Voted Approach for Part of Speech Tagging in Bengali[*]

Asif Ekbal[a], Md. Hasanuzzaman[b], and Sivaji Bandyopadhyay[c]

[a]Department of Computational Linguistics, University of Heidelberg,
1m Neuenheimer Feld 325, 69120 Heidelberg, Germany
ekbal@cl.uni-heidelberg.de, asif.ekbal@gmail.com
[b]West Bengal Industrial Development Corporation, Kolkata, India
hasanuzzaman.im@gmail.com
[c]Department of Computer Science and Engineering, Jadavpur University,
Kolkata-700032, India
sivaji_cse_ju@yahoo.com, sbandyopadhyay@cse.jdvu.ac.in

Abstract. 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. In this paper, we report about our work on POS tagging for Bengali by combining different POS tagging systems using three weighted voting techniques. The individual POS taggers are based on Maximum Entropy (ME), Conditional Random Field (CRF) and Support Vector Machine (SVM) frameworks. The POS taggers use a tag set of 27 POS tags, defined for the Indian languages. The individual system makes use of the different contextual information of the words along with the variety of word-level features that are helpful in predicting the various POS classes. The POS tagger has been trained and tested with 57,341 and 35K tokens, 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 voted POS tagger with an accuracy of 92.35%, which is an improvement of 5.29% over the least performing ME based system and 2.23% over the best performing SVM based system.

Keywords: Part of Speech (POS) tagging, Statistical techniques, Voting, 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. POS tagging is a very important preprocessing task for various language processing activities. This helps in doing deep parsing of text and in developing information extraction systems, semantic processing etc. POS tagging for natural language texts are developed using linguistic rules, stochastic models or a combination of both. Stochastic models (Cutting et al., 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 data. 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.

[*] The authors thankfully acknowledge the comments and suggestions of Anette Frank, Department of Computational Linguistics, University of Heidelberg, Germany, during preparation of the camera ready version.

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) or a SVM based system (Kudo and Matsumoto, 2001) can deal with diverse and overlapping features.

The International Institute of Information Technology (IIIT), Hyderabad, India initiated a POS tagging contest, NLPAI ML-Contest06[1] 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[2] workshop in IJCAI-07, a competition on POS tagging and chunking for south Asian languages was conducted by IIIT, Hyderabad. The best POS tagging accuracies reported were 78.66% for Hindi (Karthik, 2007), 77.37% for Telugu (Karthik, 2007) and 77.61% for Bengali (Dandapat, 2007). An HMM based POS tagger has been reported in Ekbal et al. (2007) that make use of the additional context dependent information along with word suffixes, Named Entity Recognition (NER) system and lexicon for handling of unknown words. Further, the POS taggers for Bengali can be found in Ekbal et al. (2008) with ME, Ekbal et al. (2007) with CRF and in Ekbal and Bandyopadhyay (2008a) with a SVM based approach.

2 Our Approach for POS Tagging

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 to appear 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 developed POS taggers for Bengali using ME, CRF and SVM frameworks. These POS taggers have been combined together into a final system with the help of weighted voting techniques.

We have used the C^{++} based ME package[3] for building the ME based POS tagger. A number of POS tagging models have been built that are differentiated from each other by the features, which are included in the model. The system uses L-BFGS method (Malouf, 2002) to build the ME model, which is guaranteed to converge to a solution in this kind of problem. The sequential classification approach like ME can handle many correlated features but it suffers from the *label bias* problem. Careful feature selection is very essential in the ME framework. In contrast, CRF (Lafferty et al., 2001) is a sequential modeling framework that has all the advantages of ME and also solves the problem of *label bias* in a principled way. Moreover, CRFs bring together the best of generative and classification models. We have used the OpenNLP C^{++} based CRF++ package (http://crfpp.sourceforge.net). For parameter estimation, the system uses L-BFGS method (Sha and Pereira, 2003) to build the CRF model, which is guaranteed to converge to a solution in this kind of problem.

SVM (Vapnik, 1995) achieves high generalization even with training data of a very high dimension. Further, by introducing the *Kernel function*, SVMs handle non-linear feature spaces, and carry out training considering combinations of more than one feature. We have used the YamCha toolkit (http://chasen-org/~taku/software/yamcha) for training and TinySVM-0.07 (http://cl.aist-nara.ac.jp/~taku-ku/software/TinySVM) classifier for classification. A number of experiments have been carried out with the different degrees of the *polynomial kernel function.* Both the *one vs rest* and *pairwise* multi-class decision methods have been considered in the experiments.

