Multi-Criteria-based Active Learning for Named Entity Recognition

Dan Shen[†‡1] **Jie Zhang**[†‡] **Jian Su**[†] **Guodong Zhou**[†] **Chew-Lim Tan**[‡]

[†] Institute for Infocomm Technology
21 Heng Mui Keng Terrace
Singapore 119613

[‡] Department of Computer Science
National University of Singapore
3 Science Drive 2, Singapore 117543

{shendan,zhangjie,sujian,zhougd}@i2r.a-star.edu.sg
{shendan,zhangjie,tancl}@comp.nus.edu.sg

Abstract

In this paper, we propose a multi-criteria-based active learning approach and effectively apply it to named entity recognition. Active learning targets to minimize the human annotation efforts by selecting examples for labeling. To maximize the contribution of the selected examples, we consider the multiple criteria: *informativeness*, *representativeness* and *diversity* and propose measures to quantify them. More comprehensively, we incorporate all the criteria using two selection strategies, both of which result in less labeling cost than single-criterion-based method. The results of the named entity recognition in both MUC-6 and GENIA show that the labeling cost can be reduced by at least 80% without degrading the performance.

1 Introduction

In the machine learning approaches of natural language processing (NLP), models are generally trained on large annotated corpus. However, annotating such corpus is expensive and time-consuming, which makes it difficult to adapt an existing model to a new domain. In order to overcome this difficulty, active learning (sample selection) has been studied in more and more NLP applications such as POS tagging (Engelson and Dagan 1999), information extraction (Thompson et al. 1999), text classification (Lewis and Catlett 1994; McCallum and Nigam 1998; Schohn and Cohn 2000; Tong and Koller 2000; Brinker 2003), statistical parsing (Thompson et al. 1999; Tang et al. 2002; Steedman et al. 2003), noun phrase chunking (Ngai and Yarowsky 2000), etc.

Active learning is based on the assumption that a small number of annotated examples and a large number of unannotated examples are available. This assumption is valid in most NLP tasks. Different from supervised learning in which the entire corpus are labeled manually, active learning is to select the most useful example for labeling and add the labeled example to training set to retrain model. This procedure is repeated until the model achieves a certain level of performance. Practically, a batch of examples are selected at a time, called batched-based sample selection (Lewis and Catlett 1994) since it is time consuming to retrain the model if only one new example is added to the training set. Many existing work in the area focus on two approaches: certainty-based methods (Thompson et al. 1999; Tang et al. 2002; Schohn and Cohn 2000; Tong and Koller 2000; Brinker 2003) and committee-based methods (McCallum and Nigam 1998; Engelson and Dagan 1999; Ngai and Yarowsky 2000) to select the most informative examples for which the current model are most uncertain.

这里是batch化

指出一个一个选的缺点

Being the first piece of work on active learning for name entity recognition (NER) task, we target to minimize the human annotation efforts yet still reaching the same level of performance as a supervised learning approach. For this purpose, we make a more comprehensive consideration on the contribution of individual examples, and more importantly maximizing the contribution of a batch based on three criteria: *informativeness*, *representativeness* and *diversity*.

First, we propose three scoring functions to quantify the informativeness of an example, which can be used to select the most uncertain examples. Second, the representativeness measure is further proposed to choose the examples representing the majority. Third, we propose two diversity considerations (global and local) to avoid repetition among the examples of a batch. Finally, two combination strategies with the above three criteria are proposed to reach the maximum effectiveness on active learning for NER.

代表性避免了游离点

多样性避免一个batch内有太多重复类别（即样本分布不均）

[1] Current address of the first author: Universität des Saarlandes, Computational Linguistics Dept., 66041 Saarbrücken, Germany
dshen@coli.uni-sb.de

We build our NER model using Support Vector Machines (SVM). The experiment shows that our active learning methods achieve a promising result in this NER task. The results in both MUC-6 and GENIA show that the amount of the labeled training data can be reduced by at least 80% without degrading the quality of the named entity recognizer. The contributions not only come from the above measures, but also the two sample selection strategies which effectively incorporate informativeness, representativeness and diversity criteria. To our knowledge, it is the first work on considering the three criteria all together for active learning. Furthermore, such measures and strategies can be easily adapted to other active learning tasks as well.

