A Hybrid Learning for Named Entity Recognition Systems

RAYMOND CHIONG[1]

School of Computing & Design,
Swinburne University of Technology (Sarawak Campus),
Jalan Simpang Tiga, 93576 Kuching, Sarawak, Malaysia.
[1]rchiong@swinburne.edu.my

Abstract. This paper presents a hybrid method using machine learning approach for Named Entity Recognition (NER). A system built based on this method is able to achieve reasonable performance with minimal training data and gazetteers. The hybrid machine learning approach differs from previous machine learning-based systems in that it uses Maximum Entropy Model (MEM) and Hidden Markov Model (HMM) successively. We report on the performance of our proposed NER system using British National Corpus (BNC). In the recognition process, we first use MEM to identify the named entities in the corpus by imposing some temporary tagging as references. The MEM walkthrough can be regarded as a training process for HMM, as we then use HMM for the final tagging. We show that with enough training data and appropriate error correction mechanism, this approach can achieve higher precision and recall than using a single statistical model. We conclude with our experimental results that indicate the flexibility of our system in different domains.

Keywords: Machine learning, named entity recognition, tagging.

(Received March 6, 2008 / Accepted July 16, 2008)

1 Introduction

In the field of computational linguistics, one of the very important research areas of information extraction (IE) comes in Named Entity Recognition (NER). NER is a subtask of IE that seeks to identify and classify the predefined categories of named entities in text documents. Considerable amount of work has been done on NER in recent years due to the increasing demand of automated texts and the wide availability of electronic corpora. While it is relatively easy and natural for a human reader to read and understand the context of a given article, getting a machine to understand and differentiate between words is a big challenge. For instance, the word 'brown' may refer to a person called Mr. Brown, or the colour of an item which is brown. Human readers can easily discern the meaning of the word by looking at the context of that particular sentence, but it would be almost impossible for a computer to interpret it without any additional information.

To deal with the issue, researchers in NER field have proposed various rule-based systems [15, 7, 8]. These systems are able to achieve high accuracy in recognition with the help of some lists of known named entities called gazetteers. The problem with rule-based approaches, however, is that they lack the robustness and portability. They incur steep maintenance cost especially when new rules need to be introduced for some new information or new domains.

A better option is therefore to use machine learning approaches that are trainable and adaptable with the use of statistical models. Three well-known machine learning approaches that have been used extensively in NER are Hidden Markov Model (HMM), Maximum Entropy Model (MEM) and Decision Tree. Many of the existing machine learning-based NER systems [2, 16, 3, 1, 4, 11] are able to achieve near-human performance for named entity tagging, even though the overall performance is still about 2% short from the rule-based systems.

There are also many attempts to improve the perfor-

mance of NER using hybrid approaches with the combination of handcrafted rules and statistical models [9, 13, 12]. These systems can achieve relatively good performance in the targeted domains due to the comprehensive handcrafted rules. Nevertheless, the portability problem still remains unsolved when it comes to dealing with NER in various domains.

In this paper, we propose a hybrid machine learning approach using MEM and HMM successively. The reason for using two statistical models in succession instead of one is due to the distinctive nature of the two models. HMM is able to achieve better performance than any other statistical models, and is generally regarded as one of the most successful machine learning approaches. However, it suffers from sparseness problem, which means considerable amount of data is needed for it to achieve acceptable performance. On the other hand, MEM is able to maintain reasonable performance even when there is little data available for training purpose. Our idea is therefore to walkthrough the testing corpus using MEM first in order to generate a temporary tagging result, while this procedure can be simultaneously used as a training process for HMM. During the second walkthrough, the corpus uses HMM for the final tagging. In this process, the temporary tagging result generated by MEM will be used as a reference for subsequent error checking and correction. In the case when there is little training data available, the final result can still be reliable due to the contribution of the initial MEM tagging result.

The rest of this paper is organised as follows: in the second section, some background studies are carried out and the related previous approaches in NER are mentioned. The methodology we use is then presented in section 3. Next, section 4 discusses the experimental results based on our proposed system. In the final section, conclusion is drawn with some anticipated future work being suggested.

