An experiment on automatic detection of Named Entity in Bangla

Abstract

Several preprocessing steps are necessary in many problems of automatic Natural Language Processing. One major step is named-entity detection, which is relatively simple in English, because such entities start with an uppercase character. For Indian scripts like Bangla, no such indicator exists and the problem of identification is more complex, especially for human names, which may be common nouns and adjectives. In this paper we have proposed a three-stage approach of named-entity detection. The stages are based on the use of named-entity dictionary, rules for named-entity and left-right co-occurrence statistics. Experimental results obtained on Anandabazar Patrika (Most popular Bangla newspaper) corpus are quite encouraging.

1. Introduction

The discipline of Natural Language Processing (NLP) concerns with the design and implementation of computational approaches that communicates with human using natural language. Name searching, matching and recognition have been active areas of research in the field of NLP and Information retrieval for a long period. This is an important problem since search queries are often proper nouns while all proper nouns cannot be exhaustively maintained in the dictionary for automatic identification. Moreover, human names may be picked from common noun and adjective words and hence dictionary-based syntactic information can confuse the Natural Language Processor in such a situation. Pet and other animal names, organization and place names, can also come from common nouns and adjectives. So, it becomes a non-trivial problem to automatically detect the named entity from a sentence.

This paper aims at attacking this problem for Bangla language, especially on the NE detection from newspaper text. Name recognition in English is easier since quite often the proper noun starts with an uppercase character. Bangla names cannot be identified by such case information because Bangla has single-case alphabet.

Some studies on Named Entity (NE) identification are reported in the literatures [1-4], [6-11] and [15]. The approaches mainly employ dictionary based and statistical tools for this purpose. Name searching in the context of information retrieval and query answering are also reported in the literature [5]. However, all these studies are done on non-Indian languages and to the best of our knowledge, no published study on Indian language Named Entity Recognition is available.

The NE identification approach presented here employs a combination of dictionary-based, rule-based and statistical information. The approach is employed here is elaborated in Section 2. In Section 3 the data collection and experimental setup is described. Tests have been made on a moderate size Anandabazar (most popular Bangla newspaper) news corpus. The experimental setup is described in Sec 3, while the results are presented in Section 4.

2. Proposed Named Entity (NE) detection approach

As mentioned before, our method of NE detection is a combination of dictionary-based, rule-based and statistical (n-gram based) approaches. In the dictionary based approach, we need a word-level morphological parser as well. The approaches are sequentially described here and demonstrated in Fig.1. However, at first, we describe some properties of named entity.

2.1. Properties of named entity:

If we look at a corpus of reasonable size from the perspective of NEs, we note that the words may belong to three categories: (a) words that almost never act as NE, (b) the words that almost always act as NE, (c) the words that sometimes act as names and sometimes as common nouns or adjectives. Words like *I, the, to, from, go* belong to category (a) while words like *India, Ganges, Paris, Himalayas* belong to category (b). Words like *Nirmal, Swapan, Rabi* belong to category (c). The English meaning of these words are clean, dream and sun, respectively. But they are used as names of persons in Bengali and thus can create problems for the NLP of Bangla language. In English, the names begin with uppercase.

Another point to note is that the named entity may be a single word or a multi word expression. The
multi-word names pose additional difficulty for automatic identification of NE. A multi-word may have a
component that alone is a name, like *England* in *New England* or it may consist of adjective and common noun, like *White House*. Such multi-words create additional problems for NE detection.

