Association Rules Mining for Name Entity Recognition

Indra Budi
Computer Science Faculty, University of Indonesia, indra@cs.ui.ac.id

Stéphane Bressan
School of Computing, National University of Singapore, steph@nus.edu.sg

Abstract

We propose a new name entity class extraction method based on association rules. We evaluate and compare the performance of our method with the state of the art maximum entropy method. We show that our method consistently yields a higher precision at a competitive level of recall. This result makes our method particularly suitable for tasks whose requirements emphasize the quality rather than the quantity of results.

1. Introduction

The work presented in this paper[1] is part of a larger effort to develop information retrieval and linguistic systems, tools, and techniques for an Indonesian Digital Library (see [12,16]). The application that motivates this research requires that semantic structures in XML or RDF be extracted from and superimposed on a corpus of documents written in the Indonesian language. Another motivation is the Semantic Web (abstract representation of data on the World Wide Web) can be achieved by extracting semantic structure from those documents. The natural first step of this project consists in identifying name entities from the texts in the corpus.

Name Entity Recognition (NER) is an information extraction task that is concerned with the recognition and classification of name entity from free text [11]. Name entities classes are, for instance, locations, person named, organization named, dates, and money amounts. To terms and expressions in the text correspond the entities they represent. For example, in the sentence: *"British Foreign Office Minister O'Brien (right) and President Megawati pose for photographers at the Palace",* a name entity recognition process looking for named persons and locations would identify the two persons *O'Brien* and *Megawati* and the location *Palace*. This recognition can be based on a variety of features of the terms, the sentence, the text and its syntax and could leverage external sources of information such as thesauri and dictionaries, for instance. In the example, a system may have applied a simple rule guessing that the capitalized words directly following the terms '*President*' or '*Minister*' are names of persons.

In this paper, we propose a method for name entity class recognition based on such rules in the form of association rules. The association rules defining the patterns of syntax and term features potentially defining the classes are mined from a training set of documents.

The rest of the paper is organized as follow. We present and discuss some background and related work on name entity class recognition in the next section. The method is presented in section 3 in which we detail the learning or mining phase and the testing and tagging phase. We evaluate and compare our method with the state of the art maximum entropy method [7] in section 4. Finally we conclude and identify the next steps of our research.

2. Background and Related Work

The first family of approaches to name entity recognition relies on the hand crafting of models and techniques for the recognition of entity classes. Generally the models consist of a set of patterns using grammatical (e.g. part of speech), syntactic (e.g. word precedence) and orthographic features (e.g. capitalization) in combination with dictionaries and thesauri. An example pattern in this type of system is the one we have suggested in our introductory example: "If a proper noun follows a person's title, then the proper noun is a person's name".

In this family of approaches Appelt et al. propose a name identification system based on carefully hand-crafted regular expression [2,3], while Iwanska [13] uses extensive specialized resources such as gazetteers, and white and yellow pages. Morgan, for the same purpose, uses a highly sophisticated linguistic analysis [14].

These approaches are relying on manually coded rules and manually compiled corpora. They often yield prohibitive development and maintenance costs. Furthermore, for cost reduction and effectiveness reasons, they are often domain and language specific and do not necessarily adapt well to new domains and languages.

The alternative to hand-crafted approaches is the use of data mining, knowledge discovery and machine learning techniques. Both Sekine [15] and Bennett [4] propose name identification systems based on decision trees. The decision trees in both approaches use such

[1] An extended version of this paper is available as [8]

Proceedings of the Fourth International Conference on Web Information Systems Engineering (WISE'03)
0-7695-1999-7/03 $17.00 © 2003 **IEEE**

features as part-of-speech, character, as well as dictionaries. Sekine's approach uses a single decision tree to compute the probability of a term to represent a given class while Bennet's approach combines multiple decision trees. Bikel in the popular Nymble system proposes a method based on the hidden markov model [6].

The maximum entropy approach to NER relies on the general maximum entropy technique for estimating probability distribution. Borthwick [7] described a word identification system built around a maximum entropy framework. The system uses a variety of knowledge sources, such as orthographic, lexical, section and dictionary features, to make tagging decisions. Chieu [9] uses maximum entropy and combines the local features we have mentioned before with features global to the text such as abbreviations in subsequent sentences.