2 Background

2.1 Message Understanding Conference

In 1987, the Naval Ocean Systems Center (NOSC), which is presently known as the Naval Command, Control and Ocean Surveillance Center, initiated the first Message Understanding Conference (MUC). Subsequently, a series of MUCs had been held and designed to promote and evaluate research in IE. The evaluations achieved through these MUCs have led the research program in IE until its present state. In 1995, goals and tasks were set up for MUC-6 to make the IE system more practical with an aim to achieve automatic performance with high accuracy. "Named Entity" was then developed to help identifying the names of person, organisation, and geographic location in a text. Since then, the NER tasks have become a central theme in MUC (see [5, 6] for more details).

According to the specifications defined by MUC, the NER tasks generally work on seven types of named entities as listed below with their respective markup:

- PERSON (ENAMEX)
- ORGANISATION (ENAMEX)
- LOCATION (ENAMEX)
- DATE (TIMEX)
- TIME (TIMEX)
- MONEY (NUMEX)
- PERCENT (NUMEX)

From the list above, three subtasks are derived from these seven types of named entities and assigned with three respective SGML tag elements, namely ENAMEX, TIMEX and NUMEX. As TIMEX and NUMEX are fairly easy to predict with some effective finite state methods [10], most of the current research deals only with ENAMEX that are highly variable and ambiguous.

2.2 Previous Approaches

Since MUC-6 and MUC-7, many NER systems have been proposed and proven to be successful in their targeted domains. In general, NER systems that use handcrafted rules still lead the way, with the highest F-measure score up to 96.4% achieved in MUC-6 as compared to the statistical approaches that were able to achieve 94.9% [16].

In rule-based approaches, a set of rules or patterns is defined to identify the named entities in a text. These rules or patterns consist of distinctive word format, such as capitalisation or particular preposition prior to a named entity. For instance, a string of capitalised words behind titles such as ‘Mr’, ‘Dr’, etc will be identified as name of a person, whereas a capitalised word after a preposition such as ‘in’, ‘at’, ‘near’, etc is most likely to be a location. By implementing a finite set of carefully predefined pattern matching rules, the named entities within a text could be found systematically.

There has been a substantial amount of work done using the rule-based approaches. One of the very well documented systems that followed the direction of this

approach is the framework of the LaSIE System reported by [15]. Another well-known example of rule-based system can be found in the IsoQuest's NetOwl Text Extraction System presented by [7]. Meanwhile, [8] had also built an NER system based on handcrafted rules that is able to achieve an average of 93% precision and 95% recall across a diverse text types.

Statistical approaches, on the other hand, work by using a probabilistic model containing features to the data which are similar to the rule-based approaches. The features of the data, which could be understood as rules set for the probabilistic model, are produced by learning the resulting corpora with correctly marked named entities. The probabilistic model then uses the features to calculate and identify the most probable named entities. As such, if the annotated features of the data are truly reliable, the model would have a high probability in finding almost all the named entities within a text.

In the last decade, huge amount of work in NER has been done using the statistical approaches based on some very large corpora. MEM, one of the most popular statistical models, has been applied frequently in various NER tasks. One significant account on MEM is the MENE system reported by [3]. In their system, they used four main features to identify the named entities, which they referred to as the binary features, lexical features, section features and dictionary features.

The binary features in MENE system basically deal with capitalisation in the text. Meanwhile, lexical features are concerned with the lexical terms such as list of words and their types which are used with a grammar. Section features indicate a current section of the text, whereas the dictionary features make use of a broad array of dictionaries of single or multiple terms such as first names, organisation names, corporate suffixes, etc. The dictionary features are similar to the gazetteers used for rule-based systems, except that dictionaries in MENE system require no massive maintenance effort.

Nevertheless, using MENE system alone on the MUC-7 test data as reported in [3] achieved only an F-measure of 84.22%. For MENE system to work better, [3] combined MENE with other rule-based approaches in order to achieve superior results.

