JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.6, Issue.10, Oct. 2018 E-ISSN: 2347-2693

Comparison of Generative and Discriminative Models of Part of Speech Taggers for Marathi Language

Rushali Dhumal Deshmukh

Dept. of Computer Engineering, Dr. Babasaheb Ambedkar Technological University Lonere, Raigad, India

**Corresponding Author: radesh19@gmail.com, Tel.: +91-98502-50540*

Available online at: www.ijcseonline.org

Accepted: 6/Oct/2018, Published:31/Oct/2018

Abstract— Part of Speech (POS) tagging is the process of assigning grammatical category to words. POS tagger has wide variety of applications in the field of natural language processing, speech processing, information retrieval, machine translation, sentiment analysis, question answering etc. For Indian languages, the research in the field of POS tagging is still in progress. Marathi is the fourth spoken language in India and morphologically rich language. In this paper, we compared performance of Marathi POS tagger using generative and discriminative models. Using 32 tags, specified by Unified POS standard for Marathi, POS tagged dataset of 1500 news sentences, from different domains like sports, politics, entertainment etc., is generated. The Naive Bayes, Decision Tree, Neural Network, K Nearest Neighbour, Hidden Markov Model and Conditional Random Fields give 81%, 79%, 85%, 78%, 79% and 86% accuracy on test data respectively. Results show that neural network and Conditional Random Fields give better performance.

Keywords— Part of speech tagging, Generative models, Discriminative models, Naive Bayes, Decision tree, Neural network, Hidden markov model, Conditional Random Fields.

I. INTRODUCTION

Tagging is the process of automatic assigning descriptor to given tokens. Tagging assigns most probable class to given token. POS tagger assigns one of the most probable tag to given word. POS tags include tags for adjective, adverb, noun, conjunction etc. There are mainly two type of taggers: **rule-based** and **stochastic**. Rule-based taggers uses dictionary to assign possible tags and remove tag ambiguity by using hand-written rules. Stochastic taggers use probabilistic and statistical information for assigning most probable tag. There are two types of stochastic taggers. L

1. Transformation Based Learning (TBL) taggers: First from the training corpora, it assigns most likely tag to every word. Second, it uses transformation rules to transform one state to another to find the suitable tag for given word. Learning rules are simple, but without providing tag probabilities.

2. Probabilistic taggers: Finds most likely sequence of tags T for a sequence of words W.

a) Generative (Joint) Models: Suppose we have some data {{o,c}} of paired observations o and hidden classes c. It generates conditional probability P(o|c), based on naive bayes classifiers, Hidden markov models etc..

b) Discriminative (Conditional) Models: It computes P(c|o) based on Maximum Entropy based model, Conditional Random Fields etc..

Figure 1 shows the different types of taggers.

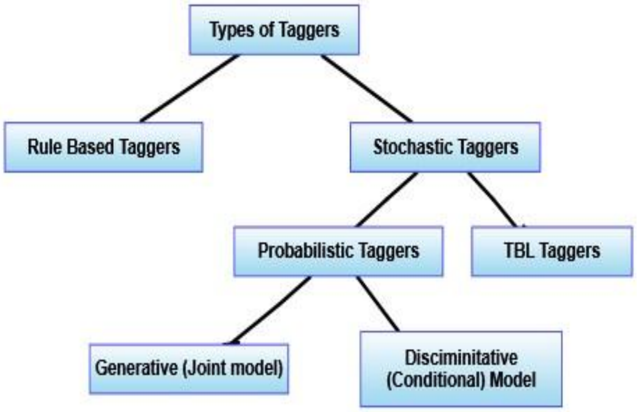

Figure 1. Types of Taggers

Marathi being free order language, existing POS taggers for English language cannot be used. Currently voluminous data is available on web which needs to be processed for number of applications like opinion mining [1] , sentiment classification[2] [3] , subjectivity analysis etc. For this POS tagging is preprocessing step.