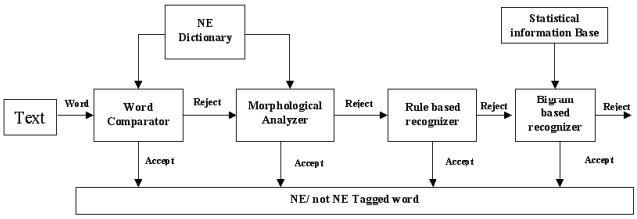

Fig 1. Flow chart for NE detection

2.2. Dictionary based NE detection:

If a dictionary is maintained where one of the above three category tags are attached to each word and if a word morphological analyzer is developed, then the combination of these two can act as a NE detector for any text file. The dictionary should be generated using a corpus of reasonable size, say 5-10 million words, as well as using conventional paper dictionary of say 50,000 root words. Normally, 10 million word corpus of Bangla contains between 100,000 and 200,000 surface words. A small fraction of these words belong to the set of NEs not found in the conventional dictionary. These words should be properly NE tagged and entered in the NE dictionary. The corpus provides important information about the inflectional nature of root words, which, in turn, helps in building the morphological analyzer. On the other hand, if we want to avoid building sophisticated morph analyzer, the most common inflected surface words of the corpus may also be included in the dictionary with the three tags described above. We have followed this procedure for our NE detection approach.

The detection algorithm will proceed as follows. Given a test word W, at first a match is searched in the NE tagged dictionary. If no match is found, W is rejected and the next word is considered for examination. But if a match occurs, we look at the tag of the matched word. If the tag is '*almost always NE*' then we declare this W as NE with weight 1. If the tag is '*almost never NE*' then W is declared as not NE (ie with weight 0). But if the tag is '*may or may not be NE*' then again W has to be rejected (say with weight 0.5), which makes this approach rather inefficient. To remedy this drawback, we employ some rule-based approach described in the next Section.

However, before sending to the rule-based module, each of the words with weight 0.5 is subject to morphological analysis. Here for each word, the suffix is stripped using a previously stored suffix database. If no database suffix matches, then the word is sent to rule based method. Else, the suffix-stripped word is again matched in the NE dictionary. If a match is found, then it is checked if the suffix can be morphologically accepted by the dictionary root word category. Then W is properly tagged with weight 1 or 0. Else, it is sent to the module for rule-based approach described below with the hope for better decision.

2.3. Rule-based NE detection:

Rule-based approaches rely on some rules, one or more of which is to be satisfied by the test word W. There may be *positive* and *negative* rules. The positive rules make the inference system biased towards NE while the negative rules tend to be biased against NE. Some small databases may be needed to execute the rules. For Bangla text NEs, some typical rules are given below. Here, rules 1-8 are positive and 9-12 are negative rules.

Rule1. If there are two or more words in a sequence that represent the characters or spell like the characters of Bangla or English, then they belong to the named entity (with high weight). For example, বি এ (B A), সি এম ডি এ (C M D A), অ গ প are all NEs. Note that the rule will not distinguish between a proper name and common name.

Rule 2. If the previous word of W is a pre-name word like শ্রী, শ্রীযুক্ত, ডঃ, মিঃ, মিস, মিসেস, বেগম, বিবি, মোঃ, মুঃ, ডক্টর, স্বামী, সৈয়দ, রেভারেন্ড, then W belongs to the named entity (with high weight). To detect them, all words of this type can be maintained in a database.

Rule 3. If after W there are title words and mid-name words to human names like ঘোষ, বোস, বসু, মিত্র, রায়, সরকার, খান, আহমেদ, রহমান, হক etc. and কুমার, চন্দ্র, রঞ্জন, শেখর, প্রসাদ, আলী, আলম etc., respectively, then W along with such words are likely to constitute a multi-word NE (with high weight).. For example, রবি বসাক, পল্লব কুমার মল্লিক are all NEs. A set of title and mid-name words should be collected and maintained in a database.

Rule 4. If a substring like -বাবু , -দাদা, -দা, -সাহেব, -কাকু , -গঞ্জ, -গ্রাম, -পুর, -গড়, -নগর occurs at the end of the word W, then W is likely to be a NE (with high weight). These strings can be collected in a database for future use.

Rule 5. If at the end of a word W there are strings like -এ-কার, -এর, -র, -রা, -এরা, -কে, -দের, -তে, -য় then W is likely to be a name (with high weight).

Rule 6. If a word like সরনী, রোড, স্ট্রিটট, লেন, থানা, স্কুল, বিদ্যালয়, কলেজ, নদী, লেক, হ্রদ, সাগর, মহাসাগর, পাহাড়, পর্বত is found after W of unknown type then W along with such word may belong to NE (with high weight). For example, বিধান সরনী , রাসেল স্ট্রিটট, ডালহৌসি পাহাড় are all NEs.

