Deep Learning Based Parts of Speech Tagger for Bengali

Md. Fasihul Kabir*, Khandaker Abdullah-Al-Mamun† and Mohammad Nurul Huda‡
Department of Computer Science and Engineering, United International University, Dhaka, Bangladesh
*aboutrafi@gmail.com, †mamun@cse.uiu.ac.bd, ‡mnh@cse.uiu.ac.bd

***Abstract*—This paper describes the Part of Speech (POS) tagger for Bengali Language. Here, POS tagging is the process of assigning the part of speech tag or other lexical class marker to each and every word in a sentence. In many Natural Language Processing (NLP) applications, POS tagging is considered as the one of the basic necessary tools. Identifying the ambiguities in language lexical items is the challenging objective in the process of developing an efficient and accurate POS Tagger. Different methods of automating the process have been developed and employed for Bengali. In this paper, we report about our work on building POS tagger for Bengali using the Deep Learning. Bengali is a morphologically rich language and our taggers make use of morphological and contextual information of the words. It is observed from the experiments based on Linguistic Data Consortium (LDC) catalog number LDC2010T16 and ISBN 1-58563-561-8 corpus that 93.33% accuracy is obtained for Bengali POS tagger using the Deep Learning.**

Keywords*—*Part of Speech Tagging, Deep Learning, Deep Belief Network, Linear Activation Function

I. Introduction

The Parts of Speech (POS) tagging is the process of assigning each word of a text with an appropriate parts of speech tag. The significance of POS for language processing is the large amount of information they give about a word and its neighbors. POS tags often signify the morphological, phonological and contextual properties of a word, and also provide information about neighbor words. POS tagging can be used in Text to Speech applications, information retrieval and extraction, shallow parsing, linguistic research for corpora and also as an intermediate step for higher level NLP tasks such as parsing, semantics, translation etc. POS tagging, thus, is a necessary application for advanced NLP applications in any language.

This paper is an investigation of Deep Learning [1] techniques applied for POS tagging of Bengali Language. For our experiments, we have used corpus developed in [2], which is a corpus developed by Microsoft Research (MSR) India to support the task of POS Tagging and other data-driven linguistic research on Bengali Language in general. Besides, we used morphological, contextual properties of word [3] with a pre-built dictionary of probable POS of words. From the experiments it is found that our POS tagging method provides significant result because of using Deep Learning and reduced feature set.

II. Previous Related Works

A. Ekbal and S. Bandyopadhyay proposed POS tagging system for Bengali news corpus using Support Vector Machine (SVM) [4] which exceed the existing systems based on the Hidden Markov Model (HMM) [5], Maximum Entropy (ME) [6] and Conditional Random Field (CRF) [7] outputted final accuracy of 86.84%. In [8], author achieved 92.35% accuracy applying Voted Approach method on corpus developed on NLPAI-2006 contest. In [9], the author proposed unsupervised method to develop Bengali POS tagger. In [10], the authors built a hybrid system using the same corpus that we have used in this study and gained F-Score of 90.84%. In [11], Global Linear Model (GLM) based Bengali POS tagger has an accuracy of 93.12%.

Deep learning has recently shown much promise for different NLP applications. Significant improvement in POS tagging had been observed by deep learning for foreign languages. Some mentionable contributions of deep learning in POS tagging are [12]–[15]. But for POS tagging in Bengali language using deep learning approach has not used yet.

III. Our Approach for POS Tagging

According to the survey review by [16], different types of approaches, such as supervised/unsupervised machine learning, rule based and combination of both (hybrid) have been applied to develop Bengali POS tagger. In our experiment we have used Deep Belief Network [17] to train our model. In machine learning, a deep belief network (DBN) is a generative graphical model, or alternatively a type of deep neural network, composed of multiple layers of latent variables ("hidden units"), with connections between the layers but not between units within each layer. DBNs can be viewed as a composition of simple, unsupervised networks such as restricted Boltzmann machines (RBMs) [17] or autoencoders [18], where each subnetwork's hidden layer serves as the visible layer for the next. This also leads to a fast, layer-by-layer unsupervised training procedure, where contrastive divergence is applied to each subnetwork in turn, starting from the "lowest" pair of layers (the lowest visible layer being a training set). In our setup we have used RBM as unsupervised networks and linear activation function [19].

