One Class per Named Entity: Exploiting Unlabeled Text for Named Entity Recognition

Yingchuan Wong and **Hwee Tou Ng**
Department of Computer Science
National University of Singapore
3 Science Drive 2, Singapore 117543
yingchuan.wong@gmail.com, nght@comp.nus.edu.sg

Abstract

In this paper, we present a simple yet novel method of exploiting unlabeled text to further improve the accuracy of a high-performance state-of-the-art named entity recognition (NER) system. The method utilizes the empirical property that many named entities occur in one name class only. Using *only* unlabeled text as the additional resource, our improved NER system achieves an F1 score of 87.13%, an improvement of 1.17% in F1 score and a 8.3% error reduction on the CoNLL 2003 English NER official test set. This accuracy places our NER system among the top 3 systems in the CoNLL 2003 English shared task.

1 Introduction

The named entity recognition (NER) task involves identifying and classifying noun phrases into one of many semantic classes such as persons, organizations, locations, etc. NER systems can be built by supervised learning from a labeled data set. However, labeled data sets are expensive to prepare as it involves manual annotation efforts by humans. As unlabeled data is easier and cheaper to obtain than labeled data, exploiting unlabeled data to improve NER performance is an important research goal.

An empirical property of named entities is that many named entities occur in one name class only. For example, in the CoNLL 2003 English NER [Sang and Meulder, 2003] training set, more than 98% of named entity types have exactly one name class. This paper presents a novel yet simple method of exploiting this empirical property of one class per named entity to further improve the NER accuracy of a high-performance state-of-the-art NER system. Using *only* unlabeled data as the additional resource, we achieved an F1 score of 87.13%, which is an improvement of 1.17% in F1 score and a 8.3% error reduction on the CoNLL 2003 English NER official test set. This accuracy places our NER system among the top 3 systems in the CoNLL 2003 English shared task.

2 System Description

The NER system we implemented is based on the maximum entropy framework similar to the MENE system of [Borthwick, 1999]. Each name class N is divided into 4 sub-classes: N_{begin}, $N_{continue}$, N_{end}, and N_{unique}. Since the CoNLL 2003 English NER shared task data has 4 name classes (person, organization, location, and miscellaneous), there is a total of 17 classes (4 name classes $\times$ 4 sub-classes + 1 not-a-name class) that a word can possibly be assigned to.

2.1 Maximum Entropy

In the maximum entropy framework, the best model is the one that has the highest entropy while satisfying the constraints imposed. These constraints are derived from training data, expressing some relationship between features and class. It is unique, agrees with the maximum-likelihood distribution, and has the exponential form [Pietra *et al.*, 1997]:

$$p(c|h) = \frac{1}{Z(h)} \prod_{j=1}^{k} \alpha_j^{f_j(h,c)},$$

where c refers to the class, h the history (or context), and $Z(h)$ is a normalization function. The features used in the maximum entropy framework are binary-valued functions which pair a class with various elements of the context. An example of a feature function is

$$f_j(h,c) = \begin{cases} 1 & \text{if } c = person_{begin}, \text{ word = } \textit{JOHN} \\ 0 & \text{otherwise} \end{cases}$$

Generalized Iterative Scaling (GIS) is used to estimate the parameters α_j [Darroch and Ratcliff, 1972].

2.2 Testing

It is possible that the classifier produces a sequence of invalid classes when assigning classes to the words in a test sentence during testing (e.g., $person_{begin}$ followed by $location_{end}$). To eliminate such sequences, we define a transition probability between classes, $P(c_i|c_j)$, to be equal to 1 if the transition is valid, and 0 otherwise. The probability of assigning the classes $c_1, \ldots, c_n$ to the words in a sentence s in a document D is defined as follows:

$$P(c_1, \ldots, c_n|s, D) = \prod_{i=1}^{n} P(c_i|s, D) * P(c_i|c_{i-1}),$$

where $P(c_i|s, D)$ is determined by the maximum entropy classifier. The sequence of classes with the highest probability is then selected using the Viterbi algorithm.