Rule 7. We note that only a few names or words in Bangla consist of characters ঁ (Chandrabindu) or ৎ (Khanda Ta)**.** So, if W does not belong to those words and has the occurrence of any of these two characters, then W may be a named entity (with high weight). For example, "অঁরি" is a French name.

Rule 8. If in the sentence containing unknown word W or a word W with *may or may not be NE* tag there are words like বলেন, বললেন, বলল, শুনল, হাসল, লিখল, লিখলেন, খেলেন, দেখল which imply action that can be done by human being, then W is likely to be a name (with high weight). A database of action verbs of various types is needed to check this rule.

Rule 9. If W of the type given in rule 8 is followed by verb not in the set of verbs described above, then W is not likely to be a NE. So, the weight should be reduced from 0.5 to a small value.

Rule 10. If there is re-duplication of W in a sentence then W is not likely to be a named entity. This is so because rarely name words are reduplicated. In fact, reduplicated name word may signify something else. For example রাম রাম is uttered to greet a person. So, the weight should be reduced in such case near zero.

Rule 11. If at the end of W there are suffixes like -টা, -খানা, -খানি, -টাতে, -টায়, -টিকে, -টাকে, -টকুন, -গুলা, -গুলো, -গুলি etc., then W is usually not a named entity.

Rule 12. If there is an echo-word after W e.g. গাছ টাছ, then none of these two words is a named entity.

The exact value of the weight for a rule is decided from training dataset. We increase or decrease the weight of the test word if a rule fires. To be consistent, we have included the dictionary-based approach under the same framework.

Thus, in our scheme, if the weight is more than certain value (say 0.75) then the word is finally accepted to be NE. Otherwise, if the weight is less than certain value (say 0.25) then the word is rejected to be NE. For intermediate cases, the word may be subject to the n-gram based technique described below.

2.4. n-gram based NE detection :

The n-gram based approach relies on the co-occurrence of other words before and after a NE. To generate the n-gram we need a corpus where the NE words are tagged manually. From these tagged words the left neighbor and right neighbor words are checked (for a 2- gram model). The frequencies of each pair of left-right neighbor are counted from the corpus. The probability of each left-right pair with respect to W may be estimated as

$P_{lr}(W)$ = No of this left-right word around W/ total no of all left-right words around W

in the training corpus.

If a particular left-right neighbors occur about a word W, then W has a *positive likelihood* of being NE, or a *negative likelihood* that W is not a NE. For example, in the sentence 'Mark is inadequate for this answer' the word 'Mark' is a negative instance for NE. But in the sentence 'Mark is a good boy', 'Mark' is a positive instance. So, we have to count from the corpus how many times the particular left-right neighbor give positive instances of W being a NE while how many are the instances of non-NE. From these positive and negative instance counts, a NE weight value is found for a particular pair of left-right word pair around W as

$$w_{lr}(W) = P_{lr}(W)\, R_{lr}(W)$$

where $R_{lr}(W)$ = No of positive instances /(No of positive instances + No of negative instances).

However, a large number of words will be negative instances at all times, so their $w_{lr}(W)$ value will come out as zero. Examples are the so-called *stop* words. They can be dealt in the dictionary itself, as discussed in Sec 2.2, reducing a lot of computational effort for this n-gram based approach. Some words which will also be positive instance, irrespective of the left right words. The dictionary can deal them as well. This fact partly justifies the scheme of having three approaches combined in our scheme for detecting NE.

Thus, the generation of training phase is completed. Now, in the test phase, if a word W has left-right neighbors whose weight is $w_{lr}(W)$ based on the training phase, then W may be assigned this weight of being named entity. This is the modified weight over and above what was given in the previous phases. For the test phase, a threshold t is set on the weight. If the

weight for the test word W is w > t then we declare W as a NE. Otherwise, we call it not-NE.

Note that, not only the words undecided by the previous stages are subject to the bigram technique, but other words in the sentence are also tested by this method. However, most of the words will be negative instances and hence their w_{lr} (W) will be zero for any pair of left-right word.

This bigram based technique is time consuming to design, since the Left-right word pairs can be many, in principle $O(N^2)$ where N is the number of surface words found in the training corpus.

