Case Study of Named Entity Recognition in Odia Using Crf++ Tool

Dr.Rakesh ch. Balabantaray
Department of Computer Science
IIIT, BBSR

Suprava Das
Department of Computer Science
IIIT, BBSR

Kshirabdhi Tanaya Mishra
Department of Computer Science
IIIT, BBSR

***Abstract*—NER have been regarded as an efficient strategy to extract relevant entities for various purposes. The aim of this paper is to exploit conventional method for NER in Odia by parameterizing CRF++ tool in different ways. As a case study, we have used gazetteer and POS tag to generate different feature set in order to compare the performance of NER task. Comparison study demonstrates how proposed NER system works on different feature set.**

Keywords—Named Entity Recognition; CRF++ Tool; Odia Named Entity

I. Introduction

NER is a subtask of information extraction that involves locating and classifying named entities such as person name, location name, organization name... etc. Besides information extraction, NER has applications in question answering (Toral et al., 2005; Molla et al., 2006), Machine translation (Babych & Hartley, 2003). In English language, recognition of named entity is easy with greater accuracy, but for Indian languages (especially for the language which are not morph analysed), recognition of named entity is challenge now. For Indian languages, many approaches have been applied for NE recognition. These approaches are: Rule based approach (krupka and Hausman, 1998) and Machine learning approach or hybrid approach Decision tree (Karkaletis et al. , 2000) , Hidden Markov model(Biker ,1997) , MEMM(Borthwick et al. ,1998) , CRF(Andrew McCallum and Wei Li , 2003)).This paper presents an overview of work done on locating named entity in a text for Odia language using conditional random field. We have used CRF++ (version 0.54) tool which is implementation of conditional random field, a machine learning approach for NE recognition. The statistical CRF model has been used for NER as it is more efficient to deal with Indian languages. Section-2 gives a brief description on conditional random field and section-3 gives brief description on Part of speech tag; section-4 describes preparation of training data and testing data for CRF based model followed by section 5 describes the features used for CRF framework, section 6 describes how CRF++ detects named entities and section 7 describes the result and accuracy. Conditional random field is a machine learning technique which overcomes the disadvantage of other machine learning approach like HMM and MEMM. In HMM, the words in input sequence are not dependant among each other. MEMM face label bias problem because of its stochastic state transmission nature. CRF overcomes these problems and gives a greater accuracy. Conditional random field are undirected graphical model used to calculate the conditional probability of values on designated output nodes given values assigned to other designated input nodes. As CRF is a discriminative, so the word identity feature is informative, this helps to label unseen words by exploiting the feature.

We have used the C++ based openNLP CRF++ package of version 0.54 (Taku Kudo, 2005). The CRF++ tool extracts the information from the training data and builds a CRF model according to weightage of information. When the test data presented with CRF model, the tool outputs the test data tagged with the labels that has been learnt.

II. Conditional Random Field

Conditional random field is a machine learning technique which overcomes the disadvantage of other machine learning approach like HMM and MEMM. In HMM, the words in input sequence are not dependant among each other. MEMM face label bias problem because of its stochastic state transmission nature. CRF overcomes these problems and gives a greater accuracy. Conditional random field are undirected graphical model used to calculate the conditional probability of values on designated output nodes given values assigned to other designated input nodes. As CRF is a discriminative, so the word identity feature is informative, this helps to label unseen words by exploiting the feature.

Conditional Random Fields can be defined as in [3] as follows: “Let **G** = (**V**, **E**) be a graph such that **Y**= (**Y**v) v **V**, so that **Y** is indexed by the vertices of **G**. Then (**X**, **Y**) is a conditional random field in case, when conditioned on **X**, the random variables **Y**v obey the Markov property with respect to the graph:

P (**Y**v|**X**, **Yw, w**? v) = p (**Y**v|**X**, **Yw, w**~v), where w~v means that w and v are neighbors in G”.

Here X might range over natural language sentences and Y denotes the label sequence.

What this means is that a CRF is an undirected graphical model whose nodes can be divided into exactly two disjoint sets X and Y, the observed and output variables, respectively; the conditional distribution is p(Y|X) is then modelled. The aim of the CRF is to find out the label sequence y ∈Y that maximizes the conditional probability p (Y|X) for a sequence X.

That is $$\mathbf{y=\underset{\mathrm{y}}{argmax}\ p(Y|X)}$$

Thus, NER task can be considered as a sequence labeling task. Hence CRF can be used for NER task.