A. Corpus & Bengali POS Tagset

We have used corpus developed by MicroSoft Research India as a part of the Indian Language Part-of-Speech Tagset (IL-POST) project. The corpus was designed based on the IL-POST framework [20]. The IL-POST was a POS-tagset framework for Indian Languages, which had been designed to cover the morpho-syntactic details of Indian Languages. It supports a three-level hierarchy of Categories, Types and Attributes.The corpus mainly comprises two different level of information for

978-1-5090-1269-5/16/$31.00 ©2016 IEEE

each lexical token, such as lexical Category and Types , and set morphological attributes and their associated values in the context. For our experiments in this study we used two top level hierarchy, Categories and Types. Categories are the top-level part-of-speech classes like noun, adjective etc. and they are obligatory. Types are the main sub-classes of categories and may be included depending on whether or not those types exist in a particular Indian language. Table I shows the POS tagset of IL-POST framework.

Categories	Types
Noun (N)	Common (NC)
	Proper (NP)
	Verbal (NV)
	Spatio-temporal (NST)
Verbs (V)	Main (VM)
	Auxiliary (VA)
Pronoun (P)	Pronominal (PPR)
	Reflexive (PRF)
	Reciprocal (PRC)
	Relative (PRL)
	Wh (PWH)
Nominal Modifier (J)	Adjectives (JJ)
	Quantifiers (JQ)
Demonstratives (D)	Absolutive (DAB)
	Relative (DRL)
	Wh (DWH)
Adverb (A)	Manner (AMN)
	Location (ALC)
Participle (L)	Relative (LRL)
	Verbal (LV)
Postposition (PP)	
Particles (C)	Coordinating (CCD)
	Subordinating (CSB)
	Classifier (CCL)
	Interjection (CIN)
	Others (CX)
Punctuation (PU)	
Residual (RD)	Foreign Word (RDF)
	Symbol (RDS)
	Other (RDX)

TABLE I. BENGALI POS TAGSET BASED ON IL-POST

B. Features of POS Tagging

According to [21], feature selection plays a crucial role in the machine learning algorithms. 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 on the different possible combination of available word and tag context. Following are the details of the set of features that have been applied for POS tagging in Bengali.

- **Length of a word:** Length of a word can be used as a feature for POS tagging. 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 by [8].
- **Word suffix and prefix:** In [22], it is mentioned that 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. 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 inflected languages like Bengali. In our experiments, we have used value of n=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.
- **Dictionary of Word vs POS Tag:** In our experiments, we have used a predefined dictionary as last feature. We have created a dictionary of word vs POS tagging using our corpus. If a word is tagged with only one POS tag, then we have saved it in our dictionary.

C. Experimental Setup

In this section, we will discuss about our experimental setup. This can be divided into two sub-sections: i) Feature Vector Construction and ii) Classifier Design.

1) Feature Vector Construction: Our feature vector is constructed as follows:

- Word length was converted into 8-bit binary number.
- The suffix and prefix that we have used in this study have three different features and each was indexed using 128-bit array.
- POS information like previous word(s) POS Tag and our dictionary based POS Tag were converted into 30-bit array each.

