Bangla Parts-of-Speech Tagging using Bangla Stemmer and Rule based Analyzer

Md. Nesarul Hoque
Dept. of Computer Science and Engineering
Port City International University
Chittagong, Bangladesh
Email: nesarul@portcity.edu.bd

Md. Hanif Seddiqui
Dept. of Computer Science and Engineering
University of Chittagong
Chittagong - 4331, Bangladesh
Email: hanif@cu.ac.bd

Abstract—**Parts-of-Speech (POS) tagging plays vital roles in the field of Natural Language Processing (NLP), such as - machine translation, spell checker, information retrieval, speech processing, emotion analysis and so on. Bangla is a very inflectional language that induces many variants from a single word. Although there is a few POS Tagger in Bangla language, very small of them address the essence of suffices to identify tag of the words. In this regard, we propose an automated POS Tagging system for Bangla language based on word-suffixes. In our system, we use our own stemming technique to retrieve a possible minimum root words and apply rules according to different forms of suffixes. Moreover, we incorporate a Bangla vocabulary that contains more than 45,000 words with their default tag and a patterned based verb-data-set. These facilitate to improve tagging efficiency of Bangla POS Tagger. We experiment our proposed system on a Bangla text corpus. The result shows that our proposed Bangla POS Tagger has outperformed the known related tagging systems.**

Keywords—Bangla Parts-of-Speech (POS) Tagger; Natural Language Processing (NLP); Morphology Bangla Stemmer

I. Introduction

Automated POS tagging is the process of detecting each word with corresponding parts-of-speech in a sentence. POS tagger is considered as the foundation of NLP. It has a diversified utilization to annotate the named entity in the large corpus, to link the named entity in the historical data, to verify and validate text data, to translate one language to another language, to correct the grammatical error of a sentence, and so on. Therefore, it will contribute to almost every branch of NLP. Different languages in the world have their own POS tagger, which enrich their languages in the linguistic aspect.

Bangla is one of the popular languages, especially in Bangladesh and part of India e.g. West Bengal, Tripura and Southern Assam. It takes the seventh position by the total number of native speakers and eleventh position by the total number of speakers[1]. Moreover, it is the national language of Bangladesh. Therefore, developing an automated Bangla POS tagger will be an essential tool to enrich the language, especially in the area of NLP.

Bangla language has a vast range of vocabulary. It is one of the most inflectional languages to form word variants by lot of the prefixes and suffixes. Prefix is an affix, which, placed at the beginning of the root word and suffix is just opposite of the prefix, which, joined with the root word at the end portion. Word with prefix is almost included into the dictionary, where all vocabulary of a language is stored. Therefore, we can get words with prefixes from the dictionary easily. However, word with different suffixes is not included into dictionary by default. Basically, noun and verb have various forms for different suffixes. Due to this, it is difficult to tag these words directly. In this context, our system handles this difficult job by combining the stemming technique [1], rule-based analysis, Bangla vocabulary and Bangla verb-data-set.

This is unsupervised learning technique. Therefore, it does not require any training data set. Preparing hand-made training data is too much assiduous task. In this paper, we have worked on eight fundamental tags, shown in Table I.

TABLE I. BANGLA PARTS-OF-SPEECH TAG LIST

	Tags	Description	Example
1	NN	Noun	জয়নুল, নজরুল
2	PRO	Pronoun	সে, তারা
3	ADJ	Adjective	বুদ্ধিমান, স□
4	VRB	Verb	করা, বলা
5	ADV	Adverb	দ্রুত, খুবই
6	PRE	Preposition	থেকে, দিয়ে
7	CON	Conjunction	এবং, ও
8	INT	Interjection	আরে, ওহে

The rest of the paper organized as follows. **Section II** points the related work with different methods for developing Bangla POS Tagger with mentioning the problems of them. **Section III** describes the overall implementation in details to assigning appropriate tag in the test data. In **Section IV**, the experiment in different phases is illustrated, at the same time we analyze the performance with some constraints. Finally, **Section V** mentions the success of this system and points

[1] www.ethnologue.com/statistics/size

978-1-4673-9930-2/15/$31.00 ©2015 IEEE

some directions to enhance this system for applying on the other fields in the Bangla NLP branches.