2 Multi-criteria for NER Active Learning

Support Vector Machines (SVM) is a powerful machine learning method, which has been applied successfully in NER tasks, such as (Kazama et al. 2002; Lee et al. 2003). In this paper, we apply active learning methods to a simple and effective SVM model to recognize one class of names at a time, such as protein names, person names, etc. In NER, SVM is to classify a word into positive class "1" indicating that the word is a part of an entity, or negative class "-1" indicating that the word is not a part of an entity. Each word in SVM is represented as a high-dimensional feature vector including surface word information, orthographic features, POS feature and semantic trigger features (Shen et al. 2003). The semantic trigger features consist of some special head nouns for an entity class which is supplied by users. Furthermore, a window (size = 7), which represents the local context of the target word *w*, is also used to classify *w*.

局部标注的缺点

However, for active learning in NER, it is not reasonable to select a single word without context for human to label. Even if we require human to label a single word, he has to make an addition effort to refer to the context of the word. In our active learning process, we select a word sequence which consists of a machine-annotated named entity and its context rather than a single word. Therefore, all of the measures we propose for active learning should be applied to the machine-annotated named entities and we have to further study how to extend the measures for words to named entities. Thus, the active learning in SVM-based NER will be more complex than that in simple classification tasks, such as text classification on which most SVM active learning works are conducted (Schohn and Cohn 2000; Tong and Koller 2000; Brinker 2003). In the next part, we will introduce informativeness, representativeness and diversity measures for the SVM-based NER.

2.1 Informativeness

The basic idea of informativeness criterion is similar to certainty-based sample selection methods, which have been used in many previous works. In our task, we use a distance-based measure to evaluate the informativeness of a word and extend it to the measure of an entity using three scoring functions. We prefer the examples with high informative degree for which the current model are most uncertain.

2.1.1 Informativeness Measure for Word

In the simplest linear form, training SVM is to find a hyperplane that can separate the positive and negative examples in training set with maximum margin. The margin is defined by the distance of the hyperplane to the nearest of the positive and negative examples. The training examples which are closest to the hyperplane are called support vectors. In SVM, only the support vectors are useful for the classification, which is different from statistical models. SVM training is to get these support vectors and their weights from training set by solving quadratic programming problem. The support vectors can later be used to classify the test data.

SVM就是要找支持向量和它们的权重，只有支持向量在分类中有用，其它点没有用？

Intuitively, we consider the informativeness of an example as how it can make effect on the support vectors by adding it to training set. An example may be informative for the learner if the distance of its feature vector to the hyperplane is less than that of the support vectors to the hyperplane (equal to 1). This intuition is also justified by (Schohn and Cohn 2000; Tong and Koller 2000) based on a version space analysis. They state that labeling an example that lies on or close to the hyperplane is guaranteed to have an effect on the solution. In our task, we use the distance to measure the informativeness of an example.

The distance of a word's feature vector to the hyperplane is computed as follows:

$$Dist(\boldsymbol{w}) = \left| \sum_{i=1}^{N} a_i y_i k(\boldsymbol{s}_i, \boldsymbol{w}) + b \right|$$

where $\boldsymbol{w}$ is the feature vector of the word, a_i, y_i, $\boldsymbol{s}_i$ corresponds to the weight, the class and the feature vector of the i^{th} support vector respectively. N is the number of the support vectors in current model.

即1或-1

We select the example with minimal *Dist*, which indicates that it comes closest to the hyperplane in feature space. This example is considered most informative for current model.

2.1.2 Informativeness Measure for Named Entity

Based on the above informativeness measure for a word, we compute the overall informativeness degree of a named entity *NE*. In this paper, we propose three scoring functions as follows. Let *NE* = $w_1 \ldots w_N$ in which w_i is the feature vector of the i^{th} word of *NE*.

- **Info_Avg**: The informativeness of *NE* is scored by the average distance of the words in *NE* to the hyperplane.

