Maximum Entropy Based Bengali Part of Speech Tagging

Asif Ekbal[1], Rejwanul Haque[2], and Sivaji Bandyopadhyay[3]

Computer Science and Engineering Department, Jadavpur University, Kolkata, India
asif.ekbal@gmail.com[1], rejwanul@gmail.com[2], sivaji_cse_ju@yahoo.com[3]

Abstract. Part of Speech (POS) tagging can be described as a task of doing automatic annotation of syntactic categories for each word in a text document. This paper presents a POS tagger for Bengali using the statistical Maximum Entropy (ME) model. 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 with a training corpus of 72, 341 word forms and it uses a tagset [1] of 26 different POS tags, defined for the Indian languages. A part of this corpus has been selected as the development set in order to find out the best set of features for POS tagging in Bengali. The POS tagger has demonstrated an accuracy of 88.2% for a test set of 20K word forms. 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. Performance of this system has been compared with a Hidden Markov Model (HMM) based POS tagger and it has been shown that the proposed ME based POS tagger outperforms the HMM based tagger.

Keywords: Part of Speech Tagging, Maximum Entropy Model, Bengali.

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. Part of speech 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 tagging for natural language texts are developed using linguistic rules, stochastic models and a combination of both. Stochastic models [1] [2] [3] 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,

[1] http://shiva.iiit.ac.in/SPSAL2007/iiit_tagset_guidelines.pdf

Advances in Natural Language Processing and Applications
Research in Computing Science 33, 2008, pp. 67-78

Received 28/10/07
Accepted 07/12/07
Final Version 18/01/08

German and other European languages, for which large labeled data are 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 also difficult and complicates the smoothing typically used in such taggers. In contrast, a Maximum Entropy (ME) based method [4] or a Conditional Random Field based method [5] can deal with diverse, overlapping features. A POS tagger has been proposed in [6] for Hindi, which uses an annotated corpus (15,562 words collected from the BBC news site), exhaustive morphological analysis backed by high coverage lexicon and a decision tree based learning algorithm (CN2). The accuracy was 93.45% for Hindi with a tagset of 23 POS tags.

International Institute of Information Technology (IIIT), Hyderabad, India initiated a POS tagging and chunking contest, NLPAI ML [2] 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 SPSAL workshop [3] 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 [7], 77.61% for Bengali [8] and 77.37% for Telugu [7].

In this paper, we have developed a POS tagger based on the statistical Maximum Entropy (ME) model. The POS tagger produces an accuracy of 84.6% for the development set with the contextual window of size three, prefixes and suffixes of length up to three characters of the current word, NE information of the current and previous words, POS information of the previous word, digit features, symbol features, length of the word and the various inflection lists. It has been experimentally shown that the accuracy of the POS tagger can be improved significantly by considering unknown word features, named entity recognizer [9] and a lexicon [10] as the means of dealing with the unknown words. Evaluation results with a test set of 20K word forms shows the effectiveness of the proposed model with an accuracy of 88.2%.

The rest of the paper is organized as follows. Section 2 describes briefly the Maximum Entropy framework. Section 3 elaborately describes our approach to the POS tagging task. Experimental results of the development and the test sets are reported in Section 4. Finally, Section 5 concludes the paper.

2 Maximum Entropy Model

The Maximum Entropy framework estimates probabilities based on the principle of making as few assumptions as possible, other than the constraints imposed. Such constraints are derived from the training data, expressing some relationships between features and outcome. The probability distribution that satisfies

[2] http://ltrc.iiitnet/nlpai_contest06/proceedings.php
[3] http://shiva.iiit.ac.in/SPSAL2007/SPSAL-Proceedings.pdf

the above property is the one with the highest entropy. It is unique, agrees with the maximum likely-hood distribution, and has the exponential form:

$$P(t|h) = \frac{1}{Z(h)} exp(\sum_{j=1}^{n} \lambda_j f_j(h,t)) \quad (1)$$

where, t is the POS tag, h is the context (or history), $f_j(h,t)$ are the features with associated weight λ_j and $Z(h)$ is a normalization function.

The problem of POS tagging can be formally stated as follows. Given a sequence of words $w_1, \ldots, w_n$, we want to find the corresponding sequence of POS tags $t_1, \ldots, t_n$, drawn from a set of tags T, which satisfies:

$$P(t_1, \ldots, t_n|w_1, \ldots, w_n) = \prod_{i=1,2\ldots,n} P(t_i|h_i) \quad (2)$$

where, h_i is the context for the word w_i.

