A Hybrid Approach for Named Entity and Sub-Type Tagging*

Rohini Srihari
Cymfony Net, Inc.
5500 Main Street
Williamsville, NY 14260
rohini@cymfony.com

Cheng Niu and Wei Li
Cymfony Net, Inc.
5500 Main Street
Williamsville, NY 14260
chengniu@cymfony.com
wei@cymfony.com

Abstract

This paper presents a hybrid approach for named entity (NE) tagging which combines Maximum Entropy Model (MaxEnt), Hidden Markov Model (HMM) and handcrafted grammatical rules. Each has innate strengths and weaknesses; the combination results in a very high precision tagger. MaxEnt includes external gazetteers in the system. Sub-category generation is also discussed.

Introduction

Named entity (NE) tagging is a task in which location names, person names, organization names, monetary amounts, time and percentage expressions are recognized and classified in unformatted text documents. This task provides important semantic information, and is a critical first step in any information extraction system.

Intense research has been focused on improving NE tagging accuracy using several different techniques. These include rule-based systems [Krupka 1998], Hidden Markov Models (HMM) [Bikel et al. 1997] and Maximum Entropy Models (MaxEnt) [Borthwick 1998]. A system based on manual rules may provide the best performance; however these require painstaking intense skilled labor. Furthermore, shifting domains involves significant effort and may result in performance degradation. The strength of HMM models lie in their capacity for modeling local contextual information. HMMs have been widely used in continuous speech recognition, part-of-speech tagging, OCR, etc., and are generally regarded as the most successful statistical modelling paradigm in these domains. MaxEnt is a powerful tool to be used in situations where several ambiguous information sources need to be combined. Since statistical techniques such as HMM are only as good as the data they are trained on, they are required to use back-off models to compensate for unreliable statistics. In contrast to empirical back-off models used in HMMs, MaxEnt provides a systematic method by which a statistical model consistent with all obtained knowledge can be trained. [Borthwick et al. 1998] discuss a technique for combining the output of several NE taggers in a black box fashion by using MaxEnt. They demonstrate the superior performance of this system; however, the system is computationally inefficient since many taggers need to be run.

In this paper we propose a hybrid method for NE tagging which combines all the modelling techniques mentioned above. NE tagging is a complex task and high-performance systems are required in order to be practically usable. Furthermore, the task demonstrates characteristics that can be exploited by all three techniques. For example, time and monetary expressions are fairly predictable and hence processed most efficiently with handcrafted grammar rules. Name, location and organization entities are highly variable and thus lend themselves to statistical training algorithms such as HMMs. Finally, many conflicting pieces of information regarding the class of a tag are

* This work was supported in part by the SBIR grant F30602-98-C-0043 from Air Force Research Laboratory (AFRL)/IFED.

frequently present. This includes information from less than perfect gazetteers. For this, a MaxEnt approach works well in utilizing diverse sources of information in determining the final tag. The structure of our system is shown in Figure 1.

