Journal of Multimedia Information System VOL. 5, NO. 3, September 2018 (pp. 147-154)
http://dx.doi.org/10.9717/JMIS.2018.5.3.147

Part-of-speech Tagging for Hindi Corpus in Poor Resource Scenario

Deepa Modi[1*], Neeta Nain[2], Maninder Nehra[3]

Abstract

Natural language processing (NLP) is an emerging research area in which we study how machines can be used to perceive and alter the text written in natural languages. We can perform different tasks on natural languages by analyzing them through various annotational tasks like parsing, chunking, part-of-speech tagging and lexical analysis etc. These annotational tasks depend on morphological structure of a particular natural language. The focus of this work is part-of-speech tagging (POS tagging) on Hindi language. Part-of-speech tagging also known as grammatical tagging is a process of assigning different grammatical categories to each word of a given text. These grammatical categories can be noun, verb, time, date, number etc. Hindi is the most widely used and official language of India. It is also among the top five most spoken languages of the world. For English and other languages, a diverse range of POS taggers are available, but these POS taggers can not be applied on the Hindi language as Hindi is one of the most morphologically rich language. Furthermore there is a significant difference between the morphological structures of these languages. Thus in this work, a POS tagger system is presented for the Hindi language. For Hindi POS tagging a hybrid approach is presented in this paper which combines "Probability-based and Rule-based" approaches. For known word tagging a Unigram model of probability class is used, whereas for tagging unknown words various lexical and contextual features are used. Various finite state machine automata are constructed for demonstrating different rules and then regular expressions are used to implement these rules. A tagset is also prepared for this task, which contains 29 standard part-of-speech tags. The tagset also includes two unique tags, i.e., date tag and time tag. These date and time tags support all possible formats. Regular expressions are used to implement all pattern based tags like time, date, number and special symbols. The aim of the presented approach is to increase the correctness of an automatic Hindi POS tagging while bounding the requirement of a large human-made corpus. This hybrid approach uses a probability-based model to increase automatic tagging and a rule-based model to bound the requirement of an already trained corpus. This approach is based on very small labeled training set (around 9,000 words) and yields 96.54% of best precision and 95.08% of average precision. The approach also yields best accuracy of 91.39% and an average accuracy of 88.15%.

Key Words: Hindi part-of-speech tagging, Hybrid approach, Probabilistic model, Rule-based model.

I. INTRODUCTION

NLP is a field of computer engineering, machine learning (artificial intelligence). NLP supports the development of an interface between human language and machine so that communication between machine and human can be easier [1]. While processing a natural language both input, as well as output, must be in the natural language. POS tagging is one of the fundamental ways to process a natural language by providing annotations to it. The activity of assigning a particular category to a word is called part-of-speech tagging. Here category means different grammatical class like pronoun, noun, conjunction, preposition, etc. Formally it can be defined as, "Given a meaningful sequence of words $w_1 \ldots w_n$, the POS process assigns respective POS tags $t_1 \ldots t_n$ to input sequence". Mathematically it can be stated as follows,

$$f(w_1 \ldots w_n) = t_1 \ldots t_n. \qquad (1)$$

POS tags also known as morphosyntactic tags impart useful statistics about a word. They tell about word's appropriate sense in the given context. POS tags also set some lexical features to the word like root, person, gender, and number, etc. POS tagging is a necessary step to perform further linguistic operations on a natural language like chunking and parsing [2]. POS tagging is a basic tool for various applications of NLP, such as text recognition, opinion mining, named-entity recognition, machine learning, machine translation, features inventory, sentiment mining, text summarization and sense-disambiguation removal, etc. The Hindi language is one of the highest morphological rich languages. So finding the ambiguity in tags is the major difficult task while talking about POS tagging for the Hindi language. For example, the word "हल्की" can be an adjective and can be an adverb too based on its context.