  越大越不确定

  $$Info(NE) = 1 - \sum_{w_i \in NE} Dist(w_i)$$

  where, w_i is the feature vector of the i[th] word in *NE*.
- **Info_Min**: The informativeness of *NE* is scored by the minimal distance of the words in *NE*.

  $$Info(NE) = 1 - \underset{w_i \in NE}{Min}\{Dist(w_i)\}$$
- **Info_S/N**: If the distance of a word to the hyperplane is less than a threshold a (= 1 in our task), the word is considered with short distance. Then, we compute the proportion of the number of words with short distance to the total number of words in the named entity and use this proportion to quantify the informativeness of the named entity.

  $$Info(NE) = \frac{NUM_{w_i \in NE}(Dist(w_i) < a)}{N}$$

In Section 4.3, we will evaluate the effectiveness of these scoring functions.

2.2 Representativeness

In addition to the most informative example, we also prefer the most representative example. The representativeness of an example can be evaluated based on how many examples there are similar or near to it. So, the examples with high representative degree are less likely to be an outlier. Adding them to the training set will have effect on a large number of unlabeled examples. There are only a few works considering this selection criterion (McCallum and Nigam 1998; Tang et al. 2002) and both of them are specific to their tasks, viz. text classification and statistical parsing. In this section, we compute the similarity between words using a general vector-based measure, extend this measure to named entity level using dynamic time warping algorithm and quantify the representativeness of a named entity by its density.

确定可以影响一大批无标签样本吗？

2.2.1 Similarity Measure between Words

In general vector space model, the similarity between two vectors may be measured by computing the cosine value of the angle between them. The smaller the angle is, the more similar between the vectors are. This measure, called cosine-similarity measure, has been widely used in information retrieval tasks (Baeza-Yates and Ribeiro-Neto 1999). In our task, we also use it to quantify the similarity between two words. Particularly, the calculation in SVM need be projected to a higher dimensional space by using a certain kernel function $K(w_i, w_j)$. Therefore, we adapt the cosine-similarity measure to SVM as follows:

越大越相似

$$Sim(w_i, w_j) = \frac{|k(w_i, w_j)|}{\sqrt{k(w_i, w_i)k(w_j, w_j)}}$$

where, w_i and w_j are the feature vectors of the words i and j. This calculation is also supported by (Brinker 2003)'s work. Furthermore, if we use the linear kernel $k(w_i, w_j) = w_i \cdot w_j$, the measure is the same as the traditional cosine similarity measure $\cos q = \frac{w_i \cdot w_j}{\|w_i\| \cdot \|w_j\|}$ and may be regarded as a general vector-based similarity measure.

2.2.2 Similarity Measure between Named Entities

In this part, we compute the similarity between two machine-annotated named entities given the similarities between words. Regarding an entity as a word sequence, this work is analogous to the alignment of two sequences. We employ the dynamic time warping (DTW) algorithm (Rabiner et al. 1978) to find an optimal alignment between the words in the sequences which maximize the accumulated similarity degree between the sequences. Here, we adapt it to our task. A sketch of the modified algorithm is as follows.

Let $NE_1 = w_{11}w_{12}\ldots w_{1n}\ldots w_{1N}$, $(n = 1,\ldots, N)$ and $NE_2 = w_{21}w_{22}\ldots w_{2m}\ldots w_{2M}$, $(m = 1,\ldots, M)$ denote two word sequences to be matched. NE_1 and NE_2 consist of M and N words respectively. $NE_1(n) = w_{1n}$ and $NE_2(m) = w_{2m}$. A similarity value $Sim(w_{1n}, w_{2m})$ has been known for every pair of words (w_{1n}, w_{2m}) within NE_1 and NE_2. The goal of DTW is to find a path, $m = map(n)$, which map n onto the corresponding m such that the accumulated similarity Sim^* along the path is maximized.

只要知道是用DP算出来的就行了，细节不清楚没关系。

$$Sim^* = \underset{\{map(n)\}}{Max}\{\sum_{n=1}^{N} Sim(NE_1(n), NE_2(map(n)))\}$$

