Part of Speech Taggers for Morphologically Rich Indian Languages: A Survey

Dinesh Kumar
Department of Information Technology
DAV Institute of Engineering & Technology
Jalandhar, Punjab, INDIA

Gurpreet Singh Josan
Department of Computer Engineering
Punjabi University Regional Campus
Talwandi Saboo, Punjab, INDIA

ABSTRACT

The problem of tagging in natural language processing is to find a way to tag every word in a text as a particular part of speech, e.g., proper pronoun. POS tagging is a very important preprocessing task for language processing activities. This paper reports about the Part of Speech (POS) taggers proposed for various Indian Languages like Hindi, Punjabi, Malayalam, Bengali and Telugu. Various part of speech tagging approaches like Hidden Markov Model (HMM), Support Vector Model (SVM), Rule based approaches, Maximum Entropy (ME) and Conditional Random Field (CRF) have been used for POS tagging. Accuracy is the prime factor in evaluating any POS tagger so the accuracy of every proposed tagger is also discussed in this paper.

Keywords

HMM, Tagging, Stochastic, Tagset, Finite State Automata, Suffix, Prefix, Support Vector Machines, Stemming, Maximum Entropy, Corpora, Tags, Morphology

1. INTRODUCTION

Part of Speech tagging is a process of marking the words in a text as corresponding to a particular part of speech, based on its definition, as well as its context [13]. POS tagging is a very important preprocessing task for language processing activities. This helps in doing deep parsing of text and in developing Information extraction systems, semantic processing etc. POS tagging for natural language texts have been developed using linguistic rule, stochastic models and a combination of both.

There are different classifications of POS tagging which are presented in following figures:

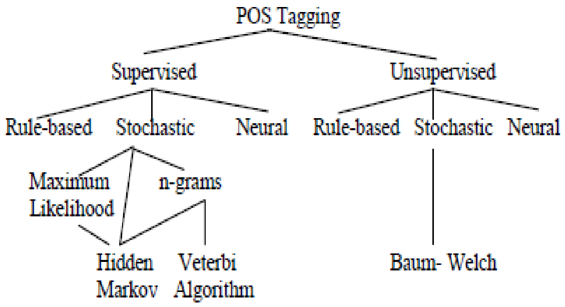


Figure 1: POS tagging Schemes

Supervised tagging method is based on pre-tagged corpora. It is a method of facilitating in the system of disambiguation or to learn the rules for tagging. *Unsupervised tagging* method on the other hand do not require pre-tagged corpus. The unsupervised POS Tagging models do not require a pre-tagged corpus. Instead, they use advanced computational techniques like the Baum-Welch algorithm to automatically induce tagsets, transformation rules, etc. Based on this information, they either calculate the probabilistic information needed by the stochastic taggers or induce the contextual rules needed by rule based systems or transformation based systems [13][14]

They are further two divided into two distinct approaches for POS Tagging-Rule based and Stochastic approaches [13]. Rule based approach uses a large database of hand-written disambiguation rules considering the morpheme ordering and contextual information. The Stochastic approach uses an unambiguously tagged text to estimate the probabilities to select the most likely sequence. For selecting the maximum likelihood probability the lexical generation probability and the n-gram probability are considered. The most common algorithm for implementing an n-gram approach is the Viterbi Algorithm which follows a Hidden Markov Model [13] [14].

2. POS TAGGING FOR INDIAN LANGUAGES

2.1 Malayalam

Malayalam is spoken primarily in Southern Coastal India by over 35 million speakers. Malayalam has its own distinct script, a syllabic alphabet consisting of independent consonant and vowel graphemes plus diacritics. Malayalam belongs to the Dravidian family of languages and is one of the four major languages of this family with a rich literary tradition. Morphologically Malayalam is richly inflected by the addition of suffixes with the root/stem word. Malayalam is a language registering a heavy amount of agglutination. The origin of Malayalam as a distinct language may be traced to the last quarter of 9th Century A.D. Malayalam has a special place in the classification of world languages. It is from Tamil that Malayalam was born. However, it is from the traditions of Sanskrit, the Indo-Aryan language, that Malayalam draws its rich diversity of words and compound alphabets (conjuncts). This dynamic synthesis of diversities has been achieved by no other Indian languages [30]

