Part-of-Speech Tagging of Hindi Corpus Using Rule-Based Method

Deepa Modi and Neeta Nain

Abstract The main goal of analysis of NLP (natural language processing) is to understand natural languages by parsing them. In the practice of analyzing natural languages there exist various sub-tasks. These sub-tasks depend on inbuilt structure of language and do not require complete knowledge and understanding of language. Part-of-speech tagging is one of them. Part-of-speech tagging is basically a practice of assigning language-specific grammatical tags to each word of language-specific input text, according to word's appearance in the text. These tags can be like noun, adverb, number, negative, etc. There exist a variety of taggers for most popular language in the world, i.e., English. But such taggers cannot be used for morphologically rich Hindi language as difference exists between structures of both languages. A "Rule-based system" is presented in this paper. 29 standard part-of-speech tags are used, including two special tags for date and time also in multiple formats. The special tags like punctuation, time, and date are based on regular expressions. Main aim of the proposed system is to increase automaticity and maintain high precision, while limiting the size of human made corpus. Proposed system uses human made corpus of around 9,000 words to increase tagging and rule-based (lexical features based) approach to decrease the size of already trained corpus. The system yields 91.84 % of average precision and 85.45 % of average accuracy.

Keywords Hindi · Probabilistic method · Rule-based method · Part-of-speech tagging

Deepa Modi (✉) · Neeta Nain
Department of Computer Science and Engineering, MNIT, Jaipur 302017, India
e-mail: deepa.modi22@gmail.com

Neeta Nain
e-mail: neetanain@yahoo.com

N. Afzalpulkar et al. (eds.), *Proceedings of the International Conference on Recent Cognizance in Wireless Communication & Image Processing*,
DOI 10.1007/978-81-322-2638-3_28

Introduction

Corpus annotation is a basic technique for processing natural languages. In the process of language annotation, input/output always appears in the pattern of a natural language like English, Hindi, etc. There exists more than one level of corpus annotation. For example, morphological analysis, POS tagging, chunk tagging, etc. POS tagging is basic step for language processing and can work as starting phase in other language processing tasks. It is a technique of assigning a token in a sentence as a particular POS tag or lexical belonging to a particular class (noun, number, adverb, time, date, etc.) based on its definition, contextual information, and morphological information. Formally it can be stated as, "While at the input a meaningful sequence of words $w_1 \ldots w_n$ is given, at the output the system have to assign respective POS tags $t_1 \ldots t_n$ to the provided sequence." These tags are useful in assigning some additional information about a word. They state about relevance of word in the given context, they tell what the word's role is in a given sentence; they give grammatical category to the word and assign grammatical features like person, number to the word. POS tagging is an intermediate step in processing of various NLP tasks, such as shallow and full parsing, word-sense disambiguation, machine learning, etc.

Hindi language is a feature rich language. So the most challenging objective in the area of POS tagging for Hindi language is identifying the ambiguities in tags.

Related Work

There exist a number of part-of-speech taggers for many languages using a number of approaches, especially for English language. Brill [1] defined a system based on transformation rules. Accuracy of this system is around 95 %. Zin and Thein [2] developed an algorithm for efficient POS tagging for Myanmar language based on pre-tagged corpus and probabilities calculated using HMM. The highest accuracy achieved by them is 97.56 % with training data of 1,000,000 words. For Hindi Language also there exists a number of implementation of POS taggers. AnnCorra, shortened for "Annotated Corpora," is a project of Lexical Resources for Indian Languages (LERIL), is a collaborative effort of several groups. They developed a system using statistical approach, which provides syntactic and semantic information [3]. Mishra and Mishra [4] designed a POS system based on manually developed database of tagged Hindi words and an approach using rule-based method. Garg et al. [5] also implemented a POS system in their research for Hindi language based on rule-based approach and achieved average precision of 85.47 % on different data sets. Singh et al. [6] defined a system based on morphological analysis and CN2 algorithm (decision tree based learning algorithm). They got 93.45 % of accuracy for part-of-speech tagging for Hindi language which is further

increased by Dalal et al. [7] to 94.38 % using maximum entropy Markov model based on different features.

There are two types of approaches for POS tagging, stochastic based and rule based. Generally, stochastic methods are used to develop POS taggers as these methods require small knowledge of language and are very easy to implement. We propose a POST system which is based on rule-based technique. Rule-based approach generally requires vast knowledge of the language and is difficult to develop.

