

# Bengali Part of Speech Tagging using Conditional Random Field

**Asif Ekbal**

Department of CSE  
Jadavpur University  
Kolkata-700032, India  
asif.ekbal@gmail.com

**Rejwanul Haque**

Department of CSE  
Jadavpur University  
Kolkata-700032, India  
rejwanul@gmail.com

**Sivaji Bandyopadhyay**

Department of CSE  
Jadavpur University  
Kolkata-700032, India  
sivaji\_cse\_ju@yahoo.com

## Abstract

This paper reports about the task of Part of Speech (POS) tagging for Bengali using the statistical Conditional Random Fields (CRFs). The POS tagger has been developed using a tagset<sup>1</sup> of 26 POS tags, defined for the Indian languages. The system makes use of the different contextual information of the words along with the variety of features that are helpful in predicting the various POS classes. The POS tagger has been trained and tested with the 72,341 and 20K wordforms, respectively. It has been experimentally verified that the lexicon, named entity recognizer and different word suffixes are effective in handling the unknown word problems and improve the accuracy of the POS tagger significantly. Experimental results show the effectiveness of the proposed CRF based POS tagger with an accuracy of 90.3%.

## 1 Introduction

Part of Speech (POS) tagging is the task of labeling each word in a sentence with its appropriate syntactic category called part of speech. POS tagging is a very important preprocessing task for language processing activities. This helps in doing deep parsing of text and in developing Information extraction systems, semantic processing etc. Part of Speech (POS) tagging for natural language texts are developed using linguistic rule, stochastic models and a combination of both. Stochastic models (Cutting, 1992; Merialdo, 1994; Brants, 2000) have been widely used in POS tagging task for simplicity and language independence of the models. Among stochastic models, Hidden

Markov Models (HMMs) are quite popular. Development of a stochastic tagger requires large amount of annotated corpus. Stochastic taggers with more than 95% word-level accuracy have been developed for English, German and other European languages, for which large labeled data is available. The problem is difficult for Indian languages (ILs) due to the lack of such annotated large corpus.

Simple HMMs do not work well when small amount of labeled data are used to estimate the model parameters. Incorporating diverse features in an HMM-based tagger is difficult and complicates the smoothing typically used in such taggers. In contrast, a Maximum Entropy (ME) based method (Ratnaparkhi, 1996) or a Conditional Random Field (CRF) based method (Lafferty et al., 2001) can deal with diverse, overlapping features. (Smriti et al., 2006) proposed a POS tagger for Hindi that has an accuracy of 93.45% with the exhaustive morphological analysis backed by high coverage lexicon and a decision tree based learning algorithm (CN2).

International Institute of Information Technology (IIIT), Hyderabad, India initiated a POS tagging contest, NLPAL Contest06<sup>2</sup> for the Indian languages in 2006. Several teams came up with various approaches and the highest accuracies were 82.22% for Hindi, 84.34% for Bengali and 81.59% for Telugu. As part of the SPSAL2007<sup>3</sup> workshop in IJCAI-07, a competition on POS tagging and chunking for south Asian languages was conducted by IIIT, Hyderabad. The best accuracies reported were 78.66% for Hindi (Avinesh and Karthick, 2007), 77.37% for Telugu (Avinesh and Karthick, 2007) and 77.61% for Bengali (Sandipan, 2007).

<sup>1</sup>[http://shiva.iiit.ac.in/SPSAL2007/iiit\\_tagset\\_guidelines.pdf](http://shiva.iiit.ac.in/SPSAL2007/iiit_tagset_guidelines.pdf)

<sup>2</sup>[http://lrc.iiitnet/nlpai\\_contest06](http://lrc.iiitnet/nlpai_contest06)

<sup>3</sup><http://shiva.iiit.ac.in/SPSAL2007>

In this paper, we have developed a POS tagger based on Conditional Random Field (CRF) that has shown an accuracy of 87.3% with the contextual window of size four, prefix and suffix of length upto three, NE information of the current and the previous words, POS information of the previous word, digit features, symbol features and the various gazetteer lists. It has been experimentally shown that the accuracy of the POS tagger can be improved significantly by introducing lexicon (Ekbal and Bandyopadhyay, 2007a), named entity recognizer (Ekbal et al., 2007b) and word suffix features for handling the unknown words. Experimental results show the effectiveness of the proposed model with an accuracy of 90.3%.