II. Related Works

There are many POS tagger have been developed for many languages such as English, Arabic, Chinese and different western languages. In all the cases, the accuracy level gains the satisfactory level i.e. above 95%. There are also some works on Bangla language to develop an automatic POS tagger. Still there is enough room to improve this with better performance.

At the early stage, Seddiqui et al [2] and Chowdhury et al. [3] propose a way and try to implement an automated Bangla POS tagger system. Chowdhury et al [3] develops Bangla POS tagger using allomorphic method with allo-rules. They use the lexicon to get the root word. It does not require any training corpus. Therefore, this is unsupervised learning method. The main suspicious thing of this paper is that there is no performance analysis, which generates a question on the level of accuracy. In addition, there is no idea about adverb, conjunction and interjection, which is considered in our system. Furthermore, we get the gratified result in tagging.

Hasan et al. [4] compares three POS tagging techniques: N-Gram, Hidden Markov Model (HMM) and Brill's tagger for both Bangla and English language. All three techniques need large set of hand-tagged training data. First two of them are stochastic based approach and last one is transformation-based approach. In the stochastic based approach, most likely tags are picked up from the training corpus. In the transformation-based approach, at first, tags are picked up according to the probability theory and then correct the wrong tag applying the set of rules. Authors suggest Brill's tagger [5] [6] [7] [8], because of higher performance compare to other two techniques (Unigram and HMM). They work on 4,484 tokens as a training data set and get 54.9% accuracy with 41-tag set. Therefore, there is a huge chance to work to develop the performance level. In Hasan et al [4], authors increase the size of the training data set (25,426 tokens), which enhances the accuracy level (69.6%). However, still it is not significantly high enough. For this reason, at the end, authors have wished to work with rule-base for Bangla POS tagging. In our system, we accept the authors' proposal and get the promising output.

Dandapat et al. [9] build a tagger using supervised and semi-supervised bi-gram HMM and a Maximum Entropy (ME) based model with morphological analysis. They work with 40K training data and get 88.75% accuracy. Although, the result of performance is at satisfactory level, but there is still space to improve the performance. In this paper, we have tried to do this.

Kumar and Josan [10] work on various Indian languages - Hindi, Punjabi, Malayalam, Bengali and Telugu. They apply different tagging models such as - Hidden Markov Model (HMM), Support Vector Model (SVM), Rule based approaches, Maximum Entropy (ME) and Conditional Random Field (CRF) and compare to each other on these languages. In Bangla language, they prefer CRF model with considering Named Entity Recognizer (NER), Lexicon and Unknown word features and work on 72,341 words as a training data to achieve 90.3% accurate result. There is no doubt that the result is good enough. However, still in backward position compare to current POS tagger available in English and other European languages. In our system, we work on suffix-based analysis with stemmer, set of rules, dictionary and patterned verb-data-set, which tends to get better result compare to this system.

Ali [11] report unsupervised learning approach using Baum-Welch trained HMM tagger for Bangla language. He has just tried to give some ideas in tagging, but cannot implement this.

Chakrabarti [12] develops rule based with four layers of Bangla POS tagging system. There is also absence performance analysis, which generates a question about accuracy level.