The *Beam search algorithm* is then used to select the sequence of word classes with the highest probability.

The features are binary/multiple valued functions, which associate a POS tag with various elements of the context. For example:

$$f_j(h,t) = 1 \text{ if } word(h) = sachin \text{ and } t = NNP \quad (3)$$

$$= 0 \quad \text{otherwise} \quad (4)$$

The general-purpose optimization method *Limited Memory BFGS method* [11] has been used for the estimation of MaxEnt parameters. We have used the C++ based Maximum Entropy package. [4]

3 Our Approach for Part Of Speech Tagging in Bengali

Bengali is one of the widely used languages all over the world. In terms of native speakers, it is the seventh popular language in the world, second in India and the national language of Bangladesh. The works on POS tagging in Indian languages, particularly in Bengali, has started very recently as there was neither any standard POS tagset nor any available tagged corpus just one/two years ago. In this work, we have used a Maximum Entropy approach for the task of POS tagging. Along with the variety of contextual and word level features, a lexicon [10] and a HMM based named entity recognizer [9] have been used to improve the accuracy of the POS tagger.

3.1 Features

Feature selection plays a crucial role in the ME framework. Experiments have been carried out to find out the most suitable features for POS tagging in Bengali. The main features for the POS tagging task have been identified based

[4] http://homepages.inf.ed.ac.uk/s0450736/software/maxent/maxent-20061005.tar.bz2

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 languages. We have considered different combinations from the following set for inspecting the best feature set for POS tagging in Bengali.

F=$\{w_{i-m}, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots w_{i+n}$, |prefix| $\leq n$, |suffix| $\leq n$, Current and/or the surrounding named entity (NE) tag(s), Previous POS tag(s), First word, Lexicon, Digit information, Length of the word, Inflection lists$\}$.

Following are details of the set of features that have been applied for POS tagging in Bengali:

• Context word feature: Preceding and following words of a particular word might be used as a feature.

• Word suffix:Word suffix information is helpful to identify POS class. This feature can be used in two different ways. The first and the naive one is, a fixed length (say, n) word suffix of the current and/or the surrounding word(s) can be treated as features. If the length of the corresponding word is less than or equal to *n-1* then the feature values are not defined and denoted by ND. The feature value is also not defined (ND) if the token itself is a punctuation symbol or contains any special symbol or digit. The second and the more helpful approach is to modify the feature as binary/mutiple valued. Variable length suffixes of a word can be matched with predefined lists of useful suffixes for different classes. Here, both the suffixes have been used. The second type of suffixes has been described later in the form of noun, verb and adjective inflections.

• Word prefix: Prefix information of a word is also helpful. A fixed length (say, n) prefix of the current and/or the surrounding word(s) can be considered as features. This feature value is not defined (ND) if the length of the corresponding word is less than or equal to *n-1* or the word is a punctuation symbol or the word contains any special symbol or digit.

• Part of Speech (POS) Information: POS information of the previous word(s) might be used as a feature. This is the only dynamic feature in the experiment.

• Named Entity Information: The named entity (NE) information of the current and/or the surrounding word(s) 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 [9] has been used. The NER system uses the four major NE classes namely, *Person name*, *Location name*, *Organization name* and *Miscellaneous name*. Date, time, percentages, numbers and monetary expressions belong to the *Miscellaneous name* category. The other words are assigned the 'NNE' tags. The NER system was developed using a portion of the Bengali news corpus [12], developed from the archive of a leading Bengali newspaper available in the web. This NER system has demonstrated 84.5% F-Score value during 10-fold cross validation test with a training corpus of 150k wordforms.

The NE information can be used in two different ways. The first one is to use the NE tag(s) of the current and/or the surrounding word(s) as the features

of the ME model. The second way is to use this NE information at the time of testing. In order to do this, the test set is passed through the HMM based NER system. Outputs of the NER system are given more priorities than the outputs of the POS tagger for the unknown words in the test set. The NE tags are then replaced appropriately by the POS tags (NNPC: Compound proper noun, NNP: Proper noun and QFNUM: Quantifier number).

• Lexicon Feature: A lexicon in Bengali has been used to improve the performance of the POS tagger. The lexicon [10] has been developed in a semi-supervised way from the Bengali news corpus [12] of 34-million word forms. It contains the Bengali root words and their basic POS information such as: noun, verb, adjective, pronoun and indeclinable. The lexicon has 100,000 entries.