## 2 Conditional Random Fields

Conditional Random Fields (CRFs) (Lafferty et al., 2001), the undirected graphical models, are used to calculate the conditional probability of values on designated output nodes given values on other designated input nodes. The conditional probability of a state sequence  $S = \langle s_1, s_2, \dots, s_T \rangle$  given an observation sequence  $O = \langle o_1, o_2, \dots, o_T \rangle$  is calculated as:

$$P_{\Lambda}(s | o) = \frac{1}{Z_o} \exp\left(\sum_{t=1}^T \sum_k \lambda_k f_k(s_{t-1}, s_t, o, t)\right),$$

where,  $f_k(s_{t-1}, s_t, o, t)$  is a feature function whose weight  $\lambda_k$  is to be learned via training. The values of the feature functions may range between  $-\infty \dots +\infty$ , but typically they are binary. To make all conditional probabilities sum up to 1, we must calculate the normalization factor,  $Z_o = \sum_s \exp\left(\sum_{t=1}^T \sum_k \lambda_k f_k(s_{t-1}, s_t, o, t)\right)$ , which,

as in HMMs, can be obtained efficiently by dynamic programming. To train a CRF, the objective function to be maximized is the penalized log-likelihood of the state sequences given the observation sequences:

$$L_{\Lambda} = \sum_{i=1}^N \log(P_{\Lambda}(s^{(i)} | o^{(i)})) - \sum_k \frac{\lambda_k^2}{2\sigma^2},$$

where,  $\{ \langle o^{(i)}, s^{(i)} \rangle \}$  is the labeled training data. The second sum corresponds to a zero-mean,  $\sigma^2$ -variance Gaussian prior over parameters, which facilitates optimization by making the like-

lihood surface strictly convex. Here, we set parameters  $\lambda$  to maximize the penalized log-likelihood using Limited-memory BFGS (Sha and Pereira, 2003), a quasi-Newton method that is significantly more efficient, and which results in only minor changes in accuracy due to changes in  $\sigma$ .

When applying CRFs to the part of speech problem, an observation sequence is a token of a sentence or document of text and the state sequence is its corresponding label sequence. While CRFs generally can use real-valued functions, in our experiments many features are binary valued. A feature function  $f_k(s_{t-1}, s_t, o, t)$  has a value of 0 for most cases and is only set to be 1, when  $s_{t-1}, s_t$  are certain states and the observation has certain properties. We have used the C++ based OpenNLP CRF++ package<sup>4</sup>.

## 3 Part of Speech Tagging in Bengali

Bengali is one of the widely used languages all over the world. It is the seventh popular language in the world, second in India and the national language of Bangladesh. In this work, we have developed a part of speech (POS) tagger for Bengali using the statistical Condition Random Field. Along with the word level suffix features, a lexicon (Ekbal and Bandyopadhyay, 2007a) and an HMM based Named Entity Recognizer (Ekbal et al., 2007b) have been used to tackle the unknown words, which in turn improve the accuracy of the POS tagger.

### 3.1 Features in Bengali POS Tagging

Feature selection plays a crucial role in the CRF framework. Experiments were carried out to find out the most suitable features for POS tagging in Bengali. The main features for the POS tagging have been identified based on the different possible combination of available word and tag context. The features also include prefix and suffix for all words. The term prefix/suffix is a sequence of first/last few characters of a word, which may not be a linguistically meaningful prefix/suffix. The use of prefix/suffix information works well for highly inflected languages like the Indian lan-

<sup>4</sup><http://crfpp.sourceforge.net>

guages. We have considered different combination from the following set for inspecting the best feature set for POS tagging:

$F = \{ w_{i-m}, \dots, w_{i-1}, w_i, w_{i+1}, \dots, w_{i+n} \}$ ,  $|\text{prefix}| \leq n$ ,  $|\text{suffix}| \leq n$ , Named Entity information, Previous POS tag, Length of a word, Lexicon, Digit features, Symbol features, Gazetteer lists}

Following is the set of features that have been applied to the POS tagging:

**Context word feature:** The surrounding words of a particular word might be used as a feature.

• **Word suffix:** Word suffix information is helpful to identify the POS information of a word. This feature can be used in two different ways. The first and the naïve one is to use a fixed length word suffix of the current and/or the surrounding word(s) as features. More helpful approach is to modify the feature as binary valued. Variable length suffixes of a word can be matched with predefined lists of useful suffixes for different classes. The different inflections that may occur with the noun, verb and adjective words have been considered.

**Word prefix:** Prefix information of a word is also helpful. A fixed length of the current and/or the surrounding word(s) might be treated as features.

• **Part of Speech (POS) Information:** The POS tag of the previous word can be used as a feature. This is the only dynamic feature in the experiment and denoted by the bigram template feature of CRF.

• **Named Entity Information:** The named entity (NE) information plays an important role in the overall accuracy of the POS tagger. In order to use this feature, an HMM based Named Entity Recognition (NER) system (Ekbal et al., 2007b) has been used to tag the training corpus (used for POS tagging) with the four major NE classes namely, ‘Person name’, ‘Location name’, ‘Organization name’ and ‘Miscellaneous name’. The NER system has been developed using a portion of the partially NE tagged Bengali news corpus (Ekbal and Bandyopadhyay, 2007c), developed from the archive of a leading Bengali newspaper available in the web. This NER system has demonstrated the F-Score value of 84.5% with 10-fold cross validation test on 150K wordforms.

The NE information can be used in two different ways. The first one is to use the NE information at the time of training the CRF model. In this case, the NE tags of the current and/or the surrounding

word (s) can be used as features of CRF. The alternative way is to use the NE information at the time of testing. In order to do this, the test set is passed through the HMM based NER system. The output of the NER system is given more priority than the output of the POS tagger for the unknown word in the test set. In the final output, these assigned NE tags are replaced appropriately by the corresponding POS tags.

**Length of a word:** Length of a word might be used as an effective feature of POS tagging. If the length of the current token is more than three then the feature ‘LengthWord’ is set to 1; otherwise, it is set to 0.

○ **Lexicon Feature:** A lexicon (Ekbal and Bandyopadhyay, 2007a) in Bengali has been used in the present POS tagging experiment. The lexicon has been developed in an unsupervised way from the Bengali news corpus (Ekbal and Bandyopadhyay, 2007c) of 34-million wordforms. Lexicon contains the Bengali root words and their basic POS information such as: noun, verb, adjective, pronoun and indeclinable. The lexicon contains 100,000 entries.

This lexicon can be used in three different ways. The first one is to use this as a binary valued feature of the CRF model. To apply this, three different features are defined for the open class of words as follows:

(a). If the current word is found to appear in the lexicon with the ‘noun’ POS, then the feature ‘nounLexicon’ is set to 1; otherwise, set to 0.

(b). If the current word is in the lexicon with the ‘verb’ POS, then the feature ‘verbLexicon’ is set to 1; otherwise, set to 0.

(c). If the current word is found to appear in the lexicon with the ‘adjective’ POS, then the feature ‘adjectiveLexicon’ is set to 1; otherwise, set to 0.

This binary valued feature is not considered for the closed classes of words like pronouns and indeclinable. The intention of using such feature was to distinguish the noun, verb and adjective words from the others.

Five different classes have been defined for using the lexicon as a feature of second category. A feature ‘LexiconClass’ is set to 1, 2, 3, 4 or 5 if the current word is in the lexicon and has ‘noun’, ‘verb’, ‘adjective’, ‘pronoun’ or ‘indeclinable’ POS, respectively.

The third way is to use this lexicon during testing. For an unknown word, the POS information extracted from the lexicon is given more priority

than the POS information assigned to that word by the CRF model. An appropriate tag conversion routine has been developed to map the five basic POS tags to the 26 POS tags.