3. Data collection

To obtain the corpus for our experiment, we browsed the net and found the site of Anandabazar Patrika, the largest Bangla daily newspaper. We downloaded the e-newspapers for the years 2001-2004. Of this huge data, a portion for the years 2001-2003 were used for training the system (about 20 million) and a portion for 2004 (about 100, 000 words) was used for testing. The data could not be utilized in a straightforward way, since the newspaper authority used a proprietary glyph code. So, we had to discover which glyph code denotes which character of Bangla script and then convert the text into ISCII coding format. After that, all the softwares were developed on these ISCII files. Once the ISCII files were generated, a software was used to collect all distinct words from this corpus of 20 million words. These distinct words were ranked in descending order of frequency and the top 20,000 ranked words were chosen for manual tagging of named entity.

The manual tagging was done by the linguists based on their global knowledge. However, if the person is in doubt, (s)he would consult a few examples in the original corpus involving the word in question. Using the contextual information, most problematic cases could be disambiguated. Those which still appeared unclear were given 'may or may not be' status. A morphological analyzer was previously developed in connection with the design of a spell checker in Bangla [13]. That analyzer has been employed for stemming of the type-words in the current NE detection problem also. Moreover, a rule-based system as described in Section 2.3 is also developed. The database needed for each rule is being continuously updated to give better experimental results.

4. Experimental results:

The software was trained with the Anandabazar Patrika web corpus of the year 2001-2003. Some geographical names were further added to enrich the database. Then several files of the corpus of the same newspaper of the year 2004 were used for testing. The results are presented in the form of recall(R), precision (P) and F-measure percentage. Here the recall is the ratio of number of NE words retrieved and the number of NE words actually present in the file, expressed in percent. In other words,

$$R\% = \frac{Number\ of\ NE\ words\ retrived}{Total\ Number\ of\ NE\ words\ in\ the\ text} X100\%$$

Precision is the number of correctly retrieved NE words to the total number of words retrieved, expressed in percent. So, we can write

$$P\% = \frac{Number\ of\ correct\ NE\ words\ retrieved}{Total\ Number\ of\ NE\ words\ retrieved} X100\%$$

The F-measure is often used in the Information Retrieval and Natural Language Processing problems. This class of measures was introduced by C. J. van Rijsbergen [14]. F_1- measure is the ratio of the twice of the multiplication of precision (P) and recall (R) and the sum of these two. In other words,

$$F1\% = \frac{2P\%R\%}{P\%+R\%} X100\%$$

F_1 measure combines recall (R) and precision (P) with an equal weight and hence is the harmonic mean of the two quantities. Note that F_1 cannot exceed 100%. Experimental results on 10 sets of test documents are shown in Table 1.

NO. OF WORDS	NO. OF NE	CORRECTLY DETECTED	NO. OF ERROR	RECALL %	PRECISION %	F_1-MEASURE %
2592	165	138	7	79.39	95.00	86.00
2938	186	157	6	81.10	96.20	88.00
2477	247	176	6	76.25	97.60	85.00
3816	336	268	7	79.76	97.40	87.00
2944	192	144	5	75.00	96.52	84.41
4843	255	210	13	82.35	93.50	87.85
2899	202	192	7	95.04	96.35	95.44
3420	232	201	9	86.63	95.52	90.85
4428	243	209	11	86.00	94.73	90.15
4228	210	177	16	84.28	90.96	87.42
4528	292	261	11	89.38	95.78	92.46
2991	193	168	5	87.04	97.02	91.75
AVERAGE				**85.50**	**94.24**	**89.51**

It is noted from Table 1 that the precision is reasonably high but the recall is somewhat moderate. The reason of moderate occurrence of recall is that the training has been done with only 20,000 corpus words, while actual number of corpus words was about 200,000. Also, we have to improve the database for rules, as well as search for other potential rules that we have not included here. The frontback 2-grams are also

at present aggregated over all NE words tagged manually. Such global occurrence statistics can mask the local phenomenon. We are working towards improving our NE detection approach and fruitful results will be communicated in a future paper.