III. Experimental Set Up

A. Part Oe Speech Tag

In corpus linguistics, part-of-speech tagging (POS tagging or POST), also called grammatical tagging or word-category disambiguation, is the process of marking up word in a text (corpus) as corresponding to a particular part of speech, based on both its definition, as well as its context—i.e. relationship with adjacent and related words in a phrase, sentence, or paragraph.

There are large numbers of POS tagger available for English language which has got satisfactory performance but cannot be applied to Hindi language due to structural differences. For our experiment we have used POS-Tagger tool for Odia language which is implemented using conditional random field. The accuracy of this tool is not high but accuracy of tagging proper noun is quite high.

B. Gazetteer

We have prepared 4 different gazetteers. The words belongs to the person, location, organization are stored in 3 different gazetteers respectively. Another gazetteer contains only NE without any classification and it contains around 730 NEs. The named entities in gazetteer are arranged in dictionary order. For morph analysis we have used another gazetteer which

D. Corpus

A corpus for Odia language is collected which contains around 45000 tokens/words from the domain of health, tourism, general. This corpus contains about 1000 named entities of PERSON, LOCATION, and ORGANIZATION. This file is split into 2 sets, 80% of words are used for training data and 20% of them used for testing data.

E. Preparation Of Training Data

For case study training data needs to be prepared in 3 different ways for 3 different cases. To make CRF++ tool learns, training data should be in a particular format. So the training file needs to be pre-processed. We have taken 3 column format training data. 1st column remains same for all cases, but 2nd and 3rd column varies. 2nd column is generated using POS Tagger tool with POS tag for two cases and for one case it is tagged with set {YES, NO}. 3rd column of training data contains all of user generated annotations for named entities. For one case the Odia named entity tagged with tag set {B-name, I-name, 00}. The tokens which are not present in the gazetteer means which are not named entities are tagged as "00". And those which are named entities , if contains single token as NE, this tagged as "B-name" , otherwise the 1st token is tagged as "B-name" and the rest tokens which are inside NE tagged as "I-name". For other two cases users are supported to label named entity by using the corresponding tags i.e. <PERSON>, <LOCATION>, <ORGANIZATION>.