This lexicon can be used in two different ways. One way is to use this as the features of the ME model. To apply this, five different features are defined for the open class of words as follows:

1. If the current word is found to appear in the lexicon with the 'noun' POS, then the feature 'Lexicon' is set to 1.
2. If the current word is found to appear in the lexicon with the 'verb' POS, then the feature 'Lexicon' is set to 2.
3. If the current word is found to appear in the lexicon with the 'adjective' POS, then the feature 'Lexicon' is set to 3.
4. If the current word is found to appear in the lexicon with the 'pronoun' POS, then the feature 'Lexicon' is set to 4.
5. If the current word is found to appear in the lexicon with the 'indeclinable' POS, then the feature 'Lexicon' is set to 5.

The intention of using this feature was to distinguish the noun, verb, adjective, pronoun and indeclinable words among themselves.

The second or the alternative 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 ME model. An appropriate mapping has been defined from these 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, it is set to 0. It helps to identify QFNUM (Quantifier number) tag.

•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 SYM (Symbols) and QFNUM (Quantifier number) 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. The motivation of using this feature was to distinguish proper nouns from the other words. We have observed that very short words are rarely proper nouns.

•Frequent word list: A list of most frequently occurring words in the training corpus has been prepared. The words that occur more than 10 times in the entire training corpus are considered to be the frequent words. The feature 'RareWord'

is set to 1 for those words that are in this list; otherwise, it is set to 0.
•Function words: A list of function words has been prepared manually. This list has 743 number of entries. The feature 'FunctionWord' is set to 1 for those words that are in this list; otherwise, the feature is set to 0.
•Inflection Lists: Various inflection lists have been created manually by analyzing the various classes of words in the Bengali news corpus [12]. A simple approach of using these inflection lists is to check whether the current word contains any inflection of these lists and to take decision accordingly. Following are the lists of inflections:

- Noun inflection list: A list of inflections that occur with noun words has been manually prepared by observing the various noun words in the Bengali news corpus. This list contains 27 entries. If the current word has any one of these inflections then the feature 'Inflection' is set to 1.
- Adjective inflection list: It has been observed that adjectives in Bengali generally occur in four different forms based on the suffixes attached. The first type of adjectives can form comparative and superlative degree by attaching the suffixes (e.g., *-tara* and *-tamo*) to the adjective word. The second set of suffixes (e.g., *-gato*, *-karo* etc.) make the words adjectives while get attached with the noun words. The third group of suffixes (e.g., *-janok*, *-sulav* etc.) identifies the POS of the word form as adjective. These three set of suffixes are included in a single list and a feature 'AdjectiveInflection' is defined as: if the current word contains any suffix of the list then 'Inflection' is set to 2. This list has been manually constructed by looking at the different adjective words of the Bengali news corpus. At present, this list has 194 entries.
- Verb inflection list: In Bengali, the verbs can be organized into 20 different groups according to their spelling patterns and the different inflections that can be attached to them. Original word-form of a verb word often changes when any suffix is attached to the verb. At present, there are 214 different entries in the verb inflection list. If the current word is in the verb inflection list then the feature 'Inflection' is set to 3.
  The feature 'Inflection' is set to 0 for those words that do not contain any of these inflections.

3.2 Unknown Word Handling Techniques

Handling of unknown word 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 tag is calculated from all words in the training set that share the same suffix.

In addition to the unknown word suffixes, a named entity recognizer [9] and a lexicon [10] have been used to tackle the unknown word problems. The details of the procedure is given below:

1. Step 1: Find the unknown words in the test set.
2. Step 2: The system assigns the POS tags, obtained from the lexicon, to those unknown words that are found in the lexicon. For noun, verb and adjective words of the lexicon, the system assigns the NN (Common noun), VFM (Verb finite main) and the JJ (Adjective) POS tags, respectively.
   Else
3. Step 3: The system considers the NE tags for those unknown words that are not found in the lexicon
   (a) Step 2.1: The system replaces the NE tags by the appropriate POS tags (NNPC [Compound proper noun] and NNP [Proper noun]).
   Else
4. Step 4: The remaining words are tagged using the unknown word features accordingly.