II. RELATED WORK

There exists plenty of research for part-of-speech tagging for the English language as it is one of the highest spoken languages in the world. The first system for English POS tagging was proposed by Brill [3], where the author defined a system by applying a hybrid approach which was

Manuscript received June 09, 2018; Revised July 03; Accepted July 18, 2018. (ID No. JMIS-2018-0035)
Corresponding Author (*): Deepa Modi, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, Rajasthan, India, 9929369501, deepa.modi22@gmail.com
[1]Department of CSE, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, Rajasthan, India, deepa.modi22@gmail.com
[2]Department of CSE, Malaviya National Institute of Technology, Jaipur, Rajasthan, India, nnain.cse@mnit.ac.in
[3]Department of CSE, Malaviya National Institute of Technology, Jaipur, Rajasthan, India, maninder4unehra@yahoo.com

based on transformation rules using a combination of stochastic and rule-based methods. The author achieved approximately 95% of accuracy for POS tagging. Zin and Thein [4] developed an effective POS tagging approach for Myanmar language. Their approach was using probabilities of Hidden Markov Model, as well as a pre-tagged corpus of size 1,000,000 words was also used in the system. With this large corpus, they achieved highest accuracy of 97.56%. Ekbal and Bandyopadhyay [5] proposed a Support Vector Machine based approach for Bengali language in their research work. Along with various word level features, a CRF-based NER (Named-entity recognition) system and a Lexicon is also used by them. Their approach acquired maximum accuracy of 86.84%.

Hindi is also a used by a large population around the world, but there exist only some implementations of part-of-speech tagging approaches for it. The most famous research project for Hindi part-of-speech tagging is developed by Lexical Resources for Indian Languages (LERIL); named as "Annotated Corpora" [6]. This project was using a statistical approach for performing POS tagging, using two kinds of tagsets which helps in providing syntactic information as well as semantic information and uses "Karaka's" to tag different tokens. Mishra and Mishra [7] used a rule-based approach for performing Hindi part-of-speech tagging. They also used a pre-tagged corpus of Hindi language for increasing the correctness of the system. Further, Garg et al. [8] also used a rule-based approach in their research to develop a POS tagging system. They considered different data sets to check the correctness of proposed approach and got an average precision of 85.47%. Singh et al. [9] performed morphological analysis upon input data and applied CN2 algorithm (learning algorithm based on decision trees) to develop a POS tagging system for Hindi and got 93.45% of accuracy. Their accuracy was further increased upto 94.38% by Dalal et al. [10]. The authors used Maximum Entropy Markov model using different features of MEM model. Narayan et al. [11] proposed an approach using the quantum neural network for Hindi POS tagging. Ghosh et al. [12] suggested a POS tagging system based on Conditional Random Field. Their system works for code-mixed text for Tamil, Hindi, and Bengali. They got highest accuracy of 75.22%.

POS tagging approaches can be broadly classified into two types as statistical-based approaches and rule-based approaches. Normally statistical approaches are used to design different POS taggers as they only need statistics about the language and do not require the in-depth grammatical knowledge. In this work, an approach is presented for part-of-speech tagging, which is a combination of probabilistic approach (which increases tagging) and rule-based approach (which bounds the size of the corpus). Ordinarily, rule-based methods are challenging to implement as they require depth grammatical knowledge about the language.

Table 1. Tagset for POS tagging for Hindi language.

Category	Annotation Convention	Category	Annotation Convention
Noun	NN	Quantifier	QF
Proper Noun	NNP	Number Quantifiers	QFNUM
Pronoun	PRP	Intensifier	INTF
Verb Finite Main	VFM	Negative	NEG
Verb Auxiliary	VAUX	Compound Common Nouns	NNC
Verb Nonfinite Adjectival	VJJ	Compound Proper Nouns	NNPC
Verb Nonfinite Adverbial	VRB	Noun in kriya mula	NVB
Verb Nonfinite Nominal	VNN	Adj in kriya mula	JVB
Adjective	JJ	Adv in kriya mula	RBVB
Adverb	RB	Interjection words	UH
Noun location	NLOC	Punctuation marks	PUNC
Postposition	PREP	Time	TIME
Particle	RP	Date	DATE
Conjunction	CC	Special Symbols	SYM
Question Words	QW		

III OUR APPROACH

In this work we have presented a Hindi POS tagging system which provides three facilities that splits input text into sentences, tokenizes input text into words and assigns POS tags to input text.