Rest of the paper is organized as follows, Section I contains the introduction of Part of speech tagging and types of POS taggers, Section II contain the related work of POS tagging for non Indian as well Indian languages, Section III contain our Approach for assigning pos-tag to Marathi sentence and algorithms used, Section IV presents results and comparison with different classifiers, Section V concludes with conclusion and future scope.

II. Related Work

Singh Jyoti et al. [4] used Trigram method for Marathi POS tagging. In trigram method, given previous two tags of word, it computes most likely tag for a word. They reported 91.63% accuracy of the system for 2000 sentences with 48,635 words. Hidden-Markov model is used by Singh Jyoti et al. [5] for assigning POS tag to Marathi word. They computed probability of tag as follows.

P(ti|wi) = P(ti|ti-1). P(ti+1|ti). P(wi|ti) (1)
P(ti|ti-1) = It is the probability of current tag given the previous tag.
P(ti+1|ti) = It is the probability of future tag given the current tag.
With 1000 sentences (25,744 words) accuracy of the system achieved is 93.82%.
Patil H. B. et al. [6] presented Part - of-Speech Tagger for Marathi language using Limited Training Corpora: Rule based approach is used for assigning pos tag to word. They used following approach.

1) Tokenization: It takes input as raw text and generates token for it.
2) Morphological analysis: After tokenization, each word is searched in the dictionary. If word not found in dictionary, then stemming is done using affix list. Again it is searched in dictionary. This is repeated until word is found in the dictionary.
3) Morphological analysis assigns pos tags to word.
4) Eleven rules are used to remove ambiguity.
They reported average accuracy of 78.82%.

Unigram, Bigram, Trigram and HMM tagging methods are applied on Marathi text by Singh Jyoti et al. [7]. In order to measure the performance of systems, they developed a test corpus of 1000 sentences (25744 words).
Unigram tagger: Here most commonly used tag is assigned to word with reported accuracy of 77.39%.
Bigram tagger: It considers context for assigning a tag to the word, considering tag of previous word.

P(ti|wi) = P(wi|ti) . P(ti|ti-1) (2)
P(ti|ti-1) is the probability of ti given previous tag ti-1.
The reported accuracy of the system is 90.30%.

Trigram Tagger: It considers context for assigning a tag to the word, considering tag of previous two words achieving the accuracy of 91.46%.

HMM tagger: A HMM is the Statistical Model used for generating tag sequences. Basic idea of HMM is to determine the most likely tag sequences using the equation (1). The accuracy of the system is 93.82%.

Das et al. [8] used Support Vector Machine to do the POS tagging of Odia text. Here they considered 10000 Odia words. They achieved accuracy of 82% using only 5 tags in the dataset. The system has been tested for newspaper text. Brill, Eric [9] presented a simple rule based POS tagger. Initially, each word is assigned its most likely tag. Then patch templates are applied to check which words are not correctly tagged. As compared to statistical tagger it does not require large table of statistics, contextual information. With a training corpus of one million words, patch corpus of 65,000 words and test corpus of 65,000 words, the error rate was only 5%. Bach et al. [10] presented POS tagging of Vietnamese social media text of 4000 sentences from Facebook. With supervised approach, they achieved accuracy of 88.26% and with unsupervised 88.92% using conditional random fields. Multilingual POS tagging with weightless neural networks is used by Carneiro et al.[11]. As training of POS tagging is time consuming they proposed multilingual Weightless Artificial Neural Network tagger for eight languages Mandarian Chinese, English, Japanese, Portugese, Italian, German, Russian, Turkish. With one pass learning capability, it matches or outperforms the state-of-art methods. Neural Network based approach is used for POS Tagging of Hindi by Narayan et al. [12]. With news data of 2600 sentences and 11500 words, they reported accuracy of 91.30% using artificial neural network. Okhovvat et al.[13] presented a HMM Persian POS tagging. First, they have discussed POS tagging challenges for Persian language. They proposed HMM with 15 fold cross validation for 10 million words of Persian language. They have experimented with homogeneous as well as heterogeneous text and reported accuracy of 98.1%. Joshi et al. [14] presented transformation based approach for POS tagging of Kadazan using lexical and contextual rules. They achieved 92% to 93% accuracy with corpus1 of 741 words and corpus2 of 1328 words. Joshi et al. [15] reported accuracy of 92.13% with 500 sentences containing 11,720 words and 24 tags using HMM tagger for Hindi. Garg et al. [16] presented Rule Based Hindi POS tagger. They achieved accuracy of 87.55% for a corpus of 26,149 words and 30 tags.

