Learning a Stopping Criterion for Active Learning for Word Sense Disambiguation and Text Classification

Jingbo Zhu   Huizhen Wang
Natural Language Processing Lab
Northeastern University
Shenyang, Liaoning, P.R.China, 110004
Zhujingbo@mail.neu.edu.cn
wanghuizhen@mail.neu.edu.cn

Eduard Hovy
University of Southern California
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695
hovy@isi.edu

Abstract

In this paper, we address the problem of knowing when to stop the process of active learning. We propose a new statistical learning approach, called minimum expected error strategy, to defining a stopping criterion through estimation of the classifier's expected error on future unlabeled examples in the active learning process. In experiments on active learning for word sense disambiguation and text classification tasks, experimental results show that the new proposed stopping criterion can reduce approximately 50% human labeling costs in word sense disambiguation with degradation of 0.5% average accuracy, and approximately 90% costs in text classification with degradation of 2% average accuracy.

1 Introduction

Supervised learning models set their parameters using given labeled training data, and generally outperform unsupervised learning methods when trained on equal amount of training data. However, creating a large labeled training corpus is very expensive and time-consuming in some real-world cases such as word sense disambiguation (WSD).

Active learning is a promising way to minimize the amount of human labeling effort by building an system that automatically selects the most informative unlabeled example for human annotation at each annotation cycle. In recent years active learning has attracted a lot of research interest, and has been studied in many natural language processing (NLP) tasks, such as text classification (TC) (Lewis and Gale, 1994; McCallum and Nigam, 1998), chunking (Ngai and Yarowsky, 2000), named entity recognition (NER) (Shen *et al.*, 2004; Tomanek *et al.*, 2007), part-of-speech tagging (Engelson and Dagan, 1999), information extraction (Thompson et al., 1999), statistical parsing (Steedman et al., 2003), and word sense disambiguation (Zhu and Hovy, 2007).

Previous studies reported that active learning can help in reducing human labeling effort. With selective sampling techniques such as *uncertainty sampling* (Lewis and Gale, 1994) and *committee-based sampling* (McCallum and Nigam, 1998), the size of the training data can be significantly reduced for text classification (Lewis and Gale, 1994; McCallum and Nigam, 1998), word sense disambiguation (Chen, et al. 2006; Zhu and Hovy, 2007), and named entity recognition (Shen *et al.*, 2004; Tomanek *et al.*, 2007) tasks.

Interestingly, deciding when to stop active learning is an issue seldom mentioned issue in these studies. However, it is an important practical topic, since it obviously makes no sense to continue the active learning procedure until the whole corpus has been labeled. How to define an adequate stopping criterion remains an unsolved problem in active learning. In principle, this is a problem of estimation of classifier effectiveness (Lewis and Gale, 1994). However, in real-world applications, it is difficult to know when the classifier reaches its maximum effectiveness before all unlabeled examples have been annotated. And when the unlabeled data set becomes very large, full annotation is almost impossible for human annotator.

In this paper, we address the issue of a stopping criterion for active learning, and propose a new statistical learning approach, called *minimum ex-*

pected error strategy, that defines a stopping criterion through estimation of the classifier's expected error on future unlabeled examples. The intuition is that the classifier reaches maximum effectiveness when it results in the lowest expected error on remaining unlabeled examples. This proposed method is easy to implement, involves small additional computation costs, and can be applied to several different learners, such as Naive Bayes (NB), Maximum Entropy (ME), and Support Vector Machines (SVMs) models. Comparing with the confidence-based stopping criteria proposed by Zhu and Hovy (2007), experimental results show that the new proposed stopping criterion achieves better performance in active learning for both the WSD and TC tasks.

2 Active Learning Process and Problem of General Stopping Criterion

2.1 Active Learning Process

Active learning is a two-step semi-supervised learning process in which a small number of labeled samples and a large number of unlabeled examples are first collected in the initialization stage, and a close-loop stage of query and retraining is adopted. The purpose of active learning is to minimize the amount of human labeling effort by having the system in each cycle automatically select for human annotation the most informative unannotated case.

```
Procedure: Active Learning Process
Input: initial small training set L, and pool of unlabeled data set U
Use L to train the initial classifier C (i.e. a classifier for uncertainty sampling or a set of classifiers for committee-based sampling)
Repeat
    • Use the current classifier C to label all unlabeled examples in U
    • Based on active learning rules R such as uncertainty sampling or committee-based sampling, present m top-ranked unlabeled examples to oracle H for labeling
    • Augment L with the m new examples, and remove them from U
    • Use L to retrain the current classifier C
Until the predefined stopping criterion SC is met.
```