[1] http://ltrc.iiitnet/nlpai_contest06

[2] http://shiva.iiit.ac.in/SPSAL2007

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

2.1 Indian Language POS Tagset

In 2006, two machine learning contests were organized on POS tagging and chunking for Indian Languages. Both the contests were conducted for three different Indian languages, namely Hindi, Bengali and Telugu. All the languages used a common tagset of 27 tags developed at the IIIT, Hyderabad (hereinafter referred as the IIIT-H tagset). The first contest was conducted by the NLP Association of India (NLPAI) and IIIT-H in the summer of 2006. In the NLPAI-2006 contest, each participating team worked on POS tagging for a single language of their choice. It was thus not easy to compare the different approaches. Keeping this in mind, the Shallow Parsing for South Asian Languages (SPSAL) contest, a workshop in the International Joint Conference on Artificial Intelligence (IJCAI) 2007, was held for a multilingual POS tagging and chunking.

All the tags used in the IIIT-H tagset[4] are broadly classified into three groups that have been listed below.

- **Group 1:**

All tags in this group are similar to the Penn tagset. Penn tagset makes finer distinction between singular and plural or comparative and superlative forms, which is not considered in the current tagset. This is in accordance with the policy about fineness and coarseness. Following are the set of tags that belong to this group.

NN-Noun, NNP-Proper Noun, PRP-Pronoun, VAUX-Verb Auxiliary, JJ-Adjective, RB-Adverb, RP-Particle, CC-Conjunction, UH-Interjection, SYM-Special Symbol.

- **Group 2:**

This group includes those tags that are a modification of some tags in the Penn tagset. The tags are listed below:

PREP-Postposition, QF-Quantifiers, QFNUM-Quantifier Number, VFM-Verb Finite Main, VJJ-Verb Non-Finite Adjectival, VRB-Verb Non-finite Adverbial, VNN-Verb Non-Finite Nominal, QW-Question Words.

- **Group 3:**

This set of new tags is designed to cater to some phenomena that are specific to Indian languages. This group contains the following tags.

NLOC-Noun Location, INTF-Intensifier, NEG-Negative, NNC-Compound Nouns, NNPC-Compound Proper Nouns, NVB-Noun in Kriyamula, JVB-Adjective in Kriyamula, RBVB-Adverb in Kriyamula, INF-Verb infinitival.

3 Features of POS Tagging

Maximum Entropy (ME) is a very flexible method of statistical modeling, which handles the sparse data problem. Under this model, a natural combination of several features can be easily incorporated, which cannot be done naturally in HMM models. Appropriate feature selection is a crucial issue in ME model as it does not provide a method for automatic selection of given feature sets. Unlike ME, CRF does not require careful feature selection in order to avoid overfitting. SVMs predict the classes depending upon the labeled word examples only. It predicts the POS tags based on feature information of words collected in a predefined window size while ME or CRF predicts them based on the information of the whole sentence. In particular, SVMs achieve high generalization even with training data of a very high dimension.

In the present work, we have used the same set of features for POS tagging using ME, CRF and SVM. Experiments have been carried out to identify the most suitable features for the POS tagging task in Bengali in each of these frameworks. The main features for POS tagging have been identified based on the different possible combinations of available words and tag context. The features also include prefix and suffix for all words. The term prefix/suffix is a sequence of

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

first/last few characters of a word, which may not necessarily be a linguistically meaningful prefix/suffix. The use of prefixes and suffixes as features has been found to be effective for highly inflected languages. A number of experiments have been carried out to find the most suitable set of features for POS tagging in each of the models from the following available set of features:

F={ $w_{i-m},...,w_{i-1},w_i,w_{i+1},...,w_{i+n}$, |prefix|≤n |suffix|≤n, Previous POS tag(s), NE tag (s) of the current and/or the surrounding words, First word, Lexicon feature, Digit information, Length of the word, Inflection feature}.

