Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 48 (2015) 507 – 512

Procedia Computer Science

International Conference on Intelligent Computing, Communication & Convergence

(ICCC-2015)

Conference Organized by Interscience Institute of Management and Technology,

Bhubaneswar, Odisha, India

Part of speech tagging in odia using support vector machine

Bishwa Ranjan Das[a], Smrutirekha Sahoo[b], Chandra Sekhar Panda[c], Srikanta Patnaik[d]

[a,d]*Department of Computer Science & Information Technology.*
Institute of Technical Education and Research. SOA University, Bhubaneswar, India
[b]*Department of Computer Science & Application, North Odisha University, Baripada, India.*
[c]*Department of Computer Science & Application, Sambalpur University, Burla, India*

Abstract

Part of Speech (POS) Tagging is a challenging task to identify the meaning of each word in a sentence. This paper shows the task of identifying each word in an odia sentence using the technique of Support Vector Machine. The POS Tagger is developed using a very small tagset of five tags. Various features sets are taken for different contextual information is helpful in predicting the POS classes. An Odia corpus of 10,000 words has taken and tested it very carefully. The previous POS Tagger was done using Artificial Neural Network (ANN) had given the accuracy of 81%. But this SVM based POS Tagger for Odia gives the result with an accuracy of 82%. It is very helpful to use in many field of natural language process. The result of this system compares with POS tagger using ANN which was previously done.

Peer-review under responsibility of scientific committee of International Conference on Computer, Communication and Convergence (ICCC 2015)

Keywords: Part of Speech: Support Vector Machine; Odia Corpus

* Bishwa Ranjan Das. Tel.:+91-8895007227;
*E-mail address:*biswadas.bulu@gmail.com

Peer-review under responsibility of scientific committee of International Conference on Computer, Communication and Convergence (ICCC 2015)
doi:10.1016/j.procs.2015.04.127

1. Introduction

POS Tagging is the process of assigning a part of speech, like noun, verb, pronoun, adverb, adverb or other lexical class marker to each word in a sentence. The solving of ambiguity in POS tagging system is challenging task for all Natural Language Processing (NLP) researchers. The input to a tagging algorithm is a string of words of a natural language sentence and a specific tagset the output is a single POS Tag for each word. There are different machine learning approaches to the problem of assigning each word of a text with a parts of speech tag, which is known as POS tagging. In this paper the performance of a POS Tagger for Odia language is shown using SVM. Support Vector Machine is basically used for classification and recognizes the pattern. [1]SVMs have high generalization performance independent of dimension of feature vectors.

Asif Ekbal[1] shows labeling each words in the corpus using SVM with accuracy 16.84%. Cutting[2] described details about POS Tagging using Hidden Markov Model. Helmut Schmid[3], a new part of speech tagging method based on neural networks (net tagger) is presented and its performance is compared to that of a HMM-Tagger (Cutting et al 1992) and a trigram based tagger (Kempe, 1993). A part-of-speech tagger based on a multilayer perceptron network is presented. It similar to the network of Nakamura et al (1990) in so far as the same training procedure (Back propagation) is used; but it differs in the structure of network and also in its purpose (Disambiguation Vs Prediction). The performance of the tagger is measured and compared to that of two other taggers (Cutting et al. 1992; Kempe 1993).

2. POS Tagging in Odia

2.1 Various terminology uses in Odia

Various terminology uses in Odia language which are used for Odia Pos Tagging like Noun, Adjective, Verb, Pronoun, Adverb, Preposition etc. in Odia.

Noun -> Bisheshya, Adjective -> Bisheshana, Verb -> Kriya, Pronoun -> word use instead of Noun, Adverb -> Kriya bisheshana etc.

2.2 Morphological Analysis

To find the root or base word in Odia many suffixes are there, these suffixes are used in verb as well as noun also. These noun suffixes are come from inflection list (Bivokti) and some suffix list are use in verb, from these suffix we find out no of nouns and verbs. Here suffix list are mentioned in which is they are used in noun and verb.

3. Support Vector Machine

Support Vector Machines is machine learning approach, basically used for classification and regression. SVMs are well known for their good generalization performance and also used for pattern recognition. The role of SVM in NLP is applied to text categorization, and gives the high accuracy with a large number of texts taken as features. I am defining very simple case, a two class problem where the classes are linearly separable. Let the data set D be given as (X_i,y_i), (X_2,y_2)…..(X_D,y_D), where X_i is the set of training tuples with associated class labels y_i,. Each y_i can take one of two values, either +1 or -1(i.e., $y_i \in \{+1,-1\}$. I see this is a 2-D data are linearly separable because a straight line can be drawn to separate all of the tuples of class +1 from all of the tuples of class -1. There are an infinite number of separating lines that could be drawn. It is to find the "best", one, that is, will have the minimum classification error on previously unseen tuples. It uses the term "hyperplane" to refer to the decision boundary that is seeking, regardless of the input attributes.
An SVM approach this problem by searching for the Maximum Marginal Hyperplane. SVM searches for the hyperplane with the largest margin, that is, the Maximum Mariginal Hyperplane(MMH). The associated margin gives the largest separation between classes. A separating hyperplane can be written as $W.X + b = 0$, Where W is a weight vector, $W= \{w_1, w_2,\ldots,w_n\}$; n is the number of attributes, and b is a bias, Let's consider two input attributes,

