# Semi-Supervised Learning for Semantic Relation Classification using Stratified Sampling Strategy

**Longhua Qian Guodong Zhou Fang Kong Qiaoming Zhu**
Jiangsu Provincial Key Lab for Computer Information Processing Technology
School of Computer Science and Technology, Soochow University
1 Shizi Street, Suzhou, China 215006
{qianlonghua,gdzhou,kongfang,qmzhu}@suda.edu.cn

## Abstract

This paper presents a new approach to selecting the initial seed set using stratified sampling strategy in bootstrapping-based semi-supervised learning for semantic relation classification. First, the training data is partitioned into several strata according to relation types/subtypes, then relation instances are randomly sampled from each stratum to form the initial seed set. We also investigate different augmentation strategies in iteratively adding reliable instances to the labeled set, and find that the bootstrapping procedure may stop at a reasonable point to significantly decrease the training time without degrading too much in performance. Experiments on the ACE RDC 2003 and 2004 corpora show the stratified sampling strategy contributes more than the bootstrapping procedure itself. This suggests that a proper sampling strategy is critical in semi-supervised learning.

## 1 Introduction

With the dramatic increase in the amount of textual information available in digital archives and the WWW, there has been growing interest in techniques for automatically extracting information from text documents. Information Extraction (IE) is such a technology that IE systems are expected to identify relevant information (usually of pre-defined types) from text documents in a certain domain and put them in a structured format.

According to the scope of the NIST Automatic Content Extraction (ACE) program (ACE, 2000-2007), current research in IE has three main objectives: Entity Detection and Tracking (EDT), Relation Detection and Characterization (RDC), and Event Detection and Characterization (EDC). This paper focuses on the ACE RDC subtask, where many machine learning methods have been proposed, including supervised methods (Miller et al., 2000; Zelenko et al., 2002; Culotta and Soresen, 2004; Kambhatla, 2004; Zhou et al., 2005; Zhang et al., 2006; Qian et al., 2008), semi-supervised methods (Brin, 1998; Agichtein and Gravano, 2000; Zhang, 2004; Chen et al., 2006; Zhou et al., 2008), and unsupervised methods (Hasegawa et al., 2004; Zhang et al., 2005).

Current work on semantic relation extraction task mainly uses supervised learning methods, since it achieves relatively better performance. However this method requires a large amount of manually labeled relation instances, which is both time-consuming and laborious. In the contrast, unsupervised methods do not need definitions of relation types and hand-tagged data, but it is difficult to evaluate their performance since there are no criteria for evaluation. Therefore, semi-supervised learning has received more and more attention, as it can balance the advantages and disadvantages between supervised and unsupervised methods. With the plenitude of unlabeled natural language data at hand, semi-supervised learning can significantly reduce the need for labeled data with only limited sacrifice in performance. Specifically, a bootstrapping algorithm chooses the unlabeled instances with the highest probability of being correctly labeled and use them to augment labeled training data iteratively.

Although previous work (Yarowsky, 1995; Blum and Mitchell, 1998; Abney, 2000; Zhang, 2004) has tackled the bootstrapping approach from both the theoretical and practical point of view, many key problems still remain unresolved, such as the selection of initial seed set. Since the size of the initial seed set is usually small (e.g.



*Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1437–1445,
Singapore, 6-7 August 2009. 

100 instances), the imbalance of relation types or manifold structure (cluster structure) in it will severely weaken the strength of bootstrapping. Therefore, it is critical for a bootstrapping approach to select the most appropriate initial seed set. However, current systems (Zhang, 2004; Chen et al., 2006) use a randomly sampling strategy, which fails to explore the affinity nature among the training instances. Alternatively, Zhou et al. (2008) bootstrap a set of weighted support vectors from both labeled and unlabeled data using SVM. Nevertheless, the initial labeled data is still randomly generated only to ensure that there are at least 5 instances for every relation subtype.

This paper presents a new approach to selecting the initial seed set based on stratified sampling strategy in the bootstrapping procedure for semi-supervised semantic relation classification. The motivation behind the stratified sampling is that every relation type should be as much as possible represented in the initial seed set, thus leading to more instances with diverse structures being added to the labeled set. In addition, we also explore different strategies to augment reliably classified instances to the labeled data iteratively, and attempt to find a stoppage criterion for the iteration procedure to greatly decrease the training time, other than using up all the unlabeled set.