Figure 1. Active learning process

In this work, we are interested in selective sampling for pool-based active learning, and focus on *uncertainty sampling* (Lewis and Gale, 1994). The key point is how to measure the uncertainty of an unlabeled example, in order to select a new example with maximum uncertainty to augment the training data. The maximum uncertainty implies that the current classifier has the least confidence in its classification of this unlabeled example *x*. The well-known *entropy* is a good uncertainty measurement widely used in active learning:

$$UM(x) = -\sum_{y \in Y} P(y \mid x) \log P(y \mid x) \qquad (1)$$

where $P(y|x)$ is the *a posteriori* probability. We denote the output class $y \in Y = \{y_1, y_2, \ldots, y_k\}$. *UM* is the uncertainty measurement function based on the entropy estimation of the classifier's posterior distribution.

2.2 General Stopping Criteria

As shown in Fig. 1, the active learning process repeatedly provides the most informative unlabeled examples to an oracle for annotation, and update the training set, until the predefined stopping criterion *SC* is met. In practice, it is not clear how much annotation is sufficient for inducing a classifier with maximum effectiveness (Lewis and Gale, 1994). This procedure can be implemented by defining an appropriate stopping criterion for active learning.

In active learning process, a general stopping criterion *SC* can be defined as:

$$SC_{AL} = \begin{cases} 1 & effectiveness(C) \geq \theta \\ 0 & otherwise, \end{cases} \qquad (2)$$

where θ is a user predefined constant and the function *effectiveness(C)* evaluates the effectiveness of the current classifier. The learning process ends only if the stopping criterion function SC_{AL} is equal to 1. The value of constant θ represents a tradeoff between the cost of annotation and the effectiveness of the resulting classifier. A larger θ would cause more unlabeled examples to be selected for human annotation, and the resulting classifier would be more robust. A smaller θ means the resulting classifier would be less robust, and less unlabeled examples would be selected to annotate.

In previous work (Shen *et al.*, 2004; Chen *et al.*, 2006; Li and Sethi, 2006; Tomanek *et al.*, 2007), there are several common ways to define the function *effectiveness(C)*. First, previous work always used a simple stopping condition, namely, when the training set reaches desirable size. However, it is almost impossible to predefine an appropriate size of desirable training data guaranteed to induce the most effective classifier. Secondly, the learning loop can end if no uncertain unlabeled examples can be found in the pool. That is, all informative examples have been selected for annotation. However, this situation seldom occurs in real-world applications. Thirdly, the active learning process can stop if the targeted performance level is achieved. However, it is difficult to predefine an appropriate and achievable performance, since it should depend on the problem at hand and the users' requirements.

2.3 Problem of Performance Estimation

An appealing solution has the active learning process end when repeated cycles show no significant performance improvement on the test set. However, there are two open problems. The first question is how to measure the performance of a classifier in active learning. The second one is how to know when the resulting classifier reaches the highest or adequate performance. It seems feasible that a separate validation set can solve both problems. That is, the active learning process can end if there is no significant performance improvement on the validation set. But how many samples are required for the pregiven separate validation set is an open question. Too few samples may not be adequate for a reasonable estimation and may result in an incorrect result. Too many samples would cause additional high cost because the separate validation set is generally constructed manually in advance.

3 Statistical Learning Approach

3.1 Confidence-based Strategy

To avoid the problem of performance estimation mentioned above, Zhu and Hovy (2007) proposed a confidence-based framework to predict the upper bound and the lower bound for a stopping criterion in active learning. The motivation is to assume that the current training data is sufficient to train the classifier with maximum effectiveness if the current classifier already has acceptably strong confidence on its classification results for all remained unlabeled data.

The first method to estimate the confidence of the classifier is based on uncertainty measurement, considering whether the entropy of each selected unlabeled example is less than a small predefined threshold. Here we call it *Entropy-MCS*. The stopping criterion $SC_{Entropy\text{-}MCS}$ can be defined as:

$$SC_{Entropy-MCS} = \begin{cases} 1 & \forall x \in U, UM(x) \le \theta_E \\ 0 & otherwise, \end{cases} \tag{3}$$

where θ_E is a user predefined entropy threshold and the function *UM(x)* evaluates the uncertainty of each unlabeled example *x*.