2.1.1 HMM based Tagging

A stochastic Hidden Markov Model (HMM) based part of speech tagger has been proposed for Malayalam. To perform parts of tagging speech using stochastic approach, an annotated corpus is needed. Due to unavailability of annotated corpus, a morphological analyzer was also developed to generate a tagged corpus from the training set [20]. The proposed architecture of the system is:

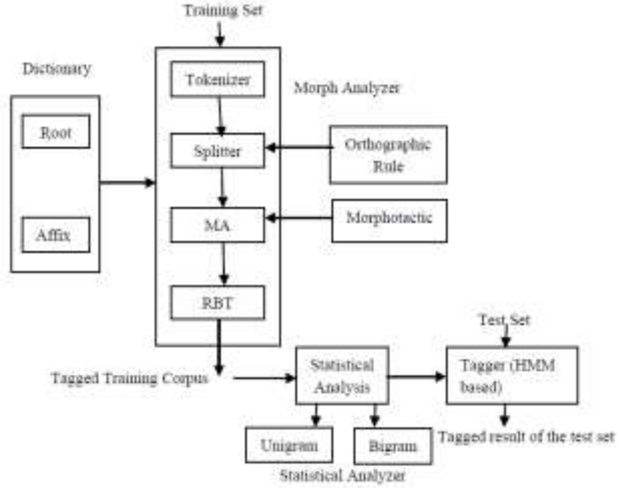


Figure 2: System Architecture [20]

The Morphological Analyzer accepts the input text which can have more than one sentence. On submitting the text, the text is transliterated to an intermediate representation and is stored as a file. This representation is used while traversing the Finite State Automata (FSA). Now each sentence is given to the Tokenizer. The token is checked with the dictionary to check if it is a valid word. If not, then the word (token) is given to the Splitter where the word is separated into root and affix based on the orthographic rules. After Identifying the Root, the analyzer searches the affix based on the morphotactics of the category of the root word. This is the morphologically Tagged result [20].

Rule based tagger was used to remove any ambiguity in the morphologically analyzed result. Special rules were written for specific cases, if any. By using the Morph Analyzer the tagged corpus is generated [20]. The statistical analyzer extracts unigram, bigram probabilities from the training corpus [14]. At the end of training phase in which a relevant statistical data was collected from the training corpus, the tagger is activated on the test corpus. To do tagging, HMM based taggers choose the tag sequence that maximizes the following formula:

P (word|tag) * P (tag | previous n tags)

And for finding the maximum probability viterbi algorithm [13] was used.

Malayalam language is a inflectionally rich in morphology [25], by adding suffixes with the root / stem word. Since words are formed by the suffix addition with root, most of the words can take the POS tag based on the root or stem. Hence in Malayalam the suffixes play major role in deciding the POS of the word. The tagset developed was based on Penn Treebank consisting of 18 tags [20].

2.1.1.1 Result Analysis

Test cases were used to test the system after training the system using the tagged corpus. For tagging the test case, both the lexical generation probability and the emission probability were used. The tagger was trained with using about 1,400 tokens. Authors claimed that the accuracy of the system can be increased by increasing the tokens. The POS Tagger developed gave an accuracy of about 90%. For performing statistical tagging, only 10 tag sequences were considered, and the result obtained from the Statistical Analyzer was very satisfactory as claimed by the authors. Almost 80% of the sequences generated automatically for the test case were found correct, when compared with the manually tagged result for those sentences [20].