The rest of this paper is organized as follows. First, Section 2 reviews related work on semi-supervised relation extraction. Then we present an underlying supervised learner in Section 3. Section 4 details various key aspects of the bootstrapping procedure, including the stratified sampling strategy. Experimental results are reported in Section 5. Finally we conclude our work in Section 6.

## 2 Related Work

Within the realm of information extraction, currently there are several representative semi-supervised learning systems for extracting relations between named entities.

DIPRE (Dual Iterative Pattern Relation Expansion) (Brin, 1998) is a system based on bootstrapping that exploits the duality between patterns and relations to augment the target relation starting from a small sample. However, it only extracts simple relations such as (*author, title*) pairs from the WWW. Snowball (Agichtein and Gravano, 2000) is another bootstrapping-based system that extracts relations from unstructured text. Snowball shares much in common with DIPRE, including the use of both the bootstrapping framework and the pattern matching approach to extract new unlabeled instances. Due to pattern matching techniques, their systems are hard to be adapted to the general problem of relation extraction.

Zhang (2004) approaches the relation classification problem with bootstrapping on top of SVM. He uses various lexical and syntactic features in the *BootProject* algorithm based on random feature projection to extract top-level relation types in the ACE corpus. Evaluation shows that bootstrapping can alleviate the burden of hand annotations for supervised learning methods to a certain extent.

Chen et al. (2006) investigate a semi-supervised learning algorithm based on label propagation for relation extraction, where labeled and unlabeled examples and their distances are represented as the nodes and the weights of edges respectively in a connected graph, then the label information is propagated from any vertex to nearby vertices through weighted edges iteratively, finally the labels of unlabeled examples are inferred after the propagation process converges.

Zhou et al. (2008) integrate the advantages of SVM bootstrapping in learning critical instances and label propagation in capturing the manifold structure in both the labeled and unlabeled data, by first bootstrapping a moderate number of weighted support vectors through a co-training procedure from all the available data, and then applying label propagation algorithm via the bootstrapped support vectors.

However, in most current systems, the initial seed set is selected randomly such that they may not adequately represent the inherent structure of unseen examples, hence the power of bootstrapping may be severely weakened.

This paper presents a simple yet effective approach to generate the initial seed set by applying the stratified sampling strategy, originated from statistics theory. Furthermore, we try to employ the same stratified strategy to augment the labeled set. Finally, we attempt to find a reasonable criterion to terminate the iteration process.

## 3 Underlying Supervised Learning

A semi-supervised learning system usually consists of two relevant components: an underlying supervised learner and a

bootstrapping algorithm on top of it. In this section we discuss the former, while the latter will be described in the following section.

In this paper, we select Support Vector Machines (SVMs) as the underlying supervised classifier since it represents the state-of-the-art in the machine learning research community, and there are good implementations of the algorithm available. Specifically, we use LIBSVM (Chang et al., 2001), an effective tool for support vector classification, since it supports multi-class classification and provides probability estimation as well.

For each pair of entity mentions, we extract and compute various lexical and syntactic features, as employed in a state-of-the-art relation extraction system (Zhou et al., 2005).

(1) Words: According to their positions, four categories of words are considered: a) the words of both the mentions; b) the words between the two mentions; c) the words before M1; and d) the words after M2.

(2) Entity type: This category of features concerns about the entity types of both the mentions.

(3) Mention Level: This category of features considers the entity level of both the mentions.

(4) Overlap: This category of features includes the number of other mentions and words between two mentions. Typically, the overlap features are usually combined with other features such as entity type and mention level.

(5) Base phrase chunking: The base phrase chunking is proved to play an important role in semantic relation extraction. Most of the chunking features concern about the headwords of the phrases between the two mentions.

In this paper, we do not employ any deep syntactic or semantic features (such as dependency tree, full parse tree etc.), since they contribute quite limited in relation extraction.

# 4 Bootstrapping & Stratified Sampling

We first present the self-bootstrapping algorithm, and then discuss several key problems on bootstrapping in the order of initial seed selection, augmentation of labeled data and stoppage criterion for iteration.

## 4.1 Bootstrapping Algorithm

Following Zhang (2004), we define a basic self-bootstrapping strategy, which keeps augmenting the labeled data set with the models straightforwardly trained from previously available labeled data as follows:

**Algorithm** self-bootstrapping

**Require**: labeled seed set L
**Require**: unlabeled data set U
**Require**: batch size S
**Repeat**
  Train a single classifier on L
  Run the classifier on U
  Find at most S instances in U that the classifier has the highest prediction confidence
  Add them into L
**Until**: no data points available or the stoppage condition is reached

Figure 1. Self-bootstrapping algorithm

In order to measure the confidence of the classifier's prediction, we compute the entropy of the label probability distribution that the classifier assigns to the class label on an example (the lower the entropy, the higher the confidence):

$$H = -\sum_{i}^{n} p_i \log p_i \qquad (1)$$

Where $n$ denotes the total number of relation classes, and $p_i$ denotes the probability of current example being classified as the $i$th class.