Below we give details about the set of features that have been applied for POS tagging in Bengali:

1. Context word feature: Preceding and following words of a particular word can be used as the features. This is based on the assumption that the surrounding words can play an effective role in deciding the POS tag of the current word.
2. Word suffix and prefix: Word suffix/prefix information is helpful to identify the POS class. A fixed length (say, *n)* word suffix/prefix of the current and/or the surrounding word(s) are used as the features. If the length of the corresponding word is less than or equal to *n-1* then the feature values are not defined (denoted by ND – Not Defined). The feature value is also not defined (ND) if the token itself is a punctuation symbol or contains any special symbol or digit. The variable length suffixes (i.e., inflection) that can appear with the different types of wordforms, particularly noun, verb and adjective words have been prepared. These have been used as the binary valued features in all the classifiers. There are 27 noun inflections, 214 verb inflections and 92 adjective inflections. The suffix/prefix has been used with the assumption that the words belonging to the same POS classes contain some common suffix/prefix. This feature works effectively for the highly inflective Indian languages like Bengali.
3. POS Information: POS information of the previous word(s) can play a crucial role in deciding the POS tag of the current word. This is the only dynamic feature in the experiment.
4. Named Entity (NE) Information: The NE information of the current and/or the surrounding word(s) does have an important role in the overall accuracy of the POS tagger. In order to use this feature, a SVM based NER system (Ekbal and Bandyopadhyay, 2008b) has been used. The NE tag(s) of the current and/or the surrounding word(s) have been used as the features in the ME/CRF/SVM based POS tagging models. The NE information has been included into the system in order to reduce the rate of errors that we faced in our earlier experiments for HMM based POS tagging (Ekbal et al., 2007). The confusion matrix of the HMM based POS tagger showed that most of the errors were concerned with NNP (Proper noun) vs. NN (Common noun).
5. Lexicon Feature: A lexicon (Ekbal and Bandyopadhyay, 2008c) in Bengali has been used to improve the performance of the POS tagger further. The lexicon has been developed using an unsupervised approach from a Bengali news corpus (Ekbal and Bandyopadhyay, 2008d), developed from the web-archive of a widely read Bengali news paper. Lexicon contains the root words and their basic POS information such as noun, verb, adjective, adverb and indeclinable (preposition, conjunction and interjection). The lexicon has 128K wordforms. This lexicon has been used in two different ways. One way is to use this as the features in any of the models. The intention of using this feature is to distinguish the noun, verb, adjective, pronoun, and indeclinable words from that of the NEs. To apply this, five different features are defined as follows:

- If the current word is found to appear in the lexicon with the ‘noun’ POS, then the feature ‘LEX’ is set to 1.
- If the current word is found to appear in the lexicon with the ‘verb’ POS, then the feature ‘LEX’ is set to 2.
- If the current word is found to appear in the lexicon with the ‘adjective’ POS, then the feature ‘LEX’ is set to 3.
- If the current word is found to appear in the lexicon with the ‘pronoun’ POS, then the feature ‘LEX’ is set to 4.
- If the current word is found to appear in the lexicon with the ‘indeclinable’ POS, then the feature ‘LEX’ is set to 5.
- If the current word appears with more than one POS then the feature ‘LEX’ is set to 0.

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 any of the models. An appropriate mapping has been defined from these five basic POS tags to the 27 POS tags.

6. Made up of digits: This is a binary valued feature and used to check whether the current token consists of digits only. It helps to identify the number expressions, particularly used for the QFNUM (Quantifier number) tags.

7. Contains symbol: This binary valued feature has been incorporated to check whether the current token contains any special symbol (e.g., %, $ etc.). This feature helps to recognize SYM (Symbols) and QFNUM (Quantifier number) tags.

8. Length of a word: Length of a word can be used as a feature for POS tagging. This is a binary valued feature that has been defined in order to check whether the length of the current token is more than *three* or not. The motivation of using this feature is to distinguish proper nouns from the other words. It has been observed that very short words are rarely proper nouns.

9. Frequent word list: A list of most frequently occurring words in the training corpus has been prepared. The words that occur with more than a particular threshold frequency in the entire training corpus are considered to be the frequent words. The value of the threshold frequency depends on the size of the training corpus. A binary valued feature is defined to check whether the current word appears in the list of frequently occurring words or not. This feature has been incorporated with the observation that the frequently occurring words are rarely proper names.

10. Function words: The list of function words has been used to extract a binary valued feature that checks whether the current word appears in this list. This list has 743 entries.

11. Inflection Lists: Various inflection lists have been created manually by analyzing the various classes of words in the Bengali news corpus (Ekbal and Bandyopadhyay, 2008d). 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. A feature ‘INF’ is defined as follows:

    - The feature ‘INF’ is set to 1 if the current word contains any noun inflection.
    - The value of the feature ‘INF’ is set to 2 if the current word contains any adjective inflection.
    - The value of the feature ‘INF’ is set to 3 if the current word contains any verb inflection.
    - The feature value of ‘INF’ is 4 if the current word contains inflection that appears in more than one inflection lists.