3. Association Rule-based NER

3.1. Mining Association Rules for NER

Association rules and association rule mining [1] has received much attention in the last decade in the database and data mining community. The model and techniques have found many applications in a variety of domains. We mention for illustration results on Web caching [5] and on query expansion in information retrieval [17] by one of the author.

An traditional association rule is a relationship of the form: $X \Rightarrow Y$, where X and Y are sets of items from the dataset to be studied. Each association rule is assigned a support factor and a confidence factor. The support is the ratio of the number of items in X and Y over the total number of items; The confidence is the ratio of the number of items in X and Y over the number items in X. The mining of association rules consists in extracting from the databases all such rules with support and confidence greater than or equal to a user-specified support and confidence.

In the name extraction task the datasets are documents which are sequences of terms with features and name classes. We use the set of features proposed by Bikel in [6]. We consider the seven name classes considered in the standard MUC-7 benchmark (see [10]) for the test on MUC-7 (English) corpus: location named, person named, organization named, dates, times, monetary amount, and percentages, and three name classes for the Indonesian corpus: location named, person named, organization named.

The items we consider are occurrences of terms. However the sets X and Y can be described in terms of terms, sequences of terms, features and name classes. In practice Y is the name class we wish to predict. Among all the possible forms for X, after informal empirical tests, we settled to consider three types of rules. Let us consider a sequence of terms $<t_1, t_2>$, where f_2 is feature of t_2 and nc_2 is the name class of t_2, we consider the following three types of association rules:

1. $<t_2> \Rightarrow nc_2$, *(support, confidence)*
2. $<t_1,t_2> \Rightarrow nc_2$, *(support, confidence)*
3. $<t_1, f_2> \Rightarrow nc_2$, *(support, confidence)*

We call rules of type 1 *dictionary rules*, rules of type 2 *bigram rules*, and rules of type 3 *feature rules*.

Let us consider the example sentence "*Prof. Hasibuan conducted a lecture on information retrieval*". In a training corpus in which name classes are given, the annotations of the corpus indicate that the term "*Hasibuan*" is name class of *person*.

We produce a dictionary rule of the form:

$\langle$Hasibuan$\rangle$ => person_named(Hasibuan)

with support and confidence depending on the number of occurrences of the term "*Hasibuan*" and the number of occurences of the term "*Hasibuan*" labelled in this entity class.

We produce a bigram rule of the form:

$\langle$Prof., Hasibuan$\rangle$ => person_named (Hasibuan)

with support and confidence depending on the number of occurrences of the expression "*Prof. Hasibuan*" and the term "*Hasibuan*" labelled in this name class.

We produce a feature rule of the form:

$\langle$Prof., Capitalized_word(X)$\rangle$ => person_named(X)

with support and confidence depending on the occurrences of the expression "*Prof. X*" with X labelled in this feature and name classes

3.2. Using Association Rules for NER

The mined rules are considered for the name entity recognition task if their support and confidence is above user defined thresholds.

We use the mined rules by type -dictionary, bigram, or feature- independently or combined. For every pair of terms in the text, the name entity recognition association rule-based algorithm, presented in figure 3.2, determines the rule with minimum support and highest confidence to be used. Ties, which are rare, are broken with a random choice. Not so rare is the case for which no rule is available. In that situation, the special class *not-a-name* is assigned to the term.

In this paper, although we have informally experimented with various combinations of the rules, we report the results of four interesting combinations of the three types of rules mined: the bigram rules alone, the combined bigram and dictionary rules, the feature rules alone and the combined feature and dictionary rules.