Besides [3], [1] also reported on an NER system that was able to achieve an F-measure score of 89.58% by using MEM. With an annotated corpus and a set of features, they first built a baseline named entity recogniser which was then used to extract the named entities and their contextual information from non-annotated data. The accuracy of their system was further improved with a final recogniser that made use of the trained data.

Another MEM-based system can be found in [4]. They presented a system that made use of global information with just one classifier called MENERGI, and showed that their system was able to achieve performance comparable to the best machine learning-based systems in MUC-6 and MUC-7.

Apart from MEM, HMM is another well-known statistical model that has been used frequently in various NER systems. The IdentiFinder reported by [2] using a modified HMM was the best-performer on the official MUC-6 and MUC-7 test data among all the machine learning-based systems. IdentiFinder employed similar features to those of MENE system, and depended on statistics to make decision in identifying the named entities. It is different in a way that it has a complete probabilistic model that governed all decisions in classifying the named entities and modelled the categories of interest and the residual input that was not of interest.

The modified HMM used by IdentiFinder was subsequently adopted by [16]. In their work, they were able to increase the performance of their NER system dramatically by introducing four sub-features with back-off modelling. Using the test data from MUC-6 and MUC-7, their system was able to achieve F-measure scores of 96.6% and 94.1% respectively.

Many more previous work was done using statistical models other than MEM and HMM. There were also many NER systems that used hybrid approaches by combining the statistical models with some rule-based learning techniques. One very successful example can be found in [9], where they used massive handcrafted rules together with MEM for partial matching. Based on our observation on the previous approaches, however, no system has tried to use MEM and HMM successively. In the next section, we will describe the methodology we adopt in this paper.

3 Methodology

As mentioned before, the NER system we present in this paper uses two statistical models - MEM and HMM - in succession. The MEM we adopt is based on the MENE system reported by [3] whereas the HMM is based on the IdentiFinder reported by [2]. Our system is built with Java using the existing implementation from the JavaNLP repository which is available at `http://nlp.stanford.edu/javanlp/`. For training and experimental purposes, we have chosen British National Corpus (BNC) which contains texts that are diverse in terms of domain, style and genre to be our testing corpus. This is to ensure that the proposed NER system is domain-independent and can adequately cope with a variety of text types.

3.1 Maximum Entropy

By following the guidelines from MUC-6 and MUC-7 for the definition of the NER task, we tokenise every word from the corpus and assign them to a desired category of named entity with the tag of either "person" (<PER>), "organisation" (<ORG>) or "location" (<LOC>). We first use MEM to estimate the probability of a given word being fallen into one of the three categories mentioned based on a set of features and some training data. Two special conditions are taken into consideration when a word falls at the beginning (<START>) and at the end (<END>) of a sentence. In the case when a given word does not fall into any of the desired categories, empty tag (< >) will be placed to indicate that the word belongs to none of the desired categories. For the purpose of finding named entities, the maximum entropy estimation process uses a model that is described below to compute the conditional probability P for all tags t based on the history h, in which every feature f_i is associated with it a weighting parameter α_i:

$$P(t|h) = \frac{\prod_i \alpha_i^{f_i(h,t)}}{Z_\alpha(h)} = \frac{\prod_i \alpha_i^{f_i(h,t)}}{\sum_t \prod_i \alpha_i^{f_i(h,t)}} \qquad (1)$$

It is necessary to note that the history h mentioned in the model refers to all the conditioning data that enable our system to make a decision on the tagging process. It comprises all information derivable from the corpus relative to a token whose tag we are trying to determine, may it be the word itself or the features. The product of the weightings for all features active on h will then be calculated, and eventually be divided by a normalisation function, $Z_\alpha(h)$.

3.1.1 Feature Function

The computation of $P(t|h)$ above is dependent on a set of feature functions $f_i(h,t)$ that carries binary values. The feature function will help to make prediction on the tagging process based on some useful word features as well as lexical features. For instance, if h is capitalised = true and the previous word is "Mr.", the feature function will be set to 1 because it is very likely that t is a <PER>. Otherwise, the feature function will be set to 0 so that it is not taken in account in the weightings. For the feature selection process, we implement a simple count-based feature reduction [1] to include only those features that have been seen at least three times on the testing corpus. Multiple features are allowed for a single word. It is necessary to note that the features we use are similar to those used by MENE and IdentiFinder.