Made up of digits: For a token if all the characters are digits then the feature ‘ContainsDigit’ is set to 1, otherwise, set to 0. It helps to identify QFNUM (Quantifier number) tags.

●Contains symbol: If the current token contains special symbol (e.g., %, \$ etc.) then the feature ‘ContainsSymbol’ is set to 1; otherwise, it is set to 0. This helps to recognize QFNUM (Quantifier number) and SYM (Symbols) tags.

Gazetteer Lists: Various gazetteer lists have been developed from the Bengali news corpus (Ekbal and Bandyopadhyay, 2007c). Gazetteer lists also include the noun, verb and adjective inflections that have been identified by analyzing the various words of the Bengali news corpus. The simplest approach of using these inflection lists is to check whether the current word contains any inflection of a particular list and make decisions. But this approach is not good, as it cannot resolve ambiguity. So, it is better to use these lists as the features of the CRF. The following is the list of gazetteers:

(i). Noun inflection list (27 entries): This list contains the inflections that occur with noun words. If the current word has any one of these inflections then the feature ‘NounInflection’ is set to 1; otherwise, set to 0.

(ii). Adjective inflection list (81 entries): It has been observed that adjectives in Bengali generally occur in four different forms based on the inflections attached. The first type of adjectives can form comparative and superlative degree by attaching the inflections (e.g., *-tara* and *-tamo* etc.) to the adjective root word. The second set of inflections (e.g., *-gato*, *-karo* etc.) make the words adjectives while get attached with the noun words. The third group of inflections (e.g., *-janok*, *-sulav* etc.) identifies the POS of the wordform as adjective. These three sets of inflections are included in a single list. A binary valued feature ‘AdjectiveInflection’ is then defined as: if the current word contains any inflection of the list then ‘AdjectiveInflection’ is set to 1; otherwise, set to 0.

(iii). Verb inflection list (327 entries): In Bengali, the verbs can be organized into twenty different groups according to their spelling patterns and the different inflections that can be attached to them. Original wordform of a verb word often changes

when any suffix is attached to the verb. If the current word contains any inflection of this list then the value of the feature ‘VerbInflection’ is set to 1; otherwise, set to 0.

(iv). Frequent word list (31, 000 entries): A list of most frequently occurring words in the Bengali news corpus (Ekbal and Bandyopadhyay, 2007c) has been prepared. The feature ‘RareWord’ is set to 1 for those words that are in this list; otherwise, set to 0.

(v). Function words: A list of function words has been prepared. The feature ‘NonFunctionWord’ is set to 1 for those words that are in this list; otherwise, the feature is set to 0.

### 3.2 Handling of Unknown Words

Handling of unknown words is an important issue in POS tagging. For words, which were not seen in the training set,  $P(t_i | w_i)$  is estimated based on the features of the unknown words, such as whether the word contains a particular suffix. The list of suffixes has been prepared. This list contains 435 suffixes; many of them usually appear at the end of verb, noun and adjective words. The probability distribution of a particular suffix with respect to specific POS tags is calculated from all words in the training set that share the same suffix.

In addition to word suffixes, a lexicon (Ekbal and Bandyopadhyay, 2007a) and a named entity recognizer (Ekbal and Bandyopadhyay, 2007b) have been used to tackle the unknown word problems. The procedure is given below:

**Step 1:** Find the unknown words in the test set.

**Step 2:** The system considers the NE tags for those unknown words that are not found in the lexicon

**Step 2.1:** The system replaces the NE tags by the appropriate POS tags (NNPC [Compound proper noun] and NNP [Proper noun]).

**Step 3:** The system assigns the POS tags, obtained from the lexicon, for those words that are found in the lexicon. The system assigns the NN (Common noun), VFM (Verb finite main), JJ (Adjective), PRP (Pronoun) and PREP (Postpositions) POS tags to the noun, verb, adjective, pronoun and indeclinable, respectively.