The second method to estimate the confidence of the classifier is based on feedback from the oracle when the active learner asks for true labels for selected unlabeled examples, by considering whether the current trained classifier could correctly predict the labels or the accuracy performance of predictions on selected unlabeled examples is already larger than a predefined accuracy threshold. Here we call it *OracleAcc-MCS*. The stopping criterion $SC_{OracleAcc\text{-}MCS}$ can be defined as:

$$SC_{OracleAcc-MCS} = \begin{cases} 1 & OracleAcc(C) \ge \theta_A \\ 0 & otherwise, \end{cases} \tag{4}$$

where θ_A is a user predefined accuracy threshold and function *OracleAcc(C)* evaluates accuracy performance of the classifier on these selected unlabeled examples through feedback of the Oracle.

3.2 Minimum Expected Error Strategy

In fact, these above two confidence-based methods do not directly estimate classifier performance that closely reflects the classifier effectiveness, because they only consider entropy of each unlabeled example and accuracy on selected informative examples at each iteration step. In this section we therefore propose a new statistical learning approach to defining a stopping criterion through estimation of the classifier's expected error on all future unlabeled examples, which we call *minimum expected error strategy* (MES). The motivation behind MES is that the classifier *C* (a classifier for uncertainty sampling or set of classifiers for committee-based sampling) with maximum effectiveness is the one that results in the lowest expected error on whole test set in the learning process. The stopping criterion SC_{MES} is defined as:

$$SC_{MES} = \begin{cases} 1 & Error(C) \le \theta_{err} \\ 0 & otherwise, \end{cases} \quad (5)$$

where θ_{err} is a user predefined expected error threshold and the function *Error(C)* evaluates the expected error of the classifier *C* that closely reflects the classifier effectiveness. So the key point of defining MES-based stopping criterion SC_{MES} is how to calculate the function *Error(C)* that denotes the expected error of the classifier *C*.

Suppose given a training set *L* and an input sample *x*, we can write the expected error of the classifier *C* as follows:

$$Error(C) = \int R(C(x) \mid x) P(x) dx \quad (6)$$

where *P(x)* represents the known marginal distribution of *x*. *C(x)* represents the classifier's decision that is one of *k* classes: $y \in Y=\{y_1, y_2, ..., y_k\}$. $R(y_i|x)$ denotes a conditional loss for classifying the input sample *x* into a class y_i that can be defined as

$$R(y_i \mid x) = \sum_{j=1}^{k} \lambda[i,j] P(y_j \mid x) \quad (7)$$

where $P(y_j|x)$ is the *a posteriori* probability produced by the classifier *C*. $\lambda[i,j]$ represents a zero-one loss function for every class pair *{i,j}* that assigns no loss to a correct classification, and assigns a unit loss to any error.

In this paper, we focus on *pool-based active learning* in which a large unlabeled data pool *U* is available, as described Fig. 1. In active learning process, our interest is to estimate the classifier's expected error on future unlabeled examples in the pool *U*. That is, we can stop the active learning process when the active learner results in the lowest expected error over the unlabeled examples in *U*. The pool *U* can provide an estimate of *P(x)*. So for minimum error rate classification (Duda and Hart. 1973) on unlabeled examples, the expected error of the classifier *C* can be rewritten as

$$Error(C) = \frac{1}{|U|} \sum_{x \in U} (1 - \max_{y \in Y} P(y \mid x)) \quad (8)$$

Assuming *N* unlabeled examples in the pool *U*, the total time is *O(N)* for automatically determining whether the proposed stopping criterion SC_{MES} is satisfied in the active learning.

If the pool *U* is very large (e.g. more than 100000 examples), it would still cause high computation cost at each iteration of active learning. A good approximation is to estimate the expected error of the classifier using a subset of the pool, not using all unlabeled examples in *U*. In practice, a good estimation of expected error can be formed with few thousand examples.