2.1.2 SVM based tagging

Another tagger for Malayalam was proposed [2] which is based on machine learning approach with Support Vector Machine (SVM) [15]. There objective was to identify the ambiguities in Malayalam lexical items, and to develop a tag set appropriate for Malayalam. Finally, to built an efficient and accurate POS Tagger. The proposed tagset for Malayalam language has 29 tags where there are 5 tags for nouns, 1 tag for pronoun, 7 tags for verbs, 3 for punctuations, two for number, and 1 for each adjective, adverb, conjunction, echo, reduplication, intensifier, postposition, emphasize, determiners, complimentizer and question word. The proposed architecture for POS tagging was:

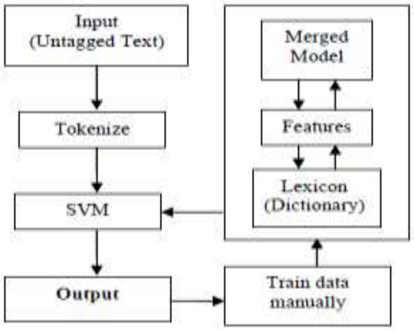


Figure 3: Architecture for POS tagging [2]

The POS tagging architecture consists of different modules which perform different functionalities to achieve better accuracy of POS tagger. They used SVM tool [15] for tokenization and the desired input in column format was given to this tool. Blank space is used as a column separator. The output of tokenize module is a corpus of untagged tokens so the corpus is manually tagged using the proposed tagset. In the initial phase, 20,000 words are tagged manually. The manually tagged corpus is trained using SVM tool [15]. This output of the tool is a dictionary with merged model and its lexicon. The remained pre-edited corpus is given to the SVM (SVMTagger, component of SVM tool) [15] for tagging in step by step. After tagging, the displayed output is checked manually and the tags are corrected properly. The proposed POS tagger has a tagged Malayalam corpus with size of 1, 80,000 tagged words [2].

2.1.2.1 Results Analysis

The performance of the POS tagger system in terms of accuracy is evaluated using SVMTeval. Initially, when the size of the lexicon is small the tagger achieves low accuracy. The following table shows the accuracy of POS tagger:

Table 1: Tagging accuracies [2]

No. of words in Lexicon	POS Tagger Accuracy
20,000	63 %

1,00,000	86 %
1,80,000	94 %

The tagger achieves 94 % accuracy when the size of lexicon was increased to 180,000 words

2.2 Bengali

Bengali, a member of the Indic group of Indo Iranian or Aryan branch of the Indo–European family of languages, originated from the eastern variety of the Magadhi Apabhramsa/Avahatta. The language has passed through two successive stages of development, namely the (a) Formative or old Bengali period, (b) Middle Bengali period. Presently Bangla is passing through its third stage of development, which is generally known as New or Modern Bengali period. [31] Bengali is a morphologically rich language. It is the seventh popular language in the world, second in India and the national language of Bangladesh [9].

In case of Bengali Language three taggers have been proposed. All the proposed taggers used different tagging approaches for doing POS tagging. Hidden Markov Model (HMM) and Maximum Entropy (ME) based stochastic taggers were proposed in the year 2007 [26]. Support vector machine based tagger was proposed in the year 2008 [9]. Both these tagging are explained in the following sections.

2.2.1 HMM & ME Based Tagging

Stochastic models (Cutting et al., [5]; Dermatas et al., [10]; Brants, [4]) have been widely used in POS tagging for simplicity and language independence of the models. Among stochastic models, bi-gram and tri-gram Hidden Markov Model (HMM) are quite popular. In this work supervised and semi-supervised bi-gram HMM & a ME based model was explored. The tagset used consists of 40 tags. The bi-gram assumption states that the POS-tag of a word depends on the current word and the POS tag of the previous word. An ME model estimates the probabilities based on the imposed constraints. Such constraints are derived from the training data, maintaining some relationship between features and outcomes. The most probable tag sequence for a given word sequence satisfies equation (1) and (2) respectively for HMM and ME model:

$$\mathbf{S} = \mathbf{argmax}_{\mathbf{t1}\dots\mathbf{tn}} \prod\nolimits_{\mathbf{i=1,n}} \mathbf{P}(\mathbf{wi}|\mathbf{ti})\mathbf{P}(\mathbf{ti}|\mathbf{ti}-\mathbf{1}) \text{ ---- (1)}$$

$$\mathbf{p}(\mathbf{t1} \dots \mathbf{tn}|\mathbf{wi} \dots \mathbf{wn}) = \prod\nolimits_{\mathbf{i=1,n}} \mathbf{p}(\mathbf{ti}|\mathbf{hi}) \text{ --- (2)}$$

Here, *hi* is the context for word *wi*. Since the basic bigram model of HMM as well as the equivalent ME models do not yield satisfactory accuracy, so the available resources like a morphological analyzer was used appropriately for better accuracy [26].