A1 and A2. Training tuples are 2-D, e.g. X={x_1, x_2}, Where x_1, x_2 are the values of attributes A1 and A2 respectively for X. The hyperplane equation $W.X + w_0 = 0$, where W is weight and w_0 is bias. The hypothesis space under consideration is the set of functions,

$$F(X, W, w0) = \text{sign}\,(W.X + w_0), \quad (1)$$

Which classify data points based on whether the quantity $W.X + w_0$ are +ve or –ve.

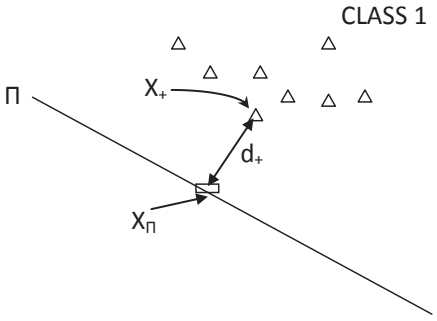

Figure 1. Points close to Hyperplane

The linear separable case is almost done. The last point is that given any unknown data point A, one can classify it using a linear indicator function. The non linear SVM classifier gives a decision making function f(x).

$$f(x) = \sum_{i=1}^{m} w_i K(x, z_i) + b\,, \quad g(x) = sign\big(f(x)\big) \quad (2)$$

If g(x) is +1, x is classified as class C_1 and -1 x is classified as class C_2. z_i are called support vectors and representative of training examples, m is the number of support vectors is a kernel that implicitly maps vectors into a higher dimensional space and can be evaluated efficiently. The polynomial kernel $K(x, z_i) = (x.z_i)^d$.

4. Methodology

4.1 Training Data

It is used in our own proposed system that was developed by ourselves. The corpus having 10000 words tested very carefully. It gives as possible as correct result for our system.

4.2 Suffix & Prefix

Some suffix & prefix alphabets are used to identify NE which are mentioned in the features. Firstly a fixed length word suffix of the current and surrounding words are used as features.

4.3 Algorithm

The proposed algorithm used for finding the POS tagging is as follows. Firstly the entire Odia Text is entered by user, and then it is divided into six steps which are described in the following algorithm.
Step 1. Enter a text.
Step 2. Convert entire text into token by tokenization.
Step 3. Finding root word using morphological analysis.
Step 3. Extract the suffix features from the each and every word.
Step 4. Compare each word with our valid suffix features.
Step 5. Identify Part of Speech.

The following flowchart to find POS tagging.

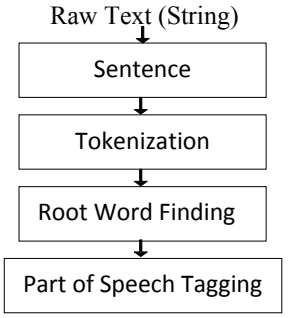

Fig. 2 Flowchart of POS tagger

4.4 Features

Here the proposed model is based on SVM
For example:-
ମୁଁ ଘରକୁ ଯାଉଛି, (mu gharaku jauchhi/ I am going to home)
ମୁଁ – ବିଶେଷ୍ୟ, ଘରକୁ – ଘର+କୁ, ଯାଉଛି – ଯିବା+ଉଛି

In the above sentence "mu gharaku jauchhi". Mu/ମୁଁ represents as noun, gharaku/ଘରକୁ represents as noun, ku/କୁ is used as inflection (Bivokti). In odia grammar, ku, nku, mananku used as Dittiya Bivokti(Inflection), Finding the root word is Ghara/ଘର using morphological analysis. Like this the word jauchhi(ଯାଉଛି) used as verb, the root word ଯିବା/jiba, the word uchhi/ଉଛି is used as suffix which is basically use in verb. In Odia, where the inflection suffix is used that word is known as Noun, and some suffix like uchhi/ଉଛି, ichhi/ଇଛି, ithili/ଇଥିଲି, uthila/ଉଥିଲା used as verb/Kriya. The suffixes –tara(ର), -ttama(ମ) are used as Adjective. There are 25 entries selected for finding adjectives used in our system. There are many more features are developed for our system for adverb, pronoun etc. Different features for part of speech tagging in Odia have been identified. The features also included prefix and suffix for all words. The prefix/suffix is a sequence of first and last few character of word. [1] Preceding and following words of a particular word are used as features. A small lexicon of 300 words is used to improve the quality of POS tagger. It contains different root words and noun, verb, adjective, pronoun etc.