## 4.2 Stratified Sampling for Initial Seeds

Normally, the number of available labeled instances is quite limited (usually less than 100 instances) when the iterative bootstrapping procedure begins. If the distribution of the initial seed set fails to approximate the distribution of the test data, the augmented data generated from bootstrapping would not capture the essence of relation types, and the performance on the test set will significantly decrease even only after one or two rounds of iterations. Therefore, the selection of initial seed set plays an important role in bootstrapping-based semantic relation extraction.

Sampling is a part of statistical practice concerned with the selection of individual observations, which is intended to yield some knowledge about a population of interest. When dealing with the task of semi-supervised semantic relation classification, the population is the training set of relation instances from the ACE RDC corpora. We compare two practical sampling strategies as follows:

(1) *Randomly sampling*, which picks the initial seeds from the training data using a random scheme. Each element thus has an equal probability of selection, and the population is not

subdivided or partitioned. Currently, most work on semi-supervised relation extraction employs this method. However, since the size of the initial seed set is very small, they are not guaranteed to capture the statistical properties of the whole training data, let alone of the test data.

(2) *Stratified sampling*. When the population embraces a number of distinct categories, *stratified sampling* (Neyman, 1934) can be applied to this case. First, the population can be organized by these categories into separate "strata", then a sample is selected within each "stratum" separately, and randomly. Generally, the sample size is normally proportional to the relative size of the strata. The main motivation for using a stratified sampling design is to ensure that particular groups within a population are adequately represented in the sample.

It is well known that the number of the instances for each relation type in the ACE RDC corpora is greatly unbalanced (Zhou et al., 2005) as shown in Table 1 for the ACE RDC 2004 corpus. When the relation instances for a specific relation type occurs frequently in the initial seed set, the classifier will achieve good performance on this type, otherwise the classifier can hardly recognize them from the test set. In order for every type of relations to be properly represented, the stratified sampling strategy is applied to the seed selection procedure.

| Types | Subtypes | Train | Test |
|---|---|---|---|
| PHYS | Located | 593 | 145 |
| | Near | 70 | 17 |
| | Part-Whole | 299 | 79 |
| PER-SOC | Business | 134 | 39 |
| | Family | 101 | 20 |
| | Other | 44 | 11 |
| EMP-ORG | Employ-Executive | 388 | 101 |
| | Employ-Staff | 427 | 112 |
| | Employ-Undetermined | 66 | 12 |
| | Member-of-Group | 152 | 39 |
| | Subsidiary | 169 | 37 |
| | Partner | 10 | 2 |
| | Other | 64 | 16 |
| ART | User-or-Owner | 160 | 40 |
| | Inventor-or-Man. | 8 | 1 |
| | Other | 1 | 1 |
| OTHER-AFF | Ethnic | 31 | 8 |
| | Ideology | 39 | 9 |
| | Other | 43 | 11 |
| GPE-AFF | Citizen-or-Resid. | 226 | 47 |
| | Based-In | 165 | 50 |
| | Other | 31 | 8 |
| DISC | | 224 | 55 |
| Total | | 3445 | 860 |

Table 1. Numbers of relations on the ACE RDC 2004: break down by relation types and subtypes

Figure 2 illustrates the stratified sampling strategy we use in bootstrapping, where *RSET* denotes the training set, *V* is the stratification variable, and *SeedSET* denotes the initial seed set. First, we divide the relation instances into different strata according to available properties, such as major relation type (considering reverse relations or not) and relation subtype (considering reverse relations or not). Then within every stratum, a certain number of instances are sampled randomly, and this number is normally proportional to the size of that stratum in the whole population. However, when this number is 0 due to the rounding of real numbers, it is set to 1. Also it must be ensured that the total number of instances being sampled is $N_S$. Finally, these instances form the initial seed set and can be used as the input to the underlying supervised learning for the bootstrapping procedure.

