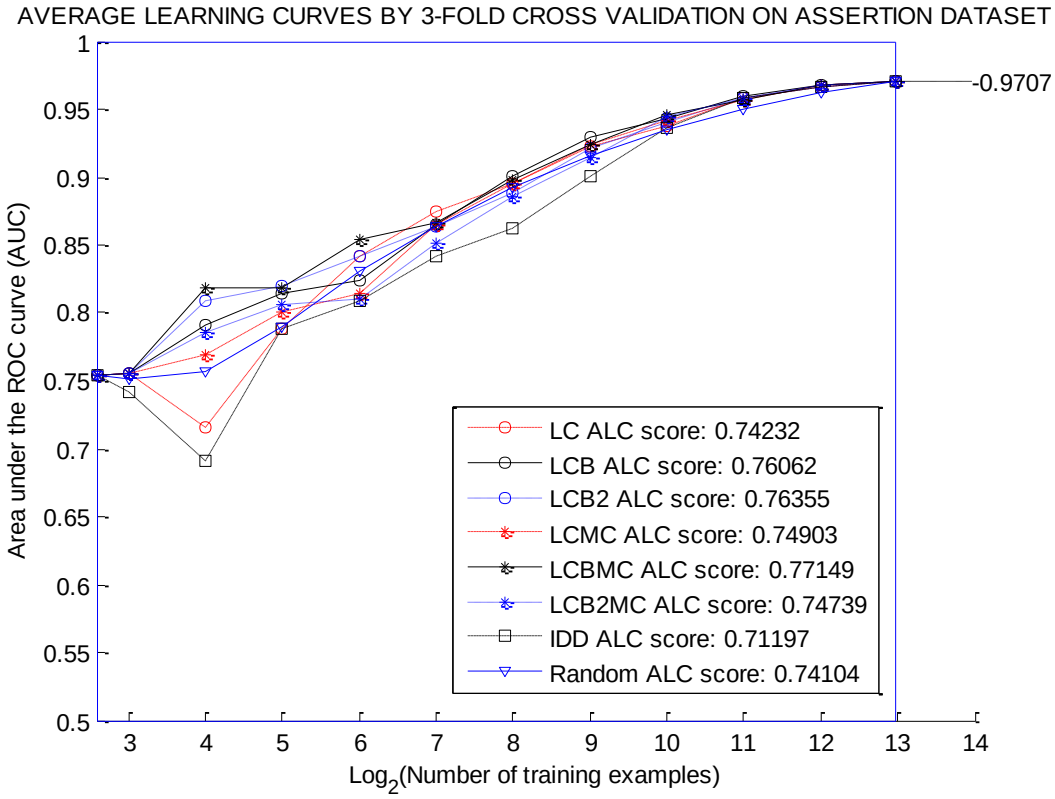

Figure 1. Average Learning Curves for 8 Querying Algorithms on the Assertion Dataset.

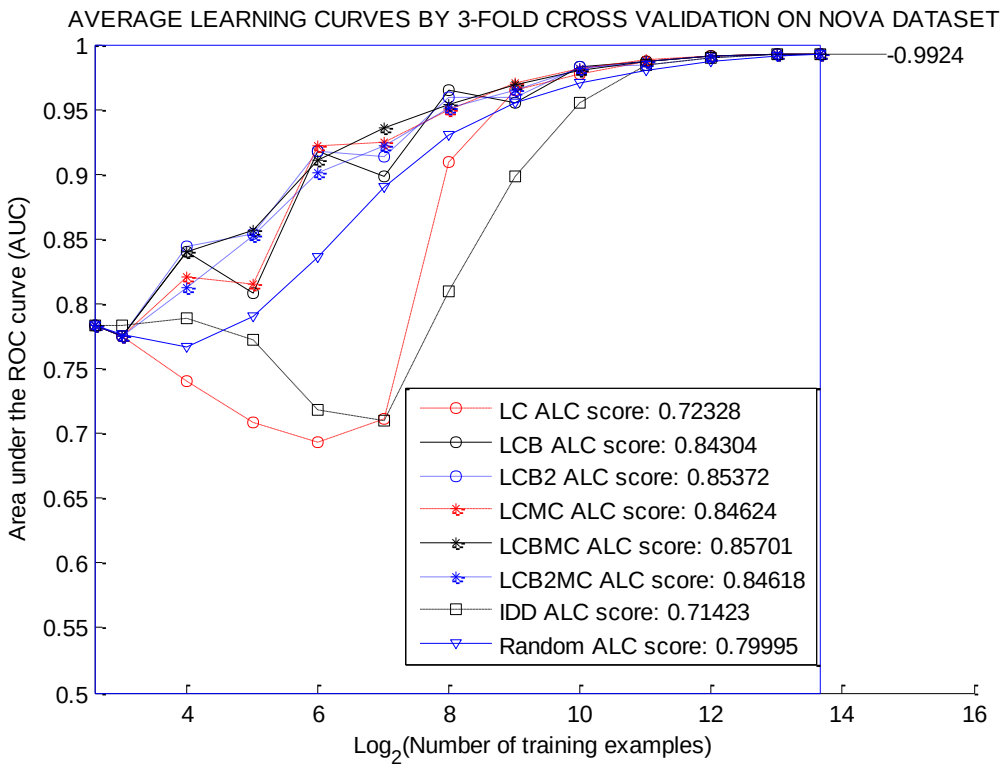

Figure 2. Average Learning Curves for 8 Querying Algorithms on the NOVA Dataset.

We can compare eight querying algorithms on the same figure vertically and horizontally. By reading vertically, we can compare the performance of eight prediction models in AUC at each stage of active learning; by reading horizontally, we can compare the costs of annotation (number of labeled samples used) by eight querying methods for each quality level of the prediction model in AUC.

Table 3 presents the evaluation of prediction models based on the average AUC score and its standard deviation when the size of querying samples was small. This table magnifies the intermediate results in the early stage of the learning curve with 16, 32, and 64 training samples. The average AUC by random querying method was not the worst in the early stage of active learning, but the standard deviation was higher compared to the other methods. The best querying method in our experiments, LCBMC, performed

reasonably well with a high average AUC and low standard deviation when only a small number of training samples was used.

Table 3. Evaluation of the classification model for eight querying algorithms and two datasets on a small training set (with 16, 32, and 64 training samples) based on average AUC score and the standard deviation.

Dataset	Size of Training Set	LC	LCB	LCB2	LCMC	LCBMC	LCB2MC	IDD	Random
ASSERTION Dataset	16	71.52% ± 2.55%	79.16% ± 3.85%	80.91% ± 1.31%	76.87% ± 5.89%	81.92% ± 1.41%	78.55% ± 4.73%	69.10% ± 2.70%	75.65% ± 5.83%
	32	78.85% ± 4.30%	81.46% ± 2.77%	82.04% ± 1.38%	80.11% ± 1.91%	81.87% ± 2.45%	80.61% ± 1.31%	78.77% ± 2.55%	79.00% ± 4.31%
	64	84.16% ± 1.42%	82.33% ± 2.05%	84.16% ± 0.74%	81.45% ± 2.34%	85.42% ± 1.03%	81.07% ± 1.45%	80.88% ± 3.18%	83.12% ± 2.25%
NOVA Dataset	16	73.98% ± 6.25%	83.99% ± 4.93%	84.42% ± 3.66%	82.01% ± 4.00%	83.98% ± 5.81%	81.30% ± 4.61%	78.91% ± 4.22%	76.70% ± 7.06%
	32	70.88% ± 2.96%	80.82% ± 5.35%	85.33% ± 1.63%	81.52% ± 6.97%	85.69% ± 3.05%	85.25% ± 3.15%	77.27% ± 2.50%	79.03% ± 6.96%
	64	69.38% ± 5.05%	91.79% ± 0.54%	91.82% ± 0.58%	92.21% ± 0.71%	91.03% ± 2.80%	90.16% ± 1.04%	71.77% ± 2.63%	83.57% ± 4.88%

Table 4 presents the evaluation of the prediction model when the training set was large (with 1024, 2048, and 4096 samples). This table magnifies the intermediate results for the late stage of active learning. In this stage, the active learners performed better when compared with the passive learner on the ASSERTION dataset. It is also true for the NOVA dataset with training sample sizes of 2048 or higher.

In addition, none of the experiments needed much computational time. The querying algorithms could rank or generate Q values for all samples in the unlabeled pool on both datasets (more than 8000 samples) in less than one second. The classifier Logistic Regression in the "Liblinear" package could complete three-fold cross validation (for the end point in the learning curve) in less than three seconds for the ASSERTION

dataset (with about 12,000 samples) and four seconds for the NOVA dataset (with about 20,000 samples).

Table 4. Evaluation of the classification model for eight querying algorithms and two datasets with a large training set (with 1024, 2048, and 4096 training samples) based on average AUC score and the standard deviation.

Dataset	Size of Training Set	LC	LCB	LCB2	LCMC	LCBMC	LCB2MC	IDD	Random
ASSERTION Dataset	1024	93.73% ± 0.48%	94.34% ± 0.53%	94.09% ± 0.47%	94.23% ± 0.57%	94.64% ± 0.26%	94.44% ± 0.41%	93.57% ± 0.56%	93.46% ± 0.46%
	2048	95.81% ± 0.26%	95.94% ± 0.24%	95.80% ± 0.41%	95.67% ± 0.47%	95.74% ± 0.29%	95.88% ± 0.54%	95.77% ± 0.11%	94.99% ± 0.33%
	4096	96.67% ± 0.28%	96.76% ± 0.28%	96.66% ± 0.36%	96.76% ± 0.26%	96.74% ± 0.30%	96.82% ± 0.32%	96.70% ± 0.24%	96.18% ± 0.26%
NOVA Dataset	1024	97.71% ± 0.62%	98.26% ± 0.36%	98.29% ± 0.22%	98.10% ± 0.18%	98.03% ± 0.51%	98.10% ± 0.39%	95.48% ± 0.46%	97.05% ± 0.42%
	2048	98.66% ± 0.23%	98.69% ± 0.25%	98.38% ± 0.26%	98.83% ± 0.22%	98.65% ± 0.23%	98.65% ± 0.41%	98.39% ± 0.26%	98.03% ± 0.28%
	4096	99.07% ± 0.20%	99.10% ± 0.19%	99.02% ± 0.28%	99.11% ± 0.20%	99.11% ± 0.25%	99.08% ± 0.24%	99.00% ± 0.20%	98.63% ± 0.21%

To assess whether there are significant differences in terms of mean ALC global scores among different active learners and the passive learner, we conducted a statistical test based on results from bootstrapping. We re-sampled the test set by random sampling with replacement for 200 times and generated 200 bootstrapping data sets. For each bootstrapping data set, we evaluated and reported ALC global scores for different active learners and the passive learner. We used Wilcoxon signed rank test, [54] a non-parametric test for paired samples, to assess whether differences between two methods are statistically significant. As there were eight different methods (28 comparisons in total), we applied Bonferroni correction [55] to adjust for multiple comparisons, with family-wise

type I error control at alpha = 0.05. Therefore, if the p-value from Wilcoxon signed rank test was less than 0.0018 (0.05/28), we claimed that there was a statistically significant difference between two methods. Table 5 shows the results of the statistical test. Except the ones between Random and LC, Random and IDD, LC and IDD, and LCMC and LCB2MC, all other comparisons showed statistically significant differences.