A dynamic programming method is used to determine the optimum path $map(n)$. The accumulated similarity Sim_A to any grid point (n, m) can be recursively calculated as

$$Sim_A(n, m) = Sim(w_{1n}, w_{2m}) + \underset{q \le m}{Max} Sim_A(n-1, q)$$

Finally, $Sim^* = Sim_A(N, M)$

Certainly, the overall similarity measure Sim^* has to be normalized as longer sequences normally give higher similarity value. So, the similarity between two sequences NE_1 and NE_2 is calculated as

$$Sim(NE_1, NE_2) = \frac{Sim*}{Max(N, M)}$$

2.2.3 Representativeness Measure for Named Entity

Given a set of machine-annotated named entities $NESet = \{NE_1, \ldots, NE_N\}$, the representativeness of a named entity NE_i in $NESet$ is quantified by its density. The density of NE_i is defined as the average similarity between NE_i and all the other entities NE_j in $NESet$ as follows.

$$Density(NE_i) = \frac{\sum_{j \neq i} Sim(NE_i, NE_j)}{N - 1}$$

If NE_i has the largest density among all the entities in $NESet$, it can be regarded as the centroid of $NESet$ and also the most representative examples in $NESet$.

2.3 Diversity

多样性的本质

Diversity criterion is to maximize the training utility of a batch. We prefer the batch in which the examples have high variance to each other. For example, given the batch size 5, we try not to select five repetitious examples at a time. To our knowledge, there is only one work (Brinker 2003) exploring this criterion. In our task, we propose two methods: local and global, to make the examples diverse enough in a batch.

Tang的工作没有涉及吗？

2.3.1 Global Consideration

For a global consideration, we cluster all named entities in $NESet$ based on the similarity measure proposed in Section 2.2.2. The named entities in the same cluster may be considered similar to each other, so we will select the named entities from different clusters at one time. We employ a K-means clustering algorithm (Jelinek 1997), which is shown in Figure 1.

Given:
$NESet = \{NE_1, \ldots, NE_N\}$
Suppose:
The number of clusters is K
Initialization:
Randomly equally partition $\{NE_1, \ldots, NE_N\}$ into K initial clusters C_j ($j = 1, \ldots, K$).
Loop until the number of changes for the centroids of all clusters is less than a threshold
- Find the centroid of each cluster C_j ($j = 1, \ldots, K$).

$$NECent_j = \arg\max_{NE \in C_j} \left(\sum_{NE_i \in C_j} Sim(NE_i, NE) \right)$$

- Repartition $\{NE_1, \ldots, NE_N\}$ into K clusters. NE_i will be assigned to Cluster C_j if

$$Sim(NE_i, NECent_j) \geq Sim(NE_i, NECent_w), w \neq j$$

Figure 1: Global Consideration for Diversity: K-Means Clustering algorithm

In each round, we need to compute the pair-wise similarities within each cluster to get the centroid of the cluster. And then, we need to compute the similarities between each example and all centroids to repartition the examples. So, the algorithm is time-consuming. Based on the assumption that N examples are uniformly distributed between the K clusters, the time complexity of the algorithm is about $\mathbf{O}(N^2/K+NK)$ (Tang et al. 2002). In one of our experiments, the size of the $NESet$ (N) is around 17000 and K is equal to 50, so the time complexity is about $\mathbf{O}(10^6)$. For efficiency, we may filter the entities in $NESet$ before clustering them, which will be further discussed in Section 3.

2.3.2 Local Consideration

When selecting a machine-annotated named entity, we compare it with all previously selected named entities in the current batch. If the similarity between them is above a threshold ß, this example cannot be allowed to add into the batch. The order of selecting examples is based on some measure, such as informativeness measure, representativeness measure or their combination. This local selection method is shown in Figure 2. In this way, we avoid selecting too similar examples (similarity value ≥ ß) in a batch. The threshold ß may be the average similarity between the examples in $NESet$.

你确定没有说错吗？按你的复杂度，不应该是min-batch内的吗？
如果用了代表性就变成O(n^2)了。

Given:
$NESet = \{NE_1, \ldots, NE_N\}$
$BatchSet$ with the maximal size K.
Initialization:
$BatchSet$ = empty
Loop until $BatchSet$ is full O(n)
- Select NE_i based on some measure from $NESet$.
- RepeatFlag = false;
- **Loop** from j = 1 to CurrentSize($BatchSet$) O(k)
  If $Sim(NE_i, NE_j) \geq b$ **Then**
    RepeatFlag = true;
    Stop the Loop;
- **If** RepeatFlag == false **Then**
  add NE_i into $BatchSet$
- remove NE_i from $NESet$

只找最大不排序则复杂度为O(n)，排序后取最大为O(1)，但排序本身要O(nlgn)

Figure 2: Local Consideration for Diversity

This consideration only requires $\mathbf{O}(NK+K^2)$ computational time. In one of our experiments (N ~ 17000 and K = 50), the time complexity is about $\mathbf{O}(10^5)$. It is more efficient than clustering algorithm described in Section 2.3.1.

