NAMED ENTITY RECOGNITION IN NATURAL LANGUAGES USING TRANSLITERATION

Sudha Morwal, Deepti Chopra and Dr. G.N. Purohit

Department Of Computer Engineering, Banasthali Vidyapith, Jaipur (Raj.), India
sudha_morwal@yahoo.co.in
deeptichopra11@yahoo.co.in

ABSTRACT

Transliteration may be defined as the process of mapping sounds in a text written in one language to another language. Current paper discusses about transliteration and its use in Named Entity Recognition. We have designed a code that executes Transliteration and assist in the process of Named Entity Recognition. We have presented some of the results of Named Entity Recognition (NER) using Transliteration.

KEYWORDS

NER, Transliteration, Unknown words, Performance Metrics

1. INTRODUCTION

Transliteration is the task of converting word, letter or group of letters written in one language to another language. In this, we map phonetic sounds of one language into another language. Some of the important applications of Transliteration include the following: Machine Translation, Information Retrieval, Named Entity Recognition and Information Extraction. Named Entity Recognition (NER) is the subtask of Information Extraction in which named entities or proper nouns are recognized from a given text and thereafter classified into various categories. Transliteration plays a key role in solving the problem of unknown words in Named Entity Recognition. If a training of English sentence in NER is performed using example based learning and in testing if a Hindi sentence is given, then only the Named Entities (words which are not tagged with OTHER tag) present in the Training sentences are transliterated to Hindi language. Also, the words which are neither present in training file nor in the transliteration file, 'OTHER' tag is assigned to them. In this way unknown words in Named Entity Recognition can be handled.

e. g. Ram/PER plays/OTHER cricket/SPORT

In the above annotated training text in English, 'PER' denotes that 'Ram' is a Person, 'plays' tagged as 'OTHER' means that 'plays' is not a Named Entity, 'cricket' is a name of a Sport, and so it tagged with 'SPORT' tag. According to our approach,

If testing sentence in NER is given as:

राम क्रिकट खेलता है

If we do not use the transliteration approach, then by testing the above mentioned Hindi sentence, राम and क्रिकट will not be identified as Named Entities. So, we must use the transliteration approach in order to identify them properly. In training sentence, since Ram and cricket are Named Entities, so they are transliterated to Hindi language and stored along with their tags in

DOI : 10.5121/ijnlc.2013.2306

transliteration file. So, now 'राम' and 'क्रिकट' can easily be identified as Named Entities and the problem of unknown word is solved.

2. RELATED WORK

Named Entity Recognition (NER) is one the most important task in NLP. It is the process in which proper nouns or Named Entities are detected and then classified into different classes of Named Entity classes e.g. Name of Person, Sport, River, Country, State, city, Organization etc. The unknown words in Named Entity Recognition can be handled using many ways.

The words that are detected as unknown or for which no training has been done but they exist in testing sentences, a specific tag e.g. unk tag can be allotted to them in order to identify them and solve the problem of unknown words in NER [1][4].

Capitalization information does not alone serve to identify the POS tag, As Capitalization can exists in unknown words also. Also, Capitalization information can identify Named Entities in English only. [2][3].

3. OUR APPROACH

In our approach, we have used transliteration approach to identify the unknown words in Named Entity Recognition. We have taken 'OTHER' tag to be a 'Not a Named Entity' tag. Figure 1 depicts our approach involved during training in NER. We process each and every token of training data and detect its tag. If the token is not tagged with 'OTHER' tag, then we simply transliterate the token into different other languages in which testing sentence appear and save the transliterated Named Entities along with their tags information in a transliterated file. If the token is attached with 'OTHER' tag, then we simply process the next token in a sentence and continue this procedure until end of the training sentences is not reached.