3 Feature Representation

Our feature representation is adapted from [Chieu and Ng, 2003], but *without* using any external name lists. We now give a general overview of the features used.

3.1 Local Features

Local features of a token w are those that are derived from only the sentence containing w. The main local features are:

Zoning Each document is segmented using simple rules into $headline$, $author$, $dateline$, and $text$ zones. The zone is used in combination with other information like capitalization and whether the token is the first word of the sentence.

Lexical Information Strings of the current, previous, and next words are used in combination with their case information. Words not found in a vocabulary list generated from training data are marked as rare.

Orthographic Information Such information includes case, digits, and the occurrence of special symbols like *$* within a word.

Word Suffixes For each named entity in the training data, 3-letter word suffixes that have high correlation score with a particular name class are collected.

Class Suffixes A list is compiled from training data for tokens that frequently terminate a particular name class. For example, the token *association* often terminates the organization class.

3.2 Global Features

Global features are derived from other sentences in the same document containing w. The global features include:

N-grams Unigrams and bigrams of another occurrence of the token.

Class Suffixes Class suffix of another occurrence of the token.

Capitalization The case information of the first occurrence of the token in an unambiguous position (non-first words in a text zone).

Acronyms Words made up of all capitalized letters are matched with sequences of initial capitalized words in the same document to identify acronyms.

Sequence of InitCaps For every sequence of initial capitalized words, its longest substring that occurs in the same document is identified.

Name Class of Previous Occurrences The predicted name classes of previous tokens are used as features.

4 Approach

4.1 One Class per Named Entity

In the CoNLL 2003 English NER training data, we have observed that around 98% of the named entity types and 91% of the named entity tokens have exactly one class. Incidentally, within natural language processing, there are similar observations proposed previously regarding the number of word senses of a collocation (one sense per collocation [Yarowsky, 1993]) and the number of word senses of occurrences of a word in a document (one sense per discourse [Gale *et al.*, 1992]). Table 1 shows the percentage of named entities that have exactly one class from the training datasets of several different shared tasks. It can be seen that this empirical property occurs in other languages as well.

4.2 Motivation

Consider the following sentence taken from a news article reporting soccer results in the CoNLL 2003 English shared task official test set:

AC Milan (9) v Udinese (11) 1330

This sentence contains two named entities: *AC Milan* and *Udinese* are organization names. If these named entities have not been seen in the training data, it might be difficult to predict their name class due to poor contextual information. Consider the following two sentences found in the English Gigaword Corpus:

> AC Milan, who beat *Udinese* 2-1 at the San Siro stadium, and Lazio, who won 1-0 at Cagliari, also have a maximum six points from two games.
>
> Milan needed a 10th minute own goal headed in by *Udinese* defender Raffaele Sergio to get off the mark, and were pegged back on the hour by Paolo Poggi.

It is easier to predict the named entities in the above sentences as tokens like *beat* and *defender* offer useful hints to the classifier.

In a data set, different sentences contain differing amounts of useful contextual information. This results in a classifier predicting a mix of correct and incorrect labels for the same named entity. If we have a reasonably accurate classifier, there will be more correct labels assigned than incorrect labels in general.

We propose a method of exploiting this empirical property of one class per named entity by using the most frequently occurring class of a named entity found in a machine-tagged data set. We call this most frequently occurring class of a named entity its *majority tag*.

4.3 Baseline Method

The CoNLL baseline system [Sang and Meulder, 2003] first determines the majority class C of a named entity N found in the training data. The baseline system then labels all occurrences of N found in the test data with class C, regardless of context. Any named entities in the test data which do not appear in the training data are not assigned any name class.