Table 5. Results of the statistical test (Wilcoxon signed rank test with Bonferroni correction for multiple testing) among ALC global scores from different active learners and the passive learner ("Y": Statistically significant; "N": Not statistically significant)

	LC	LCB	LCB2	LCMC	LCBMC	LCB2MC	IDD
Random (50 Runs)	N	Y	Y	Y	Y	Y	N
LC		Y	Y	Y	Y	Y	N
LCB			Y	Y	Y	Y	Y
LCB2				Y	Y	Y	Y
LCMC					Y	N	Y
LCBMC						Y	Y
LCB2MC							Y

Table 6 shows the variance of bootstrapping process with means and 95% confidence intervals (CIs) of the ALC scores for all querying algorithms. Table 7 shows the mean and 95% CIs in difference of ALC score from 200 bootstrapping samples between random sampling method and each other querying algorithm. Based on the result in Table 7, LCMC and LCB2MC did not perform significantly better than random sampling because 0 was within the 95% CI of the ALC difference. The difference in conclusion could be due to the following reasons. Wilcoxon signed rank test found a significant difference with respect to mean averaged over 200 runs, but did not look at the distribution of the difference. Table 7, instead, found the distribution of the difference by

using 95% confidence interval from the mean difference, which gives a better representation of what we could expect in the actual use.

Table 6. Mean and 95% confidence interval of 200 bootstrapping samples of ALC scores for 8 algorithms

	Random	LC	LCB	LCB2	LCMC	LCBMC	LCB2MC	IDD
mean	0.714	0.715	0.742	0.753	0.716	0.750	0.718	0.714
2.5% percentile	0.695	0.696	0.724	0.730	0.697	0.730	0.696	0.696
97.5% percentile	0.734	0.734	0.758	0.772	0.736	0.769	0.737	0.731

Table 7. Mean and 95% confidence interval of the ALC score difference between random sampling and other 7 querying methods from 200 bootstrapping samples

	LC	LCB	LCB2	LCMC	LCBMC	LCB2MC	IDD
mean	0.001	0.028	0.039	0.002	0.035	0.003	0.000
2.5% percentile	-0.010	0.017	0.030	-0.010	0.024	-0.008	-0.012
97.5% percentile	0.012	0.039	0.050	0.013	0.046	0.016	0.010

2.4 Discussion

For the concept assertion classification task, active learners generated better prediction models with higher AUC scores, and required less annotation effort than the passive learner (based on the results shown in Tables 3 and Table 4). Using the ASSERTION dataset, the prediction model trained by 32 randomly selected annotated samples had a 0.7900 average AUC score; however, LCBMC could achieve the prediction model with a 0.8192 average AUC by using 16 annotated samples, which saved half of the annotation cost. Overall, the active learning strategy was more efficient

in reducing annotation costs and improving prediction models for the clinical dataset ASSERTION. In Figure 1, the best learning curve by LCBMC lay above the average learning curve by random sampling. The result for the general English dataset NOVA was also consistent with the ASSERTION dataset. Such findings show that active learning strategies hold promise in solving similar clinical text classification problems when annotation is expensive and time-consuming.

To further analyze the learning curves for the ASSERTION dataset, we calculated the approximate numbers of training cases at different levels of AUC, for both active learning approaches and random sampling approaches (Table 8). In the early stage of active learning, the random sampling method used 32 samples to achieve an AUC of 0.79, while LCBMC used only 12 samples to achieve the same AUC, a 62.5% of reduction in sample size. In the middle stage of active learning, the random sampling method used 512 labeled cases to train a model with an AUC of 0.92, while LCB used about 369 samples to build the same model. In the late stage, the random sampling method required 4,096 samples to generate a model with an AUC of 0.96, while LCB used only 2,518 samples to reach the same AUC. This analysis demonstrates that active learning methods require fewer training samples than the random sampling method, with similar classification performances.

The basic uncertainty sampling algorithm LC and the information density algorithm IDD did not perform well in active learning on both datasets. LC could not find the most informative samples when the annotated instances were insufficient, because LC relies on the performance of a probabilistic model that was poor in the early stage of active learning. However, LCB and LCB2 could improve the performance for both

datasets by also considering the imbalance of class and the quality of the classification model. The model change-based method LCBMC further improved the uncertainty-based method LCB by considering the change of informative values between models. The information density-based method IDD failed to improve the global score, because the density term based on distance to center did not find the most "representative" samples for both datasets. It negatively affected the overall performance, even though the uncertainty term by LCB2 could perform reasonably well by itself on both datasets.

Table 8. Approximate numbers of training samples at different levels of AUCs for both active learning algorithms and the random sampling method.

AUC	0.79	0.83	0.86	0.89	0.92	0.93	0.95	0.96
Random	32	64	128	256	512	1024	2048	4096
LC	33	**56**	**103**	**232**	**435**	**903**	**1557**	**2768**
LCB	**16**	73	**127**	**219**	**369**	**650**	**1354**	**2518**
LCB2	**13**	**46**	129	277	**462**	**824**	**1471**	**2785**
LCMC	**26**	81	**127**	**241**	**426**	**770**	**1473**	**2843**
LCBMC	**12**	**41**	**118**	**225**	**414**	**713**	**1271**	**2784**
LCB2MC	**19**	91	166	298	524	**812**	**1330**	**2555**
IDD	35	102	269	443	694	**1002**	**1600**	**2790**

Although LCBMC was the best querying algorithm for both datasets based on the global score in our experiments, its learning curve for the ASSERTION dataset was not flawless. It could generate a classification model with 0.8192 and 0.8187 average AUC scores by using 16 and 32 annotated samples, respectively. Although the difference in AUC did not seem significant, we did not expect that the model would get worse with larger training sets. Further investigation of the querying algorithm is needed to improve the stability of the learning curve. One possible direction worth investigating is to

automatically select the batch size as a function of the probabilistic prediction and querying model in the iteration of active learning, instead of pre-setting this parameter.

CHAPTER III

STUDY II: APPLYING ACTIVE LEARNING TO SUPERVISED WORD SENSE DISAMBIGUATION IN MEDLINE

3.1 Introduction

Word sense disambiguation (WSD) is the process of identifying the appropriate sense of an ambiguous word in a given context. WSD is important for many natural language processing (NLP) tasks, such as information extraction and information retrieval. [56] The ambiguity inherent in biomedical texts is a widely recognized problem. For example, "gene", an important entity in biomedical research, can have ambiguous names referring to: 1) multiple genes; 2) a gene or an English word not related to a gene; 3) RNA, protein, or gene; or 4) genes in different species. A gene name ambiguity study showed that 85.1% of correctly retrieved mouse genes were ambiguous, easily confused with other gene names from 21 organisms in a set of 45,000 abstracts associated with mouse genes[57].

Many different approaches have been developed for biomedical WSD tasks, as described in a review paper by Schuemie et al. [58] Among them, supervised machine learning-based WSD methods have received considerable attention and have shown very good results in both general English texts [26,27,59-61] and biomedical texts such as MEDLINE abstracts. [28] Supervised WSD approaches usually build a classification model for each ambiguous word by learning from an annotated corpus containing instances of each possible sense of the word. Despite its high performance, supervised WSD has

limited scalability it is a costly and time-consuming process to build a sense-annotated corpus for each ambiguous term in biomedical texts. Researchers have investigated different automated methods to create pseudo-corpora with labeled senses and have used them for supervised WSD methods (also called semi-supervised). [62,63] Despite the successes, WSD methods based on pseudo- corpora did not perform as well as supervised WSD systems that were based on annotated instances from the real corpus. [58] An alternative new approach presented in this study is to investigate how active learning strategies can be integrated with supervised WSD methods to reduce the number of annotated samples required by a satisfactory classification model.

In this chapter, we describe a study where we applied three different active learning algorithms to Support Vector Machines (SVM) based disambiguation models for 197 ambiguous terms from MEDLINE abstracts. We compared learning curves between three active learners and a passive learner, based on random sampling across 197 WSD tasks.

3.2 Methods

Three different uncertainty sampling-based active learning algorithms (Least Confidence (LC), Margin, and Entropy) and one passive learning method (random sampling) were integrated with an SVM classifier to disambiguate 197 ambiguous words and abbreviations in the MSH WSD collection derived from MEDLINE abstracts. For each ambiguous term and for each learning algorithm, an average learning curve was generated that plots the accuracy computed from the test set as a function of the number

of annotated samples used in the model via a 10-fold cross-validation. The Area under the average Learning Curve (ALC) was used as the primary metric for evaluation.