System Description

The proposed designed POS tagging system is useful for Hindi language processing. This system is developed using rule-based approach, which includes grammatical rules (based on prefixes and suffixes) and regular expression-based rules. 29 part-of-speech tags are used in standard format. 27 POS tags are taken from IIIT—Hyderabad tagset [8] and two new special tags are included for time and date. Around 9,000 Hindi words database is prepared, which is stored in an .XML file. The proposed system is developed in Java language.

Approach Followed by the System

The proposed part-of-speech tagger (POST) system accepts data in Devanagari Hindi. The system verifies the input text, as text must be in Devanagari Hindi. After verification of input data, the system works in three sequential steps. In the very first step, it finds similarity of every word of the input data with the already trained language dependent corpus and tries to find a match. If a match is available then corresponding tag is assigned to input word. If there is no match available for a word then system goes to second step for further tagging.

In second step, the system searches various regular expressions (based on various finite state machines) in input text for numbers, punctuation marks, special symbols, time, and date like 987, *, &, 23:59, etc., and allot specific POS tag to input text. This kind of matching of regular expressions is very good as it increases the precision and accuracy of the system with significant percentage. Various string matching algorithms are applied here for matching a particular pattern.

In the final and third steps, the POST system applies various lexical rules (works on assumption that the tag for a word depends only on current word and not dependent on previous and next words and their tags) based on suffixes and prefixes of Hindi language to assign tags to the remaining unknown words. These rules are very powerful in part-of-speech tagging as there are many words in Hindi language, which start from prefixes or end with suffixes. Some of the examples of these rules are shown in Table 1.

Table 1 Some example rules based on prefixes and suffixes

S.No.	Prefix/ Suffix	Tag	Example
1	अति	Adjective	अतिशय,
2	अधि	Noun	अधिकरण, अधिकार
3	अनु	Noun	अनुकरण, अनुचर
4	अक /आक	Noun	लेखक, तैराक, लडाक
5	अक्कड	Adjective	भुलक्कड, घुमक्कड
6	आलू	Adjective	झगडालू

System Modules Description

A system can be very large to manage means as size and functionality of a system increases, it is very difficult to handle the system. So to manage the system, generally system is divided in subsystems or modules. Modularity defines the degree to which system components can be separated or recombined. As the value of the modularity increases, the system becomes more manageable and easy to handle. The presented system has following modules.

Read and Verify Hindi Text

The very first module of system reads and verifies Hindi data. It contains a text area in GUI. Here user has to enter his Hindi text. Module reads and verifies this text. Data must be Devanagari Hindi.

Split in Sentences

This module breaks input Hindi data in individual sentences according to delimiter, which can be "Puranviram" or "prashanvachak chinha." In this module input will be Hindi (Devanagari) data and output will be individual Hindi sentence.

Tokenize in Words

This module breaks input Hindi data in individual words according to delimiter "space." In this module input will be Hindi (Devanagari) data and output will be individual Hindi words. Output will be displayed in GUI.

Tag Hindi Data

This module tags each word of input Hindi (Devanagari) data with tags like pronoun, adverb, date, number, verb, time, etc. Words which are not tagged using corpus matcher or various rules are tagged as "SYM" tag. In this module input will be Devanagari Hindi data and output will be POS tagged Devanagari Hindi data. Output will be displayed in GUI.

Experiments and Results

A POS tagger system generally gives three main functionalities as breaking of input text in individual sentences and words and POS tagging for input text. Our system too provides these functionalities with 100 % correctness of split and tokenizes functionalities and 91.84 % precise POS tagging functionality. Various experiments have been performed to check validity of the system. In the following section, some experiments are stated with their respective results. Hindi text for experiments is taken from various domains like news, online data, Stories.