4 Evaluation

In this section, we evaluate the effectiveness of three stopping criteria for active learning for word sense disambiguation and text classification as follows:

- *Entropy-MCS* — stopping active learning process when the stopping criterion function $SC_{Entropy\text{-}MCS}$ defined in (3) is equal to 1, where θ_E=*0.01, 0.001, 0.0001.*
- *OracleAcc-MCS* — stopping active learning process when the stopping criterion function $SC_{OracleAcc\text{-}MCS}$ defined in (4) is equal to 1, where θ_A=*0.9, 1.0.*
- *MES* — stopping active learning process when the stopping criterion function SC_{MES} defined in (5) is equal to 1, where θ_{err}=*0.01, 0.001, 0.0001.*

The purpose of defining stopping criterion of active learning is to study how much annotation is sufficient for a specific task. To comparatively analyze the effectiveness of each stopping criterion, a *baseline* stopping criterion is predefined as when all unlabeled examples in the pool *U* are learned. Comparing with the baseline stopping criterion, a better stopping criterion not only achieves almost the same performance, but also has needed to learn fewer unlabeled examples when the active learning process is ended. In other words, for a stopping criterion of active learning, the fewer unlabeled examples that have been leaned when it is met, the bigger reduction in human labeling cost is made.

In the following active learning experiments, a 10 by 10-fold cross-validation was performed. All results reported are the average of 10 trials in each active learning process.

4.1 Word Sense Disambiguation

The first comparison experiment is active learning for word sense disambiguation. We utilize a maximum entropy (ME) model (Berger *et al.*, 1996) to design the basic classifier used in active learning for WSD. The advantage of the ME model is the ability to freely incorporate features from diverse sources into a single, well-grounded statistical model. A publicly available ME toolkit (Zhang *et. al.*, 2004) was used in our experiments. In order to extract the linguistic features necessary for the ME model in WSD tasks, all sentences containing the target word are automatically part-of-speech (POS) tagged using the Brill POS tagger (Brill, 1992). Three knowledge sources are used to capture contextual information: unordered single words in topical context, POS of neighboring words with position information, and local collocations. These are same as the knowledge sources used in (Lee and Ng, 2002) for supervised automated WSD tasks.

The data used for comparison experiments was developed as part of the OntoNotes project (Hovy *et al.*, 2006), which uses the WSJ part of the Penn Treebank (Marcus *et al.*, 1993). The senses of noun words occurring in OntoNotes are linked to the Omega ontology (philpot *et al.*, 2005). In OntoNotes, at least two human annotators manually annotate the coarse-grained senses of selected nouns and verbs in their natural sentence context. In this experiment, we used several tens of thousands of annotated OntoNotes examples, covering in total 421 nouns with an inter-annotator agreement rate of at least 90%. We find that 302 out of 421 nouns occurring in OntoNotes are ambiguous, and thus are used in the following WSD experiments. For these 302 ambiguous nouns, there are 3.2 senses per noun, and 172 instances per noun.

The active learning algorithms start with a randomly chosen initial training set of 10 labeled samples for each noun, and make 10 queries after each learning iteration. Table 1 shows the effectiveness of each stopping criterion tested on active learning for WSD on these ambiguous nouns' WSD tasks. We analyze average accuracy performance of the classifier and average percentage of unlabeled examples learned when each stopping criterion is satisfied in active learning for WSD tasks. All accuracies and percentages reported in Table 1 are macro-averages over these 302 ambiguous nouns.

Stopping Criterion	Average accuracy	Average percentage
all unlabeled examples learned	87.3%	100%
Entropy-MCS method (0.0001)	86.8%	81.8%
Entropy-MCS method (0.001)	86.8%	75.8%
Entropy-MCS method (0.01)	86.8%	68.6%
OracleAcc-MCS method (0.9)	86.8%	56.5%
OracleAcc-MCS method (1.0)	86.8%	62.4%
MES method (0.0001)	86.8%	67.1%
MES method (0.001)	86.8%	58.8%
MES method (0.01)	86.8%	52.7%

Table 1. Effectiveness of each stopping criterion of active learning for WSD on OnteNotes.

Table 1 shows that these stopping criteria achieve the same accuracy of 86.8% which is within 0.5% of the accuracy of the baseline method (all unlabeled examples are labeled). It is obvious that these stopping criteria can help reduce the human labeling costs, comparing with the baseline method. The best criterion is MES method (θ_{err}=0.01), following by OracleAcc-MCS method (θ_A=0.9). MES method (θ_{err}=0.01) and OracleAcc-MCS method (θ_A=0.9) can make 47.3% and 44.5% reductions in labeling costs, respectively. Entropy-MCS method is apparently worse than MES and OracleAcc-MCS methods. The best of the Entropy-MCS method is the one with θ_E=0.01 which makes approximately 1/3 reduction in labeling costs. We also can see from Table 1 that for Entropy-MCS and MES methods, reduction rate becomes smaller as the θ becomes smaller.