3.1.2 Gazetteers

Gazetteer has been found to be an essential element of our proposed NER system due to the limited amount of training material available for MEM. Based on the recommendation in [9], some small and well-studied lists of gazetteers are incorporated to our system. The gazetteers we use require no manual editing and are easily downloadable from the websites as listed in Table 1 below.

Description	Data Source
<PER>	`http://www.ssa.gov/OACT/babynames`
<ORG>	`http://www.fmlx.com`
<LOC>	`http://www.yahoo.com/regional`

Table 1: Sources of trained gazetteers.

3.1.3 Decoding

Once the features are trained and appropriate weight is being assigned to each feature, the final stage is to perform a viterbi search [14] to find the highest probability path through the lattice of conditional probabilities in order to mark up the correct tag for the named entities.

3.2 Hidden Markov Model

After the MEM walkthrough, all the tagged named entities in the testing corpus are used as training data for HMM to make the final tagging. Since we are confident that there will be sufficient training after parsing through the corpus using MEM, it is not necessary for our system to use the back-off models such as those used by [2] and [16].

In our system, HMM is used mainly for global context checking, that is to check the occurrences of the same named entity in different sections of the same text document. We believe that checking the context from the whole document is important as this will ensure the consistency of the tagged named entities and resolve some ambiguous cases. For instance, an organisation's name is often abbreviated especially when it has already been mentioned somewhere in a document. By checking the global information, we are able to identify the abbreviation as an organisation. Besides that, we often encounter some entities that are highly ambiguous, and their categories cannot be determined without taking the global context into consideration. The phrase 'Honda City' in sentences such as "Honda City is nice" or "Promotion for Honda City" could easily be misinterpreted as a location based on the local contextual evidence, unless we found another sentence that sounds like "I am driving Honda City".

Similar to the previously used MEM, we use HMM to compute the likelihood of words occurring within a given category of named entity. Every tokenised word is now considered to be in ordered pairs. By using a Markov chain, the likelihood of the words is calculated simply based on the previous word. For classifying the named entities, our system finds the most likely tag t for a given sequence of words w that maximises $P(t|w)$. The occurrences of the given events are counted throughout the whole text based on the calculation below:

$$P(t|t_{-1}, w_{-1}) = \frac{\text{count}(t, t_{-1}, w_{-1})}{\text{count}(t_{-1}, w_{-1})} \quad (2)$$

Finally, we use a classifier to correct the errors in the results derived from MEM to perform the final tagging process using HMM.

4 Experiments and Results

In this section, we report on the results of our proposed NER system. As mentioned earlier, we use BNC as the testing corpus in our work. Only 100 articles from BNC are extracted for the experimental purposes. This is due to the fact that we have to manually compile the key files - the files containing all the correctly annotated named entities - from all the selected articles. These key files are later used to evaluate the accuracy of the resulting marked-up files produced by our system. We reckon that it is just not possible for us to work on the whole corpus within the scope of our current research seeing that a statement from the BNC homepage at `http://www.natcorp.ox.ac.uk` reads like this:

> *"To put all the 100,106,008 words and 4,124 texts in the British National Corpus into perspective, the average paperback book has about 250 pages per centimeter of thickness; assuming 400 words a page, we calculate that the whole corpus printed in small type on thin paper would take up about ten meters of shelf space. Reading the whole corpus aloud at a fairly rapid 150 words a minute, eight hours a day, 365 days a year, would take just over four years."*