2) Classifier Design: We have used Deep Belief Network architecture of Deep Learning including three layers [866, 450, 30] of neural network. Equation-1 was used as output activation function. For each layers, learning rate was fixed to 0.3 and 25 epoch was used.

$$f(x) = x \tag{1}$$

IV. EVALUATION RESULTS & ERROR ANALYSIS

We have used 10-Fold Cross validation to evaluate our model. Evaluation is done based on the precision (P), recall (R) and F1-score (FS). We used equation 2, 3 and 4 to calculate precision, recall and F1-score respectively.

$$precision = \frac{tp}{tp + fp} \tag{2}$$

$$recall = \frac{tp}{tp + fn} \tag{3}$$

$$f1 - score = 2\frac{precision * recall}{precision + recall} \tag{4}$$

Where, tp, fp and fn represents true positive, false positive and false negative respectively.

Table II shows the precision, recall and F1-score values of our experiments. From this table we can say that average precision, recall and F1-score value of our experiments are 0.9334, 0.9333 and 0.9328 respectively.

Error analysis of our developed POS tagger has been done with the help of confusion matrix [23]. Table III gives

Fold	P	R	FS
1	0.9302	0.9299	0.9292
2	0.9323	0.9321	0.9316
3	0.9332	0.9330	0.9325
4	0.9333	0.9334	0.9329
5	0.9336	0.9335	0.9331
6	0.9344	0.9348	0.9341
7	0.9324	0.9322	0.9317
8	0.9368	0.9370	0.9365
9	0.9341	0.9341	0.9335
10	0.9338	0.9330	0.9328
	0.9334	**0.9333**	**0.9328**

TABLE II. RESULT ANALYSIS

POS Tag	TP	FP	FN
ALC	135	14	12
JJ	774	90	63
NC	2826	147	156
NST	148	39	15
VM	1048	84	59
NV	191	9	7
CSB	124	24	31
JQ	354	30	41
PU	1352	13	0
VA	189	30	34
PPR	437	35	27
PP	246	37	53
CCD	223	15	18
NP	628	47	71
AMN	128	12	22
CX	126	18	41
DAB	173	39	9
DRL	28	8	2
CCL	15	1	8
LV	2	0	6
RDX	32	1	8
PRL	23	1	12
RDS	134	4	3
LC	48	5	2
PWH	33	14	2
DWH	0	0	6
PRF	21	0	5
CIN	5	4	1
RDF	130	1	7
PRC	0	0	1

TABLE III. SUMMARY OF FOLD-1 CONFUSION MATRIX

the summary of Fold-1 confusion matrix (true-positive[TP], false-positive[FP] and false-negative[FN]). From the confusion matrix, we have observed that for less occurrence of **DWH** and **PRC** tags in our corpus, sometimes our tagger failed to recognize these two POS tags totally. As a result, the sum of true positives and false positives are equal to zero for these tags. Hence, the precision and recall are equal to zero. Moreover, miss tagging is found between **JJ** vs **NC**, **NC** vs **PP**, **VM** vs **VA** and **NC** vs **NP**.

V. PROBLEM ANALYSIS

Corpus that we have used for our experiment has followed problems:

- **Data sparsity problem:** Some POS tag classes are presented in the annotated set with very few representations. This is not enough to derive statistical information about them.
- **Class imbalance problem:** POS tag classes are highly imbalanced in their occurrence frequency. While selecting a tag this may lead to biasing towards the most frequent tags. Existing solutions of class imbalance problem typically favor rare classes.

Because of less availability of the corpus used in previous POS Tagging experiments discussed in the related works section, we could not compare our method with the above mentioned methods.

VI. CONCLUSION

This study has given a method for POS tagging using the Deep Learning. From the experiments the following are observed.

- the dimensionality of the input vector are 866.
- the total number of features are four.
- the recall of the POS tagger is 93.33%.
- some tags contains zero accuracy because of fewer frequency.
- the linear activation function works better in comparison with softmax and sigmoid functions.

The author would like to do further experiments based on the other papers and give some comparisons with other investigated methodologies.

REFERENCES

[1] L. Deng and D. Yu, "Deep learning: methods and applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.

[2] K. Bali, *Indian language part-of-speech tagset : Bengali*. Philadelphia, PA: Linguistic Data Consortium, 2010.