III. Implementation Details

The basic things in our paper is, suffix analysis, through which we have developed our desired Bangla POS tagger. In linguistic, suffix is an affix, which is added at the end of the root word according to the inflections, numbers, persons, tense, etc. Under the suffixes, a root word can have various forms with different parts-of-speech. Using suffix-based morphological analysis, a stemmer [1] has been developed, which converts the suffix-word into a root word. In addition, we develop set of rules, on the basis of - suffixes, structure of the sentences, rules of Bangla Grammar [13] and some real time observations. These rules are demonstrated below –

Rule 1: The *word*$_i$ (i= 0, 1, ..., length of the sentence - 1) is considered as always Noun/Adjective/Adverb, whatever it is found any place in any sentences, which contains the suffix, shown in Table II. For example, consider a word - "উন্নয়নশীল". Here, the suffix - "শীল", is matched with the Noun-suffix in Table II. Therefore, according to this rule, the word is counted as a Noun.

Rule 2: This rule has been built up regarding the rules of Bangla Grammar [13] and real time observation. If *word*$_i$ is Quantifier-marker (shown in Table III), then it is considered as Noun and the previous word i.e. *word*$_{i-1}$ will be Adjective. Again, if *word*$_i$ is Quantifier-marker and previous two consecutive words i.e. *word*$_{i-1}$ and *word*$_{i-2}$ are Adjective, then *word*$_i$ will be Noun and *word*$_{i-1}$ will be changed to a Noun. Now we would like to illustrate this rule for the following two sentences –

Sen 1: "আমাকে ৫ টাকা দাও।"

Sen 2: "আমাকে ৫ লক্ষ টাকা দাও।"

Here, in Sen 1, at first, a Noun tag is assigned to "৫". After that, when the word - "টাকা", is examined, then Noun tag is assigned to this word and the previous word - "৫" is changed to Adjective tag. Therefore, after applying Rule 1 the word "৫" and "টাকা" will be Adjective and Noun respectively.

Now, for Sen 2, at first the stage, "৫" is considered as a Noun. At the second stage, when the word - "লক্ষ" is examined, then a Noun tag is assigned to this word and the previous one i.e. "৫" is changed to Adjective. At the final stage, when "টাকা" is examined according to the second part of the Rule 2, it marks the current word as a Noun and checks the previous two consecutive words are Adjective or not. In this example, since previous two words ("৫" and "লক্ষ") are Adjective; therefore, immediate previous word is converted to a Noun.

Rule 3: After stemming if *word$_i$* matches to adjective word from the dictionary, then is converted into a Noun. Because, adjective words are not suffixed word i.e. they do not stemmed. E.g., consider the word "তরুণেরা". It is suffix word and does not found into the dictionary. After removing suffix using the stemmer, we get "তরুণ", which is Adjective. However, actually the original word is Noun.

Rule 4: Suppose the *word$_i$* with suffixes - "ার" or "ানোর". After stemming, if it is found into verb-data-set, then, is considered as a Noun. E.g. if a word - "দেখার"/"দেখানোর", is found into any sentence, after stemming it is converted into "দেখ"/"দেখানো", is the one kind of verb comes from root word - "দেখা". In this context, the word is encountered as a Noun.

Rule 5: After stemming if the *word$_i$* is not found into the dictionary as well as the verb-data-set, it is considered as a Noun.

Rule 6: The *word$_i$*, which is not found into the dictionary and is not stemmed, is considered as a Noun.

Rule 7: This rule comes from the observation. In dictionary, some words have different parts of speech. In this case, we change the order of the parts-of-speech according to the probability of mostly found into a large corpus. E.g. the word "ও", most of the times, found as a Conjunction rather than Pronoun. This rule has been applied for developing the dictionary data-set with POS tag.

In this system, we use a dictionary, contains more than 45,000 Bangla vocabularies with associated POS tag. Moreover, we use patterned based Bangla verb-data-set, where, all the verb-words are resided with different forms, according to person, tense and pattern of compound-verb feature [14]. Both data sets are stored in the fashion of hash map.