	Test A	Test B
All NEs	71.18	59.61
Seen NEs	83.44	80.46

Table 2: F1 score of the baseline method for all named entities and for only named entities seen in the training data

Table 2 summarizes the F1 score of the baseline method for all (seen and unseen) named entities and all seen named entities. In all tables in this paper, Test A refers to the CoNLL 2003 development test set and Test B refers to the CoNLL 2003 official test set. A named entity is seen if it exists in the training data. The baseline F1 score of the official test

Language	Shared Task	Types	Tokens
English	CoNLL 2003	98%	91%
German	CoNLL 2003	99%	96%
Dutch	CoNLL 2002	99%	98%
Spanish	CoNLL 2002	96%	76%
English (Biomedical)	BioNLP 2004	98%	85%

Table 1: Percentage of named entities that have exactly one class for several shared task training datasets.

$T \leftarrow$ training dataset
$U \leftarrow$ unlabeled dataset
$E \leftarrow$ test dataset

1. Train classifier h_1 on T
2. $U' \leftarrow$ label U using h_1
3. $L \leftarrow$ extract $(NE, MajTag)$ pairs from U'
4. Train classifier h_2 on T using L
5. $E' \leftarrow$ label E with h_2 using L

Figure 1: Pseudocode for the majority tag method

set is 59.61. However on closer examination, the F1 score is 80.46 when we only consider the named entities seen in the training data. Unseen NEs lower the F1 score since the baseline method does not assign any name class to named entities in the test set that are not found in the training data.

This shows that the poor baseline performance is primarily attributed to unseen named entities. In addition, reasonable (though not great) performance can be obtained using this baseline method if all the named entities are seen, even though contextual information is not used. This suggests that the majority tag is valuable and can be exploited further if more unseen named entities in the test data can be discovered beforehand. One way to do this is to gather named entities in machine-tagged unlabeled text so as to increase the likelihood that a named entity found in the test set is seen beforehand. Even though the NER system that labels the unlabeled text is not perfect, useful information could still be gathered.

4.4 The Majority Tag as a Feature

There are some words like *Jordan* that can refer to a person or a location depending on the context in which they are used. Hence, if the majority tag is incorporated as a feature, a classifier can be trained to take into account the context in which a named entity is used, as well as exploit the majority tag.

Figure 1 gives the high-level pseudocode of our method which is based on the maximum entropy framework and involves two stages. First, an initial-stage supervised classifier h_1 is trained with labeled data T. The classifier h_1 then labels a large unlabeled dataset U to produce a machine-tagged data set U'.

Next, the majority tag list L is produced by extracting the list of named entities with their associated majority tags from this machine-tagged data set U'. L thus contains a list of $(NE, MajTag)$ pairs, where $MajTag$ is the name class that occurs most frequently for the named entity NE in the machine-tagged data set U'. In L, the case of named entities is retained (i.e., whether a named entity appears in upper or lower case), and named entities that occur only once in U' are pruned away from L.

Lastly, a final-stage classifier h_2 is trained with labeled data T and uses the majority tag list L to generate the new feature described as follows. During training or testing, when the h_2 classifier encounters a sequence of tokens $w_1...w_n$ such that $(w_1...w_n, nc) \in L$, a feature *MJTAG-nc* will be turned on for each of the tokens $w_1, ..., w_n$ in the sequence. If there is more than one matching sequence of different lengths, preference will be given to the longest matching sequence. For example, consider the following sentence:

Udinese midfielder Fabio Rossitto has the flu.

If $(Udinese, ORG)$, $(Fabio\ \ Rossitto, PER)$, and $(Fabio, LOC)$ are present in the majority tag list, the features in the first column below will be turned on. Notice the feature that is turned on for *Fabio* is MJTAG-PER and not MJTAG-LOC, because the longest matching sequence is *Fabio Rossitto*.