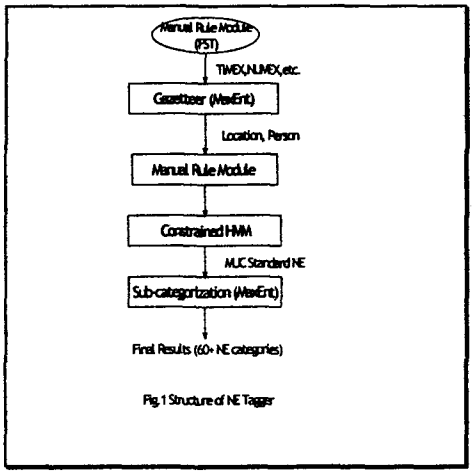

Fig.1 Structure of NE Tagger

The first module is a rule-based tagger containing pattern match rules, or templates, for time, date, percentage, and monetary expressions. These tags include the standard MUC tags [Chinchor 1998], as well as several other sub-categories defined by our organization. More details concerning the sub-categories are presented later. The pattern matcher is based on Finite State Transducer (FST) technology [Roches & Schabes 1997] that has been implemented in-house. The subsequent modules are focused on location, person and organization names. The second module assigns tentative person and location tags based on external person and location gazetteers. Rather than relying on simple lookup of the gazetteer which is very error prone, this module employs MaxEnt to build a statistical model that incorporates gazetteers with common contextual information. The core module of the system is a bigram-based HMM [Bikel et al.1997]. Rules designed to correct errors in NE segmentation are incorporated into a constrained HMM network. These rules serve as constraints on the HMM model and enable it to utilize information beyond bigrams and remove obvious errors due to the limitation of the training corpus. HMM generates the standard MUC tags, person, location and organization. Based on MaxEnt, the last module derives sub-categories such as city, airport, government, etc. from the basic tags.

Section 1 describes the FST rule module. Section 2 discusses combining gazetteer information using MaxEnt. The constrained HMM is described in Section 3. Section 4 discusses sub-type generation by MaxEnt. The experimental results and conclusion are presented finally.

1 FST-based Pattern Matching Rules for Textract NE

The most attractive feature of the FST (Finite State Transducer) formalism lies in its superior time and space efficiency [Mohri 1997] [Roche & Schabes 1997]. Applying a deterministic FST depends linearly only on the input size of the text. Our experiments also show that an FST rule system is extraordinarily robust. In addition, it has been verified by many research programs [Krupka & Hausman 1998] [Hobbs 1993] [Silberztein 1998] [Srihari 1998] [Li & Srihari 2000], that FST is also a convenient tool for capturing linguistic phenomena, especially for idioms and semi-productive expressions like time NEs and numerical NEs.

The rules which we have currently implemented include a grammar for temporal expressions (time, date, duration, frequency, age, etc.), a grammar for numerical expressions (money, percentage, length, weight, etc.), and a grammar for other non-MUC NEs (e.g. contact information like address, email).

The following sample pattern rules give an idea of what our NE grammars look like. These rules capture typical US addresses, like: *5500 Main St., Williamsville, NY14221; 12345 Xyz Avenue, Apt. 678, Los Angeles, CA98765-4321.* The following notation is used: @ for macro; | for logical OR; + for one or more; (...) for optionality.

```
0_9 =       0|1|2|3|4|5|6|7|8|9
number =    @0_9+
uppercase = A|B|C|D|E|F|G|H|I|J|
            K|L|M|N|O|P|Q|R|S|T
            U|V|W|X|Y|Z
```

```
lowercase =     a | b | c | d | e | f | g | h | i | j | k | l |
                m | n | o | p | q | r | s | t | u |v | w |
                x | y | z
letter =        @uppercase | @lowercase
word =          @letter+
delimiter =     (",") " "+
zip =    @0_9 @0_9 @0_9 @0_9 @0_9
         ("-" @0_9 @0_9 @0_9 @0_9)
street =        [[St | ST | Rd | RD | Dr | DR |
                Ave | AVE ] (".")] | Street |
                Road | Drive | Avenue
city =   @word (@word)
state =  @uppercase (".") @uppercase (".")
us =     USA | U.S.A | US | U.S. |
         (The) United States (of America)
street_addr =   @number @word @street
apt_addr =      [APT (".") | Apt (".") |
                Apartment] @number
local_addr =    @street_addr
                (@delimiter @apt_addr)
address =       @local_addr
                @delimiter @city
                @delimiter @state  @zip
                (@delimiter @us)
```

Our work is similar to the research on FST local grammars at LADL/University Paris VII [Silberztein 1998][1], but that research was not turned into a functional rule based NE system.