Split and tokenize functionality of the system is developed with the help of Unicode values. POS tagging system is designed using a combination of probability-based approach and rule-based approach. Java language is used as the developing environment, and a manually developed corpus of 9,000 words of Devanagari Hindi language is also used. The presented approach also works well in poor resources scenario.

The system works in two consecutive phases. In the first phase it tags known words (available in the trained corpus), and in the second phase it labels unknown words (not available in the trained corpus) and provides a tag sequence $t_1 \ldots . t_n$ for input word sequence $w_1 \ldots . w_n$. The following section details the tagset developed by us and the approach followed by the system.

Journal of Multimedia Information System VOL. 5, NO. 3, September 2018 (pp. 147-154)
http://dx.doi.org/10.9717/JMIS.2018.5.3.147

3.1 TAGGING CATEGORIES

We have designed a tagset which contains 29 part-

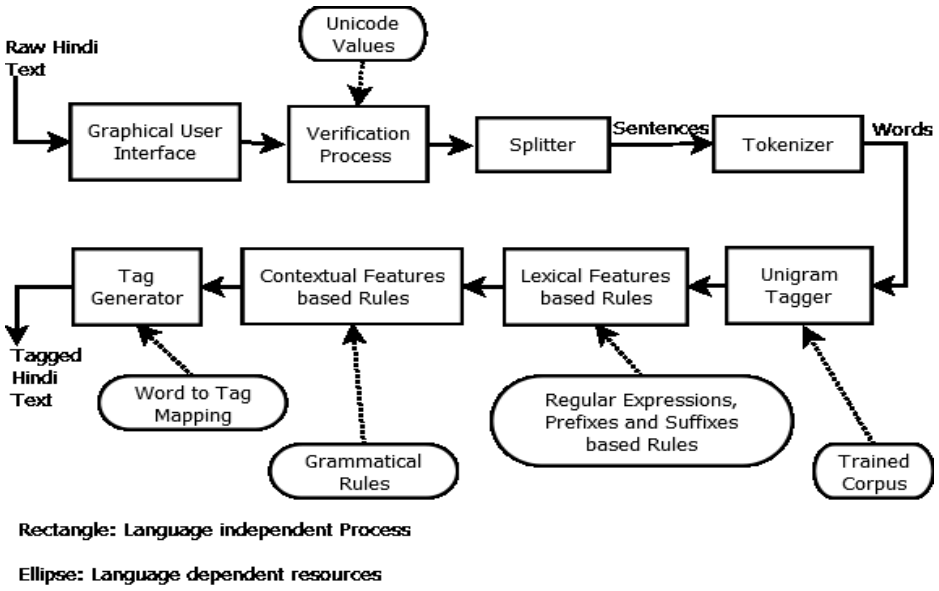

Fig. 1: Proposed approach for part-of-speech tagging for Hindi text

of-speech tags for the Hindi language. The tagset is motivated from IIIT-Hyderabad taget for Hindi [13]. This tagset also includes tags for time and date in all possible formats. The complete tagset is shown in Table 1.

3.2 THE TAGGER

The overall architecture of the presented Hindi POS tagging approach is shown in Fig. 1. The input text is given as a GUI. Verification process verifies the input text as it must be written in Devanagari Hindi only. The verification process will reject other languages text. The accepted data will be split into its constituent sentences by the Splitter. For splitting "Purnavirama (।)" and "Prashanvachak Chinha (?)" are taken into account. Split sentences will be further tokenized into component words by the Tokenizer. For tokenization "Space" and "Special symbols (@, #, +, etc.)" are taken into account. Both splitter and tokenizer are implemented using Unicode values of Hindi language.