Feature	Token
MJTAG-ORG	Udinese
-	midfielder
MJTAG-PER	Fabio
MJTAG-PER	Rossitto
-	has
-	the
-	flu

5 Experiments

Our unlabeled data set consists of 600 million words from the LDC English Gigaword Corpus. For all experiments, features that occur only once in the training data are not used and the GIS algorithm is run for 1,000 iterations. Unlabeled data is randomly selected from the English Gigaword Corpus. The reported scores are the averages of ten runs, and similar trends are observed over all runs.

5.1 Results

Table 3 shows the F1 scores of two systems trained with 5k and 204k labeled examples and exploiting increasing amounts of unlabeled data. The entire CoNLL 2003 labeled training set is used for the 204k runs. Performance peaks at 300 million words of unlabeled data on the CoNLL 2003 official test set (Test B) and drops slightly after that. The best F1 score for the official CoNLL English test set is 87.13% using

300 million words of unlabeled data. This is an improvement of 1.17% in F1 score and an error reduction of 8.3%. The +4.02% improvement in the 5k system shows that the method is even more effective on a small labeled dataset. Improvements on small labeled data sets are important for other resource-poor languages where labeled data is scarce.

Figure 2 and Figure 3 show the accuracy improvements with increasing amounts of unlabeled data, using 204k and 5k labeled examples respectively. Accuracy generally increases at a decreasing rate with additional unlabeled data for both systems.

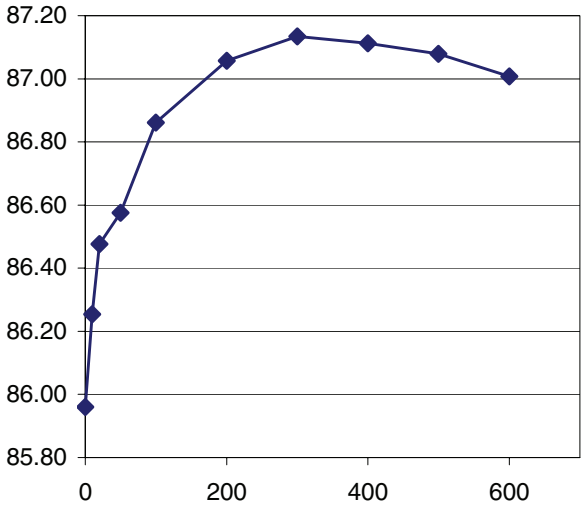

Figure 2: Test B F1 score of our system trained with the full 204k labeled training dataset from CoNLL and exploiting increasing amounts of unlabeled data.

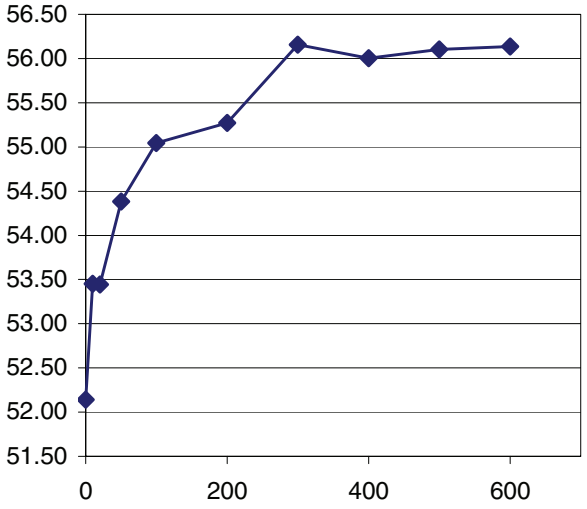

Figure 3: Test B F1 score of our system trained with 5k labeled examples from the CoNLL training dataset and exploiting increasing amounts of unlabeled data.

5.2 Comparison with Previous Best Results

Figure 4 shows the performance comparison of our system against the systems in CoNLL 2003. The top system [Florian *et al.*, 2003] uses an externally trained NER system as part of a combined system. The second best system [Chieu and Ng, 2003] uses external name lists and has a score of 88.31, but when not using any external name lists, its score drops to 86.84.