TABLE II. SUFFIXES OF NOUN, ADJECTIVE AND ADVERB

Parts of Speech	Suffixes
Noun	শীল, োয়ান, োয়ালা, খানা, গিরি, দানি, নবিশ, বন্দি, বাজি, ানো, ায়ন
Adjective	জনক, মূলক, ব্যাপী, যোগ্য, কেন্দ্রিক
Adverb	ভাবে, ভাবেই

TABLE III. MOSTLY FOUND QUANTIFIER-MARKER LIST

Parts of Speech	Suffixes
Noun	টাকা, ডলার, রুপি, দিনার, ইয়েন, ইয়রো, মাস, দিন, সপ্তাহ, সাল, বছর, যুগ, সহস্র, গজ, ফুট, ইঞ্চি, মিলিমিটার, সেন্টিমিটার, মিটার, কিলোমিটার, কেজি, লিটার, শতাংশ, দশমিক, অংশ, বার, শত, হাজার, সহস্র, লাখ, লক্ষ, কোটি, মিলিয়ন, বিলিয়ন

Now using the stemmer, set of rules, dictionary and verb-data-set, we have developed an algorithm *banglaPosTagger*, depicted in Algo. This algorithm assigns tag to each word in the Corpus *crp*, using Dictionary *dic*, VerbDataset *vset*, TagList *tagLst* and QuantifierMarker *quantMarkLst*. Here, *crp* contains any news, articles, stories, etc. and *dic*, *vset*, *tagLst* and *quantMarkLst* denote the vocabulary with corresponding POS tag, all verb with different forms, eight fundamental POS tags (Table I), and list of possible quantifier-marker (Table III) respectively.

```
Algo. banglaPosTagger(Corpus crp,
                      Dictionary dic,
                      VerbDataset vset,
                      TagList tagLst,
                      QuantifierMarker quantMarkLst)
1.  for each sentence ∈ crp
2.    for each word ∈ sentence
3.      tag=getTagFromRuleSuffices(root)
4.      if(tag ∈ tagLst) continue
5.      if(isQuantifierMarker(word, quantMarkLst) continue
6.      stemmer(word, root, suffix)
7.      if(word!=root AND isQuantifierMarker(word, quantMarkLst)) continue
8.      if(word ∈ dic) tag=getTag(word, dic)
9.      else
10.       if(word!= root)
11.         if(root ∈ dic) tag=getTag(root, dic)
12.           if(tag==‘Adjective’) tag=‘Noun’
13.         else if(getFromVerbDataset(root, vset) AND !isWrongVerbSuffices(word)) tag=‘Verb’
14.         else tag=‘Noun’
15.       else
16.         if(getFromVerbDataset(word, vset)) tag=‘Verb’
17.         else tag=‘Noun’
```

Let us comprehend the algorithm *banglaPosTagger* for the arguments *crp*, *dic*, *vset*, *tagLst* and *quantMarkLst*. In line 3, algorithm tries to check the word-suffix for Noun, Adjective and Adverb, according to the Rule 1, as illustrated earlier in this section. If the suffix matches with the Rule 1, then get the tag that can be either Noun or Adjective or Adverb and process will be continued for the next word (as line 4). Rule 2 is implemented in line 5, 6 and 7. In line 5, at first, we check the word in the Quantifier-marker list (Table III). If it is found, then the following three processes will be executed. Firstly, change the previous tag into Adjective tag. Secondly, get the current tag with Noun tag. Thirdly, if previous two tags are Adjective, then change the immediate previous tag into Noun tag. If the word is not found into the Quantifier-marker list, then it is needed to stem the word in order to convert into root word (as line 6). If word and root-word are not similar, then we do the same thing as illustrated in line 5. However, in this case, we will take the root-word (word, get after stemming) instead of word, as shown in line 7.