Three taggers have been implemented based on bigram HMM and ME model. The first tagger makes use of the supervised HMM model parameters and is named as **HMM-S,** the second tagger uses the semi supervised model parameters and is called **HMM-SS**. The third tagger is based on **ME** model and is used to find the most probable tag sequence for a given sequence of words. Morphological Analyzer was also used to further improve the accuracy of the tagger and integrated the morphological information with the model [13]. They assumed that the POS-tag of a word *w* can take values from the set TMA(*w*), where TMA(*w*) is computed by the Morphological Analyzer. The size of TMA(*w*) is much smaller than T. Thus, they have a restricted choice of tags as well as tag sequences for a given sentence. Since the correct tag *t* for *w* is always in TMA(*w*) (assuming that the morphological analyzer is complete), it is always possible to find out the correct tag sequence for a sentence even after applying the morphological restriction. Due to a much reduced set of possibilities, this model is expected to perform better for both the HMM (HMM-S and HMM-SS) and ME models even when only a small amount of labeled training text is available. They called these new models **HMM-S+MA**, **HMM-SS+ MA** and **ME+MA** [26].

To further improve the proposed models, the suffix information was also taken into consideration. Suffix information has been used during smoothing of emission probabilities for HMM models, whereas for ME models, suffix information is used as another type of feature [14]. The model with suffix information are denoted a **'+suf'** marker. Thus, They the new model are – **HMM-S+suf**, **HMMS+suf+MA**, **HMM-SS+suf** etc [26].

2.2.1.1 Experiments & Results

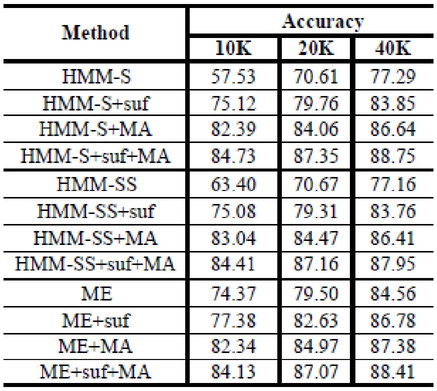
A total of 12 models were considered under different stochastic tagging schemes. To estimate the parameters for all the models the same training text has been used. The model parameters for supervised HMM and ME models are estimated from the annotated text corpus. For semi-supervised learning, the HMM learned through supervised training is considered as the initial model. Further, a larger unlabelled training data has been used to re-estimate the model parameters of the semi-supervised HMM. The experiments were conducted with three different sizes (10K, 20K and 40K words) of the training data to understand the relative performance of the models as we keep on increasing the size of the annotated data.

The training data consists of manually annotated 3625 sentences (approximately 40,000 words) for both supervised HMM and ME model. A fixed set of 11,000 unlabeled sentences (approximately 100,000 words) taken from CIIL (Central Institute of Indian Languages) corpus are used to re-estimate the model parameter during semi-supervised learning [26]. The corpus ambiguity (mean number of possible tags for each word) in the training text is 1.77 which is much larger compared to the European languages [6]

A set of randomly drawn 400 sentences (5000 words) have been used for testing all models. Out of these 14% words in the open testing text are unknown with respect to the training set, which is also a little higher compared to the European languages [6]

The results are obtained on the basis of final accuracies achieved by different models with the varying size of training data

Table 2: Tagging accuracies (in %) of different models with 10K, 20K and 40K training data [26]



Method	Accuracy		
	10K	20K	40K
HMM-S	57.53	70.61	77.29
HMM-S+suf	75.12	79.76	83.85
HMM-S+MA	82.39	84.06	86.64
HMM-S+suf+MA	84.73	87.35	88.75
HMM-SS	63.40	70.67	77.16
HMM-SS+suf	75.08	79.31	83.76
HMM-SS+MA	83.04	84.47	86.41
HMM-SS+suf+MA	84.41	87.16	87.95
ME	74.37	79.50	84.56
ME+suf	77.38	82.63	86.78
ME+MA	82.34	84.97	87.38
ME+suf+MA	84.13	87.07	88.41

The results show that the best performance is achieved for the supervised learning model along with suffix information and morphological restriction on the possible grammatical categories of a word. The use of MA in any of the models enhances the performance of the POS tagger significantly [26].