3.2.1 WSD Dataset

In this study, we used the MSH WSD dataset developed by Jimeno-Yepes et al. [64] This benchmark dataset was downloaded from the National Library of Medicine (NLM) WSD test collection collaboration. [65] The generation of MSH WSD is based on exploiting MeSH indexing in MEDLINE abstracts. It consists of 106 ambiguous abbreviations, 88 ambiguous terms and 9 of which are a combination of both, for a total of 203 ambiguous words. [64] Each instance containing the ambiguous word is assigned an appropriate sense that is represented using a Concept Unique Identifier (CUI) from the 2009AB version of the UMLS (Unified Medical Language System). For each ambiguous term/abbreviation, the dataset contains a maximum of 100 instances obtained from MEDLINE for each sense, resulting in 37,888 ambiguous cases in 37,090 MEDLINE citations. [64] Jimeno-Yepes et al. [64] also evaluated machine learning based WSD algorithms on this data set and reported an accuracy of 0.9386 for the entire MSH WSD data set, when words from titles and abstracts were used as features. To ensure that we had enough samples for training and testing, we included ambiguous words that have more than 100 instances in total for all senses in this study, resulting in 197 words. Among them, 111 are abbreviations and 86 are unabbreviated terms. In addition, 14 out of 197 words have more than two senses and the remaining 183 words have exactly two senses. Table A1 in appendix shows the frequency distribution of the senses for each ambiguous word in the data set.

3.2.2 Active Learning-Enabled Supervised WSD

3.2.2.1 The Pool-Based Active Learning Approach to Classification

An active learning-based classification system mainly consists of two core components: a classification model and an active sample selection or a querying model. The pool-based active learning approach to classification [5] was used in this study. The approach starts with a pool of unlabeled samples and it iteratively selects informative samples for annotation and model development.

In the MSH WSD dataset, the pool size varies from 100 to 500, depending on the ambiguous word. We pretended that labels of samples were not available when running the querying algorithms. For the initial training set, we randomly selected two samples from the entire pool. All experiments with different querying algorithms used the same initial training set and, therefore, have the same initial point in the learning curve. In this study, we used a batch size of one in all experiments so that we could closely monitor the performance increase by every incremental training sample. As the minimum number of training samples for an ambiguous word was 100 and we used 10-fold cross validation in the evaluation (see Section 3.2.3), we stopped the active learning process when 90 training samples were queried.

3.2.2.2 The WSD Classification Model

The WSD classification model was built on the Support Vector Machines (SVMs) algorithm with linear kernel in the package "Liblinear". [52] We used a one-vs.-all multi-

class classification model if the ambiguous term has more than two senses. As optimized parameters of SVM were different for 197 words in the data set, we used a common setting: s = 1 (L2-regularized L2-loss support vector classification) and c = 1, for all words in this study, which performed comparably to the previous study. [64] The numeric outputs by SVM were mapped into the probabilistic domain (values from 0 to 1) by a sigmoid/logistic function. All words (except the ambiguous word itself) occurring in the title and abstract of a citation where the ambiguous term appears were used as features for SVM classifiers, similarly to the previously reported study. [64]

3.2.2.3 Active Learner and Passive Learner

The second core component of active learning is the querying method. In general, there are two types of learners: active learner and passive learner. The passive learner (PL) randomly queries instances from the pool of unlabeled samples, without considering the information about samples in the pool. The active learner (AL), on the other hand, will select the instances that are the most promising, improving the predictive performance of the model. x^* is selected as the most informative sample according to the function x^*=argmax Q(x), where Q(x) is the querying algorithm that outputs the informativeness or querying value (Q value) for data matrix x in U. In this study, we implemented three uncertainty sampling-based querying algorithms that query the sample with the least certainty or closest to the decision boundary. It is applicable to multi-class classification problems such as supervised WSD tasks.

The simplest uncertainty sampling algorithm is called Least Confidence (LC), which is straightforward for the probabilistic models:

$$Q^{LC}(x) = 1 - P(y^*|x; \theta)$$

where y^* is the most likely label sequence for x. θ is the model that generates the posterior probability P of label **y** given data matrix x. In the binary classification case, LC is equivalent to querying the instance with the highest Q value (or uncertainty value) that is nearest the 0.5 posterior probability of being in the positive or negative class.

As LC only considers information about the most probable label, we also used a different multi-class uncertainty sampling method called margin sampling (Margin):

$$Q^{margin}(x) = P(y_2^*|x; \theta) - P(y_1^*|x; \theta),$$

where y_1* and y_2* are the first and second most probable class labels under the model, respectively. The intuition of this algorithm is that the samples with larger margins are easier to differentiate between the two most likely class labels, while the samples with smaller margins are more ambiguous. Thus, the margin sampling algorithm outputs the sample with the smallest difference between the two most likely class labels.

For problems with very large label sets, however, the margin method still ignores much of the output distribution for the remaining classes. Thus we implemented a more general uncertainty sampling strategy called Entropy:

$$Q^{entropy}(x) = -\sum_i P(y_i | x; \theta) \log P(y_i | x; \theta)$$

where y_i ranges over all possible labels. Entropy is a measure of uncertainty or impurity over all possible labels in a machine-learning task.

For binary classification, all three are equivalent to querying the instance with a class posterior closest to 0.5. All three querying algorithms were expected to have identical performance on 183 ambiguous words that have only two possible senses. Therefore we focused the comparison study among three querying algorithms only on the 14 ambiguous words with more than two senses.

3.2.3 Evaluation

In this study, we used the evaluation measurements similar to those in the 2010 active learning challenge. [50] The performance of the active learning-enabled classification system was evaluated by a learning curve, which plotted the accuracy (ACC) computed using the test set as a function of the number of labels annotated. ACC was defined as the ratio between the number of correctly identified samples and the number of all samples in the test set. A commonly used global measure for active learning systems, the Area under the Learning Curve (ALC), was also reported in this study. The global ALC score was normalized by the area under the best achievable learning curve (1.00 ACC on all points of the learning curve). When measuring the area under a learning curve, two neighbor points on the curve were interpolated linearly.

To evaluate a pool-based active learning framework, we need not only a pool of unlabeled samples (that will be labeled during the querying step), but also an independently labeled test set. To generate reliable results, 10-fold cross validation (CV) was performed on active learning. At each cross validation iteration, nine folds formed the pool of unlabeled samples and the remaining fold was used for the evaluation of

performance. For each ambiguous word in the MSH WSD dataset and a given querying algorithm (LC, Margin, or Entropy), ten learning curves were generated from 10-fold CV experiments. Each learning curve started from two initial training samples and stopped at 90 training samples. An average learning curve was then created by averaging the ACC scores at each corresponding point for these ten individual learning curves. The global score for each querying algorithm was then the ALC score from the averaged learning curve. Since the passive learner generated results with high variance due to random sampling, we averaged the results of the random querying method over 10 runs using the same start point, end point, and batch size.

To better summarize and compare the three querying methods and random sampling method, we generated a global average learning curve for each method from all learning curves of 197 words in the WSD data set. The global learning curve for a given method was generated by averaging points with the same number of training samples from all 197 average learning curves.

To assess whether there is a significant difference between any two learners (three active learners and one passive learner) in terms of average ALC scores from 197 ambiguous words, we used the Wilcoxon signed rank test, [54] a non-parametric test for paired samples. As there were four different methods (6 pair-wise comparisons in total), we applied a Bonferroni correction [55] to adjust for multiple comparisons, with family-wise type I error control at alpha = 0.05. Therefore, if the p-value from the Wilcoxon signed rank test was less than 0.0083 (0.05/6), we claimed that there was a statistically significant difference between two methods.

3.3 Results

For each of 197 ambiguous words, we evaluated four learning methods (three active learners and one passive learner) and generated corresponding learning curves and global ALC scores. Table 9 shows the average ALC scores for all 197 words and some subsets. Detailed ALC scores for each ambiguous word and each learning algorithm are available at the Appendix (Table A1). For any subsets in Table 9, the three active learning algorithms had close average ALC scores, but they were better than the passive learning method (random sampling). Wilcoxon signed rank tests showed that the average ALC scores generated by active learners using LC, Margin, or Entropy querying algorithms were statistically significantly better than ALC scores by the passive learner, in all subsets. However, the tests also revealed that the three active learners were not statistically significantly different. As shown in the last column of Table 9, active learners outperformed the passive learner for 177 out of all 197 words (89.84%), 101 out of 111 abbreviations (90.99%), 76 out of 86 non-abbreviated terms (88.37%), and 13 out of 14 terms with more than two senses (92.85%).

Figure 3 shows the global learning curves across 197 words for the three active learning algorithms (LC, Margin, and Entropy) and the passive learning algorithm (Random). The learning curves of the three active learning algorithms almost overlapped, but they were clearly above the random sampling curve.

Table 9. Average ALC scores for three active learning algorithms (LC, Margin, and Entropy) and one passive learning method (Random), across all 197 ambiguous words and their subsets from the MSH WSD dataset.

MSH WSD dataset (subset)	Average ALC score				Active learner advantage percentage
	LC	Margin	Entropy	Random	
197 words	0.838	0.838	0.838	0.804	177 out of 197 (89.84%)
111 abbreviations	0.885	0.885	0.885	0.845	101 out of 111 (90.99%)
86 non-abbreviated terms	0.778	0.777	0.778	0.752	76 out of 86 (88.37%)
14 words with more than 2 senses	0.764	0.761	0.761	0.723	13 out of 14 (92.85%)