---

**Require**: $RSET = \{R_1, R_2, \ldots, R_N\}$
**Require**: $V = \{v_1, v_2, \ldots, v_K\}$
**Require**: *SeedSET* with the size of $N_S$ (100)
**Initialization**:
$SeedSET = NULL$
**Steps**:

- Group *RSET* into *K* strata according to the stratified variable *V*, i.e.:
  $RSET = \{RSET_1, RSET_2, \ldots, RSET_K\}$
- Calculate the class prior probability for each stratum i={1,2,…,*K*}
  $P_i = NUM(RSET_i) / NUM(RSET)$
- Caculate the number of intances being sampled for each stratum
  $N_i = P_i * N$
  If $N_i = 0$ then $N_i = 1$
- Calculate the difference of numbers as follows:
  $$N_\Delta = N_S - \sum_{i=1}^{K} N_i$$
- If $N_\triangle > 0$ then add $N_i$ ($i$=1,2,…,$|N_\triangle|$) by 1
  If $N_\triangle < 0$ then subtract 1 from $N_i$ ($i$=1,2,...,$|N_\triangle|$)
- For each *i* from 1 to *K*
  Select $N_i$ instances from $REST_i$ randomly
  Add them into *SeedSET*

---

Figure 2. Stratefied Sampling for initial seeds

### 4.3 Augmentation of labeled data

After each round of iteration, some newly classified instances with the highest confidence can be augmented to the labeled training data. Nevertheless, just like the selection of initial seed set, we still wish that every stratum would be represented as appropriately as possible in the

instances added to the labeled set. In this paper, we compare two kinds of augmentation strategies available:

(1) *Top n* method: the classified instances are first sorted in the ascending order by their entropies (i.e. decreasing confidence), and then the top *n* (usually 100) instances are chosen to be added.

(2) *Stratified* method: in order to make the added instances representative for their stratum, we first select *m* (usually greater than *n*) instances with the highest confidence, then we choose *n* instances from them using the stratified strategy.

### 4.4 Stoppage of Iterations

In a self-bootstrapping procedure, as the iterations go on, both the reliable and unreliable instances are added to the labeled data continuously, hence the performance will fluctuate in a relatively small range. The key question here is how we can know when the bootstrapping procedure reaches its best performance on the test data. The bootstrapping algorithm by Zhang (2004) stops after it runs out of all the training instances, which may take a relatively long time. In this paper, we present a method to determine the stoppage criterion based on the mean entropy as follows:

$$H_i <= p \qquad (2)$$

Where $H_i$ denotes the mean entropy of the confidently classified instances being augmented to the labeled data in each iteration, and $p$ denotes a threshold for the mean entropy, which will be fixed through empirical experiments. This criterion is based on the assumption that when the mean entropy becomes less than or equal to a certain threshold, the classifier would achieve the most reliable confidence on the instances being added to the labeled set, and it may be impossible to yield better performance since then. Therefore, the iteration may stop at that reasonable point.

## 5 Experimentation

This section aims to empirically investigate the effectiveness of the bootstrapping-based semi-supervised learning we discussed above for semantic relation classification. In particular, different methods for selecting the initial seed set and augmenting the labeled data are evaluated.

### 5.1 Experimental Setting

We use the ACE corpora as the benchmark data, which are gathered from various newspapers, newswire and broadcasts. The ACE 2004 corpus contains 451 documents and 5702 positive relation instances. It defines 7 relation types and 23 subtypes between 7 entity types. For easy reference with related work in the literature, evaluation is also done on 347 documents (including nwire and bnews domains) and 4305 relation instances using 5-fold cross-validation. That is, these relation instances are first divided into 5 sets, then, one of them (about 860 instances) is used as the test data set, while the others are regarded as the training data set, from which the initial seed set is sampled. In the ACE 2003 corpus, the training set consists of 674 documents and 9683 positive relation instances while the test data consists of 97 documents and 1386 positive relation instances. The ACE RDC 2003 task defines 5 relation types and 24 subtypes between 5 entity types.

The corpora are first parsed using Collins's parser (Collins, 2003) with the boundaries of all the entity mentions kept. Then, the parse trees are converted into chunklink format using chunklink.pl [1]. Finally, various useful lexical and syntactic features, as described in Subsection 3.1, are extracted and computed accordingly. For the purpose of comparison, we define our task as the classification of the 5 or 7 major relation types in the ACE RDC 2003 and 2004 corpora.

For LIBSVM parameters, we adopted the polynomial kernel, and *c* is set to 10, *g* is set to 0.15. Under this setting, we achieved the best classification performance.

### 5.2 Experimental Results

In this subsection, we compare and discuss the experimental results using various sampling strategies, different augmentation methods, and iteration stoppage criterion.