- The feature ‘INF’ is set to 0 for those words that do not contain any of these inflections.

4 Unknown Word Handling for POS Tagging

Handling of unknown words is an important issue in POS tagging. In the present work, we have used the same methodologies of unknown word handling techniques for HMM, ME, CRF and SVM based POS tagging models. For unseen words, which have not been seen in the training set, P(w_i | t_i) is estimated based on features of the unknown words, such as whether the word contains a particular suffix. These suffixes may not be a meaningful unit of a word. The probability distribution of a particular suffix is generated from all words in the training set that share the same suffix. At present, we have 435 suffixes, many of them usually appear at the end of verb, noun and adjective. A null suffix has been kept for those words that do not contain any of the listed suffixes. Apart from suffix feature, two other features have been considered. They are utilized to tackle unknown digits and symbols. In addition to word suffixes, we have used a SVM based NER system (Ekbal and Bandyopadhyay, 2008b) and a lexicon (Ekbal and Bandyopadhyay, 2008c) to tackle the unknown word problems. The NER system was trained with a newspaper corpus of 150K wordforms and yielded 91.2% *F-Score* during 10-fold cross validation test.

The unknown word handling procedure is detailed below:

1. Find the unknown words in the test set.
2. Unknown words are searched in the Lexicon. If there is a match then
   2.1. POS tags obtained from the lexicon are assigned to the unknown words.
   2.2. 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. The test set is passed through the SVM based NER system.
   3.1. The system considers the NE tags for those unknown words that are not found in the lexicon
   3.2. The system replaces the NE tags by the appropriate POS tags (NNPC [Compound Proper Noun] and NNP [Proper Noun]).

   Else
4. The remaining words are tagged using the unknown word features accordingly.
   4.1. P (w_i | t_i) is estimated based on features of the unknown words, such as whether the word contains a particular suffix (may not be a meaningful unit of a word).
   4.2. The probability distribution of a particular suffix is generated from all words in the training set that share the same suffix.

5 Evaluation Results

We have developed POS taggers using ME, CRF and SVM frameworks. All the models have been evaluated with the same datasets. We have used a corpus of 72,341 tokens tagged with the 27 POS tags, defined for the Indian languages. This 27-POS tagged training corpus has been obtained through our participations in two consecutive competitions, namely NLPAI ML-2006[5] and SPSAL-2007[6]. The NLPAI ML-2006, and SPSAL-2007 contests had 46,923, 25,418 tokens, respectively. Out of 72,341 tokens, around 15K tokens are selected as the development

[5] http://ltrc.iiitnet/nlpai contest06/data2

[6] http://shiva.iiit.ac.in/SPSAL2007

set and the rest have been used as the training set. A gold standard test set of 35K tokens is used to report the evaluation results.

In the first experiment, we have implemented a *baseline* model to understand the complexity of the POS tagging task. In this model the tag probabilities depend only on the current word:

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

In the *baseline* model, each word in the test data will be assigned the POS tag, which occurred most frequently for that word in the training data.

A number of experiments have been carried out to find the most suitable set of features for POS tagging in each of the models. Results have been presented in Table 1 on the development set for the best set of features. Results show that the SVM based system performs best with an accuracy of 85.83% followed by CRF and ME. The ME based system has demonstrated the best accuracy of 81.75% for the development set with the context window of size three, i.e., previous one, current and the next one words, prefixes and suffixes of length up to three characters of the current word only, dynamic POS information of the previous word, NE tag of the current word, symbol feature, length of the word and features extracted from the lexicon and the inflection lists. The CRF based system has yielded the accuracy of 84.11% with the context window of size five, i.e., two preceding words, two following words and the current word, NE tags of the current and previous words along with the same set of features as that of the ME model. The SVM based POS tagger performs with an accuracy of 85.83% for the context window of size six, i.e., previous three, current and the next two words, POS information of the previous two words, NE tags of the previous, current and the next words along with the same set features as that of the ME and CRF based systems.

Table 1: Results on the development set

Model	Accuracy (in %)
ME	81.75
CRF	84.11
SVM	85.83

Now, the systems are tested with the test set by considering the potential set of features that yielded the best accuracies for the development set. Results of the systems along with the *baseline* models have been presented in Table 2 for the test set. There are 25.4% unknown tokens in the test set. Results show that the SVM based system performs best for the test set.