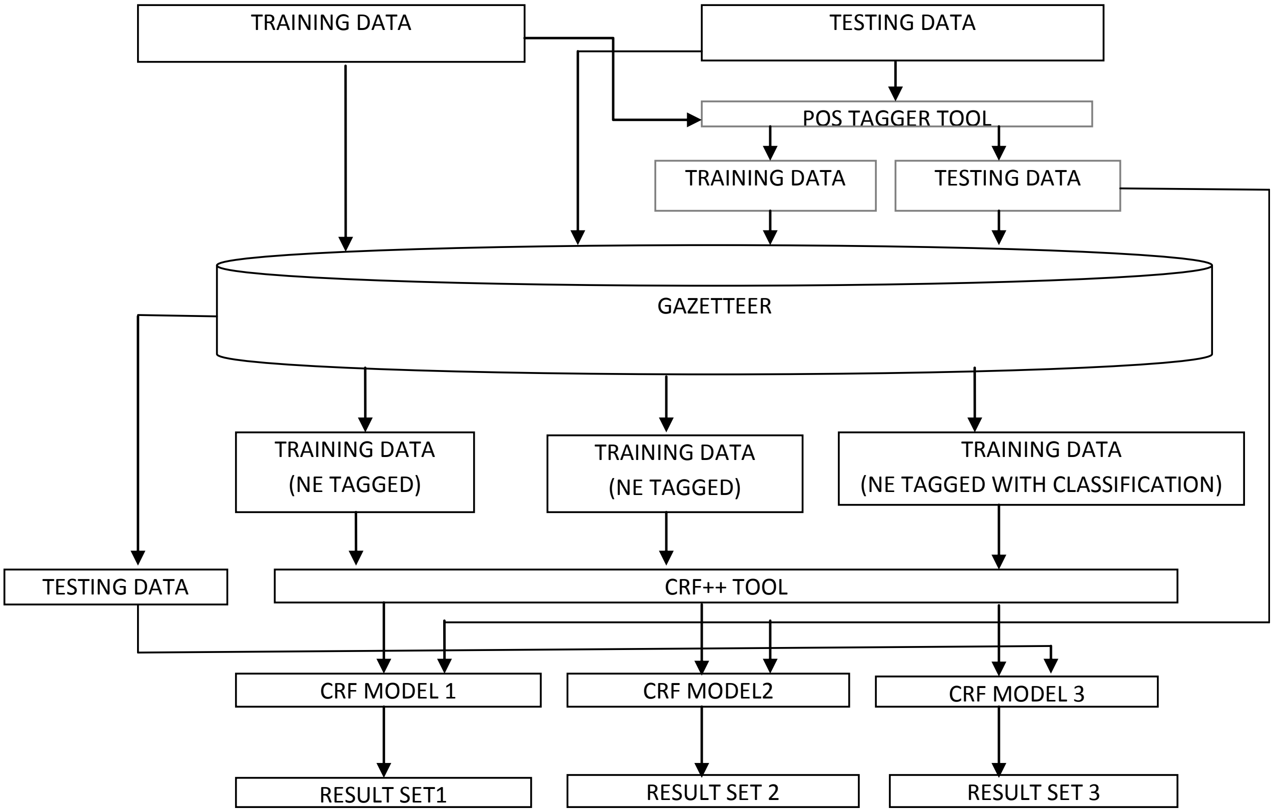

Fig. 1 [Work flow diagram]

F. Preparation Of Testing Data

Unlike the train data, the test data is in 2-column format. The test data is presented in same way as train data , only the difference is test data contains only tokens and corresponding POS tag (for two cases) and {YES, NO} tag (for one case).

The preparation of training data, testing data and analysis of NER system using CRF++ tool is schematically represented in FIGURE – 1.

IV. Result And Discussion

To evaluate the performance of NER in Odia language using CRF++ tool, we make use of 3 parameters i.e. precision, recall and f-measure.

Precision measures the percentage of correct NE tagged by CRF tool over the total number of NEs tagged by CRF tool.

$$precision = \frac{\mathrm{tp}}{\mathrm{tp} + \mathrm{fp}}$$

Recall measures the percentage of NE tagged by CRF tool over the total number of NEs in the file tagged by gazetteer.

$$\mathrm{Recall} = \frac{\mathrm{tp}}{\mathrm{tp} + \mathrm{fn}}$$

F-measure is a measure that combines precision and recall is the harmonic mean of precision and recall.

$$F\ measure = \frac{2 * \mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

The comparative study for all the three cases has done. And result for these cases are given in the table below.

Measurement	Value
Precision	0.925
Recall	0.593
F – measure	0.71

TABLE-1: [Evaluation of NEs without classification]

Table – 1 show that our proposed feature sets can effectively identify Odia named entity from testing repository.

The table-2 describes the comparison between the cases where the classification of named entity is taken into consideration. For one case gazetteer is used to parameterize CRF++ Tool and for other case POS tag along with gazetteer is used to parameterize the tool, which causes generation of different sets of feature.

Table -3 shows the actual number of NEs present in training and testing repository and the number of named entity recognized by CRF MODEL. Based upon which the performance of the system is measured.

A. Comparision Graph

We have taken different dataset to measure the performance named dataset-1, dataset-2 and dataset-3.

Two classes of parameters are most important: the combination and selection of feature and tokenization of the text.

The impact of each feature (Gazetteer and POS tag) or group of feature (Gazetteer combined with POS tag) is computed. They are displayed in following graph.

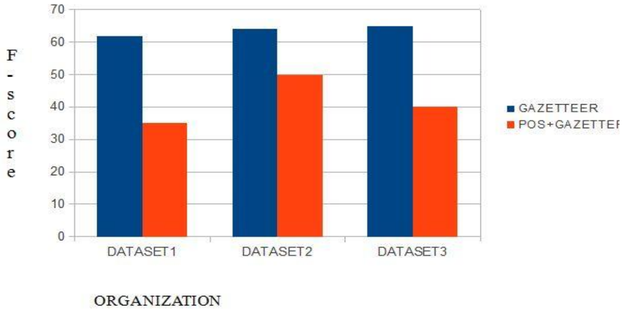

Fig. 2 [Comparison of f measure of ORGANIZATON NEs using different dataset]