4.2 Text Classification

The second data set is for active learning for text classification using the WebKB corpus [1] (McCallum *et al.*, 1998). The WebKB dataset was formed by web pages gathered from various university computer science departments. In the following active learning experiment, we use four most populous categories: *student, faculty, course* and *project*, altogether containing 4,199 web pages. Following previous studies (McCallum *et al.*, 1998), we only remove those words that occur merely once without using stemming or stop-list. The resulting vocabulary has 23,803 words. In the design of the text classifier, the maximum entropy model is also utilized, and no feature selection technique is used.

[1] See http://www.cs.cmu.edu/~textlearning

The algorithm is initially given 20 labeled examples, 5 from each class. Table 2 shows the effectiveness of each stopping criterion of active learning for text classification on WebKB corpus. All results reported are the average of 10 trials.

Stopping Criterion	Average accuracy	Average percentage
all unlabeled examples learned	93.5%	100%
Entropy-MCS method (0.0001)	92.5%	23.8%
Entropy-MCS method (0.001)	92.4%	22.3%
Entropy-MCS method (0.01)	92.5%	21.8%
OracleAcc-MCS method (0.9)	91.5%	13.1%
OracleAcc-MCS method (1.0)	92.5%	24.5%
MES method (0.0001)	92.1%	17.9%
MES method (0.001)	92.0%	15.6%
MES method (0.01)	91.5%	10.9%

Table 2. Effectiveness of each stopping criterion of active learning for TC on WebKB corpus.

From results shown in Table 2, we can see that MES method (θ_{err}=0.01) already achieves 91.5% accuracy in 10.9% unlabeled examples learned. The accuracy of all unlabeled examples learned is 93.5%. This situation means the approximately 90% remaining unlabeled examples only make only 2% performance improvement. Like the results of WSD shown in Table 1, for Entropy-MCS and MES methods used in active learning for text classification tasks, the corresponding reduction rate becomes smaller as the value of θ becomes smaller. MES method (θ_{err}=0.01) can make approximately 90% reduction in human labeling costs and results in 2% accuracy performance degradation. The Entropy-MCS method (θ_E=0.01) can make approximate 80% reduction in costs and results in 1% accuracy performance degradation. Unlike the results of WSD shown in Table 1, the OracleAcc-MCS method (θ_A=1.0) makes the smallest reduction rate of 75.5%. Actually in real-world applications, the selection of a stopping criterion is a tradeoff issue between labeling cost and effectiveness of the classifier.

5 Discussion

It is interesting to investigate the impact of performance change on defining a stopping criterion, so we show an example of active learning for WSD task in Fig. 2.

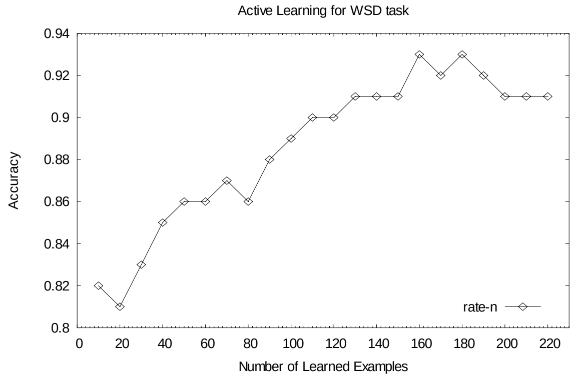

Figure 2. An example of active learning for WSD on noun "*rate*" in OntoNotes.

Fig. 2 shows that the accuracy performance generally increases, but apparently degrades at the iterations "*20*", "*80*", "*170*", "*190*", and "*200*", and does not change anymore during the iterations ["*130*"-"*150*"] or ["*200*"-"*220*"] in the active learning process. Actually the first time of the highest performance of 95% achieved is at "*450*", which is not shown in Fig. 2. In other words, although the accuracy performance curve shows an increasing trend, it is not monotonously increasing. From Fig. 2 we can see that it is not easy to automatically determine the point of no significant performance improvement on the validation set, because points such as "*20*" or "*80*" would mislead final judgment. However, we do believe that the change of performance is a good signal to stop active learning process. So it is worth studying further how to combine the factor of performance change with our proposed stopping criteria of active learning.