After splitting and tokenizing, part-of-speech tagging is performed on the input data as follows,

3.2.1 Known Word Tagging using Unigram Model

The system uses unigram (n-gram, n=1) model based approach for "known words tagging". According to Unigram model, (also known as a lexical model) class of a given word will be only determined by its own part-of-speech tag and will not depend on its contextual environment. That means part-of-speech tag of a word is not affected by the tags of its left and right neighbors. The approach of unigram model can be stated as given in equation 2.

$$P_{unigram}(t_1 t_2 t_3 t_4) = P(t_1)P(t_2)P(t_3)P(t_4) \quad (2)$$

In the above equation, t_i shows the POS tag for a word w_i and P stands for the probability. Equation 2 states that probabilities of consecutive tags will be dependent only on own likelihood and it will be independent of the probabilities of its surrounding tags. Equation 3 states the formula for calculating Unigram probability for a given word w,

$$P(t_i|w) = \frac{freq\left(\frac{w}{t_i}\right)}{freq(w)} \quad (3)$$

Here w is a random input word, and t_i represents all possible tags for word w in the pre-tagged corpus. For word w, Unigram model finds the highest probable tag ti by maximizing the value of $P(t_i|w)$ and assigns a tag to the word with the highest probability. The unigram tagger provided in our system follows this approach and assigns a unique, most probable part-of-speech tag to every known word using already trained corpus for this task.

3.2.2 Unknown Word Tagging by applying Rules based upon Lexical and Contextual Features

For handling unknown words, the proposed approach is using a rule-based method. These rules are constructed around various lexical and contextual features and are derived by calculating probabilities of various words, their left and right neighbors and their combinations in the pre-tagged corpus.

3.2.2.1. Rules based upon Lexical Features

Lexical features based rules include rules based on regular expression, rules based on prefixes and rules based on suffixes. These rules find particular patterns in an input text. Lexical features do not affect from the context of a current word.

3.2.2.1.1. Rules-based on Regular Expressions

These rules follow different finite state machine automata for matching particular patterns. Here finite state machines are abstract data processing model, generally used in simulation and consists of various states, transition function, input and output symbols. Upon receiving a (state, symbol) pair, transition function generates an output symbol. In the proposed approach, rules based on regular expressions search patterns for punctuation marks, special symbols, numeric data, date format and time format in the input text with the help of different FSM and assign tag according to them.

$$t_i(w_i) = \Re_{finder}(pattern, tag) \quad (4)$$

In the equation 4, $\Re_{finder}$ is a regular expression finder based on different FSM automaton, which finds different patterns in the input Hindi text and according to that assigns a tag t_i to the word w_i.

3.2.2.1.2. Rules-based on Prefixes

The Hindi language contains some words which start with prefixes like **"अति", "अधि"** etc. According to its prefix, a word can be tagged with relatively high probability tag like adjective, noun, etc. The approach includes 23 different rules based on prefixes. The system searches for these prefixes and tag a word according to them. These rules are extreme in POS tagging as a significant number of Hindi words starts with prefixes.

$$t_i(w_i) = search_{prefix}(word) \quad (5)$$

In the equation 5, *search* procedure checks whether the current word starts with a particular prefix, if yes then assigns a tag t_i to the word w_i according to the rules exists in the proposed approach.

3.2.2.1.3. Rules-based on Suffixes

In the same way of prefix words, there are many words in the Hindi language which end with a particular suffix like **"इयल", "ऐरा"** etc. According to its suffix, a word can be tagged as a noun, adjective, etc. with high probability. The system is having 17 different rules based on suffixes.

$$t_i(w_i) = search_{suffix}(word) \quad (6)$$

In the equation 6, *search* procedure checks whether the current word starts with a particular suffix, if yes then assigns a tag t_i to the word w_i according to the rules exists in the proposed approach. Some examples of prefix and suffix based rules are shown in Table 2.

3.2.2.2 Rules based upon Contextual Features OR Hindi Grammar

These rules are chosen from Hindi language grammar and are based upon various combinations of the current word, and its left and right neighbors mean depended upon word's context. This section states various rules which have been applied to the system so that more and more unknown words can be handled. These rules are strong enough to increase the precision and automaticity of the system.

Table 2. Examples of rules based on most common prefixes and suffixes in Hindi language.