It is important to note that the 100 articles extracted from BNC have actually been selected carefully based on a wide range of domains from different fields (see Table 2). This is to ensure that our system is flexible on various domains. The recognition task will then be performed on the 100 articles without human intervention, and the three desired categories of named entities found will be marked up within the articles. Scoring of the results is done by registering the recall and precision, and thereafter calculating the F-measure score. Here, we define recall R as the number of correct tags in the file marked up by our system over the total number of annotated tags in the key file. The purpose of recall is to measure how well our system can perform the recognition task. Meanwhile, we define precision P as the number of correct tags in the file marked up by our system over the total number of tags being marked up. This is to see how accurate our system can perform the recognition search. F-measure is then calculated based on the weighted harmonic mean of recall and precision:

$$F = \frac{(\beta^2 + 1)RP}{\beta^2 R + P} \text{ with } \beta^2 = 1 \quad (3)$$

Based on the selected articles from BNC, system performance evaluation is carried out in three steps. First, we evaluate the system performance based on a single statistical model using MEM. Subsequently, HMM alone is used for a second evaluation. After that, MEM and HMM are incorporated successively to complete the whole evaluation process. The reason for this is to make a direct comparison between a single statistical approach and a hybrid one.

With the successive use of MEM and HMM, our system is able to maintain a desirable performance regardless of the size of the training data. Table 2 below shows the F-measure scores for various domains based on <PER>, <LOC> and <ORG>, whereas Fig. 1-3 highlight the superiority of using the hybrid approach than single statistical model.

Domain		Category		
		PER	LOC	ORG
D1	Applied Science	96.15	95.49	94.19
D2	Arts	96.41	92.68	84.75
D3	Belief and Thought	94.42	92.71	86.28
D4	Commerce	93.75	94.12	87.72
D5	Imaginative	88.00	76.19	70.59
D6	Leisure	94.87	93.11	90.72
D7	Natural Science	91.76	92.44	89.56
D8	Social Science	95.37	94.90	89.00
D9	World Affairs	93.58	93.88	90.60

Table 2: F-measure score in percentage.

The results above demonstrated the portability of our system to work in different domains. In overall, we are able to achieve F-measure scores above 90% in most of the domains. For some commonly used domains, the system can even obtain a high-flying performance with F-measure scores well over 95%.

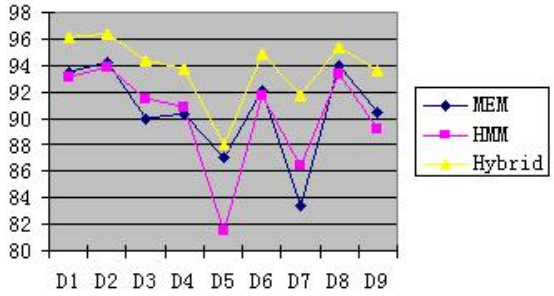

Figure 1: F-measure scores for <PER> based on different approaches.

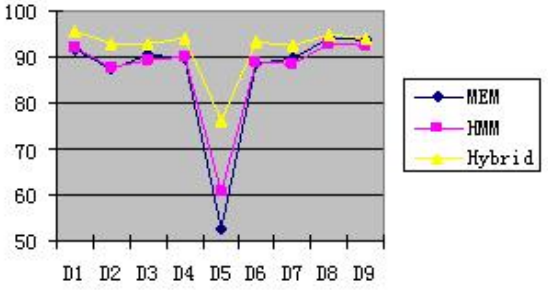

Figure 2: F-measure scores for <LOC> based on different approaches.

On the three desired categories, <PER> has achieved the highest F-measure with an average of 93.81%. It is followed by <LOC> with average of 91.72%. <ORG> is still the most ambiguous category, with only an average of 87.05% being achieved.

From our results, we notice that our system performed poorly on the 'Imaginative' domain that contains fairy tales and poetries. Due to the nature of this domain, we believe that the contextual evidence for the training features is hard to find, thus resulting in the lower scores. If we leave the 'Imaginative' domain out of the framework, the overall F-measure score can be significantly higher, with PER 94.54%, LOC 93.67% and ORG 89.10% respectively.