ALGORITHM 1: TRANSLITERATION ON TRAINING DATA

Input: Annotated Text A

Output: Transliteration file F having list of Named Entities in other language

Variables: n = No. of Tokens in A, T = t_1 t_2 t_3…..t_n tokens in A

```
1: if (A ≠ ϕ ) then
2:     for i=1 to n do
3:         if (t_i = Named Entity) then
4:             Transliterate t_i into language in which testing sentences
               appear and store transliterated token along with its tag in F
5:         end if
6:     end for
7: end if
```

Figure1 Our approach involved during training in NER

If we have mapping of phonetic sounds from one language into another, then we can easily perform transliteration task in any language. e. g. If 'Geeta/PER plays/OTHER Badminton/SPORT' is a training sentence, then the transliteration task will transliterate Geeta and Badminton into other languages, since Geeta is a Named Entity attached to 'PER' tag and Badminton is a Named Entity attached to 'SPORT' tag. 'plays' is not a Named Entity, since it is tagged by 'OTHER' tag, so transliteration is not performed on it. So, our approach is a language independent based methodology, that is capable of detecting Named Entities from a text written in any of the Natural languages, provided that it has been trained on a Named Entity Recognition based system in any of the languages and there lie some sound mappings so that transliterated Named Entities in other languages can be achieved.

Figure 2 displays the approach taken while performing testing in a NER based system. In this approach, all the tokens in a testing sentence are considered one at a time. If the present token is not an unknown entity, then a tag is alloted to it using Viterbi algorithm. If present token is an unknown entity, then we search the Transliteration file. If the unknown entity exists in transliteration file, then the Named Entity tag is attached to it, otherwise 'OTHER' tag is assigned to it. This procedure takes place until all the testing sentences are not processed.

ALGORITHM 1: TESTING IN NER

Input: Testing Data D

Output: Transliteration file F having list of Named Entities in other language, Output of Testing file O

Variables: n = No. of Tokens in D, T = t_1 t_2 t_3…..t_n tokens in D

```
1: if (D ≠ ϕ ) then
2:      for i=1 to n do
3:           if (ti ≠ Unknown Entity) then
4:                Attach Tag to ti using Viterbi Algo & store ti and tag in O
5:           else
6:                for all tokens (Tok) in F do
7:                     if (ti  Tok) then
8:                          Attach tag to ti & store ti and tag in O
9:                     else
10                         Attach 'OTHER' tag & store ti and tag in O
11:                    end if
12:               end for
13:          end if
14:     end for
15: end if
```

Figure 2 Steps taken while performing testing in Named Entity Recognition

3. RESULT

In NER, handling of unknown words is a very important issue. We have developed a code for Transliteration. These Named Entities are then transliterated into different languages and stored in the respective transliterated files of different languages. We have performed Transliteration of Named Entities in English to Hindi and it generated transliterated Named Entities with accuracy of 70.6%.

TABLE 1 Named Entities transliterated into different languages

SNO	TAGS
1	PERSON(Name of Person)
2	ORG (Name of Organization)
3	CON (Name of Country)
4	MAG (Name of Magazine)
5	WEEK (Name of Week)
6	LOC (Name of Location)
7	PC (Name of Personal Computer)
8	MONTH (Name of Month)

For a given testing sentence, we can initially search if all the tokens exist in training sentences. If yes, then using Viterbi Algorithm, we can get optimal state sequence. If any word is unknown that does not exists in training file, then we can search the transliterated file, if the word exists in transliterated file, then the correct tag can be allotted to it and hence the problem of unknown word can be solved. If we have performed training in NER in English initially and during testing if we give a word in Hindi, then this would be treated as an unknown word for which we have not performed training yet. So, we can generate the transliterated list of Named Entities from English to other languages and handle the problem of unknown words in NER. We have performed NER in English. For Training, we took English and Hindi text from NLTK and accuracy obtained with our approach is 70.6%. TABLE 1 displays the Named Entities on which transliteration is performed in different languages and then these transliterated Named Entities are stored in a file along with their tags that can be used further to resolve the problem of unknown entities in a Named Entity Recognition.