**Comparison of sampling strategies in selecting the initial seed set**

Table 2 and Table 3 show the initial and the highest classification performance of Precision/Recall/F-measure for various sampling strategies of the initial seed set on 7 major relation types of the ACE RDC 2004 corpus respectively when the size of initial seed set *L* is 100, the batch size *S* is 100, and the top 100

[1] http://ilk.kub.nl/~sabine/chunklink/

instances with the highest confidence are added at each iteration. Table 2 also lists the number of strata for stratified sampling methods from which the initial seeds are randomly chosen respectively. Table 3 additionally lists the time needed to complete the bootstrapping process (on a PC with a Pentium IV 3.0G CPU and 1G memory). In this paper, we consider the following five experimental settings when sampling the initial seeds:

- *Randomly* Sampling: as described in Subsection 4.2.
- *Stratified-M* Sampling: the strata are grouped in terms of major relation types without considering reverse relations.
- *Stratified-MR* Sampling: the strata are grouped in terms of major relation types, including reverse relations.
- *Stratified-S* Sampling: the strata are grouped in terms of relation subtypes without considering reverse subtypes.
- *Stratified-SR* Sampling: the strata are grouped in terms of relation subtypes, including reverse subtypes.

For each sampling strategies, we performed 20 trials and computed average scores and the total time on the test set over these 20 trials.

| Sampling strategies for initial seeds | # of strat. | P(%) | R(%) | F |
|---|---|---|---|---|
| *Randomly* | 1 | 66.1 | 65.9 | 65.9 |
| *Stratified-M* | 7 | 69.1 | 66.5 | 67.7 |
| *Stratified-MR* | 13 | 69.3 | 67.3 | 68.2 |
| *Stratified-S* | 30 | 69.8 | 67.7 | 68.7 |
| *Stratified-SR* | 39 | 69.9 | 68.5 | 69.2 |

Table 2. The initial performance of applying various sampling strategies to selecting the initial seed set on the ACE RDC 2004 corpus

| Sampling strategies for initial seeds | Time (min) | P(%) | R(%) | F |
|---|---|---|---|---|
| *Randomly* | 52 | 68.6 | 66.2 | 67.3 |
| *Stratified-M* | 65 | 71.0 | 66.9 | 68.8 |
| *Stratified-MR* | 65 | 71.6 | 67.0 | 69.2 |
| *Stratified-S* | 71 | 72.7 | 67.8 | 70.1 |
| *Stratified-SR* | 77 | 72.9 | 68.4 | 70.6 |

Table 3. The highest performance of applying various sampling strategies in selecting the initial seed set on the ACE RDC 2004 corpus

These two tables jointly indicate that the self-bootstrapping procedure for all sampling strategies can moderately improve the classification performance by ~1.2 units in F-score, which is also verified by Zhang (2004). Furthermore, they show that:

- The most improvements in performance come from improvements in precision. Actually, for some settings the recalls even decrease slightly. The reason may be that due to the nature of self-bootstrapping, the instances augmented at each iteration are always those which are the most similar to the initial seed instances, therefore the models trained from them would exhibit higher precision on the test set, while it virtually does no help for recall.
- All of the four stratified sampling methods outperform the randomly sampling method to various degrees, both in the initial performance and the highest performance. This means that sampling of the initial seed set based on stratification by major/sub relation types can be helpful to relation classification, largely due to the performance improvement of the initial seed set, which is caused by adequate representation of instances for every relation type.
- Of all the four stratified sampling methods, the Stratified-SR sampling achieves the best performance of 72.9/68.4/70.6 in P/R/F. Moreover, the more the number of strata generated by the sampling strategy, the more appropriately they would be represented in the initial seed set, and the better performance it will yield. This also implies that the hierarchy of relation types/subtypes in the ACE RDC 2004 corpus is fairly reasonably defined.
- An important conclusion, which can be draw accordingly, is that the F-score improvement of *Stratified-SR* sampling over Randomly sampling in initial performance (3.3 units) is significantly greater than the F-score improvement gained by bootstrapping itself using *Randomly* sampling (1.4 units). This means that the sampling strategy of the initial seed set is even more important than the bootstrapping algorithm itself for relation classification.
- It is interesting to note that the time needed to bootstrap increases with the number of strata. The reason may be that due to more diverse structures in the labeled data for stratified sampling, the SVM needs more time to differentiate between instances, i.e. more time to learn the models.