Table 2: Results on the test set

Model	Accuracy (in %)
Baseline	54.7
ME	81.91
CRF	84.23
SVM	85.92

We included various techniques for handling the unknown words into the system. Results are reported in Table 3 by including the various unknown word handling techniques. Results show the effectiveness of the various unknown word handling techniques with the significant improvement in the accuracies in all the systems. It is evident from the evaluation results of Table 2 and Table 3 that the unknown word handling techniques are very effective in improving the POS tagging accuracy in each of the systems. Results of Table 2 and Table 3 show that the various unknown word handling techniques increase the accuracy by 5.15% in the ME based system, 5.61% in the CRF based system and 4.2% in the SVM based system. Results also demonstrate that all the techniques are not equally important to handle the unknown word problems. Lexicon is the most effective followed by word suffixes and the NER system.

Table 3: Overall results on the test set

Model	Accuracy (in %)
ME + Lexicon	84.93
ME + Lexicon + NER	85.69
ME + Lexicon + NER + Word suffixes	87.06
CRF+ Lexicon	86.79
CRF + Lexicon + NER	87.51
CRF + Lexicon + NER + Word suffixes	89.84
SVM + Lexicon	88.09
SVM + Lexicon + NER	89.01
SVM + Lexicon + NER + Word suffixes	90.12

5.1 Voting

Voting is a technique that combines more than one classifier in order to obtain higher accuracy. A close scrutiny to the evaluation results in each of the POS tagging systems suggests that a particular word wrongly POS tagged by any system may be correctly tagged by the other system. This observation leads us to decide that rather than selecting the POS tag from a particular classifier, it may be more effective if all the classifiers are considered in the final tag assignment. In our experiments, in order to obtain higher performance, we have applied weighted voting to the three systems, namely ME, CRF and SVM based POS taggers. We have used following weighting methods in our experiments:

a. Uniform weights (Majority voting): The same voting weight is assigned to all the systems. The combined system selects the classifications, which are proposed by the majority of the models. In case of tie, the output of the SVM classifier has been selected as the final output.

b. Cross validation precision values: The training corpus is divided into 10 equal subsets. In cross validation test, one subset is withheld for testing while the remaining 9 subsets are used for training. This process is repeated 10 times to yield the average precision values. The voting weight for a particular system is determined by assigning the corresponding average precision value of the 10-fold cross validation precisions. We have defined two different types of weights depending on the 10-fold cross validation precision as follows:

   (i).Total Precision: In this method, we have assigned the overall average precision of any classifier as the weight for it.

   (ii).Tag Precision: Here, we have assigned the average precision value of the individual POS tag as the weight.

Experimental results of the voted system are presented in Table 4. Evaluation results show that the system achieves the highest performance for the voting scheme 'Tag Precision', which considers the individual tag precision value as the weight of the corresponding system. Voting shows the improvement in accuracies by **2.23%** over the best performing SVM based system and **5.29%** over the least performing ME based system.

Table 4: Experimental results of the voted system

Voting Scheme	Accuracy (in %)
Majority	91.05
Total Precision	92.01
Tag Precision	92.35

6 Error Analysis

In order to improve the computational model, it is necessary to analyze the errors. We have conducted an error analysis via the confusion matrix, also called a contingency table. A confusion matrix for an n-way classification task is an n-by-n matrix, C, where the cell C (*x: y*) contains the number of times (in percentage) an item with correct classification *x* was classified by the model as *y* with respect to the total number of errors during the classification task. The row labels indicate correct tags, column labels indicate the tagger's hypothesized tags, and each cell indicates the percentage of the overall tagging error. A portion of the confusion matrix is shown in Table 5 for the test set. For example, C (NNC, NN) indicates the number of times (in percentage) an item with (actual) tag NNC has been assigned the tag NN by the model with respect to the total number of errors.

C (NNC, NN) = (number of times NNC →NN) / (total number of errors)= 79 / 874 = 9.03 %

Table 5: Confusion matrix

	JJ	NN	NNC	NNP	VFM	…
JJ	-	5.34	0	0.08	0.27	
NN	2.04	-	1.33	0.59	0.32	
NNC	1.71	9.03	-	0.24	0	
NNP	0.05	1.13	0	-	0	
VFM	0.36	1.19	0	0	-	
..						