4. Experimental Results

In order to evaluate the effectiveness of our method we use the standard name entity recognition corpus, the MUC-7 corpus (see [9]). The corpus contains news articles in the English language in which terms representing name entities of seven classes (see section 3) have been labeled. We use a training set of 200 articles to learn the association rules and a testing set of 100 articles. The algorithm is implemented in C++ .

```
for every pair of terms ⟨t1, t2⟩
   find the set R of rules X => nc
    such that ⟨t1, t2⟩ matches X
    and support and confidence are above threshold
   if R is not empty
   then
    Chose in R the rule X => nc with highest confidence
    Assign nc as the name class of t2
   Else
    Assign not-a-name as name class of t2
endfor
```

Figure 3.2 Recognition algorithm

We also ran experiments on a corpus of news articles in the Indonesian language. The corpus consists of 55 manually labelled news articles collected for eight days, between April 2nd 2001 and April 10th 2001, from the online version of the Indonesian newspaper Kompas (http://www.kompas.com) (see [16]). We took the articles vary in size from 201 to 1532 terms. As mentioned above, we use three name classes.

The effectiveness of the method is measured in terms of recall and precision. Recall is defined as the number of correct responses divided by number of answers. Precision is defined as the number of correct responses divided by the number of responses. A response is a term labeled by the algorithm with a name class. An answer is the name class of the term as labeled in the corpus. A response is correct if it corresponds to an answer.

In this series of experiment we compare our method with the maximum entropy method. A maximum entropy solution to NER is finding the probability of $p(f|h)$ for any f from the space of possible futures (seven name entities as MUC-7) and for every h from the space of possible histories. A "history" in maximum entropy is all of the conditioning data which enables to make a decision among the space of futures, in this term is token features.

For each type of association rules or combination of types of association rules that we use, we use the corresponding terms and features in the history of the maximum entropy method: one term for the dictionary method, two consecutive terms for the bigram method and a term and the feature of its successor for the feature method.

The maximum entropy method is implemented using the Java-based opennlp maximum entropy package (http://maxent.sourforge.net) on the same machine.

Table 4.2. Results of experiment with MUC-7

Rule	Rule Association		Maximum Entropy	
	Recall	Precision	Recall	Precision
Bigram	34.37	93.21	57.40	65.03
Feature	44.84	67.75	49.56	58.99
Bigram+Dict	60.44	89.59	53.72	69.48
Feature+Dict	66.34	83.43	43.70	60.89

Tables 4.1 and 4.2 report the results of the experiments on the MUC-7 and Kompas corpora, respectively. The Association rule based methods consistently yields a higher precision than the corresponding maximum entropy based method.

The dictionary rules can be used in combination with the bigram or the feature rules to significantly increase the recall (namely many terms occur without any particular contextual features). The various combinations of features also seem to impact consistently in both association rule and maximum entropy based methods.

For the Indonesian corpus, with only three name entity classes, the association rule based methods have difficulties to reach a competitive level of recall.

Table 4.2 Results of experiment with Kompas

Rule	Rule Association		Maximum Entropy	
	Recall	Precision	Recall	Precision
Bigram	28.45	86.52	52.53	72.14
Feature	51.58	77.65	58.39	70.01
Bigram+Dict	48.42	82.61	78.09	70.20
Feature+Dict	62.80	81.35	63.10	61.13

5. Conclusions and Future Work

We have presented a new name entity class recognition method based on association rules. We compared our method with the maximum entropy method. Our experiments showed that association rules consistently yield a higher precision than the maximum entropy method. In the English corpus, under the appropriate combination of types of rules it is possible to improve the recall so that the association rule method is strictly more effective that the maximum entropy.

The next step in our project is to devise a method to reconstruct structured elements from the elementary name entities identified. Our target language is XML. To illustrate our idea, let us consider the motivating example from which we wish to extract an XML document describing the meeting taking place:

"British Foreign Office Minister O'Brien (right) and President Megawati pose for photographers at the Palace."

Figure 5.1 contains the manually constructed XML we hope to obtain. In italic are highlighted the components that require global, ancillary, or external knowledge. Indeed, although, we expect similar methods (association rules and maximum entropy) can be used to learn the model of combination of elementary entities into complex elements, we also expect that global, ancillary, and external knowledge will be necessary such as lists of names of personalities (Mike O'Brien, Megawati Soekarnoputri), gazetteers (Jakarta is in Indonesia), document temporal and geographical context (Jakarta, 05/06/2003), etc.

```
<meeting>
 <date>05/06/2003</date format=europe>
 <location>
   <name>State Palace</name>
   <city>Jakarta</city>
   <country>Indonesia</country>
 </location>
 <participants>
   <person>
     <name>Megawati Soekarnoputri</name>
     <quality>President </quality>
     <country>Indonesia</country>
   </person>
   <person>
     <name>Mike O'Brien</name>
     <quality>Foreign Office Minister</quality>
     <country>Britain</country>
   </person>
 </participants>
</meeting>
```