Now the algorithm tries to find the word into dictionary and get the required tag if it is there, as given in line 8. If it is not there (as line 9), then the algorithm checks whether this word is stemmed or not i.e. word and root-word are similar or not (as line 10). If word is stemmed, line 11 finds the root-word from the dictionary and gets the corresponding tag if it is located there. In this case, for implementing Rule 3 (discussed earlier in this section), convert the tag into a Noun tag, if it is Adjective (as line 12). However, if the root-word is not found into the dictionary, the algorithm then tries to find this into the verb-data-set as well as check that it is not wrong-verb-suffixes (explained in Rule 4). If both two conditions are satisfied, then the tag for this word will be Verb (as illustrated in line 13), otherwise the it will be Noun (according to Rule 5), as given in line 14 in this algorithm. The line 15 shows that, if the word is not stemmed i.e. word and root-word are same, then search the word into verb-data-set and take the Verb tag if it is found there (as given in line 16), otherwise the word will be treated as Noun (applying Rule 6), as demonstrated in line 17. The process will be continued until all the word of each sentence in the corpus has been encounted (as line 1 and 2).

Example: Considering the two sentences, collected from the test data –

"বাংলাদেশের ক্রিকেট আর বাংলাদেশের তারুণ্যের প্রশংসা করলেন নরেন্দ্র মোদি। আজ বঙ্গবন্ধু আন্তর্জাতিক সম্মেলন কেন্দ্রে ঢাকা বিশ্ববিদ্যালয় আয়োজিত জনবক্তৃতায় এ কথা বলেন ভারতীয় প্রধানমন্ত্রী।"

We get the output through the following ways by applying this algorithm –

বাংলাদেশের_NN ক্রিকেট_NN আর_CON বাংলাদেশের_NN তারুণ্যের_NN প্রশংসা_NN করলেন_VRB নরেন্দ্র_NN মোদি_NN

আজ_ADV বঙ্গবন্ধু_NN আন্তর্জাতিক_ADJ সম্মেলন_NN কেন্দ্রে_NN ঢাকা_NN বিশ্ববিদ্যালয়_NN আয়োজিত_ADJ জনবক্তৃতায়_NN এ_PRO কথা_NN বলেন_VRB ভারতীয়_ADJ প্রধানমন্ত্রী_NN

In the first sentence, the words - "বাংলাদেশের", "ক্রিকেট" and "মোদি" are not found into the dictionary as well as into the verb-data-set, whether it is suffix-word or not. Therefore, these words are treated as Noun (as Rule 5 and 6). However, the words - "আর", "প্রশংসা" and "নরেন্দ্র", directly picked up from the dictionary with associated tag. In case of the word, "তারুণ্যের", it is the suffix-word and converted into the root word, "তারুণ্য", which is Adjective. According to the Rule 3, it is transformed into Noun.

In the second sentence, the words - "আজ", "আন্তর্জাতিক", "সম্মেলন", "ঢাকা", "বিশ্ববিদ্যালয়", "আয়োজিত", "এ", "কথা", "ভারতীয়" and "প্রধানমন্ত্রী", directly obtained from the dictionary with associated tag. For the word - "কেন্দ্রে", after stemming it is converted into root word - "কেন্দ্র", which, resided into dictionary and get the tag with corresponding POS tag. In case of word - "বঙ্গবন্ধু", does not found into dictionary and this is not suffix-word, therefore, with the Rule 6, Noun tag is assigned here. The word - "জনবক্তৃতায়", is the suffix-word, but stemmer cannot convert it into root word. In this case, after stemming, the obtained word - "জনবক্তৃত", considered as a Noun word through the Rule 5. In this sentence, the word - "বলেন", is not the dictionary word, also stemmer cannot stem this word. The tag for this word is taken from the pattern based verb-data-set.