2.2.2 Support Vector Machine based tagging

Support vector machine is a new generation learning system based on recent advances in statistical learning theory. It gives excellent performance in the applications like text categorization, hand-written character recognition, natural language processing, etc. It has many advantages over conventional statistical learning algorithms. Simple HMMs do not work well when small amount of labeled data are used to estimate the model parameters. Incorporating diverse features in an HMM based tagger is difficult and complicates the smoothing typically used in such taggers. In contrast, a ME [23] or a CRF [18] or a SVM [17] can deal with the diverse and overlapping features more efficiently. A POS tagger has been proposed in [27] that has shown an accuracy of 93.45% for Hindi with a tagset of 23 POS tags.

SVMs have advantages over conventional statistical learning algorithms, such as Decision Tree, HMMs, ME from the following two aspects [9]:

1. SVMs have high generalization performance independent of dimension of feature vectors. Other algorithms require careful feature selection, which is usually optimized heuristically, to avoid over fitting.

2. SVMs can carry out their learning with all combinations of given features without increasing computational complexity by introducing the Kernel function. Conventional algorithms cannot handle these combinations efficiently.

In this work, SVM based approach was used for the task of POS tagging. To improve the accuracy of the POS tagger, a lexicon [7] and a CRF-based NER system [8] have been used, along with the variety of contextual and word level features. The SVM based POS tagger has been developed using a corpus 72,341 word forms tagged with the 26 POS tags, defined for the Indian languages. Out of 72,341 word forms, around 15K word forms have been selected as the development set and the rest, i.e., 57,341 word forms have been used as the training set of the SVM based tagger in order to find out the best set of features for POS tagging in Bengali.
The baseline model has been defined as the one where the POS tag probabilities depend only on the current word:

$$P(t1, t2 \dots, tn|w1, w2.., wn) = \prod_{i=1,n} P(ti|wi) \text{ --- (3)}$$

In this model, each word in the test data will be assigned the POS tag, which occurred most frequently for that word in the training data.

Features for part of speech (POS) tagging in Bengali have been identified based on the different possible combination of available word and tag context. The features also include prefix and suffix for all words. The term prefix/suffix is a sequence of first/last few characters of a word, which may not be a linguistically meaningful prefix/suffix. The use of prefix/suffix information works well for highly inflected languages like the Indian languages [9]. Numbers of experiments were conducted taking the different combinations from the set 'F' to identify the best-suited set of features for the POS tagging task. From the analysis, the following combinations were found to give the best result:

F={ $w_{i-2}w_{i-1}w_iw_{i+1}w_{i+2}$, |prefix|<=3, |suffix|<=3, Dynamic POS tags of the previous two words, NE tags of the current and the previous words, Lexicon feature, Symbol feature, Digit feature, Length feature, Inflection lists}.

2.2.2.1 Result analysis

A standard test set of 20K word forms has been used in order to report the evaluation results of the system. The POS tagger has demonstrated the overall accuracy of 86.84% for the test set by including the unknown word handling mechanisms. There are 23% words are unknown in the test set.

Table 3: Comparative evaluation results [9]

Baseline	Accuracy (in %)
HMM (with unknown word handling)	78.59
ME(with unknown word handling)	83.32
CRF(with unknown word handling)	85.61
SVM(with unknown word handling)	86.84

Results demonstrate the fact that the proposed SVM based POS tagger outperforms the least performing HMM based system by 8.24% in accuracy and the best performing CRF based system by 1.13% [9].

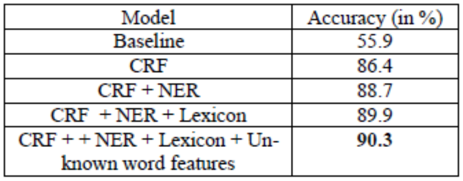
2.2.3CRF based tagging

Authors of [11] have developed Conditional Random Fields (CRF) based approach for the development of POS tagger for Bengali. Since, features selection plays a very important role in the CRF framework. The authors have identified the main features for POS tagging in Bengali based on the different possible combination of available word and tag context including prefix & suffix for all words.