III. Methodology

In our experimentation, Marathi news sentences from online news paper are used for creating corpus. Pos-tagged corpus containing 1500 sentences with 10115 word is created using

Unified POS standard in Indian Languages, Department of Information Technology, Ministry of Communications & Information Technology Govt. of India. Total 32 tags are used as shown in table 1, are used for creating POS tagged dataset. Dataset is verified from Marathi Linguistic Expert.
Example of a tagged sentence.

घरातल्या_NN सांस्कृतिक_JJ वातावरणामुळे_NN निर्माण_NN झालेली_VM संगीताची_NN गोडी_JJ आणि_CC त्यानंतर_NST सारेगमप_NNP ,_PUNC वर्ल्ड_UNK अंताक्षरी_NN या_DM स्पर्धेतून_NN भारताचं_NNP प्रतिनिधीत्व_NN करणाऱ्या_VM स्वरूप_NNP यांना_PR ओळख_NN आणि_CC नाव_NN मिळवून_VM दिलं_VM ._PUNC.

A. POS Tagset used

Table 1. POS Tagset used for tagging

Sr. No.	Category	Label	Examples
1.	Common Noun	NN	वर्ग, योजना, कुलूप, चोरी etc
2.	Proper Noun	NNP	लालासाहेब, दत्तात्रय, वैशाली etc
3.	Nloc Noun	NST	तेथील, जवळच, वर, पुढे, etc
4.	Pronoun	PR	येथे, तेथे, जो, तो etc
5.	Personal Pronoun	PRP	तो, मी, तू, ते, तुम्ही etc
6.	Reflexive Pronoun	PRF	स्वत: , आपण etc
7.	Relative Pronoun	PRL	ज्याने, जेव्हा, जिथे etc
8.	Reciprocal Pronoun	PRC	परस्पर, एकमेक etc
9.	Wh-word Pronoun	PRQ	कोण, केव्हा, कुठे etc
10.	Demonstrative	DM	हा, जो etc
11.	Deictic Demonstrative	DMD	इथे, तिथे etc
12.	Relative Demonstrative	DMR	ज्याने
13.	Wh-word Demonstrative	DMQ	कोणता, कोणी etc
14.	Main Verb	VM	आहे, केली, झोपला etc
15.	Auxiliary Verb	VAUX	लागला, आले etc
16.	Adjective	JJ	सुंदर, चांगला, मोठा etc
17.	Adverb	RB	लवकर, हळूहळू etc
18.	Conjunction	CC	आणि, कारण etc
19.	Conjunction Coordinator	CCD	पण, परंतु etc
20.	Conjunction Subordinator	CCS	कारण की, जर-तर, का की etc
21.	Particles	RP	तर
22.	Intensifier particles	INTF	खूप, बराच, अतिशय etc
23.	Negation particles	NEG	नको, न
24.	Quantifiers	QT	थोडे, जास्त, काही,पहिला etc
25.	General Quantifiers	QTF	थोडे, जास्त, काही etc
26.	Cardinals Quantifiers	QTC	एक, दोन, तीन etc
27.	Ordinals Quantifiers	QTO	पहिला, दुसरा etc
28.	Foreign word Residuals	RDF	A word written in script other than the script of the original text.
29.	Symbol Residuals	SYM	$, &, *, (,) etc
30.	Punctuation Residuals	PUNC	. , : ;
31.	Unknown Residuals	UNK	
32.	Echowords Residuals	ECH	आजूबाजूला, झेलाझेली, डोकेबिके etc

B. *Working of POS tagger*

Figure 2 shows working of our POS tagging system. First, corpus of 1500 sentences is created from Marathi news papers. Second, it is manually tagged using tagset mentioned in Table 1. From the dataset of 1500 tagged sentences 80% of dataset is used for training with 5 fold cross validation and rest for testing.