The confusion matrix suggests that most of the probable tagging errors faced by the current POS tagger are NNC vs. NN and JJ vs. NN. A multiword extraction unit for Bengali would have taken care of the NNC vs. NN problem. The problem of JJ vs. NN is hard to resolve and probably requires the use of linguistic rules.

7 Conclusion

In this paper, we have reported our work on POS tagging for Bengali by combining the ME, CRF and SVM classifiers using weighted voting techniques. Each of the classifiers makes use of the different contextual information of the words along with a variety of orthographic word-level features. A number of experiments have been carried out to find the best set of features in each of the models. Various techniques for handling unknown words have been devised to improve the performance in each of the individual systems. Finally, the three systems have been combined together into a final system using weighted voting techniques. The system has been trained with the datasets obtained from the NLPAI ML-2006 and SPSAL-2007 contests. Evaluation results on a gold standard test set of 35 tokens yield an accuracy of 92.35%, which is an improvement **2.23%** over the best performing SVM based system and **5.29%** over the least performing ME based system. Results also show that various unknown word handling techniques increase the accuracy by 5.15% in the ME based system, 5.61% in the CRF based system and 4.2% in the SVM based system.

Future works include the development of POS taggers in other Indian languages. We would also like to investigate other efficient voting methods to combine the systems.

References

Brants, T. 2000. TNT-A Statistical Part of Speech Tagger. In *Proc. of the 6th ANLP Conference,* 224-231.

Cutting, D., J. Kupiec, J. Pederson and P. Sibun. 1992. A Practical Part of Speech Tagger. In *Proc. of the 3rd ANLP Conference*, 133-140.

Dandapat, Sandipan. 2007. Part Of Specch Tagging and Chunking with Maximum Entropy Model. In *Proceedings of SPSAL2007, IJCAI-07*, India, 29-32.

Ekbal, Asif, R. Haque and S. Bandyopadhyay. 2008. Maximum Entropy based Bengali Part of Speech Tagging. In A. Gelbukh *(Ed.), Advances in Natural Language Processing and Applications*, *Research in Computing Science (RCS) Journal*, Volume (33), 67-78.

Ekbal, Asif, R. Haque and S. Bandyopadhyay. 2007. Bengali Part of Speech Tagging using Conditional Random Field. In *Proceedings of the 7th International Symposium on Natural Language Processing (SNLP-07),* Thailand, 131-136.

Ekbal, Asif and S. Bandyopadhyay. 2008a. Part of Speech Tagging in Bengali using Support Vector Machine. In *Proceedings of the International Conference on Information Technology (ICIT 2008),* 106-111, IEEE CS Press.

Ekbal, Asif and S. Bandyopadhyay. 2008b. Bengali Named Entity Recognition using Support Vector Machine. In *Proceedings of the Workshop on Named Entity Recognition for South and South East Asian Languages, 3rd International Joint Conference on Natural Language Processing (IJCNLP-08)*, India, PP: 51-58.

Ekbal, Asif and S. Bandyopadhyay. 2008c. Web-based Bengali News Corpus for Lexicon Development and POS Tagging. In *POLIBITS, an International Journal, ISSN 1870-9044,* Vol. 37(2008), PP. 20-29, National Polytechnic Institute, Mexico.

Ekbal, Asif and S. Bandyopadhyay. 2008d. A Web-based Bengali News Corpus for Named Entity Recognition. *Language Resources and Evaluation Journal*, Volume (40), 173-182.

Ekbal, Asif, S. Mondal and S. Bandyopadhyay. 2007. POS Tagging using HMM and Rule-based Chunking. In *Proceedings of SPSAL-2007*, *IJCAI-07*, 25-28.

Karthik, Avinesh PVS G. 2007. Part Of Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning. In *Proceedings of SPSAL2007, IJCAI,* India, 21-24.

Kudo, T and Y. Matsumoto. 2001. Chunking with Support Vector Machines. In *Proceedings of NAACL*, 192-199.

Lafferty, J., A. McCallum and F. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. of the 18th ICML'01*, 282-289.

Malouf, R. 2002. A Comparison of Algorithms for Maximum Entropy Parameter Estimation. In *Dan Roth and Antal Van Den Bosch* (*eds.*), *Proceedings of the Sixth Conference on Natural Language Learning (CoNLL-2002)*, 2002, 49–55.

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.

Sha, Fei and Fernando Pereira. 2003. Shallow Parsing with Conditional Random Fields. In P*roceedings of HLT-NAACL*, 213–220.

Vapnik, Valdimir N. 1995. *The Nature of Statistical Learning Theory*. Springer.