4 Experimental Results

The ME based POS tagger has been trained on a corpus of 72,341 word forms tagged with the 26 POS tags, defined for the Indian languages. This 26-POS tagged training corpus was obtained from the NLPAI ML Contest-2006 [5] and SPSAL-2007 [6] contest data. The NLPAI ML 2006 contest data was tagged with 27 different POS tags and had 46,923 wordforms. This POS tagged data was converted into the 26-POS [7]tagged data by defining appropriate mapping. The SPSAL-2007 contest data was tagged with 26 POS tags and had 25,418 wordforms. Out of 72,341 word forms, around 15K word forms have been selected as the development set and the rest has been used as the training set of the ME based tagger in order to find out the best set of features for POS tagging in Bengali.

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

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

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 suffixes.

Thirty two different experiments were conducted taking the different combinations from the set 'F' to identify the best-suited set of features for POS tagging in Bengali. From our empirical analysis, we have found that the following combination gives the best result for the development set:
F=[w_{i-1}, w_i, w_{i+1}, |prefix| ≤ 3, |sufix| ≤ 3, NE tags of the current and previous

[5] http://ltrc.iiitnet/nlpai_contest06/data2
[6] http://shiva.iiit.ac.in/SPSAL2007/check_login.php
[7] http://shiva.iiit.ac.in/SPSAL2007/iiit_tagset_guidelines.pdf

words, POS tag of the previous word, Lexicon feature, Symbol feature, Digit feature, Length feature, Inflection 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; pre, suf: Prefix and suffix of the current word; pp: POS tag of the previous word; ppi: POS tag of the previous ith word; pn, cn, nn: NE tags of the previous, current and the next word; pni: NE tag of the previous ith word. Evaluation results of the system for the development set are presented in Tables 1-2.

Table 1. Experimental results for the development set

Feature (word, tag)	Accuracy (in %)
pw, cw, nw	63.11
pw2, pw, cw, nw, nw2	**67.22**
pw3, pw2, pw, cw, nw, nw2, nw3	65.12
pw2, pw, cw, nw, nw2, pp	69.72
pw2, pw, cw, nw, nw2, pp, pp2	69.12
pw2, pw, cw, nw, nw2, pp, $\|pre\| \leq 6$, $\|suf\| \leq 6$	70.42
pw2, pw, cw, nw, nw2, pp, $\|pre\| \leq 5$, $\|suf\| \leq 5$	71.7
pw2, pw, cw, nw, nw2, pp, $\|pre\| \leq 4$, $\|suf\| \leq 4$	72.10
pw2, pw, cw, nw, nw2, pp, $\|suf\| \leq 3$, $\|pre\| \leq 3$	73.81
pw, cw, nw, pp, $\|suf\| \leq 3$, $\|pre\| \leq 3$	**75.16**
pw, cw, nw, pp, $\|pre\| \leq 3$, $\|ppre\| \leq 3$, $\|psuf\| \leq 3$, $\|suf\| \leq 3$	73.36

Evaluation results (3rd row) of Table 1 show that word window [−2,+2] gives the best result without any feature. Results also show the fact that further increase (4th row) or decrease (2nd row) in window size reduces the accuracy of the POS tagger. Experimental results (5th and 6th rows) show that the accuracy of the POS tagger can be improved by including the POS information of the previous word(s). Clearly, it is seen that POS information of the previous word only is more helpful. Experimental results (7th-11th rows) show the effectiveness of prefixes and suffixes up to a particular length for the highly inflective Indian languages as like Bengali. Results (11th row) also show that the window $[-1, +1]$ yields better result than the window $[-2, +2]$ with the same set of features. Observations from the results (12th row) suggest that the surrounding word suffixes and/or prefixes do not increase the accuracy.

It can be decided from the results (2nd-5th rows) of Table 2 that the named entity (NE) information of the current and/or the surrounding word(s) improves the overall accuracy of the POS tagger. It is also indicative from this results (3rd row) that the NE information of the previous and current words, i.e, within the window [−1,0] is more effective than the NE information of the windows [−1,+1], [0,+1] or the current word alone. An improvement of 2.1% in the overall accuracy