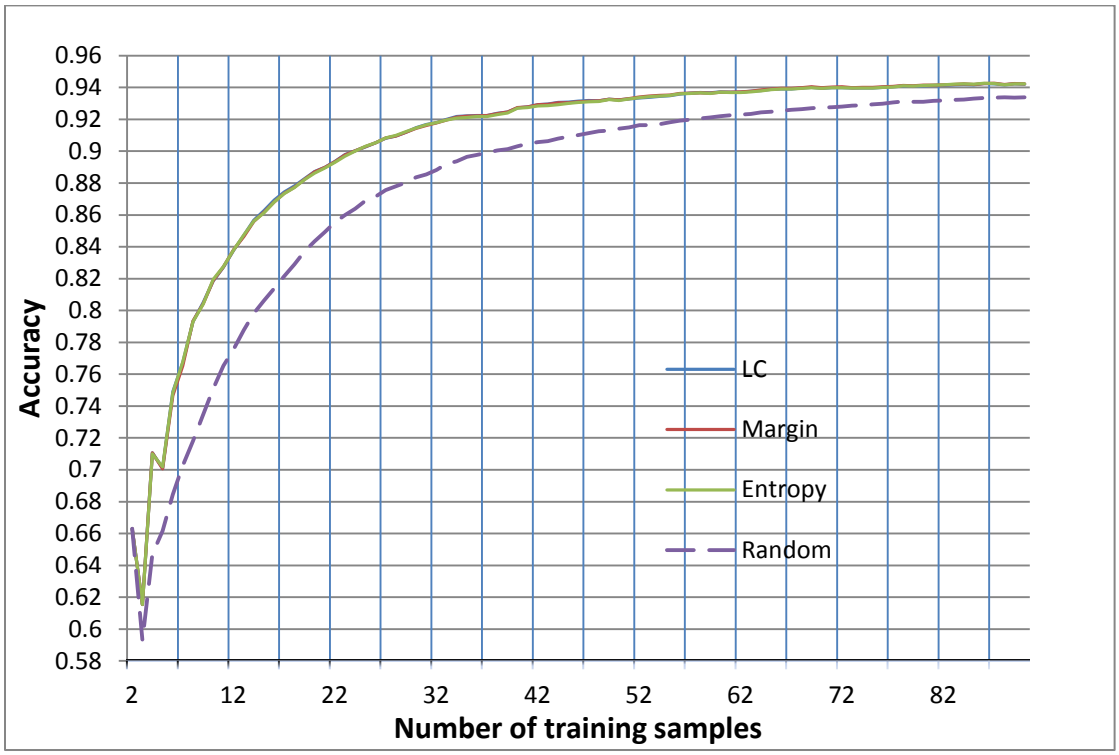

Figure 3. Average Learning Curves comparison over 197 words in MSH WSD dataset

Based on the learning curves, we further reported the approximate numbers of training samples needed on average at different performance levels of supervised WSD systems for both active learning algorithms and random sampling (Table 10). We

calculated the numbers of required training samples for different methods at different ACC values (0.75-0.90). It was clear that the active learners required fewer annotated training samples than the passive learner in order to reach the same accuracy for WSD tasks. For example, to train the WSD system to achieve an accuracy of 0.90, we needed 38 training samples for the random sampling method. But the active learners needed 24 training samples only, indicating a 37% (14/38) decrease in annotated training samples.

Table 10. Approximate numbers of training samples needed on average at different accuracy values for both active learners and passive learner.

Accuracy	LC	Margin	Entropy	Random
0.70	5	5	5	7
0.75	6	6	6	10
0.80	9	9	9	15
0.85	13	13	13	21
0.90	24	24	24	38

Furthermore, we also reported the performance of WSD systems integrated with different active learners and the passive learner when the number of training samples was fixed. Table 11 shows the accuracy of active learners and the passive learner when the number of training samples was set from 10 to 90, in increments of 10 samples. Our results showed that active learners could always generate higher accuracy than the passive learner when the same number of training samples was used. Additionally, the improvement of active learners was greater in the early stage (lower numbers of training samples needed).

Table 11. Accuracy of active learners and the passive learner across 197 ambiguous words when different numbers of training samples were used

Number of Training Samples	LC	Margin	Entropy	Random
10	0.819	0.819	0.820	0.751
20	0.887	0.887	0.886	0.844
30	0.915	0.914	0.915	0.884
40	0.927	0.928	0.927	0.903
50	0.932	0.932	0.932	0.914
60	0.937	0.937	0.937	0.922
70	0.939	0.940	0.940	0.927
80	0.941	0.942	0.941	0.931
90	0.942	0.942	0.942	0.934

As active learners may perform differently on multi-class classification tasks, we further conducted stratified analysis on the subset of 14 ambiguous words that had more than two senses. Table 12, Table 13, and Table 14 show the detailed ALC scores of four learners for these individual words. Active learners consistently showed better performance than the passive learner. Although LC achieved a slightly higher average ALC score (0.764) than Margin and Entropy (0.761), these differences were still not statistically significant according to the test. We also conducted stratified analysis on 111 abbreviations and detailed results can be found in Table 15 and Table 16. We noticed that abbreviations were relatively easier to disambiguate; active learners needed 50% fewer training samples on average than passive learner (33 versus 67) in order to achieve an accuracy of 96%. This could be because acronyms are often accompanied by the expanded (unambiguous) forms, e.g., "extraction of acylcarnitine (AC) and amino acids (AA)". In addition, senses of the same abbreviation are generally quite unrelated, which probably makes the disambiguation task easier than others.

Table 12. Active learning result for 14 words with more than two senses in MSH WSD test collection

Words	Sense Distribution					10-fold CV ALC scores			
	S1	**S2**	**S3**	**S4**	**S5**	**LC**	**Margin**	**Entropy**	**Random**
Ala	98	97	98	0	0	0.825	0.812	0.824	0.762
Ca	89	98	98	98	0	0.615	0.620	0.601	0.580
Cold	93	96	62	0	0	0.686	0.683	0.683	0.668
Cortical	95	99	98	0	0	0.748	0.727	0.733	0.675
CP	97	99	98	0	0	0.868	0.864	0.866	0.822
DDS	99	98	20	0	0	0.827	0.829	0.828	0.772
Ice	98	37	98	0	0	0.755	0.759	0.762	0.759
Lens	97	99	99	0	0	0.716	0.681	0.689	0.662
Lupus	99	99	91	0	0	0.671	0.671	0.671	0.659
PCA	99	99	99	95	98	0.796	0.827	0.808	0.769
PCP	99	99	54	0	0	0.865	0.862	0.869	0.814
RA	99	99	99	0	0	0.869	0.872	0.872	0.795
TAT	99	99	99	0	0	0.719	0.714	0.711	0.672
THYMUS	99	96	99	0	0	0.735	0.734	0.743	0.711
Average						0.764	0.761	0.761	0.723

Table 13, Approximate numbers of training samples needed on average at different accuracy values for both active learners and passive learner over 14 words with more than two senses

Accuracy	LC	Margin	Entropy	Random
0.50	4	4	4	6
0.60	6	7	6	10
0.70	11	11	10	17
0.75	14	15	15	22
0.80	19	19	20	27
0.85	28	27	27	38
0.90	49	49	53	69
0.91	59	60	61	80
0.92	86	82	81	>90

Table 14. Accuracy of active learners and the passive learner across 14 words with more than two senses when different numbers of training samples were used

Number of Training Samples	LC	Margin	Entropy	Random
10	0.6862	0.6880	0.6997	0.5913
20	0.8149	0.8105	0.7998	0.7310
30	0.8659	0.8608	0.8644	0.8158
40	0.8836	0.8870	0.8812	0.8542
50	0.9019	0.9006	0.8957	0.8731
60	0.9116	0.9102	0.9082	0.8927
70	0.9120	0.9190	0.9130	0.9009
80	0.9157	0.9206	0.9169	0.9104
90	0.9226	0.9232	0.9241	0.9128

Table 15. Approximate numbers of training samples needed on average at different accuracy values for both active learners and passive learner over 111 abbreviations

Accuracy	LC	Margin	Entropy	Random
0.75	5	5	5	7
0.80	6	6	6	10
0.85	8	8	8	14
0.90	13	13	13	21
0.95	26	26	26	46
0.96	32	33	32	67

Table 16. Accuracy of active learners and the passive learner across 111 abbreviations when different numbers of training samples were used

Number of Training Samples	LC	Margin	Entropy	Random
10	0.8808	0.8821	0.8826	0.8011
20	0.9377	0.9379	0.9380	0.8948
30	0.9570	0.9563	0.9575	0.9302
40	0.9637	0.9641	0.9642	0.9449
50	0.9669	0.9670	0.9666	0.9524
60	0.9692	0.9698	0.9691	0.9585
70	0.9701	0.9710	0.9700	0.9612
80	0.9707	0.9710	0.9707	0.9637
90	0.9707	0.9707	0.9707	0.9662

In addition, we tested the SVM-based WSD system alone by using all samples in the dataset, similar to the experiment in the previous study. Our SVM-based WSD system achieved an average accuracy of 0.944 (via 10-fold cross validation) for all 197 words, which was similar to the previously reported result. [64]

3.4 Discussion

In this study, we applied three different active learning algorithms to word sense disambiguation tasks in the MEDLINE corpus. To the best of our knowledge, this is the first attempt to explore the use of active learning in supervised WSD tasks in the biomedical domain. Our results based on the MSH WSD dataset showed that WSD systems integrated with active learners significantly outperformed the one with the passive learner (random sampling) in terms of average ALC score. Further analysis demonstrated that active learning strategies could not only reduce the number of training

samples required for supervised WSD systems, but could also improve classification models when the same number of training samples was used. These findings suggest the great potential of active learning in improving the scalability of supervised WSD approaches in the biomedical domain. To achieve high performance on this data set (over 90% accuracy), supervised WSD systems would require a few dozens of sense-tagged instances for each ambiguous term when random sampling was used (Table 10). By applying our current active learning strategies, we observed a reduction of 30-40% in annotation labor, which is promising. However, it is still not clear if such a reduction is good enough for building supervised WSD systems with a broad coverage, because the ambiguity problem is pervasive in the biomedical domain. For example, Fundel and Zimmer [66] found that approximately 65% of 2.2 million human or rat related MEDLINE abstracts contained protein names that are ambiguous between the human and rat synonym lists. Liu and colleagues[67] also reported that 33.1% of clinical abbreviations found in the UMLS 2001 were ambiguous. In addition, annotation cost is also highly associated with the required performance of a task. If a WSD accuracy of 85% is good enough for a specific task, our active learning strategies would require only about a dozen of sense-tagged instances on this data set (see Table 10). Therefore, a formal study is needed to further assess the feasibility of developing real-world WSD systems based on active learning, which should evaluate annotation costs at different levels of required performance.

We implemented three different querying algorithms for multi-class WSD tasks: LC, Margin, and Entropy. Although they are all uncertainty sampling-based algorithms, they are different when computing the uncertainty based on probabilities generated by the

classifier: the LC algorithm considers the sense with the most confidence only; the Margin algorithm considers the two most likely senses; and the Entropy algorithm considers information for all possible senses. For the 14 words that had more than two senses in the dataset, we noticed a slight difference between LC and Margin/Entropy, but it was not statistically significant based on the statistical test, likely due to the small sample size (N=14). Another limitation of this study was that the pools for active learning were relatively small (maximum pool size was 500), as we used annotated samples in the MSH WSD dataset only. In a real-world application of active learning, we could collect a large number of unlabeled samples from MEDLINE for each ambiguous term, thus forming a much bigger pool for active learning experiments. We expect that larger pools will make the performance of active learning even better. We also noticed that most words in the MSH WSD data set had almost equally distributed senses and only 17 out of 197 words had highly skewed senses. During the creation of MSH WSD data set, some minor senses were removed according to the procedure. In practice, imbalanced sense distribution will be observed more often, which could make WSD tasks more challenging.