The OracleAcc-MCS method would not work if only one or too few informative examples are queried at the each iteration step in the active learning. There is an open issue how many selected unlabeled examples at each iteration are adequate for the batch-based sample selection.

For these stopping crieria, there is no general method to automatically determine the best threshold for any given task. It may therefore be necessary to use a dynamic threshold change technique in which the predefined threshold can be automatically modified if the performance is still significantly improving when the stopping criterion is met during active learning process.

6 Conclusion and Future Work

In this paper, we address the stopping criterion issue of active learning, and analyze the problems faced by some common ways to stop the active learning process. In essence, defining a stopping criterion of active learning is a problem of estimating classifier effectiveness. The purpose of defining stopping criterion of active learning is to know how much annotation is sufficient for a special task. To determine this, this paper proposes a new statistical learning approach, called minimum expected error strategy, for defining a stopping criterion through estimation of the classifier's expected error on future unlabeled examples during the active learning process. Experimental results on word sense disambiguation and text classification tasks show that new proposed minimum expected error strategy outperforms the confidence-based strategy, and achieves promising results. The interesting future work is to study how to combine the best of both strategies, and how to consider performance change to define an appropriate stopping criterion for active learning.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant (60473140), the National 863 High-tech Project (2006AA01Z154); the Program for New Century Excellent Talents in University(NCET-05-0287).

References

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

E Brill. 1992. *A simple rule-based part of speech tagger*. In the Proceedings of the Third Conference on Applied Natural Language Processing.

J. Chen, A. Schein, L. Ungar, M. Palmer. 2006. *An empirical study of the behavior of active learning for word sense disambiguation*. In Proc. of HLT-NAACL06

R. O. Duda and P. E. Hart. 1973. *Pattern classification and scene analysis*. New York: Wiley.

S. A. Engelson and I. Dagan. 1999. *Committee-based sample selection for probabilistic classifiers*. Journal of Artificial Intelligence Research.

E. Hovy, M. Marcus, M. Palmer, L. Ramshaw and R. Weischedel. 2006. *Ontonotes: The 90% Solution*. In Proc. of HLT-NAACL06.

Y.K. Lee and. H.T. Ng. 2002. *An empirical evaluation of knowledge sources and learning algorithm for word sense disambiguation*. In Proc. of EMNLP02

D. D. Lewis and W. A. Gale. 1994. *A sequential algorithm for training text classifiers*. In Proc. of SIGIR-94

M. Li, I. K. Sethi. 2006. *Confidence-based active learning*. IEEE transaction on pattern analysis and machine intelligence, 28(8):1251-1261.

M. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. *Building a large annotated corpus of English: the Penn Treebank*. Computational Linguistics, 19(2):313-330

A. McCallum and K. Nigram. 1998. *Employing EM in pool-based active learning for text classification*. In Proc. of 15th ICML

G. Ngai and D. Yarowsky. 2000. *Rule writing or annotation: cost-efficient resource usage for based noun phrase chunking*. In Proc. of ACL-02

A. Philpot, E. Hovy and P. Pantel. 2005. *The Omega Ontology*. In Proc. of ONTOLEX Workshop at IJCNLP.

D. Shen, J. Zhang, J. Su, G. Zhou and C. Tan. 2004. *Multi-criteria-based active learning for named entity recognition*. In Prof. of ACL-04.

M. Steedman, R. Hwa, S. Clark, M. Osborne, A. Sakar, J. Hockenmaier, P. Ruhlen, S. Baker and J. Crim. 2003. *Example selection for bootstrapping statistical parsers*. In Proc. of HLT-NAACL-03

C. A. Thompson, M. E. Califf and R. J. Mooney. 1999. *Active learning for natural language parsing and information extraction*. In Proc. of ICML-99.

K. Tomanek, J. Wermter and U. Hahn. 2007. *An approach to text corpus construction which cuts annotation costs and maintains reusability of annotated data*. In Proc. of EMNLP/CoNLL07

L. Zhang, J. Zhu, and T. Yao. 2004. *An evaluation of statistical spam filtering techniques*. ACM Transactions on Asian Language Information Processing, 3(4):243–269.

J. Zhu, E. Hovy. 2007. *Active learning for word sense disambiguation with methods for addressing the class imbalance problem*. In Proc. of EMNLP/CoNLL07