Prefix/Suffix	Tag	Example
अति	Adjective	अतिशय
अधि	Noun	अधिकरण, अधिकार
उप	Noun	उपवन, उपकार
आ	Noun	आहार
आलू	Adjective	झगड़ालू
इयल	Adjective	मरियल, सड़ियल
वाला	Adjective	किस्मतवाला
ऐरा	Noun	लुटेरा

$$t_i(w_i) = Apply_{rules\_context}(tag_{i-1} tag_i tag_{i+1}) \quad (7)$$

Here tag_i is the tag of current word w_i, tag_{i-1} and tag_{i+1} are tags of left and right neighbors of the current word w_i. According to these rules if we know the tags of the preceding and the succeeding of the current word, then we can apply the following grammatical rules to find the tag for the current word.

These rules are stated as follows:

1. If current tag is post position, then the previous tag will be probably a noun.

   Example: **राम ने पानी में कमल देखा।**

   Explanation: In this example, **"में"** is a post position and **"पानी"** is a noun.

2. If current tag is an adjective, then the next tag will be probably a noun.

   Example: **सीता अच्छी लड़की हैं।**

   Explanation: In this example, **"अच्छी"** is an adjective and **"लड़की"** is a noun.

3. If current tag is a pronoun, then next tag will be probably a noun.

   Example: **यह तुम्हारा बस्ता हैं।**

   Explanation: In this example, **"तुम्हारा"** is a pronoun and **"बस्ता"** is a noun.

4. If current tag is a verb (Finite Main, Nonfinite Adjectival, Nonfinite Adverbial or Nonfinite Nominal), then the previous tag will be probably a noun.

   Example: **वह बाजार जा रहा हैं।**

   Explanation: In this example, **"जा"** is a verb and

"बाजार" is a noun.

5. If two following tags are a noun, then the first tag will be probably a compound common noun.
   Example: **राज्य सरकार ने अच्छा काम किया।**
   Explanation: In this example, **"राज्य"** is a compound common noun and **"सरकार"** is a noun.

6. If current tag is a noun and the next tag is a proper noun, then the current tag will be probably compound proper noun.
   Example: **राम गोयल जा रहे हैं।**
   Explanation: In this example, **"राम"** is a compound proper noun and **"गोयल"** is a proper noun.

7. If current tag is an auxiliary verb, then the previous tag will be probably a finite main verb.
   Example: **रमा जा रही हैं।**
   Explanation: In this example **"रही"** is an auxiliary verb and **"जा"** is a main verb.

8. If current tag is a verb and the previous tag is a noun, adjective or adverb, then the previous tag is changed to a noun in kriya mula, an adjective in kriya mula or an adverb in kriya mula respectively.
   Example: **उसने फल हरा होते ही तोड़ लिया।**
   Explanation: In this example, **"होते"** is a verb and **"हरा"** is an adjective in kriya mula.

IV EXPERIMENTS AND RESULTS

To check the correctness, performance, and validity of the proposed approach, various experiments have been conducted. Some examples of part-of-speech tagging from the presented system are given as follows:

1. Input Text: **अभ्यर्थियों को दो अतिरिक्त मौके परीक्षा देने के लिए मिलेंगे।**
   Output Text: **अभ्यर्थियों_NN को_PREP दो_QFNUM अतिरिक्त_JJ मौके_NN परीक्षा_NN देने_VNN के_PREP लिए_PREP मिलेंगे_VFM ।_PUNC**
2. Input Text: **देश में अंग्रेजों को आने से रोकने के लिए टीपू सुल्तान ने बहुत कुर्बानी दी थीं। विशाखापट्नम की लड़ाई में उनकी मौत हुई थी, जिसे लोग शहादत मानते हैं।**
   Output Text: **देश_NN में_PREP अंग्रेजों_NN को_PREP आने_VNN से_PREP रोकने_VNN के_PREP लिए_PREP टीपू_SYM सुल्तान_NNP ने_PREP बहुत_QF कुर्बानी_NN दी_VFM थीं_VAUX ।_PUNC विशाखापट्नम_NN की_PREP लड़ाई_NN में_PREP उनकी_PRP मौत_NN हुई_VFM थी_VAUX ,_PUNC जिसे_NNC लोग_NN शहादत_NN मानते_VFM हैं_VAUX ।_PUNC**
3. Input Text: **बिहार चुनाव में विशेष पैकेज की घोषणा कर पीएम मोदी ने बढ़त हासिल कर ली थी, लेकिन चुनाव में अपने सहयोगियों के बयान और आरएसएस के आरक्षण के मुद्दे पर हो रहे नुकसान को देख पीएम ने अपना संयम खो दिया। मोदी के बयान पीएम पद की गरिमा के मुताबिक नहीं थे, ये सवाल उठने लगे।**
   Output Text: **बिहार_NNP चुनाव_NN में_PREP विशेष_JJ पैकेज_NN की_PREP घोषणा_NN कर_VFM पीएम_NNPC मोदी_NNP ने_PREP बढ़त_NVB हासिल_NN कर_VFM ली_VFM थी_VAUX ,_PUNC लेकिन_CC चुनाव_NN में_PREP अपने_PRP सहयोगियों_NN के_PREP बयान_NN और_CC आरएसएस_NN के_PREP आरक्षण_NN के_PREP मुद्दे_NN पर_PREP हो_VFM रहे_VAUX नुकसान_NN को_PREP देख_VFM पीएम_NN ने_PREP अपना_PRP संयम_NN खो_VFM दिया_VAUX ।_PUNC मोदी_NNP के_PREP बयान_NN पीएम_NNC पद_NN की_PREP गरिमा_NN के_PREP मुताबिक_JJ नहीं_NEG थे_VFM ,_PUNC ये_PRP सवाल_NN उठने_VFM लगे_VAUX ।_PUNC**

In the above examples, input Devanagari Hindi texts are tagged with the respective class of part-of-speech according to Hindi grammar. For tagging, Unigram model from probability class is applied as well as Hindi grammar rules are also applied to tag the unknown words (words which does not exist in the pre-tagged corpus). The following section describes the corpus dataset used in the experiments, the performance measures used for evaluation, the performance of the proposed approach and comparative analysis of evaluation results with previously available work.

4.1 TEST DATA SETS

The corpus contains around 9,000 words, and complete corpus belongs to news domain. For all experiments, data is collected from various domains like history, news, politics, science, and literature. All the corpus data is collected from online sources like online newspapers, story books, sites and open articles. Test data is around 22% in size as compared to the training set. Fig. 2 shows the size of various data sets from different domains.

The test datasets follow the Gaussian distribution with a mean of 566 words about history domain. 72.39% of the data is captured under 1-standard deviation, and 100% data is captured under 2-standard deviation. So it is

showing Bell's curve.

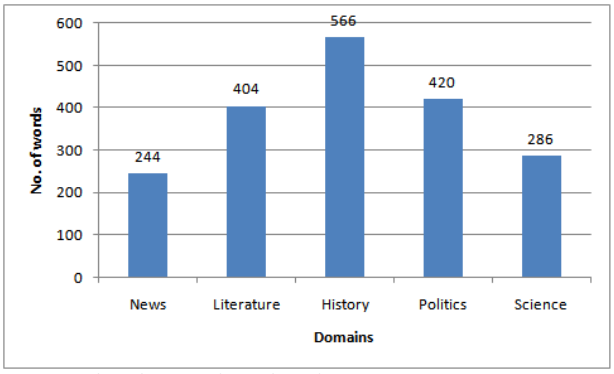

Fig.2. Test data from various domains.

4.2 EVALUATION MEASURES

To judge the significance and quality of the approach, various types of evaluation measures are used in general are precision, recall, true positive rate, false positive rate etc.

Table 3. Confusion matrix for a two-class problem

	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Three evaluation parameters viz. recall, precision and accuracy are used to check the performance of the approach. All three parameters are generated from the Table 3 [14] and shown in equation 8, equation 9 and equation 10 respectively.