**Comparison of different augmentation strategies of training data**

Figure 3 compares the performance of F-score for two augmentation strategies: the *Top n* method and the *stratified* method, over various initial seed sampling strategies on the ACE RDC 2004 corpus. For each iteration, a variable

number (*m* is ranged from 100 to 500) of classified instances in the decreasing order of confidence are first chosen as the base examples, then at most 100 examples are selected from the base examples to be augmented to the labeled set. Specifically, when *m* is equal to 100, the whole set of the base example is added to the labeled data, i.e. degenerated to the *Top n* augmentation strategy. On the other hand, when *m* is greater than 100, we wish we would select examples of different major relation types from the base examples according to their distribution in the training set, in order to achieve the performance improvement as much as the stratified sampling does in the selection of the initial seed set.

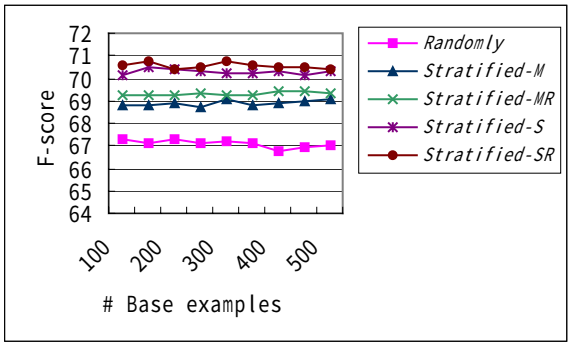


Figure 3. Comparison of two augmentation strategies over different sampling strategies in selecting the initial seed set.

This figure shows that, except for *randomly* sampling strategy, the stratified augmentation strategies improve the performance. Nevertheless, this result is far from our expectation in two ways:

- The performance improvement in F-score is trivial, at most 0.4 units on average. The reason may be that, although we try to add as many as 100 classified instances to the labeled data according to the distribution of every major relation type in the training set, the top *m* instances with the highest confidence are usually focused on certain relation types (e.g. PHSY and PER-SOC), this leads to the stratified augmentation failing to function effectively. Hence, all the following experiments will only adopt *Top n* method for augmenting the labeled data.
- With the increase of the number of the base examples, the performance fluctuates slightly, thus it is relatively difficult to recognize where the optima is. We think there are two contradictory factors that affect the performance. While the reliability of the instances extracted from the base examples decreases with the increase of the number of base examples, the probability of extracting instances of more relation types increases with the increase of the number of the base examples. These two factors inversely interact with each other, leading to the fluctuation in performance.

**Comparison of different threshold values for stoppage criterion**

We compare the performance and bootstrapping time (20 trials with the same initial seed set) when applying stoppage criterion in Formula (2) with different threshold *p* over various sampling strategies on the ACE RDC 2004 corpus in Figure 4 and Figure 5 respectively. These two figures jointly show that:

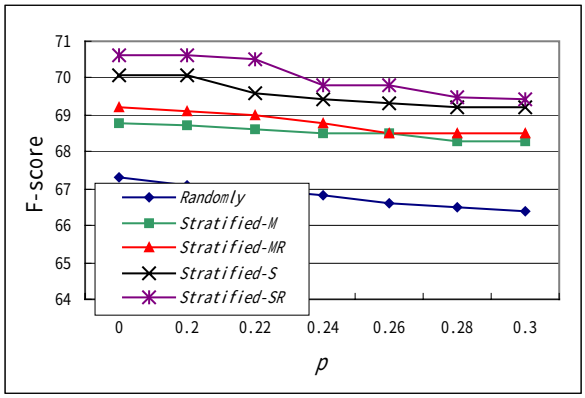


Figure 4. Performance for different *p* values

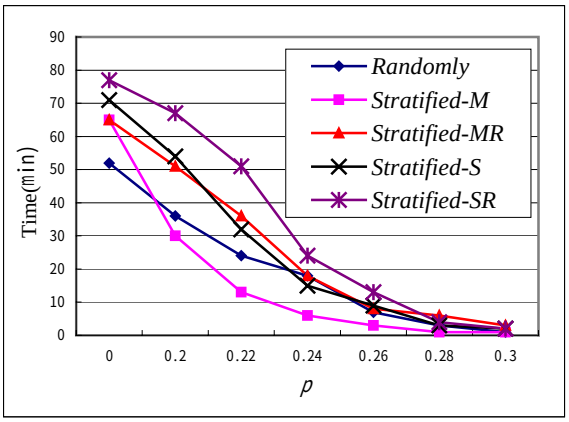


Figure 5. Bootstrapping time for different *p* values

- The performance decreases slowly while the bootstrapping time decreases dramatically with the increase of p from 0 to 0.3. Specifically, when the p equals to 0.3, the bootstrapping time tends to be neglected, while the performance is almost similar to the initial performance. It implies that we can find a reasonable point for each sampling strategy, at which the time falls greatly while the performance nearly does not degrade.