绝对算错了，你K^2哪来的？

3 Sample Selection strategies

In this section, we will study how to combine and strike a proper balance between these criteria, viz. informativeness, representativeness and diversity,

to reach the maximum effectiveness on NER active learning. We build two strategies to combine the measures proposed above. These strategies are based on the varying priorities of the criteria and the varying degrees to satisfy the criteria.

- **Strategy 1**: We first consider the informativeness criterion. We choose *m* examples with the most informativeness score from *NESet* to an intermediate set called *INTERSet*. By this preselecting, we make the selection process faster in the later steps since the size of *INTERSet* is much smaller than that of *NESet*. Then we cluster the examples in *INTERSet* and choose the centroid of each cluster into a batch called *BatchSet*. The centroid of a cluster is the most representative example in that cluster since it has the largest density. Furthermore, the examples in different clusters may be considered diverse to each other. By this means, we consider representativeness and diversity criteria at the same time. This strategy is shown in Figure 3. One limitation of this strategy is that clustering result may not reflect the distribution of whole sample space since we only cluster on *INTERSet* for efficiency. The other is that since the representativeness of an example is only evaluated on a cluster. If the cluster size is too small, the most representative example in this cluster may not be representative in the whole sample space.

结合了代表性和多样性，由于聚类的是最不确定的那些数据，也考虑到了不确定性，但是不确定有点弱。

缺点：（1）使用子集聚类的结果没有反映整个样本空间的分布，多样性都是局部的。（2）这应该是所有k-means都有的：代表性是在聚类内部算出来的，无法反映出样本在整个空间的代表性。聚类越小，这个缺点越放大。

Given:
NESet = {NE_1, …, NE_N}
BatchSet with the maximal size *K*.
INTERSet with the maximal size *M*
Steps:
- *BatchSet* = ∅
- *INTERSet* = ∅
- Select *M* entities with most *Info* score from *NESet* to *INTERSet*.
- Cluster the entities in *INTERSet* into *K* clusters
- Add the centroid entity of each cluster to *BatchSet*

Figure 3: Sample Selection Strategy 1

如何正则化？每种指标的取值范围要一致，且都应该是越大越...

计算样本密度的时间复杂度是O(n^2)，但这是一种全局密度

- **Strategy 2** (Figure 4) We combine the informativeness and representativeness criteria using the function $l\,Info(NE_i)+(1-l)Density(NE_i)$, in which the *Info* and *Density* value of NE_i are normalized first. The individual importance of each criterion in this function is adjusted by the trade-off parameter l ($0 \le l \le 1$) (set to 0.6 in our experiment). First, we select a candidate example NE_i with the maximum value of this function from *NESet*. Second, we consider diversity criterion using the local method in Section 3.3.2. We add the candidate example NE_i to a batch only if NE_i is different enough from any previously selected example in the batch. The threshold ß is set to the average pair-wise similarity of the entities in *NESet*.

这种方法的优缺点是什么呢？
优点：不确定性，代表性，多样性同时考虑到了，且都很充足。

那你的时间复杂度不是O(n^2)吗

Given:
NESet = {NE_1, …, NE_N}
BatchSet with the maximal size *K*.
Initialization:
BatchSet = ∅
Loop until *BatchSet* is full
- Select NE_i which have the maximum value for the combination function between Info score and Density socre from *NESet*.

$$NE_i = \arg\max_{NE_i \in NESet} (l\,Info(NE_i) + (1-l)\,Density(NE_i))$$

- RepeatFlag = false;
- **Loop** from j = 1 to CurrentSize(*BatchSet*)
  If $Sim(NE_i, NE_j) \ge b$ **Then**
    RepeatFlag = true;
    Stop the Loop;
- **If** RepeatFlag == false **Then**
  add NE_i into *BatchSet*
- remove NE_i from *NESet*