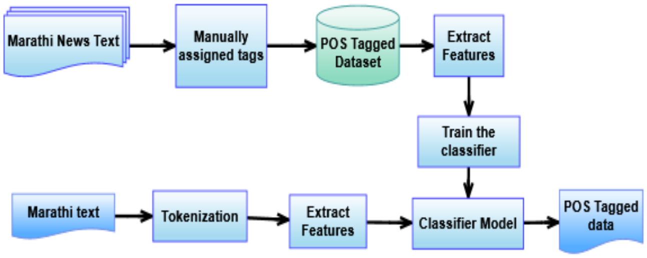

Figure 2. Flowchart of POS tagger

Following 13 features are extracted by feature extractor.
word : word itself
is_first : true if word is first else false
is_last : true if word is last else false
prefix-1 : first prefix of word
prefix-2 : first two prefixes of word
prefix-3 : first three prefixes of word
suffix-1: first suffix of word
suffix-2 : first two suffixes of word
suffix-3 : first three suffixes of word
prev_word : previous word
next_word : next word
has_hyphen : true if word has hyphen else false
is_numeric : true if word is numeric else false

e.g. "तो हुशार आहे."
Feature set extracted for above sentence is as follows.
[{'word': 'तो', 'is_first': True, 'is_last': False, 'prefix-1': '', 'prefix-2': 'तो', 'prefix-3': 'तो', 'suffix-1': 'ो', 'suffix-2': 'तो', 'suffix-3': 'तो', 'prev_word': '', 'next_word': 'हुशार', 'has_hyphen': False, 'is_numeric': False, 'prev-word': '<START>'}, {'word': 'हुशार', 'is_first': False, 'is_last': False, 'prefix-1': '', 'prefix-2': 'हु', 'prefix-3': 'हुश', 'suffix-1': 'र',

```
'suffix-2': 'ार', 'suffix-3': 'शार', 'prev_word': 'तो', 'next_word': 'आहे', 'has_hyphen': False, 'is_numeric': False, 'prev-word': 'तो'}, {'word': 'आहे', 'is_first': False, 'is_last': False, 'prefix-1': '', 'prefix-2': 'आह', 'prefix-3': 'आहे', 'suffix-1': 'े', 'suffix-2': 'हे', 'suffix-3': 'आहे', 'prev_word': 'हुशार', 'next_word': '.', 'has_hyphen': False, 'is_numeric': False, 'prev-word': 'हुशार'}, {'word': '.', 'is_first': False, 'is_last': True, 'prefix-1': '', 'prefix-2': '.', 'prefix-3': '.', 'suffix-1': '.', 'suffix-2': '.', 'suffix-3': '.', 'prev_word': 'आहे', 'next_word': '', 'has_hyphen': False, 'is_numeric': False, 'prev-word': 'आहे'}]
```

C. Algorithms used

1. **Naive Bayes Classifier**
   It is based on Bayesian classifier and assumes all features to be conditionally independent.
2. **Decision Tree**
   Internal node denotes a test on an attribute. Each branch represents outcome of test. Leaf nodes denote predicted class. At each node it chooses best attribute for splitting based on information gain computed using entropy which is a measure of impurity.
3. **Hidden Markov Model**
   It is an extension of markov chain. In hidden markov model hidden states are pos tags and output symbols are words. It uses emission probabilities i.e. word likelihood probabilities p(wi|ti) and tag transition probabilities p(ti|ti-1).
4. **Conditional Random Fields**
   CRFs are conditionally trained, undirected graphical models. CRF's are globally normalized. These are widely used and applied
5. **Neural Network**
   It consists of neurons arranged in three layers input, hidden and output layers. Several hidden layers can exist between input and output layers.
6. **K Nearest Neighbour**
   For classification it considers K nearest neighbours using distance metric. Euclidean distance is commonly used in continuous attributes. In discrete attributes hamming distance is used as distance metric. The most frequent label from K nearest neighbours is assigned to unlabeled data.