	GAZETTEER			All features (Gazetteer and POS tag)			F measure comparison
	P	R	F	P	R	F	
PERSON	0.87	0.81	0.84	0.97	0.44	0.63	25% decrease
LOCATION	0.88	0.82	0.85	0.75	0.50	0.60	18% decrease
ORGANIZATION	0.50	0.82	0.62	0.66	0.25	0.35	43% decrease

TABLE – 2: [Results for PERSON, LOCATION and ORGANIZATION using gazetteer and all features]
P-Precision R-Recall F-F measure

	person		location		organization	
	TRN	*TST*	*TRN*	*TST*	*TRN*	*TST*
Gazetteer	382	180	248	175	183	70
Gazetteer and POS		109		121		43

TABLE – 3: [Calculation of total number of NEs for all cases]
TRN – Training Data, TST – Testing Data

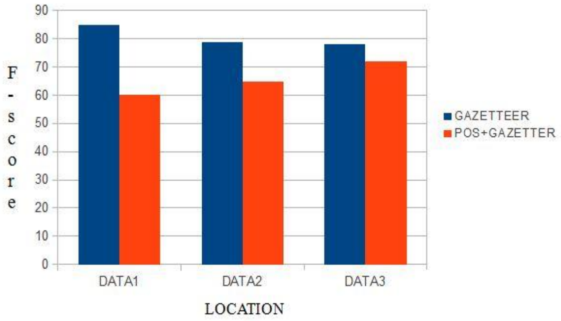

Fig. 3 [Comparison of f measure of LOCATION NEs using different dataset]

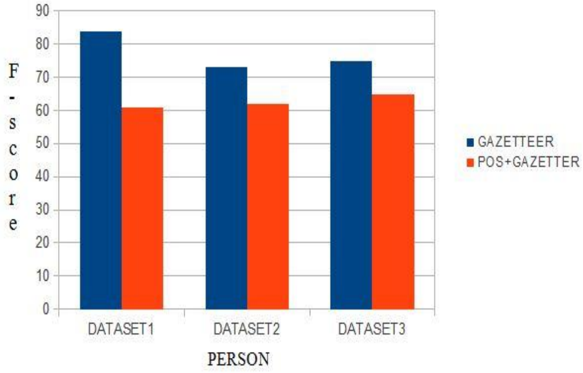

Fig. 4 [Comparison of f measure of PERSON NEs using different dataset]

VIII. Conclusion

In this paper we have shown a novel NER system based on conditional random field by generating various type of feature set. We have used CRF based POS tagger tool and gazette file to parameterize CRF++ Tool. The performance of the system is quite good when we experiment with individual case (f-measure for NEs only is 71% and f-measure for NEs with classification is 84% for PER, 85% for LOC and 62% for ORG). The performance of system decreases when we combine both POS tag and Gazetteer to generate feature. The reason for decrease in performance may be the average accuracy of POS Tagger tool. The accuracy may be increased if accuracy of POS Tagger tool is good. Morphological analysis has also shown a small contribution to the performance of the system. The current work is limited to recognizing the named entities which does not have nested structure.

References

[1] Wallach, H. M. 2004. Conditional random fields: An introduction, Technical Report MS-CIS-04-21, University of Pennsylvania, Department of Computer and Information Science, University of Pennsylvania.

[2] Li Wei and McCallum Andrew. 2004. Rapid Development of Hindi Named Entity Recognition using Conditional Random Fields and Feature Induction (Short Paper). In *ACM Transactions on Computational Logic*.

[3] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labelling Sequence Data,"*Proc. 18th Int'l Conf. Machine Learning*, 2001.

[4] Sotirios P. Chatzis, Yiannis Demiris, "The Conditional Random Field Model for Sequential Data Modelling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 02 Oct. 2012. IEEE computer Society Digital Library.

[5] Georgios Paliouras, Vangelis Karkaletsis, Georgios Petasis and Constantine D. Spyropoulos," Learning Decision Trees for Named-Entity Recognition and Classification", ECRAN, 2000.

[6] Colmenar, J.M., Abanades, M.A., Poza, F., Martin, D., Cuesta, A., Herran, A., Hidalgo, J.I., "On a generalized name entity recognizer based on Hidden Markov Models", *Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference* , On page(s): 952 - 958

[7] Andrew McCallum, Dayne Freitag, and Fernando Pereira," Maximum Entropy Markov Models for Information Extraction and Segmentation", 17th International Conf. on Machine Learning, 2000, 591-598

[8] CRF++.http://crfpp.sourceforge.net/