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

$$Precision = \frac{TP}{TP+FN} \quad (9)$$

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (10)$$

Here recall also called sensitivity is the measure of "how many words got tagged correctly by the presented approach from the complete set of input Hindi text." Precision is the measure of "how many words are tagged correctly from the complete set of data which have been tagged by the approach" whereas accuracy can be defined as "total number of correctly handled words by the presented approach out of total input data." Both precision and recall are therefore based on the understanding and measure of relevance of tags here, as precision is indicative of "how useful the tagged results are" and recall measures "how complete the tags are". A high value of both is preferred.

4.3 PERFORMANCE ANALYSIS

To judge the correctness of presented approach validation is performed using holdout method of cross-validation where complete data set is divided into two different sets. Here testing data used is around 22% in size as compared to the training corpus. Upcoming data is tagged according to the rules described in section 3.

Achieved results for POS tagging are shown in Table 4. The system yields 95.08% of average precision, 88.15% of average accuracy and 92.70% of average sensitivity. The system also yields 96.54% of best precision, 91.39% of best accuracy and 94.67% of best sensitivity for News domain. The system shows lowest results for Literature domain because text belonging to this domain is slightly different from general text. The results for POS tagging are illustrated graphically also as shown in Fig. 3.

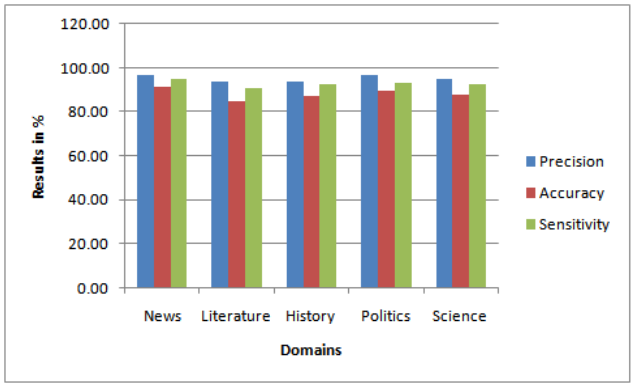

Fig.3. POS tagging: precision, accuracy and sensitivity for various domains.

4.4 COMPARATIVE ANALYSIS

There exists moderated work in the area of part-of-speech tagging for the Hindi language. But to the best of our knowledge, achieved precisions in this work are highest with good accuracy while having the smallest dataset of an already tagged corpus of the Hindi language.

Table 4. Performance of the proposed POS tagging approach in terms of precision and recall.

Domains	No of words	No of tagged words	Correctly tagged words	Precision	Accuracy	Recall/ Sensitivity
News	244	231	223	96.54%	91.39%	94.67%
Literature	404	367	343	93.46%	84.90%	90.84%
History	566	524	492	93.89%	86.93%	92.58%
Politics	420	391	377	96.42%	89.76%	93.10%
Science	286	264	251	95.08%	87.76%	92.41%

The presented approach gives the average precision of 95.08% while using a small-sized pre-tagged corpus of 9,000 words. Previously Garg et al. [8] reported 85.47% of precision with training data of around 18,000 words. Dalal et al. [10] and Singh et al. [9] achieved 94.38% and 93.45 % respective accuracies with around training data of 15,500. All data sets used by all these authors are larger as compare to the data set used in the presented approach.

V CONCLUSION AND FUTURE WORK

In this work, we presented an approach for part-of-speech tagging for Hindi Devanagari script. A combination of probabilistic approach and rule-based approach was used while developing the system. For tagging known words a Unigram model of probability class was applied, and for unknown words, various rules were derived from Hindi grammar. These rules were based on prefixes, suffixes and contextual environment of the word. Contextual rules were obtained by calculating probabilities of the current word along with its left and right neighbors if any. All types of rules were based on regular expressions and implemented using different finite machine automaton. We achieved an average precision of 95.08% for upcoming new Hindi data.

As a future work, we would increase the correctness of the system by emphasizing on more hybrid approaches as well as by expanding our rule-set rather than emphasizing on the size of the data-set. We would also propose an algorithm for removing ambiguity in POS tagging as well as we would provide some additional facilities like chunking and parsing of Hindi text.

REFERENCES

[1] C. D. Manning and H. Schutze, Foundations of Statistical Natural Language Processing. USA, MIT Press, 1999.