IV. Results and Discussion

Accuracy, precision, recall, f1-score, confusion matrix are used to measure performance of classifier.

True Positives (TP): These are actually true items classified as true by classifier.

False Positives (FP): These are actually false items but classified as true by classifier.

False Negatives (FN): These are actually true items classified as false by classifier.

True Negatives (TN): These are actually false items classified as false by classifier.

N: It is total number of items.

Accuracy: It is the number of correct predictions made over all predictions made.

Precision = True Positives / (True Positives + False Positives)

Recall = True Positives / (True Positives + False Negatives)

Accuracy = (True Positives + True Negatives)/N

Error = (False Positives + False Positives)/N

F1 = 2*Recall*Precision/(Recall + Precision)

Table 2. Results with all classifiers using 5 fold cross validation

Results	NB	DT	NN	KNN	HMM	CRFs
accuracy in fold 1	0.79	0.77	0.81	0.75	0.79	0.84
accuracy in fold 2	0.79	0.77	0.82	0.78	0.79	0.85
accuracy in fold 3	0.79	0.77	0.83	0.77	0.79	0.84
accuracy in fold 4	0.80	0.78	0.83	0.77	0.78	0.84
accuracy in fold 5	0.80	0.79	0.84	0.78	0.78	0.86
accuracy on train data	0.79	0.78	0.83	0.77	0.78	0.85
accuracy on test data	0.81	0.79	0.85	0.78	0.79	0.86
Average precision	0.80	0.82	0.85	0.78	0.78	0.86
Average Recall	0.81	0.80	0.85	0.78	0.79	0.86
Average f1-score	0.79	0.80	0.85	0.78	0.78	0.86
Average precision score, micro-averaged over all classes	0.81	0.80	0.85	0.79	0.79	0.87

Table 2 shows results with Naive Bayes(NB), Decision tree(DT) , Neural Network(NN), K Nearest Neighbour(KNN), Hidden Markov Model(HMM) and Conditional Random Fields(CRFs).

With 1500 sentences with 10115 words we used 5 fold cross validation for above classifiers with 80%-20% train-test split.

Above results shows that 81% accuracy is obtained for Naive Bayes, 79% accuracy is obtained for decision tree, 85% accuracy is obtained for neural network, 78% accuracy is obtained for K nearest neighbour, 79% accuracy is obtained for Hidden Markov Models, 86% accuracy is obtained for Conditional Random Fields(CRFs) on test data.
Average F1-score which is weighted average of precision and recall is 0.86 for conditional random fields.

Figure 3 gives comparison of top 3 classifiers among six classifiers used in our study.

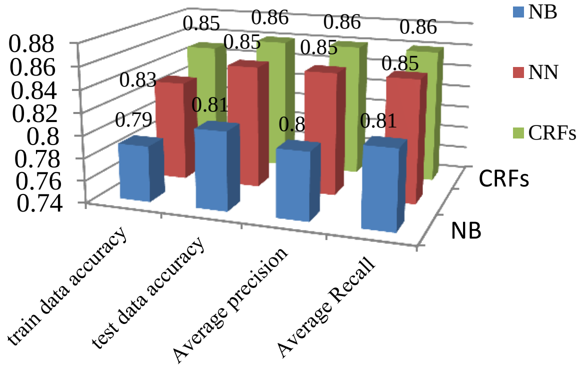

Figure 3. Result Comparison of top 3 classifiers

For all classifiers train and test data accuracy is almost same. There is no problem of overfitting.
Table 3 gives details of tag wise precision, recall, f1-score and support using CRFs.