We analyzed the learning curves of the 20 words where active learners did not perform better than the passive learner. We categorized the patterns of these cases as follows. (1) Poor model in the early stage: there was a cutoff point where the learning curves of AL and PL crossed over in the early stage of learning. AL performed poorly in the early stage before the cutoff but could outperform PL in the later stage. This pattern happened in 11 out of 20 cases. The reason could be that uncertainty sampling algorithms are sensitive to the quality of models. When the model is poor, the learning curve could be very unstable. The "hasty generalization" problem pointed out by Wallace et al. [68]

could be one of the reasons for poor models in early stage. Samples selected based on early uncertainty models could be not representative enough, especially for cases with skewed class distribution. As suggested by Wallace et al, one solution could be applying diversity-based algorithms in the early stage. When the learning process passes the cutoff, active learning performs better than random because the classification model gets better. (2) Easy WSD cases: for some ambiguous words, high performance WSD models could be built based on only a small number of labeled samples. Basically they are easy WSD cases. For these cases, the informativeness or informative value of each sample is equally high and active learning is not necessary, as random sampling does the same job. We found 3 easy cases (*lymphogranulomatosis*, *PCD*, and *SLS*) out of 20 words. (3) Difficult WSD cases: this pattern was almost opposite to the second one. Even though we used all available samples with labels in the training set, the performance was not improved much. This indicates that the difference in informativeness or informative value among samples is small, and the informative value of each sample is equally low. We found three of these difficult cases (*Coffee*, *TMJ*, and *veterinary*). For the remaining three cases, the learning curves between AL and PL looked very similar. This could be due to the equal informativeness or informative value of each sample, or the querying algorithms failed to distinguish the difference of informativeness among unlabeled samples. These cases are also difficult cases because it is difficult to distinguish their samples.

Based on the above analysis, in order to further improve active learning for WSD tasks, we should investigate more robust active learning algorithms that can tolerate low quality models, or methods that can select good initial samples to build high quality models in the early stage. In addition to uncertainty sampling algorithms, other methods

that consider different types of information (e.g., sample diversity [44]) also need to be studied. We also plan to look into other available WSD datasets that have more multiple senses so that we can test active learning algorithms on multi-class classification problems. Moreover, we are interested in applying active learning to real-world WSD tasks by developing an annotation interface that implements active learning querying algorithms for sample selection.

CHAPTER IV

CONCLUSION

Both studies demonstrated that active learning algorithms can be applied to clinical text and biomedical literature classification tasks effectively. Overall, active learning algorithms improved the performance of supervised machine learning models while reduced annotation effort. Results for concept assertion classification task showed that when the same number of annotated samples was used, active learning strategies could generate better classification models (best ALC – 0.77) than the passive learning method (random sampling) (ALC – 0.74). Result for supervised WSD tasks showed that active learners significantly outperformed the passive learner for 177 out of 197 (89.8%) ambiguous terms.

Active learning strategies enable the learning machine to build the predictive models with required quality much faster than traditional machine learning or passive learning. For concept assertion classification task, to achieve an AUC of 0.79, the random sampling method used 32 samples, while our best active learning algorithm required only 12 samples, a reduction of 62.5% in manual annotation effort. For supervised WSD task, to achieve an average accuracy of 90%, the passive learner needed 38 samples, while the active learners needed only 24 annotated samples, a 37% reduction of annotation effort.

In addition, we analyzed cases where active learning algorithms did not achieve superior performance and analyzed three causes for supervised WSD tasks: (1) poor

model in early learning stage; (2) easy WSD cases; and (3) difficult WSD cases, which provide useful insight for future improvements. For concept assertion classification class, we developed a new "model change" querying method which could improve the cases caused by poor models in the early stage. It was, however, limited to binary classification tasks.

In the future, we will be exploring the application of active learning in clinical named entity recognition tasks, which is a sequencing labeling task that is different from the classification tasks discussed in this thesis. Uncertainty sampling algorithms could find the most informative samples, labeling these samples, however, may take much more effort than samples which are less informative. As the cost of labeling a sentence or a document may not be uniformly distributed, it is interesting to investigate cost-sensitive active learning which takes into account the real annotation cost for each sample instead of just the number of annotated samples. Moreover, uncertainty sampling would perform slowly in the sequencing labeling tasks because they rely on learning algorithm that is not fast when the size of labeled samples is large. Therefore, we are interested in discovering diversity based active learning algorithms, which do not depend on the classifier with annotation information but just the clinical text itself such as distribution of samples and semantic and syntactic information. Finally, we will develop a real-time active learning enabled annotation system which could assist to build high-performance supervised learning model as the ultimate solution for clinical text classification problems.

REFERENCES

1. Settles B. Active Learning Literature Survey. *Computer Sciences Technical Report 1648, University of Wisconsin-Madison.* 2009.

2. Tong S, Chang E. Support vector machine active learning for image retrieval. *Proceedings of the ACM International Conference on Multimedia.* 2001:107-118.

3. Liu Y. Active learning with support vector machine applied to gene expression data for cancer classification. *J Chem Inf Comp Sci.* Nov-Dec 2004;44(6):1936-1941.

4. Forman G. Incremental machine learning to reduce biochemistry lab costs in the search for drug discovery. Paper presented at: BIOKDD022002; Edmonton, Alberta, Canada.

5. Lewis DD, Gale WA. A sequential algorithm for training text classifiers. *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval.* 1994:3-12.

6. Tong S, Koller D. Support vector machine active learning with applications to text classification. *J Mach Learn Res.* Win 2002;2(1):45-66.

7. Settles B, Craven M. An analysis of active learning strategies for sequence labeling tasks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP).* 2008:1069-1078.

8. D. Lewis JC. Heterogeneous uncertainty sampling for supervised learning. Paper presented at: Proceedings of the Eleventh International Conference on Machine Learning1994.

9. H.S. Seung MO, and H. Sompolinsky. Query by committee. Paper presented at: Proceedings of the ACM Workshop on Computational Learning Theory1992.

10. B. Settles MC, and S. Ray. Multiple-instance active learning. *Advances in Neural Information Processing Systems (NIPS).* Vol volume 20: MIT Press; 2008:pages 1289–1296.

11. Chaloner K, Verdinelli I. Bayesian experimental design: A review. *Stat Sci.* Aug 1995;10(3):273-304.

12. McCallum NRaA. Toward optimal active learning through sampling estimation of error reduction. Paper presented at: Proceedings of the International Conference on Machine Learning (ICML)2001.

13. Scheffer T, Decomain C, Wrobel S. Active Hidden Markov Models for Information Extraction. *Proceedings of the 4th International Conference on Advances in Intelligent Data Analysis*: Springer-Verlag; 2001:309-318.

14. Shannon CE. A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.* 2001;5(1):3-55.

15. Dagan I, Engelson PS. Committee-based sampling for training probabilistic classifiers. Paper presented at: In Proceedings of the International Conference on Machine Learning (ICML)1995.

16. Nigam AMaK. Employing EM in pool-based active learning for text classification. Paper presented at: Proceedings of the International Conference on Machine Learning (ICML)1998.

17. Ringger E, McClanahan P, Haertel R, et al. Active learning for part-of-speech tagging: accelerating corpus annotation. *Proceedings of the Linguistic Annotation Workshop*. Prague, Czech Republic: Association for Computational Linguistics; 2007:101-108.

18. Becker M, Osborne M. A two-stage method for active learning of statistical grammars. *Proceedings of the 19th international joint conference on Artificial intelligence*. Edinburgh, Scotland: Morgan Kaufmann Publishers Inc.; 2005:991-996.

19. Chen J, Schein A, Ungar L, Palmer M. An Empirical Study of the Behavior of Active Learning for Word Sense Disambiguation. *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference.* 2006:120-127.

20. Kuo J-S, Li H, Yang Y-K. Learning transliteration lexicons from the web. *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Sydney, Australia: Association for Computational Linguistics; 2006:1129-1136.

21. Sassano M. An empirical study of active learning with support vector machines for Japanese word segmentation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Philadelphia, Pennsylvania: Association for Computational Linguistics; 2002:505-512.

22. Denny JC. Chapter 13: Mining electronic health records in the genomics era. *PLoS Comput Biol.* 2012;8(12):e1002823.

23. Prokosch HU, Ganslandt T. Perspectives for medical informatics. Reusing the electronic medical record for clinical research. *Methods Inf Med.* 2009;48(1):38-44.

24. Tannen RL, Weiner MG, Xie DW. Use of primary care electronic medical record database in drug efficacy research on cardiovascular outcomes: comparison of database and randomised controlled trial findings. *Brit Med J.* Jan 27 2009;338.

25. McCarty CA, Chisholm RL, Chute CG, et al. The eMERGE Network: a consortium of biorepositories linked to electronic medical records data for conducting genomic studies. *BMC Med Genomics.* 2011;4:13.

26. Merkel M, Andersson M. Combination of contextual features for word sense disambiguation: LIU-WSD. Paper presented at: SENSEVAL-2 Workshop2001.

27. Bruce RaJW. Word-sense disambiguation using decomposable models. Paper presented at: Proceedings of the 32nd annual meeting on Association for Computational Linguistics1994; Morristown, NJ, USA.

28. Liu H, Teller V, Friedman C. A multi-aspect comparison study of supervised word sense disambiguation. *J Am Med Inform Assoc.* Jul-Aug 2004;11(4):320-331.

29. Hripcsak G, Friedman C, Alderson PO, DuMouchel W, Johnson SB, Clayton PD. Unlocking clinical data from narrative reports: a study of natural language processing. *Ann Intern Med.* May 1 1995;122(9):681-688.

30. Friedman C, Alderson PO, Austin JH, Cimino JJ, Johnson SB. A general natural-language text processor for clinical radiology. *J Am Med Inform Assoc.* Mar-Apr 1994;1(2):161-174.