2.2.3.1 Evaluation &Result analysis

The POS tagger was developed using a tagset of 26 POS tags. The system makes use of the different contextual information of the words along with the variety of features that are helpful in predicting the various POS classes. The POS tagger has been trained and tested with the 72,341 and 20K word forms, respectively. With lexicon, Named Entity Recognizer (NER) and unknown word features, the accuracy of the POS tagger improves significantly. The following results were obtained by the authors:

Table 4: Overall evaluation results [11]



Model	Accuracy (in %)
Baseline	55.9
CRF	86.4
CRF + NER	88.7
CRF + NER + Lexicon	89.9
CRF + + NER + Lexicon + Un-known word features	**90.3**

It was found from the results that CRF model with the consideration of NER, Lexicon and Unknown word features outperforms the other variation of CRF model. The authors have achieved an accuracy of 90.3% with CRF model [11].

2.3 Hindi

Hindi is the official language of India. About 182 million people speak Hindi as their native language and many others speak Hindi as a second language-some estimates say that around 350 million people speak Hindi. Hindi is a morphologically rich language. Different POS tagging approaches have been proposed for Hindi Language [1][29]. A tagging method for Hindi was proposed in [29] that overcome the troubles in accurate tagging due to the scarcity of large sized training corpora.

2.3.1 Morphology driven tagger

In this work, authors have proposed a new POS tagging methodology which can be used by languages having lack of resources. The methodology makes use of locally annotated modestly-sized corpora (15,562 words), exhaustive morphological analysis backed by high-coverage lexicon and a decision tree based learning algorithm (CN2) [29]. The proposed tagger uses the affix information stored in a word and assigns a POS tag using no contextual information by taking in consideration the previous and the next word in the Verb Group (VG) to correctly identify the main verb and the auxiliaries. Lexicon lookup was used for identifying the other POS categories. The architecture of the proposed tagger is given below

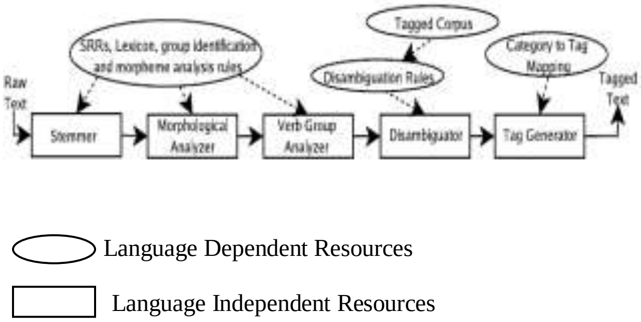


Figure 4: Tagger Architecture [29]

The process does not involve learning or disambiguation of any sort and is completely driven by hand-crafted morphology rules. The work progresses at two levels [29]:

1. **At Word Level**: To out all possible root-suffix pairs along with POS category label for a word, a stemmer is used in conjunction lexicon and Suffix Replacement Rules (SRRs). If the input word is not found in the lexicon and does not carry any inflectional suffix, than, *derivational morphology rules* are applied.

2. **At Group Level**: At this level a *Morphological Analyzer* (MA) uses the information encoded in the extracted suffix to add morphological information to the word.

2.3.1.1 Evaluation and Result analysis

The tests were performed on contiguous partitions of the corpora (15,562 words) that are 75% training set and 25% testing set. The results are obtained by performing a 4-fold cross validation over the corpora. The average accuracy of the learning based (LB) tagger after 4-fold cross validation is 93.45% [29].

2.3.2 Maximum Entropy Based Tagger

Maximum entropy (ME) principle states that the least biased model which considers all known information is the one which maximizes entropy. The ME technique builds a model which assumes nothing other than the imposed constraints. To build such a model, we define feature functions. A feature function is a boolean function which captures some aspect of the language which is relevant to the sequence labeling task [1]. The author presented the feature function for POS tagging is

$$\mathbf{fj(l|c)} = \begin{cases} \mathbf{1} & \textbf{if the current word is alphanumeric} \\ \mathbf{0} & \textbf{oherwise} \end{cases} \quad \text{-----}\ (4)$$

Where l is one of the possible labels and c is the context.