| Relation types | *Bootproject* | | | *LP-js* | | | *Stratified Bootstrapping* | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| ROLE | 78.5 | 69.7 | 73.8 | 81.0 | 74.7 | 77.7 | 74.7 | 86.3 | 80.1 |
| PART | 65.6 | 34.1 | 44.9 | 70.1 | 41.6 | 52.2 | 66.4 | 47.0 | 55.0 |
| AT | 61.0 | 84.8 | 70.9 | 74.2 | 79.1 | 76.6 | 74.9 | 66.1 | 70.2 |
| NEAR | - | - | - | 13.7 | 12.5 | 13.0 | 100.0 | 2.9 | 5.6 |
| SOC | 47.0 | 57.4 | 51.7 | 45.0 | 59.1 | 51.0 | 65.2 | 79.0 | 71.4 |
| Average | 67.9 | 67.4 | 67.6 | 73.6 | 69.4 | 70.9 | 73.8 | 73.3 | 73.5 |

Table 4. Comparison of semi-supervised relation classification systems on the ACE RDC 2003 corpus

● Clearly, if the performance is the primary concern, then $p$=0.2 may be the best choice in that we can get ~30% saving on the time at the cost of only ~0.08 loss in F-score on average. If the time is a primary concern, then $p$=0.22 is a reasonable threshold in that we get ~50% saving on the time at the cost of ~0.25 units loss in F-score on average. This suggests that our proposed stoppage criterion is effective to terminate the bootstrapping procedure with minor performance loss.

**Comparison of *Stratified Bootstrapping* with *Bootproject* and *Label propagation***

Table 4 compares *Bootproject* (Zhang, 2004), *Label propagation* (Chen et al., 2006) with our *Stratified Bootstrapping* on the 5 major types of the ACE RDC 2003 corpus.

Both *Bootproject* and *Label propagation* select 100 initial instances randomly, and at each iteration, the top 100 instances with the highest confidence are added to the labeled data. Differently, we choose 100 initial seeds using stratified sampling strategy; similarly, the top 100 instances with the highest confidence are augmented to the labeled data at each iteration. Due to the lack of comparability followed from the different size of the labeled data used in (Zhou et al., 2008), we omit their results here.

This table shows that our *stratified bootstrapping* procedure significantly outperforms both *Bootproject* and *Label Propagation* methods on the ACE RDC corpus, with the increase of 5.9/4.1 units in F-score on average respectively. *Stratified bootstrapping* consistently outperforms *Bootproject* in every major relation type, while it outperforms *Label Propagation* in three of the major relation types, especially SOC type, with the exception of AT and NEAR types. The reasons may be follows. Although there are many AT relation instances in the corpus, they are scattered divergently in multi-dimension space so that they tend to be relatively difficult to be recognized via SVM. For the NEAR relation instances, they occur least frequently in the whole corpus, so it is very hard for them to be identified via SVM. By contrast, even small size of labeled instances can be fully utilized to correctly induce the unlabeled instances via LP algorithm due to its ability to exploit manifold structures of both labeled and unlabeled instances (Chen et al., 2006).

In general, these results again suggest that the sampling strategy in selecting the initial seed set plays a critical role for relation classification, and stratified sampling can significantly improve the performance due to proper selection of the initial seed set.

## 6 Conclusion

This paper explores several key issues in semi-supervised learning based on bootstrapping for semantic relation classification. The application of stratified sampling originated from statistics theory to the selection of the initial seed set contributes most to the performance improvement in the bootstrapping procedure. In addition, the more strata the training data is divided into, the better performance will be achieved. However, the augmentation of the labeled data using the stratified strategy fails to function effectively largely due to the unbalanced distribution of the confidently classified instances, rather than the stratified sampling strategy itself. Furthermore, we also propose a mean entropy-based stoppage criterion in the bootstrapping procedure, which can significantly decrease the training time with little loss in performance. Finally, it also shows that our method outperforms other state-of-the-art semi-supervised ones.