31. Hripcsak G, Austin JHM, Alderson PO, Friedman C. Use of natural language processing to translate clinical information from a database of 889,921 chest radiographic reports. *Radiology.* Jul 2002;224(1):157-163.

32. Savova GK, Masanz JJ, Ogren PV, et al. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *J Am Med Inform Assoc.* Sep-Oct 2010;17(5):507-513.

33. Xu H, Stenner SP, Doan S, Johnson KB, Waitman LR, Denny JC. MedEx: a medication information extraction system for clinical narratives. *J Am Med Inform Assoc.* Jan-Feb 2010;17(1):19-24.

34. Aronson AR, Lang FM. An overview of MetaMap: historical perspective and recent advances. *J Am Med Inform Assoc.* May-Jun 2010;17(3):229-236.

35. Denny JC, Smithers JD, Miller RA, Spickard A. "Understanding" medical school curriculum content using KnowledgeMap. *J Am Med Inform Assn.* Jul-Aug 2003;10(4):351-362.

36. Uzuner O, South BR, Shen S, DuVall SL. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *J Am Med Inform Assoc.* Sep-Oct 2011;18(5):552-556.

37. Zweigenbaum P, Demner-Fushman D, Yu H, Cohen KB. Frontiers of biomedical text mining: current progress. *Brief Bioinform.* Sep 2007;8(5):358-375.

38. Lu Z. PubMed and beyond: a survey of web tools for searching biomedical literature. *Database (Oxford).* 2011;2011:baq036.

39. Islamaj Dogan R, Yeganova L. Topics in machine learning for biomedical literature analysis and text retrieval. *J Biomed Semantics.* Oct 5 2012;3 Suppl 3:S1.

40. Kim S, Song Y, Kim K, Cha J, Lee G. MMR-based Active Machine Learning for Bio Named Entity Recognition. *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL.* June 2006 2006:pages 69 - 72.

41. Wallace BC, Small K, Brodley CE, Trikalinos TA. Active learning for biomedical citation screening. *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. Washington, DC, USA: ACM; 2010:173-182.

42. Miller TA, Dligach D, Savova GK. Active learning for coreference resolution. *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*. Montreal, Canada: Association for Computational Linguistics; 2012:73-81.

43. Figueroa RL, Zeng-Treitler Q, Ngo LH, Goryachev S, Wiechmann EP. Active learning for clinical text classification: is it better than random sampling? *J Am Med Inform Assoc.* Jun 15 2012.

44. Kim S, Song Y, Kim K, Cha J-w, Lee GG. MMR-based active machine learning for bio named entity recognition. *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL.* 2006:69-72.

45. Kim JD, Ohta T, Tateisi Y, Tsujii J. GENIA corpus--semantically annotated corpus for bio-textmining. *Bioinformatics.* 2003;19 Suppl 1:i180-182.

46. Chen Y, Mani S, Xu H. Applying active learning to assertion classification of concepts in clinical text. *J Biomed Inform.* Apr 2012;45(2):265-272.

47. Chen Y, Cao H, Mei Q, Zheng K, Xu H. Applying active learning to supervised word sense disambiguation in MEDLINE. *J Am Med Inform Assoc.* Jan 30 2013.

48. Y. Chen SM. Study of active learning in the challenge. Paper presented at: Proc. 2010 International Joint Conference on Neural Networks; 18-23 July 2010, 2010; Barcelona, Spain.

49. Jiang M, Chen Y, Liu M, et al. A study of machine-learning-based approaches to extract clinical entities and their assertions from discharge summaries. *J Am Med Inform Assoc.* Sep-Oct 2011;18(5):601-606.

50. Guyon I, Cawley G, Dror G, Lemaire V. Results of the Active Learning Challenge. *Journal of Machine Learning Research:Workshop and Conference Proceedings: Workshop on Active Learning and Experimental Design.* 2011;16:19-45.

51. Joachims T. *A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization*: Carnegie Mellon University;1996.

52. Fan RE, Chang KW, Hsieh CJ, Wang XR, Lin CJ. LIBLINEAR: A Library for Large Linear Classification. *J Mach Learn Res.* Aug 2008;9:1871-1874.

53. Mays DP. Bayesian application to the two-stage near-saturated experimental design method with dispersion effects. *J Stat Comput Sim.* May 2006;76(5):459-473.

54. Wilcoxon F. Individual comparisons of grouped data by ranking methods. *J Econ Entomol.* Apr 1946;39:269.

55. Hochberg Y, Tamhane AC. Multiple Comparison Procedures. *John Wiley & Sons, New York.* 1987.

56. Ide N, Veronis J. Introduction to the special issue on word sense disambiguation: The state of the art. *Comput Linguist.* Mar 1998;24(1):1-40.

57. Chen L, Liu H, Friedman C. Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics.* Jan 15 2005;21(2):248-256.

58. Schuemie MJ, Kors JA, Mons B. Word sense disambiguation in the biomedical domain: An overview. *J Comput Biol.* Jun 2005;12(5):554-565.

59. Lee YKaHTN. An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. Paper presented at: Proceedings of the ACL-02 conference on Empirical methods in natural language processing, Association for Computational Linguistics2002; Morristown, NJ, USA.

60. Mooney RJ. Comparative Experiments on Disambiguating Word Senses: An Illustration of the Role of Bias in Machine Learning. Paper presented at: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-96)1996; Philadelphia, PA.

61. Ng HTaHBL. Integrating multiple knowledge sources to disambiguate word sense: an exemplar-based approach. Paper presented at: Proceedings of the 34th annual meeting on Association for Computational Linguistics1996; Morristown, NJ, USA.

62. Yu H, Kim W, Hatzivassiloglou V, Wilbur WJ. Using MEDLINE as a knowledge source for disambiguating abbreviations and acronyms in full-text biomedical journal articles. *J Biomed Inform.* Apr 2007;40(2):150-159.

63. Liu H, Johnson SB, Friedman C. Automatic resolution of ambiguous terms based on machine learning and conceptual relations in the UMLS. *J Am Med Inform Assoc.* Nov-Dec 2002;9(6):621-636.

64. Jimeno-Yepes AJ, McInnes BT, Aronson AR. Exploiting MeSH indexing in MEDLINE to generate a data set for word sense disambiguation. *Bmc Bioinformatics.* 2011;12:223.

65. Li L, Chase HS, Patel CO, Friedman C, Weng C. Comparing ICD9-encoded diagnoses and NLP-processed discharge summaries for clinical trials pre-screening: a case study. *AMIA Annu Symp Proc.* 2008:404-408.

66. Fundel K, Zimmer R. Gene and protein nomenclature in public databases. *Bmc Bioinformatics.* 2006;7:372.

67. Liu H, Lussier YA, Friedman C. A study of abbreviations in the UMLS. *Proc AMIA Symp.* 2001:393-397.

68. Wallace BC, Trikalinos TA, Lau J, Brodley C, Schmid CH. Semi-automated screening of biomedical citations for systematic reviews. *Bmc Bioinformatics.* 2010;11:55.

Appendix

Table A1. Active learning results for 197 ambiguous words. (Note: Type "A" represents Abbreviation; type "T" represents Term; type "AT" represents Abbreviation and Term.)

ID	Words	Type	Number of Sentences						10-fold CV ACC	Average ALC score				ALVSPL
			Total	M1	M2	M3	M4	M5		LC	Margin	Entropy	Random (10 runs)	
1	AA	A	192	99	93	0	0	0	0.9948	0.9169	0.9169	0.9169	0.8705	1
2	ADA	A	198	99	99	0	0	0	0.9949	0.9179	0.9179	0.9179	0.8888	1
3	ADH	A	197	98	99	0	0	0	0.9898	0.9066	0.9066	0.9066	0.8741	1
4	ADP	A	139	89	50	0	0	0	0.9424	0.8926	0.8926	0.8926	0.8174	1
5	Adrenal	T	186	93	93	0	0	0	0.8710	0.6877	0.6877	0.6877	0.6704	1
6	Ala	A	293	98	97	98	0	0	0.9727	0.8252	0.8115	0.8244	0.7624	1
7	ALS	A	191	92	99	0	0	0	0.9686	0.9236	0.9236	0.9236	0.8865	1
8	ANA	A	193	97	96	0	0	0	0.9896	0.9642	0.9642	0.9642	0.9267	1
9	Arteriovenous Anastomoses	T	129	99	30	0	0	0	0.9380	0.8844	0.8844	0.8844	0.8628	1
10	Astragalus	T	195	96	99	0	0	0	0.9795	0.8806	0.8806	0.8806	0.8355	1
11	B-Cell Leukemia	AT	157	65	92	0	0	0	0.8153	0.7033	0.7033	0.7033	0.6908	1
12	BAT	T	192	98	94	0	0	0	0.9740	0.8435	0.8435	0.8435	0.8505	0
13	BLM	A	194	96	98	0	0	0	0.9948	0.9220	0.9220	0.9220	0.8794	1
14	Borrelia	T	196	98	98	0	0	0	0.7959	0.6861	0.6861	0.6861	0.6463	1
15	BPD	A	196	98	98	0	0	0	0.9949	0.9304	0.9304	0.9304	0.9018	1
16	BR	A	166	69	97	0	0	0	0.9699	0.8362	0.8362	0.8362	0.7863	1
17	Brucella abortus	T	177	98	79	0	0	0	0.9266	0.8027	0.8027	0.8027	0.7551	1
18	BSA	A	185	98	87	0	0	0	1.0000	0.9377	0.9377	0.9377	0.8801	1
19	BSE	A	197	98	99	0	0	0	1.0000	0.9546	0.9546	0.9546	0.9115	1
20	Ca	A	383	89	98	98	98	0	0.8564	0.6148	0.6201	0.6008	0.5800	1
21	CAD	A	194	98	96	0	0	0	0.9845	0.9159	0.9159	0.9159	0.8997	1
22	Callus	T	150	99	51	0	0	0	0.9333	0.8958	0.8958	0.8958	0.8301	1
23	CAM	A	195	97	98	0	0	0	0.9897	0.9071	0.9071	0.9071	0.8606	1
24	Cardiac pacemaker	T	198	99	99	0	0	0	0.9091	0.8333	0.8333	0.8333	0.7996	1
25	CCD	A	137	95	42	0	0	0	1.0000	0.9516	0.9516	0.9516	0.9045	1
26	CCl4	A	195	97	98	0	0	0	0.9897	0.9356	0.9356	0.9356	0.8983	1
27	CDA	A	190	99	91	0	0	0	1.0000	0.9470	0.9470	0.9470	0.8970	1
28	CDR	A	143	48	95	0	0	0	1.0000	0.9214	0.9214	0.9214	0.8614	1
29	Cell	AT	193	97	96	0	0	0	0.9637	0.8494	0.8494	0.8494	0.8028	1
30	Cement	T	174	86	88	0	0	0	0.9195	0.7479	0.7479	0.7479	0.7310	1
31	CH	A	148	91	57	0	0	0	0.9324	0.8150	0.8150	0.8150	0.7647	1
32	Cholera	T	196	98	98	0	0	0	0.9541	0.8228	0.8228	0.8228	0.7714	1
33	CI	A	183	99	84	0	0	0	0.9617	0.8608	0.8608	0.8608	0.8004	1