The rules in our NE grammars cover expressions with very predictable patterns. They were designed to address the weaknesses of our statistical NE tagger. For example, the following missings (underlined) and mistagging originally made by our statistical NE tagger have all been correctly identified by our temporal NE grammar.

began <TIMEX TYPE="DATE">Dec. 15, the</TIMEX> space agency
on Jan. 28, <TIMEX TYPE="DATE">1986</TIMEX>,
in September <TIMEX TYPE="DATE">1994</TIMEX>on <TIMEX TYPE="TIME">Saturday at</TIMEX> 2:42 a.m. ES<ENAMEX TYPE="PERSON">T.</ENAMEX>
He left the United States in <TIMEX TYPE="DATE">1984 and</TIMEX> moved
in early <TIMEX TYPE="DATE">1962 and</TIMEX>
in <TIMEX TYPE="DATE">1987 the Bonn</TIMEX> government ruled

[1] They have made public their research results at their website (http://www.ladl.jussieu.fr/index.html), including a grammar for certain temporal expressions and a grammar for stock exchange sub-language.

2 Incorporating Gazetteers with the Maximum Entropy Model

We use two gazetteers in our system, one for person and one for location. The person gazetteer consists of 3,000 male names, 5,000 female names and 14,000 family names. The location gazetteer consists of 250,000 location names with their categories such as CITY, PROVINCE, COUNTRY, AIRPORT, etc. The containing and being-contained relationship among locations is also provided.

The following is a sample line in the location gazetteer, which denotes "Aberdeen" as a city in "California", and "California" as a province of "United States".

Aberdeen (CITY) California (PROVINCE) United States (COUNTRY)

Although gazetteers obviously contain useful name entity information, a straightforward word match approach may even degrade the system performance since the information from gazetteers is too ambiguous. There are a lot of common words that exist in the gazetteers, such as "I", "A", "Friday", "June", "Friendship", etc. Also, there is large overlap between person names and location names, such as "Clinton", "Jordan", etc.

Here we propose a machine learning approach to incorporate the gazetteer information with other common contextual information based on MaxEnt. Using MaxEnt, the system may learn under what situation the occurrence in gazetteers is a reliable evidence for a name entity.

We first define "LFEATURE" based on occurrence in the location gazetteer as follows:

COUNTRY (country name)
USSTATE (US state name)
MULTITOKEN (a location name consisting of multiple tokens)
BIGCITY (a location name occurring in OXFD dictionary)
COEXIST (where COEXIST(A,B) is true iff A and B are in the same US state, or in the same foreign country)
OTHER

There is precedence from the first LFEATURE to the last one. Each token in the input document is assigned a unique "LFEATURE". We also define "NFEATURE" based on occurrence in the name gazetteer as follows:

FAMILY (family name)
MALE (male name)
FEMALE (female name)
FAMILYANDMALE (family and male name)
FAMILYANDFEMALE (family and female name)
OTHER

With these two extra features, every token in the document is regarded as a three-component vector (word, LFEATURE, NFEATURE). We can build a statistical model to evaluate the conditional probability based on these contextual and gazetteer features. Here "tag" represents one of the three possible tags (Person, Location, Other), and history represents any possible contextual history. Generally, we have:

$$p(tag \mid history) = \frac{p(tag, history)}{\sum_{tag'} p(tag', history)} \qquad (1)$$

A maximum entropy solution for probability has the form [Rosenfeld 1994] [Ratnaparkhi 1998]

$$p(tag, history) = \frac{\prod_i \alpha_i^{f_i(history, tag)}}{Z(history)} \qquad (2)$$

$$Z(history) = \sum_{tag'} \prod_i \alpha_i^{f_i(history, tag')} \qquad (3)$$

where $f_i(history, tag)$ are binary-valued feature functions that are dependent on whether the feature is applicable to the current contextual history. Here is an example of our feature function:

$$f(history, tag) = \begin{cases} 1 & \text{if current token is a country name, and tag is location} \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

In (2) and (3) α_i are weights associated to feature functions.

The weight evaluation scheme is as follows: We first compute the average value of each feature function according to a specific training corpus. The obtained average observations are set as constraints, and the Improved Iterative Scaling (IIS) algorithm [Pietra et al. 1995] is employed to evaluate the weights. The resulting probability distribution (2) possesses the maximum entropy among all the probability distributions consistent with the constraints imposed by feature function average values.