**Acknowledgments**

This research is supported by Project 60673041 and 60873150 under the National Natural Science Foundation of China, Project 2006AA01Z147 under the “863” National High-Tech Research and Development of China,

Project BK2008160 under the Jiangsu Natural Science Foundation of China, and the National Research Foundation for the Doctoral Program of Higher Education of China under Grant No. 20060285008. We would also like to thank the excellent and insightful comments from the three anonymous reviewers.

## References

S. Abney. Bootstrapping. 2002. *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL 2002)*.

ACE 2002-2007. The Automatic Content Extraction (ACE) Projects. 2007. http//www.ldc.upenn.edu/Projects/ACE/.

E. Agichtein and L. Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the 5th ACM international Conference on Digital Libraries (ACMDL 2000)*.

A. Blum and T. Mitchell. 1996. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the workshop on Computational Learning Theory*. Morgan Kaufmann Publishers.

S. Brin. 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at 6th International Conference on Extending Database Technology (EDBT 98)*.

C.C. Chang and C.J. Lin. 2001. LIBSVM: a library for support vector machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm.

M. Collins. 2003. Head-Driven Statistics Models for Natural Language Parsing. *Computational linguistics*, 29(4): 589-617.

J.X. Chen, D.H. Ji, and L.T. Chew. 2006. Relation Extraction using Label Propagation Based Semi supervised Learning. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association of Computational Linguistics (COLING/ACL 2006)*, pages 129-136. July 2006, Sydney, Australia.

A. Culotta and J. Sorensen. 2004. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting of the Association of Computational Linguistics (ACL 2004)*, pages 423-439. 21-26 July 2004, Barcelona, Spain.

T. Hasegawa, S. Sekine, and R. Grishman. 2004. Discovering Relations among Named Entities from Large Corpora. In *Proceedings of the 42nd Annual Meeting of the Association of Computational Linguistics (ACL 2004)*. 21-26 July 2004, Barcelona, Spain.

N. Kambhatla. Combining lexical, syntactic and semantic features with Maximum Entropy models for extracting relations. In *Proceedings of the 42nd Annual Meeting of the Association of Computational Linguistics (ACL 2004)(posters)*, pages 178-181. 21-26 July 2004, Barcelona, Spain.

S. Miller, H. Fox, L. Ramshaw, and R. Weischedel. 2000. A novel use of statistical parsing to extract information from text. In *Proceedings of the 6th Applied Natural Language Processing Conference*. 29 April-4 May 2000, Seattle, USA.

J. Neyman. 1934. On the Two Different Aspects of the Representative Method: The Method of Stratified Sampling and the Method of Purposive Selection. *Journal of the Royal Statistical Society*, 97(4): 558-625.

L.H. Qian, G.D. Zhou, Q.M. Zhu, and P.D Qian. 2008. Exploiting constituent dependencies for tree kernel-based semantic relation extraction. In *Proceedings of The 22nd International Conference on Computational Linguistics (COLING 2008)*, pages 697-704. 18-22 August 2008, Manchester, UK.

D. Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *the Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL 95)*, pages 189-196. 26-30 June 1995, MIT, Cambridge, Massachusetts, USA.

D. Zelenko, C. Aone, and A. Richardella. 2003. Kernel Methods for Relation Extraction. *Journal of Machine Learning Research*, (2): 1083-1106.

M. Zhang, J. Zhang, J. Su, and G.D. Zhou. 2006. A Composite Kernel to Extract Relations between Entities with both Flat and Structured Features. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association of Computational Linguistics (COLING/ACL 2006)*, pages 825-832. Sydney, Australia.

M. Zhang, J. Su, D. M. Wang, G. D. Zhou, and C. L. Tan. 2005. Discovering Relations between Named Entities from a Large Raw Corpus Using Tree Similarity-Based Clustering. In *Proceedings of the 2nd international Joint Conference on Natural Language Processing (IJCNLP-2005)*, pages 378-389. Jeju Island, Korea.

Z. Zhang. 2004. Weakly-supervised relation classification for Information Extraction. In *Proceedings of ACM 13th conference on Information and Knowledge Management (CIKM 2004)*. 8-13 Nov 2004, Washington D.C., USA.

G.D. Zhou, J. Su, J. Zhang, and M. Zhang. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd Annual Meeting of the Association of Computational Linguistics (ACL 2005)*, pages 427-434. Ann Arbor, USA.

G.D. Zhou, J.H. Li, L.H. Qian, and Q.M. Zhu. 2008. Semi-Supervised Learning for Relation Extraction. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing (IJCNLP-2008)*, page 32-38. 7-12 January 2008, Hyderabad, India.