**Step 4:** The remaining unknown words are tagged using the word suffixes.

## 4 Experimental Results

The CRF based POS tagger has been trained on a corpus of 72,341 wordforms. This 26-POS tagged training corpus was obtained from the NLPAL\_Contest06 and SPSAL2007 contest data. The NLPAL\_Contest06 data was tagged with a tagset of 27 POS tags and had 46,923 wordforms. This data has been converted into the 26-POS<sup>5</sup> tagged data by defining appropriate mapping. The SPSAL2007 contest data was tagged with 26 POS tags and had 25,418 wordforms. Out of 72,341 wordforms, around 15K POS tagged corpus has been selected as the development set and the rest has been used as the training set of the CRF based POS tagger.

We define the *baseline* model as the one where the NE tag probabilities depend only on the current word:

$$P(t_1, t_2, t_3, \dots, t_n | w_1, w_2, w_3, \dots, w_n) = \prod_{i=1..n} P(t_i, w_i)$$

In this model, 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, named entity recognizer and word suffix lists.

Fifty four different experiments were conducted taking the different combinations from the set ‘F’ to identify the best-suited set of features for the POS tagging task. From our empirical analysis, we found that the following combination gives the best result with 685 iterations:

F={  $w_i - 2w_i - 1w_iw_{i+1}$ , |prefix|≤3, |suffix|≤3, POS tag of the previous word, NE tags of the current and the previous words, Lexicon features, Symbol feature, Digit feature, Gazetteer lists }

The meanings of the notations, used in the experiments, are defined below:

pw, cw, nw: Previous, current and the next word;  
 pwi, nwi: Previous and the next ith word from the current word;  
 pre, suf: Prefix and suffix of the current word;  
 pp: POS tag of the previous word;  
 pn2, pn, cn, nn: NE tag of the previous to previous, previous, current and the next word.

Evaluation results of the system for the development set are presented in Table 1. It is observed from the experimental results (from 2<sup>nd</sup> -5<sup>th</sup> rows) that the word window [-2, +1] gives the best result.

The accuracy of the POS tagger increases to 73.12% by including the POS information of the previous word. Results show that inclusion of prefix and suffix features improve the accuracy. Observations from the evaluation results (7<sup>th</sup> and 8<sup>th</sup> rows) suggest that prefix and suffix of length upto three of the current word is more effective. In another experiment, we have also observed that the surrounding word prefixes and/or suffixes do not increase the accuracy. The accuracy of the POS tagger is further increased by 1.61% (8<sup>th</sup> and 9<sup>th</sup> rows) with the introduction of digit, symbol and word-length features.

Feature (word, tag)	Accuracy (in %)
pw, cw, nw	66.31
pw2, pw, cw, nw, nw2	69.23
pw3, pw2, pw, cw, nw, nw2, nw3	68.12
pw2, pw, cw, nw	70.13
pw2, pw, cw, nw, pp	73.12
pw2, pw, cw, nw, pp,  suf ≤4,  pre ≤4	76.30
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3	78.71
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word	80.32
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, pn2, pn, cn, nn	81.31
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, pn, cn, nn	82.23
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, pn, cn	83.88
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, cn, nn	83.56
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, cn	83.43
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, pn, cn, Lexicon features	85.61
pw2, pw, cw, nw, pp,  suf ≤3,  pre ≤3, ContainsDigit, ContainsSYM, Length-Word, pn, cn, Lexicon features, Gazetteer lists	<b>88.30</b>

Table1: Results on the Development Set

Experimental results clearly show that the accuracy of the tagger can be improved significantly with the NE information. It is also indicative (10<sup>th</sup>-

<sup>5</sup>[http://shiva.iit.ac.in/SPSAL2007/iit\\_tagset\\_guidelines.pdf](http://shiva.iit.ac.in/SPSAL2007/iit_tagset_guidelines.pdf)

14<sup>th</sup> rows) that the NE information of the window [-1, 0] is more effective than the NE information of the window [-2, +1], [-1, +1], [0, +1] or the current word. It is observed from the evaluation results (12<sup>th</sup> and 15<sup>th</sup> rows) that the accuracy can be increased by 1.73% with the lexicon features, particularly ‘nounLexicon’, ‘verbInflection’, ‘adjectiveInflection’ and ‘LexiconClass’ features. Finally, an accuracy of 88.3% is obtained with the inclusion of various gazetteer lists in the form of noun, verb and adjective inflection lists along with the frequent word and function word lists.