5. Automatic Evaluation Approach :

Every detection system is to be judged by some evaluation techniques, e.g. BLEU (Bi-lingual Evaluation Understudy) [16] and several others. So, in case of ours we introduced an Automatic Evaluation System to the main detection system. The evaluation system is actually based upon a manually annotated dataset of almost 70,000 words. These datasets are tagged manually in a "non-NE <NE *Name* NE> non-NE" format and can be viewed in [17]. After the system detects and tags the names, the detection system treats the NE-detected file location as the *"Target Location"*. In our annotated dataset the annotated corpus is available for the same documents. That location is treated as the "*Annotated Location"*. As the evaluation system starts evaluating, a word by word comparison is done between the target and annotated locations. At the end of evaluation number of correctly detected words, the number of wrong detection and the number of real NE is found and so the Precision, Recall and F1-Measure is calculated easily.
We have also found that our evaluation system gives almost the same result as found by manual evaluation.

References:

[1] C. L. Borgman, and S. L Siegfried. 1992. *Getty's Synoname and its cousins: A survey of applications of personal namematching algorithms,* Journal of the American Society of Information Science, Vol. 43: 459-476.

[2] J. C. Harmansen, 1985 *Automatic name searching in large databases of international names*, Ph.D. thesis, Georgetown University.

[3] P. Hayes, 1994. *NameFinder: software that finds names in text,* Proceedings RIAO 94, 11-13 October, New York, Vol. 1, pp. 762-774.

[4] U. Pfeiffer; T. Poersch, and N. Fuhr. 1996. *Retrieval effectiveness of proper name search methods,* Information Processing & Management, Vol. 32, pp. 667-679.

[5] P. Thompson and C. C. Dozier, West Group, 610 Opperman Drive, Eagan, MN 55123, USA on *Name Searching and Information Retrieval.* Website Reference: http://arxiv.org/html/cmplg/ 9706017

[6] D. Marynard, V.Tablan, K. Cunningham and Y. Wilks. 2003. *Muse : a multisource entity recognition system.* Computers and the Humanities. Website Reference: http://gate.ac.uk/sale/muse/muse.pdf

[7] D. D. Palmer and D. S. Day. 1997. *A statistical profile of the named entity task.* Proceedings of Fifth ACL Conference for Applied Natural Language Processing (ANLP-97).

[8] G. S. Mann and D. Yarowsky. 2003. *Unsupervised personal name disambiguation.* Proceeding of CONILL-2003.

[9] H. Cunningham. 2002. *Gate, a general architecture for text engineering.* Computers and the Humanities, Vol. 36, pp. 223- 254.

[10] M. Narayanswamy, K.E. Ravikumar, and V.K. Shanker. 2003. *A biological named entity recognizer.* Proceedings of the Pacific Symposium on Biocomputing, Hawaii.

[11] R. Yangarber, W. Lin and R. Grishman. 2002. *Unsupervised Learning of Generalized Names.* Proceedings of
the 19th International Conference on Computational Linguistics (COLING-2002).

[12] S. Cucerzon and D. Yarowsky. 1999. *Language independent named entity recognition combining morphological*
and contextual evidence. Proceedings of the 1999 Joint SIGDAT conference on EMNLP and VLC.

[13] B.B. Chaudhuri. 2001. *A Novel Spell-checker for Bangla Text Based on Reversed-Word Dictionary.* Vivek Vol. 14 No. 4,pp. 3-12.

[14] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, London, 1979.

[15] D. Okanohara, Y. Miyao, Y. Tsuruoka and J. Tsujii. 2006. *Improving the Scalability of Semi-Markov Conditional Random Fields for Named Entity Recognition.* Proceedings of the COLING-ACL, Sydney, Australia, 17-21 July, pp. 465-472.

[16] Papineni, K., Roukos, S., Ward, T., and Zhu, W. J. 2002. *BLEU: a method for automatic evaluation of machine translation* in ACL-2002: 40th Annual meeting of the Association for Computational Linguistics pp. 311--318

[17] Annotated Bengali Corpus by Prof. B. B. Chaudhuri, ISI, Calcutta and Suvankar Bhattacharya. Website Reference : http://cybersuv.googlepages.com/Annotated.zip