34	Cilia	T	151	96	55	0	0	0	0.9272	0.8514	0.8514	0.8514	0.8070	1
35	CIS	A	153	99	54	0	0	0	1.0000	0.9535	0.9535	0.9535	0.8849	1
36	CNS	A	195	99	96	0	0	0	0.9795	0.8720	0.8720	0.8720	0.8494	1
37	Coffee	T	193	96	97	0	0	0	0.7772	0.6476	0.6476	0.6476	0.6695	0
38	Cold	AT	251	93	96	62	0	0	0.8884	0.6861	0.6831	0.6833	0.6683	1
39	Compliance	T	196	99	97	0	0	0	0.9133	0.7603	0.7603	0.7603	0.7315	1
40	Cortex	T	152	99	53	0	0	0	0.9737	0.8756	0.8756	0.8756	0.8231	1
41	Cortical	T	292	95	99	98	0	0	0.9623	0.7477	0.7271	0.7330	0.6747	1
42	CP	A	294	97	99	98	0	0	1.0000	0.8681	0.8636	0.8655	0.8224	1
43	Crack	T	163	64	99	0	0	0	0.9755	0.8594	0.8594	0.8594	0.8301	1
44	CRF	A	196	97	99	0	0	0	1.0000	0.9317	0.9317	0.9317	0.8806	1
45	cRNA	A	197	99	98	0	0	0	0.9949	0.7354	0.7354	0.7354	0.8251	0
46	Crown	T	185	96	89	0	0	0	0.8703	0.7431	0.7431	0.7431	0.7142	1
47	CTX	A	179	95	84	0	0	0	1.0000	0.9350	0.9350	0.9350	0.8788	1
48	DAT	A	196	99	97	0	0	0	0.9949	0.9388	0.9388	0.9388	0.8863	1
49	DBA	A	179	96	83	0	0	0	1.0000	0.9469	0.9469	0.9483	0.9028	1
50	dC	A	198	99	99	0	0	0	0.9899	0.9077	0.9077	0.9077	0.8452	1
51	DDD	A	198	99	99	0	0	0	0.9495	0.8516	0.8516	0.8516	0.8133	1
52	DDS	A	217	99	98	20	0	0	0.9954	0.8270	0.8287	0.8281	0.7719	1
53	DE	A	124	27	97	0	0	0	0.8710	0.7582	0.7582	0.7582	0.7950	0
54	DI	A	194	96	98	0	0	0	0.9948	0.9606	0.9606	0.9606	0.9117	1
55	Digestive	T	195	98	97	0	0	0	0.8667	0.7440	0.7440	0.7440	0.7050	1
56	DON	A	126	99	27	0	0	0	0.9762	0.9335	0.9335	0.9335	0.8874	1
57	drinking	T	198	99	99	0	0	0	0.9646	0.8072	0.8072	0.8072	0.7800	1
58	eCG	A	198	99	99	0	0	0	0.9798	0.9021	0.9021	0.9021	0.8832	1
59	Eels	AT	126	98	28	0	0	0	0.9524	0.8972	0.8972	0.8972	0.8394	1
60	EGG	T	187	95	92	0	0	0	0.8877	0.7369	0.7369	0.7369	0.7216	1
61	EM	A	127	30	97	0	0	0	0.9843	0.9433	0.9433	0.9433	0.8793	1
62	EMS	A	197	99	98	0	0	0	0.9848	0.9049	0.9049	0.9049	0.8408	1
63	Epi	A	192	97	95	0	0	0	0.9896	0.8530	0.8530	0.8530	0.8265	1
64	ERP	A	193	99	94	0	0	0	1.0000	0.9665	0.9665	0.9665	0.9130	1
65	ERUPTION	T	193	98	95	0	0	0	0.9741	0.8843	0.8843	0.8843	0.8416	1
66	Erythrocytes	T	183	95	88	0	0	0	0.8033	0.6972	0.6972	0.6972	0.6668	1
67	Exercises	T	194	96	98	0	0	0	0.8711	0.6928	0.6928	0.6928	0.6878	1
68	FA	A	195	97	98	0	0	0	1.0000	0.9365	0.9365	0.9365	0.8848	1
69	Familial Adenomatous Polyposis	T	198	99	99	0	0	0	0.8333	0.7475	0.7475	0.7475	0.7347	1
70	FAS	A	197	99	98	0	0	0	1.0000	0.9636	0.9636	0.9636	0.9138	1
71	Fe	A	188	89	99	0	0	0	0.9153	0.8251	0.8251	0.8251	0.7988	1
72	Fish	AT	186	93	93	0	0	0	0.9624	0.8568	0.8568	0.8568	0.7874	1
73	Follicles	T	194	96	98	0	0	0	0.9897	0.8673	0.8673	0.8673	0.8282	1
74	FTC	A	198	99	99	0	0	0	1.0000	0.9672	0.9672	0.9672	0.9193	1

75	GAG	A	177	78	99	0	0	0	0.9887	0.8882	0.8882	0.8882	0.8347	1
76	Gamma-Interferon	T	194	98	96	0	0	0	0.8557	0.7136	0.7136	0.7136	0.6728	1
77	Ganglion	T	197	98	99	0	0	0	0.9188	0.8215	0.8215	0.8215	0.7842	1
78	Gas	T	193	98	95	0	0	0	0.9326	0.7944	0.7944	0.7944	0.7861	1
79	Glycoside	T	197	99	98	0	0	0	1.0000	0.8788	0.8788	0.8788	0.8084	1
80	Haemophilus ducreyi	T	153	54	99	0	0	0	0.9216	0.8641	0.8641	0.8641	0.8263	1
81	HCl	A	195	96	99	0	0	0	1.0000	0.9382	0.9382	0.9382	0.8810	1
82	Heregulin	T	173	99	74	0	0	0	0.8786	0.7194	0.7194	0.7194	0.7078	1
83	HGF	A	190	93	97	0	0	0	0.9368	0.8064	0.8064	0.8064	0.7835	1
84	HHV 8	A	171	76	95	0	0	0	0.8596	0.6698	0.6698	0.6698	0.7216	0
85	Hip	T	164	98	66	0	0	0	0.7805	0.6668	0.6668	0.6668	0.6851	0
86	HIV	A	194	96	98	0	0	0	0.8557	0.6801	0.6801	0.6801	0.6972	0
87	HPS	A	177	98	79	0	0	0	1.0000	0.9670	0.9670	0.9670	0.9314	1
88	HR	A	106	10	96	0	0	0	0.9528	0.9292	0.9292	0.9292	0.9069	1
89	Hybridization	T	191	97	94	0	0	0	0.9372	0.8362	0.8362	0.8362	0.7864	1
90	IA	A	134	99	35	0	0	0	0.9776	0.8764	0.8764	0.8764	0.8464	1
91	Ice	AT	233	98	37	98	0	0	0.9614	0.7553	0.7586	0.7622	0.7594	0
92	INDO	A	121	98	23	0	0	0	0.9835	0.9213	0.9213	0.9213	0.8717	1
93	Ion	T	196	97	99	0	0	0	0.9133	0.7996	0.7996	0.7996	0.7604	1
94	IP	A	196	97	99	0	0	0	1.0000	0.9490	0.9490	0.9490	0.8739	1
95	Iris	T	156	94	62	0	0	0	0.9231	0.7947	0.7947	0.7947	0.7660	1
96	ITP	A	186	99	87	0	0	0	0.9946	0.8843	0.8843	0.8843	0.8622	1
97	JP	A	192	99	93	0	0	0	0.9948	0.9263	0.9263	0.9263	0.8815	1
98	LABOR	T	195	97	98	0	0	0	0.8974	0.7714	0.7714	0.7714	0.7229	1
99	Lactation	T	167	83	84	0	0	0	0.9162	0.7482	0.7482	0.7482	0.7682	0
100	Language	T	197	98	99	0	0	0	0.9645	0.8202	0.8202	0.8202	0.7775	1
101	Laryngeal	T	197	98	99	0	0	0	0.8680	0.6899	0.6899	0.6899	0.6847	1
102	Lawsonia	T	115	99	16	0	0	0	0.9565	0.9420	0.9420	0.9420	0.8853	1
103	Leishmaniasis	T	161	99	62	0	0	0	0.9317	0.8152	0.8152	0.8152	0.7891	1
104	lens	T	295	97	99	99	0	0	0.8780	0.7160	0.6813	0.6889	0.6615	1
105	Lupus	T	289	99	99	91	0	0	0.8893	0.6711	0.6713	0.6711	0.6586	1
106	lymphogranulomatosis	T	119	99	20	0	0	0	0.9748	0.7946	0.7946	0.7946	0.8664	0
107	MAF	A	119	98	21	0	0	0	0.9832	0.9331	0.9331	0.9331	0.8833	1
108	Malaria	T	196	97	99	0	0	0	0.9388	0.7005	0.7005	0.7005	0.7468	0
109	MBP	A	140	96	44	0	0	0	0.9786	0.8966	0.8966	0.8966	0.8284	1
110	MCC	A	131	99	32	0	0	0	1.0000	0.9495	0.9495	0.9495	0.8821	1
111	Medullary	T	197	99	98	0	0	0	0.9695	0.8226	0.8226	0.8226	0.7707	1
112	MHC	A	193	96	97	0	0	0	0.9896	0.9172	0.9172	0.9172	0.8760	1
113	Milk	T	193	96	97	0	0	0	0.8964	0.7768	0.7768	0.7768	0.7414	1
114	Moles	T	171	72	99	0	0	0	0.9357	0.8152	0.8152	0.8152	0.7625	1
115	MRS	A	163	97	66	0	0	0	1.0000	0.9487	0.9487	0.9487	0.9029	1