5 Conclusions

In this paper, we presented a hybrid machine learning approach that used MEM and HMM successively. We showed that with the preliminary data training through MEM and appropriate classifier for error correction in the final recognition process through HMM, the performance of our proposed NER system can be greatly

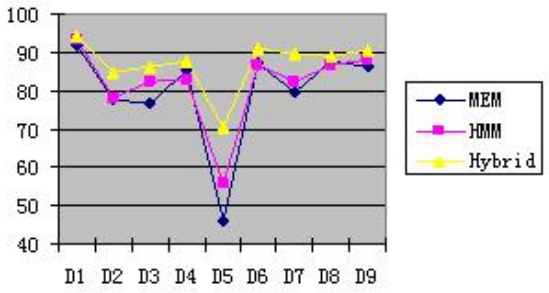

Figure 3: F-measure scores for <ORG> based on different approaches.

enhanced as compared to using only a single statistical model. Moreover, our system is also able to adapt to different domains without human intervention, and maintain desirable performance regardless of the size of the training corpus.

While our experimental results have been quite positive, we reckon that our proposed approach is still fairly immature. Much work needs to be done to make the performance of our system more robust.

For future work, we would like to see how our NER system can be trained on corpora with foreign languages such as the Malay language or the Chinese language. Meanwhile, we are also interested to see how more sophisticated features can be incorporated to improve the performance of the system further.

References

[1] Bender, O., Och, F. J., and Ney, H. Maximum entropy models for named entity recognition. In *Proceedings of the 7th Conference on Computational Natural Language Learning (CoNLL '03)*, pages 148–151, 2003.

[2] Bikel, D. M., Schwartz, R. L., and Weischedel, R. M. An algorithm that learns what's in a name. *Machine Learning*, 34(1-3):211–231, 1999.

[3] Borthwick, A., Sterling, J., Agichten, E., and Grisham, R. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In *Proceedings of the 6th Workshop on Very Large Corpora*. Association for Computational Linguistics, 1998.

[4] Chieu, H. L. and Ng, H. T. Named entity recognition: A maximum entropy approach using global information. In *Proceedings of the 19th Interna-*

tional Conference on Computational Linguistics (COLING '02), 2002.

[5] Chinchor, N. *MUC-6 Named Entity Task Definition (Version 2.1)*. MUC-6, 1995.

[6] Chinchor, N. *MUC-7 Named Entity Task Definition (Version 3.5)*. MUC-7, 1998.

[7] Krupka, G. R. and Hausman, K. Isoquest inc: Description of the netowl text extraction system as used for muc-7. In *Proceedings of the 7th Message Understanding Conference (MUC-7)*, 1998.

[8] Maynard, D., Tablan, V., Ursu, C., Cunningham, H., and Wilks, Y. Named entity recognition from diverse text types. In *Recent Advances in Natural Language Processing 2001 Conference*, 2001.

[9] Mikheev, A., Moens, M., and Grover, C. Named entity recognition without gazetteers. In *Proceedings of the 9th European Chapter of the Association of Computational Linguistics (EACL '99)*, pages 1–8, 1999.

[10] Roche, E. and Schabes, Y. *Finite-State Language Processing*. The MIT Press, 1997.

[11] Sekine, S., Grishman, R., and Shinnou, H. A decision tree method for finding and classifying names in japanese texts. In *Proceedings of the 6th Workshop on Very Large Corpora*, 1998.

[12] Seon, C., Ko, Y., Kim, J., and Seo, J. Named entity recognition using machine learning methods and pattern-selection rules. In *Proceedings of the Natural Language Processing Pacific Rim Symposium (NLPRS '01)*, pages 229–236, 2001.

[13] Srihari, R. N. and Li, W. A hybrid approach for named entity and sub-type tagging. In *Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP '00)*, pages 247–254, 2000.

[14] Viterbi, A. J. Error bounds for convolutional codes and an asympotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, 1967.

[15] Wakao, T., Gaizauskas, R., and Wilks, Y. Evaluation of an algorithm for the recognition and classification of proper names. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING '96)*, pages 418–423, 1996.

[16] Zhou, G. D. and Su, J. Named entity recognition using a hmm-based chunk tagger. In *Proceedings of the 40th Annual meeting of the Association for Computational Linguistics (ACL '02)*, pages 473–480, 2002.