Figure 4: Sample Selection Strategy 2

4 Experimental Results and Analysis

4.1 Experiment Settings

In order to evaluate the effectiveness of our selection strategies, we apply them to recognize protein (PRT) names in biomedical domain using GENIA corpus V1.1 (Ohta et al. 2002) and person (PER), location (LOC), organization (ORG) names in newswire domain using MUC-6 corpus. First, we randomly split the whole corpus into three parts: an initial training set to build an initial model, a test set to evaluate the performance of the model and an unlabeled set to select examples. The size of each data set is shown in Table 1. Then, iteratively, we select a batch of examples following the selection strategies proposed, require human experts to label them and add them into the training set. The batch size *K* = 50 in GENIA and 10 in MUC-6. Each example is defined as a machine-recognized named entity and its context words (previous 3 words and next 3 words).

Domain	Class	Corpus	Initial Training Set	Test Set	Unlabeled Set
Biomedical	PRT	GENIA1.1	10 sent. (277 words)	900 sent. (26K words)	8004 sent. (223K words)
Newswire	PER	MUC-6	5 sent. (131 words)	602 sent. (14K words)	7809 sent. (157K words)
	LOC		5 sent. (130 words)		7809 sent. (157K words)
	ORG		5 sent. (113 words)		7809 sent. (157K words)

Table 1: Experiment settings for active learning using GENIA1.1(PRT) and MUC-6(PER,LOC,ORG)

可能吗？

The goal of our work is to minimize the human annotation effort to learn a named entity recognizer with the same performance level as supervised learning. The performance of our model is evaluated using "precision/recall/F-measure".

4.2 Overall Result in GENIA and MUC-6

In this section, we evaluate our selection strategies by comparing them with a random selection method, in which a batch of examples is randomly selected iteratively, on GENIA and MUC-6 corpus. Table 2 shows the amount of training data needed to achieve the performance of supervised learning using various selection methods, viz. *Random*, *Strategy1* and *Strategy2*. In GENIA, we find:

- The model achieves 63.3 F-measure using 223K words in the supervised learning.
- The best performer is *Strategy2* (31K words), requiring less than 40% of the training data that *Random* (83K words) does and 14% of the training data that the supervised learning does.
- *Strategy1* (40K words) performs slightly worse than *Strategy2*, requiring 9K more words. It is probably because *Strategy1* cannot avoid selecting outliers if a cluster is too small.
- *Random* (83K words) requires about 37% of the training data that the supervised learning does. It indicates that only the words in and around a named entity are useful for classification and the words far from the named entity may not be helpful.

达到同样的性能可能是因为同局部标注那样用单词作为指标，有些词对于任务是无用的。

Class	Supervised	Random	Strategy1	Strategy2
PRT	223K (F=63.3)	83K	40K	31K
PER	157K (F=90.4)	11.5K	4.2K	3.5K
LOC	157K (F=73.5)	13.6K	3.5K	2.1K
ORG	157K (F=86.0)	20.2K	9.5K	7.8K

Table 2: Overall Result in GENIA and MUC-6

Furthermore, when we apply our model to newswire domain (MUC-6) to recognize person, location and organization names, *Strategy1* and *Strategy2* show a more promising result by comparing with the supervised learning and *Random*, as shown in Table 2. On average, about 95% of the data can be reduced to achieve the same performance with the supervised learning in MUC-6. It is probably because NER in the newswire domain is much simpler than that in the biomedical domain (Shen et al. 2003) and named entities are less and distributed much sparser in the newswire texts than in the biomedical texts.

生物更像一个特定的领域。而新闻是通用文本。

新闻比生物更有效的原因是：
（1）NER在新闻中更简单
（2）NER在新闻中更少
（3）NER分布更稀疏

4.3 Effectiveness of Informativeness-based Selection Method

In this section, we investigate the effectiveness of informativeness criterion in NER task. Figure 5 shows a plot of training data size versus F-measure achieved by the informativeness-based measures in Section 3.1.2: *Info_Avg*, *Info_Min* and *Info_S/N* as well as *Random*. We make the comparisons in GENIA corpus. In Figure 5, the horizontal line is the performance level (63.3 F-measure) achieved by supervised learning (223K words). We find that the three informativeness-based measures perform similarly and each of them outperforms *Random*. Table 3 highlights the various data sizes to achieve the peak performance using these selection methods. We find that *Random* (83K words) on average requires over 1.5 times as much as data to achieve the same performance as the informativeness-based selection methods (52K words).