Table 2. Experimental results for the development set

Feature (word, tag)	Accuracy (in %)
pw, cw, nw, pp, pn, cn, nn, $\|suf\| \leq 3$, $\|pre\| \leq 3$	77.1
pw, cw, nw, pp, pn, cn, $\|suf\| \leq 3$, $\|pre\| \leq 3$	**78.2**
pw, cw, nw, pp, cn, nn, $\|pre\| \leq 3$, $\|suf\| \leq 3$	77.8
pw, cw, nw, pp, cn, $\|pre\| \leq 3$, $\|suf\| \leq 3$	76.2
pw, cw, nw, pp, pn, cn, $\|pre\| \leq 3$, $\|suf\| \leq 3$, Digit, Symbol, Length, FunctionWord	80.3
pw, cw, nw, pp, pn, cn, $\|pre\| \leq 3$, $\|suf\| \leq 3$, Digit, Symbol, Length, FunctionWord, Lexicon	82.1
pw, cw, nw, pp, pn, cn, $\|pre\| \leq 3$, $\|suf\| \leq 3$, Digit, Symbol, FunctionWord, Lexicon, Inflection	84.2
pw, cw, nw, pp, pn, cn, $\|pre\| \leq 3$, $\|suf\| \leq 3$, Digit, Symbol, Length, Lexicon, Inflection, FunctionWord, RareWord	**84.6**

is observed (6th row) with the introduction of the 'Symbol', 'Digit', 'Length' and 'FunctionWord' features. The use of lexicon as the features of ME model further improves the accuracy by 1.8% (7th row). Accuracy of the POS tagger rises to 84.2% (8th row) by including the noun, verb and adjective inflections. Finally, an overall accuracy of 84.6% is obtained with the inclusion of 'RareWord' feature.

Evaluation results of the POS tagger by including the various mechanisms for handling the unknown words are presented in Table 3 for the development set. The table also shows the results of the *baseline* model.

Table 3. Overall evaluation results of the POS tagger for the development set

Model	Accuracy (in %)
Baseline	55.9
ME	84.6
ME + Lexicon (used for unknown word handling)	85.8
ME + Lexicon (used for unknown word handling) + NER (used for unknown word handling)	87.1
ME + Lexicon (used for unknown word handling) + NER (used for unknown word handling)+ Unknown word features	**88.9**

Clearly, the results of Table 3 show the effectiveness of employing the various strategies for handling the unknown words with the increase in the accuracy values. The ME model exhibits an accuracy of 88.9% with the lexicon, named entity recognizer and unknown word features. The system has demonstrated

an improvement in the overall accuracy by 1.2% with the use of lexicon [10] as a mean to handle the unknown word problems. The accuracy of the tagger is further improved by 1.3% when NER system [9] is included additionally to handle the unknown words. This improvement shows the effectiveness of NER during POS tagging. The overall accuracy of the POS tagger increases by 1.8% with the inclusion of unknown word features.

Now, the development set is included as part of the training set and thus the resultant training set consists of 72,341 word forms. A gold standard test set of 20K word forms has been used in order to report the evaluation results of the system. Experimental results of the system along with the *baseline* model are demonstrated in Table 4 for the test set. The POS tagger has demonstrated the overall accuracy of 88.2% for the test set.

Table 4. Experimental results of the system for the test set

Model	Accuracy (in %)
Baseline	54.7
ME	83.8
ME + Lexicon	84.9
ME + Lexicon + NER	86.3
ME + Lexicon + NER + Unknown word features	**88.2**

The performance of the proposed ME based POS tagger has been compared with a HMM based POS tagger [13], in which trigram model was considered. Additional context dependent features were considered for the emission probabilities in this HMM based POS tagger . Also, the lexicon [10], named entity recognizer [9] and unknown word features were considered in a similar way in order to handle the unknown word problems. This system has been trained and tested with the same data. Evaluation results of the HMM based POS tagger is presented in Table 5.

Table 5. Evaluation results of the HMM based POS tagger for the test set

Model	Accuracy (in %)
HMM	75.8
HMM + Lexicon	76.9
HMM + Lexicon + NER	78.1
HMM + NER + Lexicon + Unknown word features	**80.3**

Evaluation results of Table 4 and Table 5 show that the proposed ME based POS tagger outperforms the HMM based POS tagger [13] with the same training and test sets. So, it can be decided that the ME framework is more effective than the HMM framework in handling the morphologically complex Indian languages that contain diverse, overlapping features.