116	Murine sarcoma virus	T	180	81	99	0	0	0	0.8000	0.6645	0.6645	0.6645	0.6412	1
117	NBS	A	145	98	47	0	0	0	1.0000	0.9550	0.9550	0.9550	0.8943	1
118	NEUROFIBROMATOSIS	T	197	99	98	0	0	0	0.8782	0.7861	0.7861	0.7861	0.7377	1
119	NM	A	122	38	84	0	0	0	0.9590	0.8943	0.8943	0.8943	0.8311	1
120	NPC	A	162	98	64	0	0	0	1.0000	0.9604	0.9604	0.9604	0.9201	1
121	Nurse	T	192	94	98	0	0	0	0.8542	0.6836	0.6836	0.6836	0.6782	1
122	Nursing	T	198	99	99	0	0	0	0.8990	0.7837	0.7837	0.7837	0.7441	1
123	OCD	A	198	99	99	0	0	0	0.9949	0.8997	0.8997	0.8997	0.8742	1
124	OH	A	198	99	99	0	0	0	0.9899	0.8952	0.8952	0.8952	0.8669	1
125	Orf	AT	197	99	98	0	0	0	0.9695	0.8391	0.8391	0.8391	0.8528	0
126	ORI	A	123	24	99	0	0	0	0.9919	0.9662	0.9662	0.9662	0.9345	1
127	PAF	A	115	99	16	0	0	0	1.0000	0.9830	0.9830	0.9830	0.9205	1
128	Parotitis	T	193	94	99	0	0	0	0.9016	0.7174	0.7174	0.7174	0.7145	1
129	PCA	A	490	99	99	99	95	98	0.9939	0.7964	0.8270	0.8077	0.7694	1
130	PCB	A	127	99	28	0	0	0	0.9921	0.9267	0.9267	0.9267	0.8790	1
131	PCD	A	197	99	98	0	0	0	1.0000	0.9160	0.9160	0.9160	0.9217	0
132	PCP	A	252	99	99	54	0	0	0.9881	0.8647	0.8618	0.8691	0.8137	1
133	PEP	A	198	99	99	0	0	0	0.9899	0.9376	0.9376	0.9376	0.8768	1
134	PHA	A	110	11	99	0	0	0	0.9909	0.9500	0.9500	0.9500	0.9210	1
135	Pharmaceutical	T	195	96	99	0	0	0	0.9282	0.8346	0.8346	0.8346	0.7883	1
136	Phosphorus	T	181	93	88	0	0	0	0.8508	0.7618	0.7618	0.7618	0.6988	1
137	Phosphorylase	T	166	99	67	0	0	0	0.8614	0.7400	0.7400	0.7400	0.7147	1
138	pI	A	156	99	57	0	0	0	0.9808	0.9067	0.9067	0.9067	0.8637	1
139	Plague	T	167	98	69	0	0	0	0.9162	0.7740	0.7740	0.7740	0.7490	1
140	Plaque	T	196	97	99	0	0	0	0.9898	0.9205	0.9205	0.9205	0.8881	1
141	Platelet	T	196	98	98	0	0	0	0.8316	0.7077	0.7077	0.7077	0.6910	1
142	Pleuropneumonia	T	197	99	98	0	0	0	0.9137	0.7897	0.7897	0.7897	0.7580	1
143	Pneumocystis	T	198	99	99	0	0	0	0.9091	0.7750	0.7750	0.7750	0.7558	1
144	POL	A	162	99	63	0	0	0	0.9877	0.9266	0.9266	0.9266	0.8863	1
145	Polymyalgia Rheumatica	T	198	99	99	0	0	0	0.9394	0.7504	0.7504	0.7504	0.7649	0
146	posterior pituitary	T	194	99	95	0	0	0	0.9021	0.8224	0.8224	0.8224	0.7790	1
147	Potassium	T	172	86	86	0	0	0	0.8953	0.7403	0.7403	0.7403	0.7222	1
148	PR	A	164	65	99	0	0	0	0.9939	0.8840	0.8840	0.8840	0.8319	1
149	Projection	T	194	99	95	0	0	0	0.9433	0.8108	0.8108	0.8108	0.7956	1
150	PVC	A	195	96	99	0	0	0	0.9949	0.9148	0.9148	0.9148	0.8581	1
151	RA	A	297	99	99	99	0	0	0.9933	0.8691	0.8724	0.8719	0.7953	1
152	Radiation	T	195	96	99	0	0	0	0.8513	0.7104	0.7104	0.7104	0.6990	1
153	RB	A	197	99	98	0	0	0	1.0000	0.9010	0.9010	0.9010	0.8471	1
154	RBC	A	195	99	96	0	0	0	0.8615	0.7329	0.7329	0.7329	0.7036	1
155	rDNA	A	198	99	99	0	0	0	0.9192	0.7626	0.7626	0.7626	0.7368	1
156	Respiration	T	196	98	98	0	0	0	0.9541	0.7965	0.7965	0.7965	0.7766	1

157	Retinal	T	193	94	99	0	0	0	0.9067	0.7835	0.7835	0.7835	0.7676	1
158	Root	T	194	99	95	0	0	0	0.9691	0.8461	0.8461	0.8461	0.7832	1
159	RSV	A	134	99	35	0	0	0	0.9776	0.9188	0.9188	0.9188	0.8301	1
160	SARS-associated coronavirus	A	118	47	71	0	0	0	0.9322	0.8236	0.8236	0.8236	0.7960	1
161	SARS	T	197	99	98	0	0	0	0.9492	0.8551	0.8551	0.8551	0.8362	1
162	SCD	A	196	99	97	0	0	0	0.9949	0.9307	0.9307	0.9307	0.8783	1
163	Schistosoma mansoni	T	198	99	99	0	0	0	0.8737	0.7588	0.7588	0.7588	0.7091	1
164	Semen	T	186	87	99	0	0	0	0.9247	0.7581	0.7581	0.7581	0.7430	1
165	sex factor	T	131	96	35	0	0	0	0.9313	0.8006	0.8006	0.8006	0.8030	0
166	SLS	A	163	65	98	0	0	0	1.0000	0.9604	0.9604	0.9604	0.9670	0
167	Sodium	T	194	96	98	0	0	0	0.8660	0.7233	0.7233	0.7233	0.6846	1
168	SPR	A	198	99	99	0	0	0	1.0000	0.9479	0.9479	0.9479	0.9043	1
169	SS	A	144	98	46	0	0	0	1.0000	0.9696	0.9696	0.9696	0.9253	1
170	Staph	T	187	95	92	0	0	0	0.8663	0.7479	0.7479	0.7479	0.7386	1
171	STEM	T	198	99	99	0	0	0	0.9596	0.8806	0.8806	0.8806	0.8320	1
172	Sterilization	T	196	98	98	0	0	0	0.9286	0.7473	0.7473	0.7473	0.7178	1
173	Strep	T	197	98	99	0	0	0	0.8274	0.7581	0.7581	0.7581	0.7276	1
174	Synapsis	T	134	35	99	0	0	0	0.9328	0.7369	0.7369	0.7369	0.7740	0
175	TAT	A	297	99	99	99	0	0	0.8822	0.7186	0.7135	0.7113	0.6722	1
176	Tax	AT	178	97	81	0	0	0	0.9663	0.9258	0.9258	0.9258	0.8572	1
177	TEM	A	196	99	97	0	0	0	0.9847	0.7848	0.7848	0.7848	0.8194	0
178	THYMUS	T	294	99	96	99	0	0	0.8980	0.7352	0.7338	0.7426	0.7112	1
179	TLC	A	197	98	99	0	0	0	1.0000	0.9460	0.9460	0.9460	0.9008	1
180	TMJ	A	197	98	99	0	0	0	0.7868	0.6537	0.6537	0.6537	0.6548	0
181	TMP	A	150	51	99	0	0	0	0.9867	0.8786	0.8786	0.8786	0.7953	1
182	TNC	A	164	68	96	0	0	0	1.0000	0.9236	0.9236	0.9236	0.8691	1
183	TNT	A	197	99	98	0	0	0	1.0000	0.9263	0.9263	0.9263	0.8625	1
184	Tolerance	T	198	99	99	0	0	0	0.9293	0.8010	0.8010	0.8010	0.7792	1
185	tomography	T	198	99	99	0	0	0	0.8838	0.7941	0.7941	0.7941	0.7393	1
186	Torula	T	122	34	88	0	0	0	0.8607	0.7228	0.7228	0.7228	0.7090	1
187	TPA	A	198	99	99	0	0	0	0.9848	0.9429	0.9429	0.9429	0.9018	1
188	TPO	A	198	99	99	0	0	0	0.9949	0.8575	0.8575	0.8575	0.8292	1
189	TRF	A	179	80	99	0	0	0	0.9944	0.9380	0.9380	0.9380	0.8975	1
190	TYR	A	190	92	98	0	0	0	0.9421	0.8370	0.8370	0.8370	0.7759	1
191	US	A	197	98	99	0	0	0	0.9594	0.8139	0.8139	0.8139	0.7726	1
192	Ventricles	T	197	99	98	0	0	0	0.9543	0.8137	0.8137	0.8137	0.7789	1
193	veterinary	T	181	99	82	0	0	0	0.8232	0.6311	0.6311	0.6311	0.6586	0
194	Wasp	AT	198	99	99	0	0	0	0.9747	0.8949	0.8949	0.8949	0.8332	1
195	WBS	A	128	35	93	0	0	0	0.9922	0.9316	0.9316	0.9316	0.8780	1
196	WT1	A	198	99	99	0	0	0	0.8485	0.7213	0.7213	0.7213	0.6779	1
197	Yellow Fever	T	181	98	83	0	0	0	0.9061	0.7488	0.7488	0.7488	0.7171	1