IV. Experiment and Evaluation

Here we collect different types of news, as a test data from the site www.prothom-alo.com, is the leading on-line news in Bangladesh. There are around 8,155 words of 585 sentences, get from this site. In this paper, we experiment on these words in three different contexts. At first, we consider dictionary and stemmer. Secondly, we examine by adding verb-data-set with the dictionary and stemmer. Finally, we do the experiment with dictionary, stemmer, verb-data-set and set of rules and get a satisfactory output through this. Each cases, we calculate the performance using the following formula –

$$\text{Accuracy} = \frac{Number\ of\ correctly\ detected\ POS\ tag}{\text{Total words in the corpus}}$$

In the first case, we detect only 3,842 words with appropriate POS tag. Lots of word is undetected. The main reason is many words are not included into the dictionary, because of Proper noun. Proper nouns are not the dictionary word. In addition, some words are not converted into the root words. Another reason is our stemmer does not work properly on all the form of verb word. In this level of experiment, our accuracy (47.1%) is not good enough.

In the second case, we emphasize on different forms of verb word. Through the analysis of various forms of verb-suffix according to person, tense and pattern of compound verb, we build a pattern based verb-data-set. This data set is added with the first case and we get better result than the previous one. Here we correctly detect 5,169 words. Therefore, our performance is improved and gets 63.4% accurate result. However, still this is not in satisfactory level.

Finally, we concentrate on the rule-based analysis with some real time observations and propose seven rules, as discussed in the previous section. With the help of these rules and other resources (i.e. dictionary, stemmer and verb-data-set), we get more desired output. In this experimental phase, we appropriately detect 7,637 words as a correct POS tag by examining 8,155 words. Therefore, this system cannot properly detect only 518 words. Our obtained accuracy reaches at 93.7%.

The experimental result of above three cases has been shown is Table IV.

TABLE IV. Test result of Bangla POS tag detection

Total Words	**Experiment Type**	**Detected Words**	**Accuracy**
8155	Dictionary + Stemmer	3842	47.1%
	Dictionary + Stemmer + Verb Dataset	5169	63.4%
	Dictionary + Stemmer + Verb Dataset + Rules	7637	93.7%

By analyzing the last context, where we get the maximum accuracy, some constraints are identified, for which we could not get 100% accurate result. These are explained below-

Firstly, some words (comes from English language) are not found into the dictionary. Here, these types of words are counted as Noun. There is not any specific rules can be applied on them. E.g. "এভারেস্টজয়ী", "অলরাউন্ডার", "টেস্ট", "ওয়ানডেতে", "টি-টোয়েন্টিতেও", "টেস্টটা", "ড্র", "গ্রাফটি", "সিরিজ", "মাল্টিমিডিয়ার", etc.

Secondly, there are few suffix-words (i.e. not root word), where we cannot apply any morphological rule to stem that words. For this reason, we cannot properly detect these words. E.g. "গর্বই", "এরই", "এটাই", "আমিও", "বাড়লেও", etc.

Thirdly, some suffix-words, which are stemmed but not converted into root word i.e. these words are not properly stemmed. In that case, if these words are not found into verb-data-set, then, they are considered as Noun, using Rule 5, whatever it is correct or not. E.g. "বক্তৃতায়"->"বক্তৃত", "ছুটির"->"ছু", "পড়ে"->"প", "হতে"->"হ", "ধরা"->"ধ", "নিলে"->"ন", etc.

Fourthly, some words have different parts of speech according to the structure of the sentence. In that situation, a POS tag is assigned to a word, which comes first in the dictionary. That's why, a word is tagged incorrectly if it does not place into the first position. E.g. "ঢাকা", "হারে", "ও", "তারা", "মেলা", "করে", etc. have different parts-of-speech in different sentences.

Fifthly, some suffix-words are stemmed, but wrongly converted into dictionary word, which tend to assign incorrect POS tag. E.g. "মেতে" (Verb)->"মে" (Noun), "নামে" (Verb)-> "নাম" (Noun), "ছুটে" (Verb)->"ছুট" (Noun) etc.

Sixthly, in some cases, two or more words are joined by hyphen (-), considered as one word (E.g. "টাইপ-১", "ই-বুক", "ই-বই", "তা-ই", "তা-ও", "দূর-ভবিষ্যতে", "হুয়াং-তি", "গ্রেড-১", etc.), in other cases it is considered as different words (E.g. "পেয়েছেন-এই", "কারণে-যে", "হয়েছে-অদূর", "উচিত-এটা", "পরিমাণ-দুই", "না-পাওয়া", etc.). In this system, we have considered the first case. For this reason, there happens a wrong tagging sometimes, when, second case is appeared.