Note that the NER system reported in this paper does *not* utilize any external name lists. Rather, the majority tag list was generated automatically from machine-tagged unlabeled data, without manual labeling of name class for names found in this list. In addition, the names found in our majority tag list may contain errors in general, since they are automatically determined. Despite this, we have achieved performance at the third place. Our results are encouraging given that we use unlabeled data as the only additional resource.

6 Analysis

Table 4 shows the number of named entities in Test B discovered due to adding 300 million words of unlabeled data. In the rightmost column of Table 4, named entities are considered seen if they are found in either the training data or the machine-tagged unlabeled data. The unlabeled data is labeled with a classifier trained with 204k labeled examples. It can be seen that adding unlabeled data has successfully led to a large proportion of named entities previous unseen to be discovered.

	0 M	300M
Seen NEs	3148	5164
Unseen NEs	2500	484

Table 4: Seen and unseen counts of named entities in Test B before and after using 300 million words of unlabeled data. In this table, named entities are seen if they exist in the training data or machine-tagged unlabeled data. The classifier is trained with 204k labeled examples.

Table 5 shows the Test B F1 score of the two systems before and after using 300 million words of unlabeled text.

(a) 204k Labeled Training Examples

	0 M	300 M	Change
All	85.96	87.13	+1.17
Seen NEs	91.77	92.12	+0.35
Unseen NEs	75.23	77.29	+2.06

(b) 5k Labeled Training Examples

	0 M	300 M	Change
All	52.14	56.16	+4.02
Seen NEs	71.91	72.52	+0.61
Unseen NEs	44.53	49.14	+4.61

Table 5: Breakdown of improvement in F1 score for seen and unseen named entities in Test B using 300 million words of unlabeled text.

	204k (All)		5k	
MWords	Test A	Test B	Test A	Test B
0	90.80	85.96	57.09	52.14
10	91.16	86.25	58.84	53.45
20	91.20	86.48	58.96	53.44
50	91.29	86.58	59.80	54.38
100	91.36	86.86	60.40	55.04
200	91.47	87.06	60.97	55.27
300	91.50	87.13	61.21	56.16
400	91.55	87.11	61.37	56.00
500	91.59	87.08	61.13	56.10
600	91.64	87.01	61.15	56.14
Max Improvement	+0.84	+1.17	+4.28	+4.02

Table 3: F1 scores for different amounts of labeled and unlabeled data. Unlabeled data are randomly selected from the English Gigaword Corpus. Reported F1 scores are obtained using an average of 10 runs.

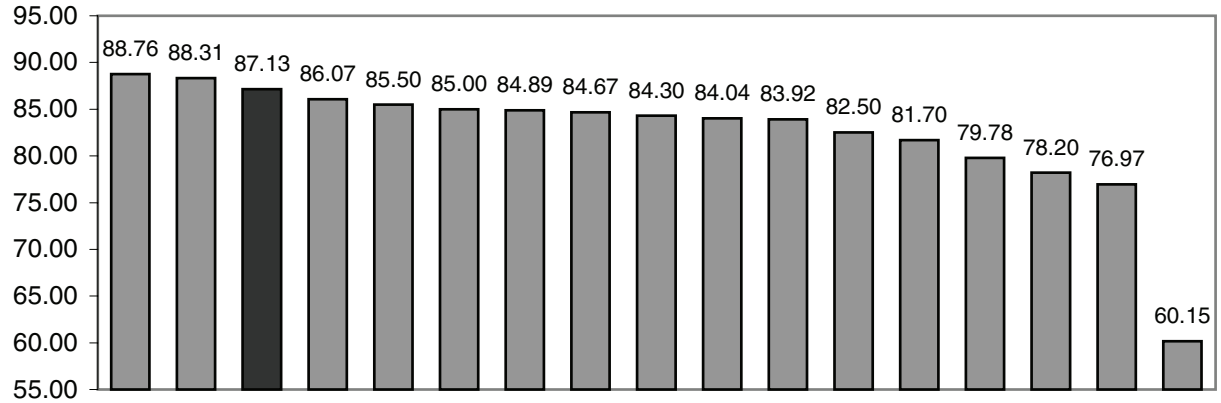

Figure 4: Comparison of F1 scores on the official test set against the systems in the CoNLL 2003 shared task. Our system's performance is represented by the black shaded column at the third place.