5. Discussion and Results

SVM based POS Tagger has developed using a small corpus tagged with 5 tags, we already prepared a small amount of training corpus. In this system, each word in the test data will be assigned the POS tag which occurred most frequently for that word in the training data. The unknown word is assigned the pos tag with the help of lexicon, NER system and word suffixes. The result of this system is being calculated by precision, recall and f_score. Individually result being calculated for noun, verb, adjective, pronoun, adverb.

Precision = |AT ∩ OT | / |OT|
Recall = |AT ∩ OT| / |AT|
F_Score = 2(Recall * Precision) / (Recall + Precision)
Here AT - Actual Tag, OT - Obtained Tag. Precision means how many correct entities from whatever has been obtained are. Recall means out of the correct once how many have been obtained named entities. Here accuracy is calculated through F_Score in percentage. With the help of harmonic mean (HM) more accurate result also calculated.
HM< GM< AM

Whenever harmonic mean will be increased, automatically geometric mean as well as arithmetic mean will be increased. Individually we calculate the accuracy for noun, verb, adjective, pronoun and adverb.

6. Conclusion

Our proposed system tries to identify POS tags nearly accurately with a success rate of 81% without any error. Although this system worked fine on the Odia newspaper text, we are not sure if this will work equally well in other types of Odia text. Since Odia is a resource-poor as well as less-researched language, it is obvious that we need more exhaustive research in this direction before we can claim appreciable success in finding the POS tags used in Odia written texts. The performance of this system has been compared with the existing one Odia POS tag system and one Bengali POS Tag system. There are many linguistic and stylistic issues (e.g., agglutinative nature and different writing style, etc) that also need careful attention for developing tagger system for the Odia language. Definitely, the availability of an Odia text corpus of only five lakh words collected from Odia newspapers cannot be the benchmark trial database for systems like this, even if SVM system works fine on our database. With this limited success we propose to move further as application relevance of POST is proved in many domains of NLP: parsing, word sense disambiguation, information retrieval, question answering, machine learning – to mention a few.

References

1. Ekbal A. and Bandyopadhyay, S. (2008), "Part of Speech Tagging in Bengali using Support Vector Machine", Proceedings of the International Conference on Information Technology (ICIT 2008), pp.106-111, IEEE.
2. Cutting D., J. Kupiec, J. Pederson and P. Sibun, "A Practical Part of Speech Tagger", In Proc of the 3rd Conference on Applied Natural Language Processing, Pp 133-144,1992
3. Ankur parikh, "Part-of-speech Tagging using neural network", Proceedings of ICON-2009: 7th International Conference on Natural Language Processing, Report No: IIIT/TR/2009/232.
4. Helmut Schmid, "Part-Of- Speech Tagging with Neural Networks", COLING '94 Proceedings of the 15th conference on Computational linguistics - Volume 1, Pages 172-176.
5. Developing Oriya Morphological Analyzer Using Lt-toolbox, Itisree Jena, Sriram Chaudhury, Himani Chaudhry, Dipti M., Information Systems for Indian Languages Communications in Computer and Information Science Volume 139, 2011, pp 124-129
6. Smriti Singh, Kuhoo Gupta, Manish Shrivastava, and Pushpak Bhattacharyya(2006), "Morphological richness offsets resource demand – experiences in constructing a pos tagger for Hindi", Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, Sydney, Australia, pp. 779–786.
7. Ekbal, Asif, Mondal, S., and S. Bandyopadhyay (2007) "POS Tagging using HMM and Rule-based Chunking", In Proceedings of SPSAL-2007, IJCAI-07, pp. 25-28.
8. Ekbal, R. Haque and S. Bandyopadhyay (2008), "Maximum Entropy Based Bengali Part of Speech Tagging", Advances in Natural Language Processing and Applications, Research in Computing Science (RCS) Journal, \Vol. (33), pp. 67-78
9. Ekbal, R. Haque and S. Bandyopadhyay (2007), "Bengali Part of Speech Tagging using Conditional Random Field", Proceedings of the 7th International Symposium on Natural Language Processing (SNLP-07), Thailand, pp.131-136.
10. Ekbal , M. Hasanuzzaman and S. Bandyopadhyay (2009), "Voted Approach for Part of Speech Tagging in Bengali", Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC-09), December 3-5, Hong Kong, pp.120-129.