[2] A. Voutilainen. Part-of-speech tagging. The Oxford handbook of computational linguistics, 219-232, 2003.

[3] E. Brill, “A Simple Rule-based Part of Speech Tagger,” in Proceedings of the Third Conference on Applied Natural Language Processing (ANLC ’92), Proc ACL, Italy, pp. 152–155, 1992.

[4] K.K. Zin and N.L. Thein,“Part of speech Tagging for Myanmar using Hidden Markov Model,” In Proceedings of the International Conference on the Current Trends in Information Technology (CTIT), Dubai, UAE, pp. 1–6, 2009.

[5] A. Ekbal and S. Bandyopadhyay, “Part of Speech Tagging in Bengali Using Support Vector Machine,” In Proceedings of the International Conference on Information Technology, ICIT ’08, India, pp. 106–111, 2008.

[6] “AnnCorra: An Introduction,” 2010; https://www.sketchengine.eu/wp-content/uploads/posguidelines_indian_languages.pdf.

[7] N. Mishra and A. Mishra, “Part of Speech Tagging for Hindi Corpus,” In Proceedings of the International Conference on Communication Systems and Network Technologies (CSNT), India, pp. 554–558, 2011.

[8] V. Goyal, N. Garg and S. Preet, “Rule Based Hindi Part of Speech Tagger,” in Proceedings of the Coling, pp. 163-174, 2012.

[9] S. Singh, K. Gupta, M. Shrivastava and P. Bhattacharyya, “Morphological Richness Offsets Resource Poverty- an Experience in Building a POS Tagger for Hindi,” in Proceedings of the COLING/ACL, Sydney, Australia, 2006.

[10] A. Dalal, K. Nagaraj, U. Sawant, S. Shelke, and P. Bhattacharyya, “Building Feature Rich POS Tagger for Morphologically Rich Languages: Experiences in Hindi,” in Proceedings of the ICON, 2007.

[11] R. Narayan, V. P. Singh, and S. Chakraverty, "Quantum neural network based parts of speech tagger for Hindi," International Journal of Advancements in Technology, vol 5, no. 2, pp. 137-152, 2014.

[12] S. Ghosh, S. Ghosh, and D. Das, "Part-of-speech Tagging of Code-Mixed Social Media Text," In Proceedings of the Second Workshop on Computational Approaches to Code Switching, pp. 90-97, 2016.

[13] “A Part of Speech Tagger for Indian Languages (POS tagger)”, 2007; http://shiva.iiit.ac.in/SPSAL2007/iiit tagset guidelines.pdf.

[14] U. M. Fayyad, P. Smyth, and R. Uthurusamy (Eds.),

Advances in Knowledge Discovery and Data Mining, USA, American Association for Artificial Intelligence, 1996.

Authors

Neeta Nain, presently working as Associate Professor, Department of Computer Science and Engineering, Malaviya National Institute of Technology Jaipur, has a teaching experience of over 21 years. Her research area is Pattern Recognition, Machine Learning and Biometrics. She has published more than seventy papers on these topics for various International Journals and conferences like Elsevier Journal of Visual Communication and Information Representation, ACM Transactions on Asian Language Information Processing, IETE Journal of Research, Springer Multimedia Tools and Applications etc. She was also Program Chair, of The 13th IEEE International Conference on SIGNAL IMAGE TECHNOLOGY & INTERNET BASED SYSTEMS (SITIS2017).

Maninder Nehra was born in India. He received his M.Tech degree in computer engineering stream from Malaviya National Institute of Technology, Jaipur, India and now pursuing Ph.D. in computer engineering department from the same institute. His research interests include Hindi language processing, image processing, and pattern recognition.

Deepa Modi received her B.Tech degree in the Department of Computer Science and Engineering from Rajasthan Technical University, Rajasthan, India in 2010. She received her M.Tech degree in the stream of Computer Engineering from Malaviya National Institute of Technology, Jaipur, India, in 2015. Her research interests are artificial intelligence, natural language processing, genetic algorithms and pattern recognition.