Error analysis of the POS tagger has been done with the help of confusion matrix. Since nouns appear most frequently in a corpus, unknown words have a tendency of being assigned noun tags (NN, in most cases). A close scrutiny of the confusion matrix suggests that some of the probable tagging errors facing the current POS tagger are NNC (Compound common noun) vs NN (Common noun), JJ (Adjective) vs NN (Common noun) and VFM (Verb finite main) vs VAUX (Common noun). A multiword extraction unit for Bengali would have taken care of the NNC vs NN problem. The problems of JJ vs NN is hard to resolve and probably requires the use of linguistic rules. The problems of VFM vs VAUX can be solved with the use of linguistic rules. Errors involving NNP vs NN are reduced but not fully removed even after the use of NER system. We need to improve the accuracy of the NER system further in order to remove this problem.

A close investigation to the evaluation results show that approximately 92% of the errors are involved with the unseen words in the test set. We have incorporated several mechanisms, as discussed earlier, for dealing with the unknown word problems. Experiments have been conducted to observe the accuracies of the unknown words separately. Evaluation results are presented in Table 6. It is observed from the evaluation results that unknown word features are the most effective in handling the unseen words followed by the NER system and the lexicon.

Table 6. Evaluation results of the POS tagger for the unknown words in the test set

Model	Accuracy (in %)
ME	68.4
ME + Lexicon	73.3
ME + Lexicon + NER	79.7
ME + Lexicon + NER + Unknown word features	**88.1**

5 Conclusion

We have developed a POS tagger using Maximum Entropy model that has good accuracy with the contextual window of size three, prefix and suffix of length three, NE information of the current and previous words, POS information of the previous word, digit features, symbol features, length of the word and the various inflection lists. The accuracy of this system has been improved significantly by

incorporating several techniques such as word level features, named entity recognizer and lexicon for handling unknown word problem. The performance of the proposed model has been compared with a HMM based POS tagger. The proposed system has demonstrated an accuracy of 88.2%, which is an improvement of approximately 8% over the HMM based tagger.

Analyzing the performance using other methods like Conditional Random Fields and Support Vector Machines (SVMs) will be other interesting experiments.

References

1. Cutting, D., Kupiec, J., Pedersen, J., Sibun, P.: A practical part-of-speech tagger. In: Proceedings of the Third Conference on Applied Natural Language Processing. (1992) 133–140
2. Merialdo, B.: Tagging english text with a probabilistic model. Comput. Linguist. **20**(2) (1994) 155–171
3. Brants, T.: TnT a Statistical Parts-of-Speech Tagger. In: Proceedings of the Sixth Conference on Applied Natural Language Processing ANLP-2000. (2000) 224–231
4. Ratnaparkhi, A.: A maximum entropy part-of -speech tagger. In: Proc. of EMNLP'96. (1996)
5. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In: ICML. (2001) 282–289
6. Singh, S., Gupta, K., Shrivastava, M., Bhattacharyya, P.: Morphological richness offsets resource demand-experiences in constructing a pos tagger for hindi. In: Proceedings of the COLING/ACL 2006. (2006) 779–786
7. Avinesh, P., Karthik, G.: Part Of Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning. In: Proceedings of IJCAI Workshop on Shallow Parsing for South Asian Languages, India (2007) 21–24
8. Dandapat, S.: Part Of Specch Tagging and Chunking with Maximum Entropy Model. In: Proceedings of the IJCAI Workshop on Shallow Parsing for South Asian Languages, Hyderabad, India (2007) 29–32
9. Ekbal, A., Naskar, S., Bandyopadhyay, S.: Named Entity Recognition and Transliteration in Bengali. Named Entities: Recognition, Classification and Use, Special Issue of Lingvisticae Investigationes Journal **30**(1) (2007) 95–114
10. Ekbal, A., Bandyopadhyay, S.: Lexicon Development and POS Tagging using a Tagged Bengali News Corpus. In: Proceedings of the 20th International Florida AI Research Society Conference (FLAIRS-2007), Florida (2007) 261–263
11. Malouf, R.: A comparison of algorithms for maximum entropy parameter estimation. In: Proceedings of Sixth Conf. on Natural Language Learning. (2002) 49–55
12. Ekbal, A., Bandyopadhyay, S.: A Web-based Bengali News Corpus for Named Entity Recognition. Language Resources and Evaluation Journal (accepted) (2007)
13. Ekbal, A., Mondal, S., Bandyopadhyay, S.: POS Tagging using HMM and Rule-based Chunking. In: Proceedings of the IJCAI Workshop on Shallow Parsing for South Asian Languages, Hyderabad, India (2007) 31–34