Experiment 1: Splitting

This experiment states system functionality of "split input Hindi text in individual sentences according to delimiter full stop or question mark." Consider the following example,

Input text to the system:

"नई दिल्ली। सिविल सेवा देने वाले अभ्यर्थियों को इस साल से आयु में दो साल का फायदा मिलेगा। उन्हें दो अतिरिक्त मौके परीक्षा देने के लिए मिलेंगे।

Output of the system:

1. नई दिल्ली।
2. सिविल सेवा देने वाले अभ्यर्थियों को इस साल से आयु में दो साल का फायदा मिलेगा।
3. उन्हें दो अतिरिक्त मौके परीक्षा देने के लिए मिलेंगे।

Experiment 2: Tokenization

This experiment states system functionality of "Tokenize input Hindi text in individual words according to space delimiter." Consider the following example,
Input Text to the system:

"दो साल का फायदा मिलेगा"

Output of the system:

दो, साल, का, फायदा, मिलेगा

Experiment 3: POS Tagging

This experiment shows system functionality of "Part-of-speech tagging of input Hindi sequence." Consider the following example,

Input Text to the system:

"सिविल सेवा देने वाले अभ्यर्थियों को इस साल से आयु में दो साल का फायदा मिलेगा।

Output of the system:

सिविल_NNC सेवा_NN देने_VNN वाले_PREP अभ्यर्थियों_NN को_PREP इस_PRP साल_NN से_PREP आयु_NN में_PREP दो_QFNUM साल_NN का_PREP फायदा_NN मिलेगा_VFM ।_PUNC

Evaluation

The system has validated on various data sets. We performed validation through holdout method. Evaluation measures of a system as Precision and Accuracy can be defined as [9],

$$Precision = \frac{\text{Number of correctly agged words}}{\text{Total number of tagged words}} \quad (1)$$

$$Accuray = \frac{\text{Number of correctly tagged words}}{\textit{Total number of words}} \quad (2)$$

The system yields 91.84 % of average precision and 85.45 % of average accuracy. To the best of our knowledge, achieved precisions through this system are highest with good accuracies while having smallest database of already tagged corpus. In past Garg et al. [5] reported 85.47 % of precision with training data of around 18,000 words. Dalal et al. [7] and Singh et al. [6] achieved 94.38 % and 93.45 % respective accuracies with around training data of 15,500.

Conclusion and Future Work

The presented POST system is designed with the help of rule-based approach. Corpus matching is applied while tagging known words. For unknown words tagging various Hindi grammar rules are applied. These rules increase precision as well as accuracy of the system. The system can split and tokenize input Hindi text successfully. Input Hindi text can be tagged by presented system with average precision of 91.84 %. In future we would increase the precision and accuracy of the implemented POS system by focusing on increasing the number of effective grammatical rules or by applying more hybrid techniques, instead of increasing the size of already tagged corpus. We would also provide some additional functionality with POS tagging.

Reference

1. Brill, E.: A simple rule-based part of speech tagger. In: Proceedings of the Third Conference on Applied Natural Language Processing ANLC '92, Stroudsburg, PA, USA, pp. 152–155 (1992)
2. Zin, K.K., Thein, N.L.: Part of speech tagging for Myanmar using hidden markov model. In: Proceedings of International Conference on the Current Trends in Information Technology (CTIT), Dubai, Dec 2009, pp. 1–6 (2009)
3. Bharati, A., Sharma, D.M., and Sangal, R.: AnnCorra: An Introduction (Vol. 14), Technical Report no: TR-LTRC (2001)
4. Mishra, N., Mishra, A.: Part of speech tagging for Hindi corpus. In: Proceedings of the International Conference on Communication Systems and Network Technologies (CSNT), Katra, Jammu, India, June 2011, pp. 554–558 (2011)
5. Garg, N., Goyal, V., Preet, S.: Rule based Hindi part of speech tagger. In: Proceedings of Coling, Mumbai, India (2012)
6. Singh, S., Gupta, K., Shrivastava, M., Bhattacharyya, P.: Morphological richness offsets resource poverty—an experience in building a POS tagger for Hindi. In: Proceedings of Coling, Sydney, Australia (2006)
7. Dalal, A., Nagaraj, K., Sawant, U., Shelke, S., Bhattacharyya, P.: Building feature rich POS tagger for morphologically rich languages: experiences in Hindi. In: Proceedings of ICON (2007)
8. A part of speech tagger for Indian languages (pos tagger) (2007)
9. Fayyad, U.M., Shapiro, G., Smyth, P., Uthurusamy, R.: Advances in Knowledge Discovery and Data Mining, American Association for Artificial Intelligence, Menlo Park, CA, USA (1996)