[3] N. S. Dash, "Some techniques used for processing bengali corpus to meet new demands of linguistics and language technology," *SKASE Journal of Theoretical Linguistics*, vol. 4, no. 2, 2007.

[4] A. Ekbal and S. Bandyopadhyay, "Part of speech tagging in bengali using support vector machine," in *Information Technology, 2008. ICIT'08. International Conference on*. IEEE, 2008, pp. 106–111.

[5] A. Ekbal, S. Mondal, and S. Bandyopadhyay, "Pos tagging using hmm and rule-based chunking," *The Proceedings of SPSAL*, pp. 25–28, 2007.

[6] A. Ekbal, R. Haque, and S. Bandyopadhyay, "Maximum entropy based bengali part of speech tagging," *A. Gelbukh (Ed.), Advances in Natural Language Processing and Applications, Research in Computing Science (RCS) Journal*, vol. 33, pp. 67–78, 2008.

[7] ——, "Bengali part of speech tagging using conditional random field," *Proceedings of Seventh International Symposium on Natural Language Processing (SNLP2007)*, pp. 131–136, 2007.

[8] A. Ekbal and M. Hasanuzzaman, "Voted approach for part of speech tagging in bengali." 2009.

[9] H. Ali, "An unsupervised parts-of-speech tagger for the bangla language," 2010.

[10] M. M. Yoonus and S. Sinha, "A hybrid pos tagger for indian languages." *Language in India*, vol. 11, no. 9, 2011.

[11] S. Mukherjee and S. Das Mandal, "Bengali parts-of-speech tagging using global linear model," in *India Conference (INDICON), 2013 Annual IEEE*, Dec 2013, pp. 1–4.

[12] X. Zheng, H. Chen, and T. Xu, "Deep learning for chinese word segmentation and pos tagging." in *EMNLP*, 2013, pp. 647–657.

[13] Y. Tsuboi, "Neural networks leverage corpus-wide information for part-of-speech tagging," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 938–950.

[14] C. D. Santos and B. Zadrozny, "Learning character-level representations for part-of-speech tagging," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1818–1826.

[15] A. Bordes, X. Glorot, J. Weston, and Y. Bengio, "A semantic matching energy function for learning with multi-relational data," *Machine Learning*, vol. 94, no. 2, pp. 233–259, 2014.

[16] P. Antony and K. Soman, "Parts of speech tagging for indian languages: a literature survey," *International Journal of Computer Applications (0975-8887)*, vol. 34, no. 8, 2011.

[17] G. E. Hinton, "Deep belief networks," vol. 4, no. 5, p. 5947, 2009.

[18] Y. Bengio, P. Lamblin, D. Popovici, H. Larochelle *et al.*, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

[19] F. Agostinelli, M. Hoffman, P. Sadowski, and P. Baldi, "Learning activation functions to improve deep neural networks," *arXiv preprint arXiv:1412.6830*, 2014.

[20] B. Sankaran, K. Bali, M. Choudhury, T. Bhattacharya, P. Bhattacharyya, G. N. Jha, S. Rajendran, K. Saravanan, L. Sobha, and K. V. Subbarao, "A common parts-of-speech tagset framework for indian languages," in *Proceedings of LREC 2008*. European Language Resources Association, 2008, pp. 1331–1337. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=138364

[21] A. Navot, "On the role of feature selection in machine learning," Ph.D. dissertation, Hebrew University, 2006.

[22] B. G. P. K. D. Dipankar and D. S. Bandyopadhyay, "Part of speech (pos) tagger for kokborok," in *24th International Conference on Computational Linguistics*, 2012, p. 923.

[23] A. H. Fielding and J. F. Bell, "A review of methods for the assessment of prediction errors in conservation presence/absence models," *Environmental conservation*, vol. 24, no. 01, pp. 38–49, 1997.