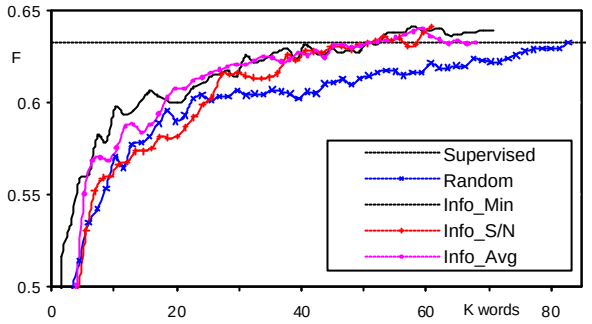

Figure 5: Active learning curves: effectiveness of the three informativeness-criterion-based selections comparing with the *Random* selection.

Supervised	Random	Info_Avg	Info_Min	Info_ S/N
223K	83K	52.0K	51.9K	52.3K

Table 3: Training data sizes for various selection methods to achieve the same performance level as the supervised learning

4.4 Effectiveness of Two Sample Selection Strategies

In addition to the informativeness criterion, we further incorporate representativeness and diversity criteria into active learning using two strategies described in Section 3. Comparing the two strategies with the best result of the single-criterion-based selection methods *Info_Min*, we are to justify that representativeness and diversity are also important factors for active learning. Figure 6 shows the learning curves for the various methods: *Strategy1*, *Strategy2* and *Info_Min*. In the beginning iterations (F-measure < 60), the three methods performed similarly. But with the larger training set, the efficiencies of *Strategy1* and *Strategy2* begin to be evident. Table 4 highlights the final result of the three methods. In order to reach the performance of supervised learning, *Strategy1* (40K words) and *Strategyy2* (31K words) require about 80% and 60% of the data that *Info_Min* (51.9K) does. So we believe the effective combinations of informativeness, representativeness and diversity will help to learn the NER model more quickly and cost less in annotation.

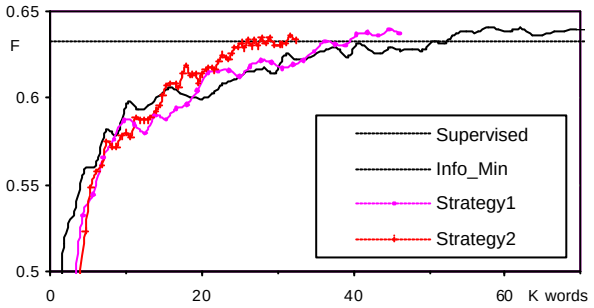

Figure 6: Active learning curves: effectiveness of the two multi-criteria-based selection strategies comparing with the informativeness-criterion-based selection (*Info_Min*).

Info_Min	Strategy1	Strategy2
51.9K	40K	31K

Table 4: Comparisons of training data sizes for the multi-criteria-based selection strategies and the informativeness-criterion-based selection (*Info_Min*) to achieve the same performance level as the supervised learning.

5 Related Work

Since there is no study on active learning for NER task previously, we only introduce general active learning methods here. Many existing active learning methods are to select the most uncertain examples using various measures (Thompson et al. 1999; Schohn and Cohn 2000; Tong and Koller 2000; Engelson and Dagan 1999; Ngai and Yarowsky 2000). Our informativeness-based measure is similar to these works. However these works just follow a single criterion. (McCallum and Nigam 1998; Tang et al. 2002) are the only two works considering the representativeness criterion in active learning. (Tang et al. 2002) use the density information to weight the selected examples while we use it to select examples. Moreover, the representativeness measure we use is relatively general and easy to adapt to other tasks, in which the example selected is a sequence of words, such as text chunking, POS tagging, etc. On the other hand, (Brinker 2003) first incorporate diversity in active learning for text classification. Their work is similar to our local consideration in Section 2.3.2. However, he didn't further explore how to avoid selecting outliers to a batch. So far, we haven't found any previous work integrating the informativeness, representativeness and diversity all together.