The F1 scores for seen and unseen named entities are separately listed. In this table, a named entity is considered seen if it is present in the labeled training data. In both systems, the improvement of F1 score on unseen named entities is much greater than that of the seen named entities. This demonstrates that we are successful in what we set out to achieve: using unlabeled data to improve the accuracy of unseen named entities.

7 Related Work

[Zhu, 2005] gave a comprehensive summary of recent work in learning with labeled and unlabeled data. There is much research on co-training, such as [Blum and Mitchell, 1998; Collins and Singer, 1999; Pierce and Cardie, 2001]. Our work does not fall under the co-training paradigm. Instead of using two cooperating classifiers, we use two classifiers in two stages. Co-training produces a final system combining two different sets of features that outperforms either system alone. In our work, however, the final-stage classifier has the features of the initial-stage classifier with an additional majority tag feature.

[Ando and Zhang, 2005] presents a semi-supervised learning paradigm called *structural learning*. This method identifies the common predictive structure shared by multiple classification problems, which can then be used to improve performance on the target problem. The auxiliary classification problems are automatically generated on unlabeled data. Our proposed method in this paper to exploit unlabeled data is complementary to their proposed structural learning method.

Although prior research efforts on bootstrapping methods for NER [Collins and Singer, 1999; Cucerzan and Yarowsky, 1999] also rely on the empirical property of one class per named entity in some way, our work reported in this paper is different in that it exploits this property to make a very good NER system even better. In particular, its simplicity allows it to be easily applied to other resource-poor languages when labeled data is scarce.

8 Conclusion

In this paper, we have presented a novel yet simple technique applied to the named entity recognition task using majority tags. By using unlabeled data as the only additional resource, we have achieved an error reduction of 8.3% and an F1 score of 87.13%. Our NER system has achieved performance among the top 3 systems of the CoNLL 2003 shared task.

References

[Ando and Zhang, 2005] Rie Kubota Ando and Tong Zhang. A high-performance semi-supervised learning method for text chunking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2005.

[Blum and Mitchell, 1998] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, 1998.

[Borthwick, 1999] Andrew Borthwick. *A Maximum Entropy Approach to Named Entity Recognition*. PhD thesis, New York University, 1999.

[Chieu and Ng, 2003] Hai Leong Chieu and Hwee Tou Ng. Named entity recognition with a maximum entropy approach. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*, 2003.

[Collins and Singer, 1999] Michael Collins and Yoram Singer. Unsupervised models for named entity classification. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

[Cucerzan and Yarowsky, 1999] Silviu Cucerzan and David Yarowsky. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

[Darroch and Ratcliff, 1972] J. N. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. *The Annals of Mathematical Statistics*, 43(5), 1972.

[Florian *et al.*, 2003] Radu Florian, Abe Ittycheriah, Hongyan Jing, and Tong Zhang. Named entity recognition through classifier combination. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*, 2003.

[Gale *et al.*, 1992] William A. Gale, Kenneth W. Church, and David Yarowsky. One sense per discourse. In *Proceedings of the Speech and Natural Language Workshop*, 1992.

[Pierce and Cardie, 2001] David Pierce and Claire Cardie. Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, 2001.

[Pietra *et al.*, 1997] Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4), 1997.

[Sang and Meulder, 2003] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL)*, 2003.

[Yarowsky, 1993] David Yarowsky. One sense per collocation. In *Proceedings of the ARPA Human Language Technology Workshop*, 1993.

[Zhu, 2005] Xiaojin Zhu. *Semi-Supervised Learning with Graphs*. PhD thesis, Carnegie Mellon University, 2005.