Evaluation results of the POS tagger by employing various mechanisms for handling the unknown words are presented in Table 2. The POS tagger has shown the highest accuracy of 92.1% for the development set by introducing the various mechanisms, such as word suffix features, named entity recognizer and lexicon, for handling the unknown words.

Finally, the POS tagger has been tested with the test set of 20K wordforms. Evaluation results of the POS tagger along with the *baseline* model are presented in Table 3. The system has demonstrated an accuracy of 90.3%, which is an improvement of 3.9% with the inclusion of different mechanisms for handling unknown words.

Model	Accuracy (in %)
CRF	88.3
CRF + NER	90.4
CRF + NER + Lexicon	91.6
CRF + NER + Lexicon + Unknown word features	<b>92.1</b>

Table 2: Overall Evaluation Results of the Development Set

Model	Accuracy (in %)
Baseline	55.9
CRF	86.4
CRF + NER	88.7
CRF + NER + Lexicon	89.9
CRF + NER + Lexicon + Unknown word features	<b>90.3</b>

Table 3: Overall Evaluation Results of the Test Set

## 5 Conclusion

We have developed a POS tagger using the statistical CRF framework that has good accuracy with the contextual window [-2, +1], prefix and suffix of length upto three, NE information of the current

and the previous words, POS information of the previous word, digit features, symbol features and the various gazetteer lists. The accuracy of this system has been improved significantly by incorporating several techniques for handling unknown word problem. Developing POS taggers using other methods like ME and SVM will be other interesting experiments.

## References

- Avinesh PVS, Karthik G. 2007. Part Of Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning. In *Proc. of SPSAL2007, IJCAI*, India, 21-24.
- Brants, T. 2000. TnT-A Statistical Part of Speech Tagger. In *Proc. of the 6<sup>th</sup> ANLP Conference*, 224-231.
- Cutting, D., J. Kupiec, J. Pederson and P. Sibun. 1992. A Practical Part of Speech Tagger. In *Proc. of the 3<sup>rd</sup> ANLP Conference*, 133-140.
- Ekbal, A., and S. Bandyopadhyay. 2007a. Lexicon Development and POS tagging using a Tagged Bengali News Corpus. In *Proc. of FLAIRS-2007*, Florida, 261-263.
- Ekbal, A., S. Naskar and S. Bandyopadhyay. 2007b. Named Entity Recognition and Transliteration in Bengali. *Named Entities: Recognition, Classification and Use, Special Issue of Lingvisticae Investigationes Journal*, 30:1 (2007), 95-114.
- Ekbal, A., and S. Bandyopadhyay. 2007c. A Web-based Bengali News Corpus for Named Entity Recognition. *Language Resources and Evaluation Journal*, To appear by December 2007.
- Lafferty, J., McCallum, A., and Pereira, F. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. of the 18<sup>th</sup> ICML'01*, 282-289.
- Merialdo, B. 1994. Tagging English Text with Probabilistic Model. *Computational Linguistics*, 20(2): 155-172.
- Ratnaparkhi, A. 1996. A Maximum Entropy Part of Speech Tagger. In *Proc. of the EMNLP Conference*, 133-142.
- Sandipan Dandapat. 2007. Part Of Specch Tagging and Chunking with Maximum Entropy Model. In *Proc. of SPSAL2007, IJCAI*, India, 29-32.
- Sha, F. and Pereira, F. 2003. Shallow Parsing with Conditional Random fields. In *Proc. of NAACL-HLT*, Canada, 134-141.
- S. Singh , K. Gupta , M. Shrivastava and P. Bhattacharya. 2006. Morphological Richness Offsets Resource Demand - Experiences in Constructing a POS Tagger for Hindi. In *Proc. of COLING/ACL*, 779-786.