4. PERFORMANCE METRICS

Performance Metrics is measure to estimate the performance of a NER based system. Performance Metrics can be calculated in terms of 3 parameters: Precision, Accuracy and F-Measure. The output of a NER based system is termed as “response” and the interpretation of human as the “answer key” [9]. Consider the following terms:

1. Correct-If the response is same as the answer key.
2. Incorrect-If the response is not same as the answer key.
3. Missing-If answer key is found to be tagged but response is not tagged.
4. Spurious-If response is found to be tagged but answer key is not tagged. [6]

Hence, we define Precision, Recall and F-Measure as follows: [5]7][8]

Precision (P): Correct / (Correct + Incorrect + Missing)
Recall (R): Correct / (Correct + Incorrect + Spurious)
F-Measure: (2 * P * R) / (P + R)

5. CONCLUSION

In this paper, we have discussed about Transliteration. We have obtained 70.58% accuracy by performing Named Entity Recognition in English and handling unknown words using transliteration approach. We have also discussed about Performance Metrics which is a very important measure to judge the performance of a Named Entity Recognition based system.

ACKNOWLEDGEMENT

I would like to thank all those who helped me in accomplishing this task.

REFERENCES

[1] http://www.cse.iitb.ac.in/~cs626-460-2012/seminar_ppts/ner.pdf
[2] http://www.sigkdd.org/explorations/issues/7-1-2005-06/4-Fu.pdf
[3] http://acl.ldc.upenn.edu/acl2002/MAIN/pdfs/Main036.pdf
[4] http://staff.um.edu.mt/cabe2/publications/nfHmms.pdf
[5] G.V.S.RAJU, B.SRINIVASU, Dr.S.VISWANADHA RAJU, 4K.S.M.V.KUMAR "Named Entity Recognition for Telugu Using Maximum Entropy Model"
[6] B. Sasidhar, P. M. Yohan, Dr. A. Vinaya Babu3, Dr. A. Govardhan,."A Survey on Named Entity Recognition in Indian Languages with particular reference to Telugu" IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March 2011.
[7] Asif Ekbal, Rejwanul Haque, Amitava Das, Venkateswarlu Poka and Sivaji Bandyopadhyay "Language Independent Named Entity Recognition in Indian Languages" .In Proceedings of the IJCNLP-08 Workshop on NER for South and South East Asian Languages, pages 33–40,Hyderabad, India, January 2008.Available at: http://www.mt-archive.info/IJCNLP-2008-Ekbal.pdf
[8] Darvinder kaur, Vishal Gupta. "A survey of Named Entity Recognition in English and other Indian Languages" .IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 6, November 2010.
[9] Shilpi Srivastava, Mukund Sanglikar & D.C Kothari. "Named Entity Recognition System for Hindi Language: A Hybrid Approach" International Journal of Computational Linguistics (IJCL), Volume (2) : Issue (1) : 2011.Available at http://cscjournals.org/csc/manuscript/Journals/IJCL/volume2/Issue1/IJCL-19.pdf

About Authors

Sudha Morwal is an active researcher in the field of Natural Language Processing. Currently working as Associate Professor in the Department of Computer Science at Banasthali University (Rajast han), India. She has done M.Tech (Computer Science) , NET, M.Sc (Computer Science) and her PhD is in progress from Banasthali University (Rajasthan), India. She has published many papers in International Conferences and Journals.

Deepti Chopra received B.Tech degree in Computer Science and Engineering from Rajasthan College of Engineering for Women, Jaipur, Rajasthan in 2011.Currently she has done M.Tech in Computer Science and Engineering from Banasthali University, Rajasthan. Her research interests include Artificial Intelligence, Natural Language Processing, and Information Retrieval. She has published many papers in International journals and conferences.

Dr. G. N. Purohit is a Professor in Department of Mathematics & Statistics at Banasthali University (Rajasthan). Before joining Banasthali University, he was Professor and Head of the Department of Mathematics, University of Rajasthan, Jaipur. He had been Chief-editor of a research journal and regular reviewer of many journals. His present interest is in O.R., Discrete Mathematics and Communication networks. He has published around 40 research papers in various journals