In the training stage, our gazetteer module contains two sub-modules: feature function induction and weight evaluation [Pietra et al. 1995]. The structure is shown in Figure 2.

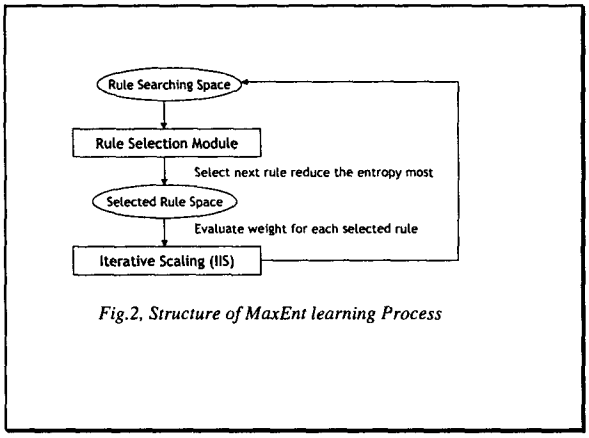

Fig.2, Structure of MaxEnt learning Process

We predefine twenty-four feature function templates. The following are some examples and others have similar structures:

$$f(history, tag) = \begin{cases} 1 & \text{if LFEATURE = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if NFEATURE = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if current word = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if previous word = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if following word = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

where the symbol "_" denotes any possible values which may be inserted into that field. Different fields will be filled different values.

Then, using a training corpus containing 230,000 tokens, we set up a feature function candidate space based on the feature function templates. The "Feature Function Induction Module" can select next feature function that reduces the Kullback-Leibler divergence the most [Pietra et al. 1995]. To make the weight evaluation computation tractable at the feature function induction stage, when trying a new feature function, all previous computed weights are held constant, and we only fit one new constraint that is imposed by the candidate feature function. Once the next feature function is selected, we recalculate the weights by IIS to satisfy all the constraints, and thus obtain the next tentative probability. The feature function induction module will stop when the Log-likelihood gain is less than a pre-set threshold.

The gazetteer module recognizes the person and location names in the document despite the fact that some of them may be embedded in an organization name. For example, "New York Fire Department" may be tagged as <LOCATION> New York </NE> Fire Department. In the input stream for HMM, each token being tagged as location is accordingly transformed into one of the built-in tokens "CITY","PROVINCE", "COUNTRY". The HMM may group "CITY Fire Department" into an organization name. A similar technique is applied for person names.

Since the tagged tokens from the gazetteer module are regarded by later modules as either person or location names, we require that the current module generates results with the highest possible precision. For each tagged token we will compute the entropy of the answer. If the entropy is higher than a pre-set threshold, the system will not be certain enough about the answer, and the word will be untagged. The missed location or person names may be recognized by the following HMM module.

3 Improving NE Segmentation through constrained HMM

Our original HMM is similar to the Nymble [Bikel et al. 1997] system that is based on bigram statistics. To correct some of the leading errors, we incorporate manual segmentation rules with HMM. These syntactic rules may provide information beyond bigram and balance the limitation of the training corpus.

Our manual rules focus on improving the NE segmentation. For example, in the token sequence "College of William and Mary", we have rules based on global sequence checking to determine if the words "and" or "of" are common words or parts of organization name.

The output of the rules are some constraints on the HMM transition network, such as "same tags for tokens A, B", or "common word for token A". The Viterbi algorithm will select the optimized path that is consistent with such constraints.

The manual rules are divided into three categories: (i) preposition disambiguation, (ii) spurious capitalized word disambiguation, and (iii) spurious NE sequence disambiguation.

The rules of preposition disambiguation are responsible for determination of boundaries involving prepositions ("of", "and", "'s", etc.). For example, for the sequence "A of B", we have the following rule: A and B have same tags if the lowercase of A and B both occur in OXFD dictionary. A "global word sequence checking" [Mikheev, 1999] is also employed. For the sequence "Sprint and MCI", we search the document globally. If the word "Sprint" or

"MCI" occurs individually somewhere else, we mark "and" as a common word.