Table 3. Tag wise precision, recall, f1-score and support using CRFs

Tag	Precision	Recall	F1-score	Support
CC	0.89	0.94	0.91	125
CCD	1.00	0.10	0.18	10
CCS	1.00	1.00	1.00	1
DM	0.87	0.95	0.91	122
DMD	0.33	0.33	0.33	3
DMQ	1.00	1.00	1.00	4
DMR	0.00	0.00	0.00	0
ECH	0.75	0.25	0.38	12
INTF	0.88	0.50	0.64	14
JJ	0.77	0.65	0.71	253
NEG	1.00	0.83	0.91	6
NN	0.84	0.93	0.88	1204
NNP	0.81	0.73	0.77	217
NST	0.87	0.72	0.79	36
PR	0.50	0.20	0.29	10
PRC	0.00	0.00	0.00	0
PRF	0.92	0.81	0.86	27
PRL	0.73	0.39	0.51	28
PRP	0.80	0.83	0.82	134
PRQ	0.73	1.00	0.84	8
PUNC	1.00	0.99	0.99	431
QT	1.00	0.50	0.67	2
QTC	0.94	0.92	0.93	107
QTF	0.82	0.78	0.80	60
QTO	1.00	0.65	0.79	23
RB	0.85	0.72	0.78	87
RDF	0.00	0.00	0.00	0
RP	0.22	0.20	0.21	10
SYM	0.95	0.98	0.97	88
UNK	0.75	0.17	0.27	18
VAUX	0.85	0.81	0.83	178
VM	0.88	0.89	0.89	625
Avg.	**0.86**	**0.86**	**0.86**	**3840**

Table 3 shows that total 3840 tags are tested. For NN 1204 tags tested. System reported precision of 0.84 and recall of 0.93 reason is that it is predicted as NNP or NST. Similarly some of NNP, NST are predicted as NN. For DMD precision, recall, f1-score achieved is 0.33. The reason is that it gets classified as DM. Also for RP tag precision, recall value is around 0.21. For PUNC precision is 1.00 and recall is 0.99, almost all PUNC tag are correctly identified.

Figure 4 shows precision recall (PR) curve using CRFs.

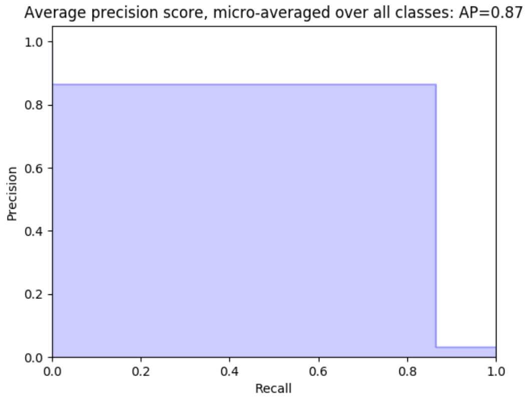

Figure 4. PR curve using CRFs

Precision indicates accuracy and recall indicates completeness. Precision and Recall are inversely related. Higher area under curve denotes higher precision and higher recall. In PR curve area under curve is 0.87 for CRFs.

V. Conclusion and Future Scope

Our system achieved highest POS tagging accuracy of 86% using CRFs with 1500 sentences with 10115 words. Though our dataset is for Marathi news paper, this can work for other Marathi text also. In many applications like word sense disambiguation, information retrieval, machine translation etc we need POS tagged dataset.

Previous work of Marathi POS tagging has been done with 26 tags, tagset developed by Bharati, Akshar, et al.[17], IIIT Hyderabad . Our results cannot be compared directly with existing POS taggers of Marathi Language as their datasets, POS tagsets are different. Our main contribution is the creation of Marathi POS tagged dataset using 32 tags specified by Unified POS standard in Indian Languages and testing its performance using generative as well as discriminative models. Results show that discriminative model performs better than generative models. Performance of system can be improved by increasing size of labelled dataset considering data from different domains. Other machine learning algorithms can also be applied.