6 Conclusion and Future Work

In this paper, we study the active learning in a more complex NLP task, named entity recognition. We propose a multi-criteria-based approach to select examples based on their informativeness, representativeness and diversity, which are incorporated all together by two strategies (local and global). Experiments show that, in both MUC-6 and GENIA, both of the two strategies combining the three criteria outperform the single criterion (informativeness). The labeling cost can be significantly reduced by at least 80% comparing with the supervised learning. To our best knowledge, this is not only the first work to report the empirical results of active learning for NER, but also the first work to incorporate the three criteria all together for selecting examples.

Although the current experiment results are very promising, some parameters in our experiment, such as the batch size K and the l in the function of strategy 2, are decided by our experience in the domain. In practical application, the optimal value of these parameters should be decided automatically based on the training process. Furthermore, we will study how to overcome the limitation of the strategy 1 discussed in Section 3 by using more effective clustering algorithm. Another interesting work is to study when to stop active learning.

References

R. Baeza-Yates and B. Ribeiro-Neto. 1999. Modern Information Retrieval. *ISBN 0-201-39829-X.*

K. Brinker. 2003. Incorporating Diversity in Active Learning with Support Vector Machines. In *Proceedings of ICML*, 2003.

S. A. Engelson and I. Dagan. 1999. Committee-Based Sample Selection for Probabilistic Classifiers. *Journal of Artifical Intelligence Research*.

F. Jelinek. 1997. Statistical Methods for Speech Recognition. *MIT Press.*

J. Kazama, T. Makino, Y. Ohta and J. Tsujii. 2002. Tuning Support Vector Machines for Biomedical Named Entity Recognition. In *Proceedings of the ACL2002 Workshop on NLP in Biomedicine*.

K. J. Lee, Y. S. Hwang and H. C. Rim. 2003. Two-Phase Biomedical NE Recognition based on SVMs. In *Proceedings of the ACL2003 Workshop on NLP in Biomedicine.*

D. D. Lewis and J. Catlett. 1994. Heterogeneous Uncertainty Sampling for Supervised Learning. In *Proceedings of ICML*, 1994.

A. McCallum and K. Nigam. 1998. Employing EM in Pool-Based Active Learning for Text Classification. In *Proceedings of ICML*, 1998.

G. Ngai and D. Yarowsky. 2000. Rule Writing or Annotation: Cost-efficient Resource Usage for Base Noun Phrase Chunking. In *Proceedings of ACL*, 2000.

T. Ohta, Y. Tateisi, J. Kim, H. Mima and J. Tsujii. 2002. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of HLT 2002.*

L. R. Rabiner, A. E. Rosenberg and S. E. Levinson. 1978. Considerations in Dynamic Time Warping Algorithms for Discrete Word Recognition. In *Proceedings of IEEE Transactions on acoustics, speech and signal processing. Vol. ASSP-26, NO.6.*

D. Schohn and D. Cohn. 2000. Less is More: Active Learning with Support Vector Machines. In *Proceedings of the 17th International Conference on Machine Learning.*

D. Shen, J. Zhang, G. D. Zhou, J. Su and C. L. Tan. 2003. Effective Adaptation of a Hidden Markov Model-based Named Entity Recognizer for Biomedical Domain. In *Proceedings of the ACL2003 Workshop on NLP in Biomedicine*.

M. Steedman, R. Hwa, S. Clark, M. Osborne, A. Sarkar, J. Hockenmaier, P. Ruhlen, S. Baker and J. Crim. 2003. Example Selection for Bootstrapping Statistical Parsers. In *Proceedings of HLT-NAACL*, 2003.

M. Tang, X. Luo and S. Roukos. 2002. Active Learning for Statistical Natural Language Parsing. In *Proceedings of the ACL 2002.*

C. A. Thompson, M. E. Califf and R. J. Mooney. 1999. Active Learning for Natural Language Parsing and Information Extraction. In *Proceedings of ICML* 1999.

S. Tong and D. Koller. 2000. Support Vector Machine Active Learning with Applications to Text Classification. *Journal of Machine Learning Research*.

V. Vapnik. 1998. Statistical learning theory. *N.Y.:John Wiley.*