The rules of spurious capitalized word disambiguation are designed to recognize the first word in the sentence. If the first word is unknown in the training corpus, but occurs in OXFD as a common word in lowercase, HHM's unknown word model may be not accurate enough. The rules in the following paragraph are designed to treat such a situation.

If the second word of the same sentence is in lowercase, the first word is tagged as a common word since it never occurs as an isolated NE token in the training corpus unless it has been recognized as a NE elsewhere in the document. If the second word is capitalized, we will check globally if the same sequence occurs somewhere else. If so, the HMM is constrained to assign the same tag to the two tokens. Otherwise, the capitalized token is tagged as a common word.

The rules of spurious NE sequence disambiguation are responsible for finding spurious NE output from HMM, adding constraints, and re-computing NE by HMM. For example, in a sequence "Person Organization", we will require the same output tag for these two tokens and run HMM again.

4 NE Sub-Type Tagging using Maximum Entropy Model

The output document from constrained HMM contains MUC-standard NE tags such as person, location and organization. However, for a real information extraction system, the MUC-standard NE tag may not be enough and further detailed NE information might be necessary. We have predefined the following sub-types for person, location and organization:

Person: Military Person
Religious Person
Man
Woman
Location: City
Province
Country
Continent
Lake
River
Mountain
Road
Region
District
Airport
Organization: Company
Government
Army
School
Association
Mass Medium

If a NE is not covered by any of the above sub-categories, it should remain a MUC-standard tag. Obviously, the sub-categorization requires much more information beyond bigram than MUC-standard tagging. For example, it is hard to recognize CNN as a Mass Media company by bigram if the token "CNN" never occurs in the training corpus. External gazetteer information is critical for some sub-category recognition, and trigger word models may also play an important role.

With such considerations, we use the Maximum entropy model for sub-categorization, since MaxEnt is powerful enough to incorporate into the system gazetteer or other information sources which might become available at some later time.

Similar to the gazetteer module in Section 2, the sub-categorization module in the training stage contains two sub-modules, (i) feature function induction and (ii) weight evaluation. We have the following seven feature function templates:

$$f(history, tag) = \begin{cases} 1 & \text{if MUC\_tag = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if MUC\_tag = \_, LFEATURE = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if contain\_word(\_), MUC\_tag(history) = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if Previous\_Word = \_, MUC\_tag = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if following\_Word = \_, MUC\_tag = \_, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if MUC\_tag = \_, contain\_male\_name, and tag = \_} \\ 0 & \text{else} \end{cases}$$

$$f(history, tag) = \begin{cases} 1 & \text{if MUC\_tag = \_, contain\_female\_name, and tag = \_} \\ 0 & \text{else} \end{cases}$$

We have trained 1,000 feature functions by the feature function induction module according to the above templates.

Because much more external gazetteer information is necessary for the sub-categorization and there is an overlap between male and female name gazetteers, the result from the current MaxEnt module is not sufficiently accurate. Therefore, a conservative strategy has been applied. If the entropy of the output answer is higher than a threshold, we will back-off to the MUC-standard tags. Unlike MUC NE categories, local contextual information is not sufficient for sub-categorization. In the future more external gazetteers focusing on recognition of government, company, army, etc. will be incorporated into our system. And we are considering using trigger words [Rosenfeld, 1994] to recognize some sub-categories. For example, "psalms" may be a trigger word for "religious person", and "Navy" may be a trigger word for "military person".