REFERENCES

[1] Vadivukarassi, M., N. Puviarasan, and P. Aruna. "Identification of Opinion Words and Polarity of Reviews in Tweets using Aspect Based Opinion Mining." International Journal of Scientific Research in Computer Science, Engineering and Information Technology pp.282-289, 2017.

[2] Vidya, S. "Cross Domain Sentiment Classification Using Natural Language Processing." IJSRCSEIT pp.348-353,2018.

[3] Bollegala, Danushka, David Weir, and John Carroll. "Using multiple sources to construct a sentiment sensitive thesaurus for cross-domain sentiment classification." In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, pp. 132-141. Association for Computational Linguistics, 2011.

[4] Singh, Jyoti, Nisheeth Joshi, and Iti Mathur. "Part of speech tagging of Marathi text using trigram method." arXiv preprint arXiv:1307.4299,2013.

[5] Singh, Jyoti, Nisheeth Joshi, and Iti Mathur. "Marathi Parts-of-Speech Tagger Using Supervised Learning." Intelligent Computing, Networking, and Informatics. Springer, New Delhi, pp.251-257,2014.

[6] Patil, H. B., A. S. Patil, and B. V. Pawar. "Part-of-Speech Tagger for Marathi Language using Limited Training Corpora." IJCA Proceedings on National Conference on Recent Advances in Information Technology NCRAIT (4). pp.33-37, 2014.

[7] Singh, Jyoti, Nisheeth Joshi, and Iti Mathur. "Development of Marathi part of speech tagger using statistical approach." In Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on, pp. 1554-1559. IEEE, 2013.

[8] Das, Bishwa Ranjan, Smrutirekha Sahoo, Chandra Sekhar Panda, and Srikanta Patnaik. "Part of speech tagging in odia using support vector machine." Procedia Computer Science 48 .pp. 507-512,2015.

[9] Brill, Eric. "A simple rule-based part of speech tagger." In Proceedings of the third conference on Applied natural language processing, pp. 152-155. Association for Computational Linguistics, 1992.

[10] Bach, Ngo Xuan, Nguyen Dieu Linh, and Tu Minh Phuong. "An empirical study on POS tagging for Vietnamese social media text." Computer Speech & Language 50. pp. 1-15,2018.

[11] Carneiro, Hugo CC, Felipe MG França, and Priscila MV Lima. "Multilingual part-of-speech tagging with weightless neural networks." Neural Networks 66 pp.11-21,2015.

[12] Narayan, Ravi, S. Chakraverty, and V. P. Singh. "Neural network based parts of speech tagger for Hindi." IFAC Proceedings Volumes 47.1.pp.519-524,2014.

[13] Okhovvat, Morteza, and Behrouz Minaei Bidgoli. "A hidden Markov model for Persian part-of-speech tagging." Procedia Computer Science 3.pp. 977-981,2011.

[14] Alex, Marylyn, and Lailatul Qadri Zakaria. "Kadazan Part of Speech Tagging Using Transformation-based Approach." Procedia Technology 11.pp. 621-627,2013.

[15] Joshi, Nisheeth, Hemant Darbari, and Iti Mathur. "HMM based POS tagger for Hindi." Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing (AISC-2013). 2013.

[16] Garg, Navneet, Vishal Goyal, and Suman Preet. "Rule based Hindi part of speech tagger." Proceedings of COLING 2012: Demonstration Papers.pp.163-174,2012.

[17] Bharati, Akshar, et al. "Anncorra: Annotating corpora guidelines for pos and chunk annotation for indian languages." LTRC-TR31 (2006).

Authors Profile

Mrs. Rushali Dhumal(Deshmukh) pursed Bachelor of Computer Engineering from Pune University, India in 1999 and Master of Computer Engineering from Pune University, India in year 2007. She is currently pursuing Ph.D. in the area of Semantic Analysis of Natural Languages. Her main research work focuses on Natural Language Processing using Machine learning and Data Mining.