Figure 5.1. Extracted XML data

References

[1] Agrawal, R., Tomasz I., and Arun S., "Mining Association Rule betweens Sets of Items in Large Databases", Proceeding of the 1993 ACM SIGMOID International Conference on Management of Data, Washington DC, 1993.

[2] Appelt, D., and et. al., "FASTUS: A finite state processor for information extraction from real-world text", Proceedings of IJCAI 1993.

[3] Appelt, D., and et. al., "SRI International FASTUS system MUC-6 test results and analysis", Proceedings of the MUC-6, NIST, Morgan-Kaufmann Publisher, Columbia, 1995.

[4] Bennett, S.W., Chinatsu A., and Craig L., "Learning to Tag Multilingual Texts through Observation", Proceedings of the Second Conference on Empirical Methods in Natural Language Processing, pp. 109-116, Somerset, New Jersey, 1997.

[5] Bin, L., Bressan, S., and Ooi, B.C., "Making Web Servers Pushier", In Proceedings of the Workshop on Web Usage Analysis and User Profiling, Springer Verlag, August 1999.

[6] Bikel, D., S. Miller, R. Schwartz and R. Weischedel, "NYMBLE: A High Performance Learning Name-Finder", Proceeding of the fifth Conference on Applied Natural Language Processing, pp 194-201, 1997.

[7] Borthwick, A., Sterling, J., Agichtein, E., and Grishman, R., "Exploiting diverse knowledge sources via maximum entropy in named entity recognition", Proceedings of the Sixth Workshop on Very Large Corpora, Montreal, Canada, 1998.

[8] Budi, I. and Bressan S., "Association Rules Mining for Name Entity Recognition", technical report number TRA6/03 of the School of Computing at the National University of Singapore, 2003.

[9] Chieu, H.L., and Hwee Tou Ng, "Named Entity Recognition: A Maximum Entropy Approach Using Global Information", Proceedings of the 19th International Conference on Computational Linguistics, 2002.

[10] Chinchor, N., Erica B., Lisa F., and Patty R., "1999 Named Entity Recognition Task Definition Version 1.4", MITRE Corporation and SAIC, August 1999.

[11] Grishman, R., "Information Extraction: Techniques and Challenges", Lecture Notes in Computer Science, Vol. 1299, Springer-Verlag, 1997.

[12] Indradjaja, L. and Bressan, S., "Automatic Learning of Stemming Rules for the Indonesian Language", In Proc. of the The 17th Pacific Asia Conference on Language, Information and Computation (PACLIC), 2003.

[13] Iwanska, L., Croll, M., Yoon, T., and Adams, M., "Wayne state university: Description of the UNO natural language processing system as used for MUC-6", In Proc. of the MUC-6, NIST, Morgan-Kaufmann Publishers, Columbia, 1995.

[14] Morgan, R., and et. al., "University of durham: Description of the LOLITA system as used for MUC-6" In Proc. of the MUC-6, NIST, Morgan-Kaufmann Publishers, Columbia, 1995.

[15] Sekine, S., Grishman, R., and Shinnou, H., "A Decision Tree Method for Finding and Classifying Names in Japanese Texts", Proceedings of the Sixth Workshop on Very Large Corpora, Montreal, Canada, 1998.

[16] Vinsensius, V., and Bressan, S., "Temu-Kembali Informasi untuk Dokumen-dokumen dalam Bahasa Indonesia", In Electronic proceedings of Indonesia DLN Seminar, 2001. (In Indonesian).

[17] Wei, J., Qin, Z., Bressan, S. and Ooi, B.C., "Mining Term Association Rules for Automatic Global Query Expansion: A Case Study with Topic 202 from TREC4", In Proceedings of the Americas Conference on Information Systems 2000 Data Mining and Information Retrieval in Business, 2000.