Experiment and Conclusion

We have tested our system on MUC-7 dry run data; this data consists of 22,000 words and represents articles from The New York Times. Since a key was provided with the data, it is possible to properly evaluate the performance of our NE tagger. The scoring program computes both the precision and recall, and combines these two measures into f-measure as the weighted harmonic mean [Chinchor, 1998]. The formulas are as follows:

$$\text{Precision} = \frac{\text{number of correct responses}}{\text{number responses}}$$

$$\text{Recall} = \frac{\text{number of correct responses}}{\text{number correct in key}}$$

$$\text{F} = \frac{(\beta^2 + 1)\text{Precision} * \text{Recall}}{(\beta^2 \text{Recall}) + \text{Precision}}$$

The score of our system is as follows:

	Recall	Precision
Organization	95	95
Person	96	93
Location	96	94
Date	92	91
Time	92	91
Money	100	86
Percentage	100	75

F-measure =93.39

If the gazetteer module is removed from our system, and the constrained HMM is restored to the standard HMM, the f-measures for person, location, and organization are as follows:

	Recall	Precision
Organization	94	92
Person	95	91
Location	95	92

Obviously, our gazetteer model and constrained HMM have greatly increased the system accuracy on the recognition of persons, locations, and organizations. Currently, there are some errors in our gazetteers. Some common words such as "Changes", "USER", "Administrator", etc. are mistakenly included in the person name gazetteer. Also, too many person names are included into the location gazetteer. By cleaning up the gazetteers, we can continue improving the precision on person name and locations.

We also ran our NE tagger on the formal test files of MUC-7. The following are the results:

	Recall	Precision
Person	92	95
Organization	85	86
Location	90	92
Date	95	85

Time	79	72
Money	95	82
Percentage	97	80
Overall F-measure	89	

There is some performance degradation in the formal test. This decrease is because that the formal test is focused on satellite and rocket domains in which our system has not been trained. There are some person/location names used as spacecraft or robot names (ex. Mir, Alvin, Columbia...), and there are many high-tech company names which do not occur in our HMM training corpus. Since the finding of organization names totally relies on the HMM model, it suffers most from domain shift (10% degradation). This difference implies that gazetteer information may be useful in overcoming the domain dependency.

This paper has demonstrated improved performance in an NE tagger by combining symbolic and statistical approaches. MaxEnt has been demonstrated to be a viable technique for integrating diverse sources of information and has been used in NE sub-categorization.

References

G. R Krupka and K. Hausman, "IsoQuest Inc: Description of the NetOwl Text Extraction System as used for MUC-7" in Proceedings of Seventh Machine Understanding Conference (MUC-7) (1998)

D. M. Bikel, "Nymble: a high-performance learning name-finder" in Proceedings of the Fifth Conference on Applied Natural Language Processing, 1997, pp. 194-201, Morgan Kaufmann Publishers.

A. Borthwick, et al., Description of the MENE named Entity System, In Proceedings of the Seventh Machine Understanding Conference (MUC-7) (1998)

R. Rosenfeld, Adaptive Statistical language Modeling, PHD thesis, Carnegie Mellon University, (1994)

A. Ratnaparkhi, Maximum Entropy Models for Natural Language Ambiguity resolution, PHD thesis, Univ. of Pennsylvania, (1998)

S. D. Pietra, Vincent Della Pietra, and John Lafferty, Inducing Features of Random Fields, Tech Report, Carnegie Mellon University, (1995)

A. Mikheev, A Knowledge-free Method for Capitalized Word Disambiguation, in *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, (1999), pp. 159-166

J. R. Hobbs, 1993. FASTUS: A System for Extracting Information from Text, Proceedings of the DARPA workshop on Human Language Technology", Princeton, NJ, pp. 133-137.

Emmanuel Roche & Yves Schabes, 1997. Finite-State Language Processing, The MIT Press, Cambridge, MA.

Li, W & Srihari, R. 2000. *Flexible Information Extraction Learning Algorithm*, Final Technical Report, Air Force Research Laboratory, Rome Research Site, New York

M. Silberztein, 1998. Tutorial Notes: Finite State Processing with INTEX, COLING-ACL'98, Montreal (also available at http://www.ladl.jussieu.fr)

M. Mohri,. 1997. Finite-State Transducers in Language and Speech Processing, Computational Linguistics, Vol. 23, No. 2, pp. 269-311.

R. Srihari, 1998. A Domain Independent Event Extraction Toolkit, AFRL-IF-RS-TR-1998-152 Final Technical Report, Air Force Research Laboratory, Rome Research Site, New York