Finally, since Bangla has a huge range of vocabulary, sometimes we missed some root words, which is not belonging in our dictionary data set. In this test corpus, "কর্মযজ্ঞ" is that type of word.

V. Conclusion and Futerue Direction

We have successfully developed a word-label Bangla POS tagging system by using stemmer and set of rules. Here, the dictionary and patterned based verb-data-set supports to increase the performance in satisfactory level (93.7%), which outperformed than any other existing systems. The main focusing point of this paper is suffix based analysis, through which, we build-up set of rules and verb-data-set. In this system, we work on eight fundamental parts-of-speech tags. Therefore, in future we will concentrate on all the subcategories of each base tag with punctuation. Punctuation plays important role to give expression in the NLP field. Moreover, we will try to improve the accuracy level more in higher position by combining the probability model.

References

[1] M. H. Seddiqui, A. M. S. Rana, A. Mamud, and T. Sayed, “Morphological analysis of bangla words,” in *6th International Conference on Computer and Information Technology(ICCIT)*, 2003, pp. 313–316.

[2] M. H. Seddiqui, A. M. S. Rana, A. Al Mahmud, and T. Sayeed, “Parts of speech tagging using morphological analysis in bangla,” in *Proceeding of the 6th International Conference on Computer and Information Technology (ICCIT)*, 2003.

[3] M. S. A. Chowdhury, N. M. M. Uddin, M. Imran, M. M. Hassan, and M. E. Haque, “Parts of speech tagging of bangla sentence,” in *Proceeding of the 7th International Conference on Computer and Information Technology (ICCIT), Bangladesh*, 2004.

[4] F. M. Hasan, N. UzZaman, and M. Khan, “Comparison of different pos tagging techniques (n-gram, hmm and brillŠs tagger) for bangla,” in *Advances and Innovations in Systems, Computing Sciences and Software Engineering*. Springer, 2007, pp. 121–126.

[5] E. Brill, “A simple rule-based part of speech tagger,” in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 112–116.

[6] ——, “Automatic grammar induction and parsing free text: A transformation-based approach,” in *Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, 1993, pp. 237–242.

[7] ——, “Transformation-based error-driven parsing,” in *Proceedings of the Third International Workshop on Parsing Technologies, Tilburg, The Netherlands*. Citeseer, 1993.

[8] ——, “Some advances in transformation-based part of speech tagging,” *arXiv preprint cmp-lg/9406010*, 1994.

[9] S. Dandapat, S. Sarkar, and A. Basu, “Automatic part-of-speech tagging for bengali: An approach for morphologically rich languages in a poor resource scenario,” in *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*. Association for Computational Linguistics, 2007, pp. 221–224.

[10] D. Kumar and G. S. Josan, “Part of speech taggers for morphologically rich indian languages: a survey,” *International Journal of Computer Applications (0975–8887) Volume*, pp. 1–9, 2010.

[11] H. Ali, “An unsupervised parts-of-speech tagger for the bangla language,” *Department of Computer Science, University of British Columbia*, 2010.

[12] D. Chakrabarti and P. CDAC, “Layered parts of speech tagging for bangla,” *Language in India, www. languageinindia. com, Special Volume: Problems of Parsing in Indian Languages*, 2011.

[13] R. Islam, P. Sarkar, and H. Mahbubul, *BANGLA ACADEMY PROMITO BANGLA BYABAHARIK BYAKARAN*. Wahab, Abdul, 2014.

[14] P. Dasgupta, “The internal grammar of compound verbs in bangla,” *Indian linguistics*, vol. 38, no. 3, pp. 68–85, 1977.