The authors have used following main feature functions for POS tagging:

1. Context based features
2. Word features
3. Dictionary features
4. Corpus-based features

2.3.2.1 Experiments and Results

Authors have conducted experiments for different split of training and test data.

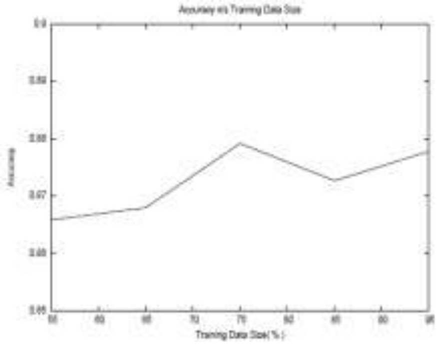


Figure 5: POS tagging accuracy [1]

From Figure 5, it is found that, POS tagging accuracy increases with increase in proportion of training data till it reaches 75%, after which there is a reduction in accuracy due to over fitting of the trained model to training corpus. Beyond a split of 85-15, increasing training corpus proportion increases the accuracy as the test corpus size becomes very small. This prompted us to use a 75-25 split for training and test data in our experiments. The results were averaged out across different runs, each time randomly picking training and test data.

The best POS tagging accuracy of the system in these runs was found to be 89.34% and the least accuracy was 87.04%. The average accuracy over 10 runs was 88.4% [1].

2.3.3 HMM Based Tagger

Hidden Markov Model (HMM) based tagger for Hindi was proposed by [21]. The authors attempted to utilize the morphological richness of the languages without resorting to complex and expensive analysis. The core idea of their approach was to explode the input in order to increase the length of the input and to reduce the number of unique types encountered during learning. This in turn increases the probability score of the correct choice while simultaneously decreasing the ambiguity of the choices at each stage. This also decreases data sparsity brought on by new morphological forms for known base words [21].But the problem with this approach was that it also loses all the information contained in the suffixes. As suffix contains good information of the category of the word so it is primary requirement to preserve the suffix and it is also used for further disambiguation.

The authors have used simple longest suffix removal technique for doing stemming. After this stemming and exploding of input, the exploded inflected tokens result in 2 tokens in the new corpus: the stem and the suffix. After the stemming the next steps is to assign appropriate tag to words. For doing this HMM based tagging approach was used. The accuracy of Simple HMM and Exploded Input HMM model was calculated.

2.3.3.1 Evaluation & Results

The corpus used for the training and testing purposes contains 66900 words. This data was 'exploded' resulting in a new corpus of 81751 tokens which was divided into 80% and 20% parts. The test set contains 13500 words which resulted in an exploded test set of 16000 tokens (stem and suffix tokens). The accuracy is calculated after imploding the output considering the assigned tag of the stem as the correct tag.

Table 5: Comparison between HMM & EI-HMM [21]

	HMM	EI-HMM CaTags	EI-HMM SuffTags
Accuracy	83.26	93.12	93.05

The data shows that the accuracy of Exploding Input HMM is much better than the Simple HMM based model

2.3.4 CRF Based Tagger

Conditional random field [16] is a probabilistic framework for labeling and segmenting data. It is a form of undirected graphical model that defines a single log-linear distribution over label sequences given a particular observation sequence. CRFs define conditional probability distributions P (**Y**|**X**) of label sequences given input sequences. Lafferty et al. defines the probability of a particular label sequence Y given observation sequence X to be a normalized product of potential functions each of the form

$$\mathbf{exp}\ (\sum \boldsymbol{\lambda j tj}(\boldsymbol{Yi}-\mathbf{1},\boldsymbol{Yi},\boldsymbol{X},\boldsymbol{i}) + \sum \boldsymbol{\mu k Sk}(\boldsymbol{Y},\boldsymbol{X},\boldsymbol{i}))\text{------ }(5)$$

where $\boldsymbol{tj}(\boldsymbol{Yi}-\mathbf{1},\boldsymbol{Yi},\boldsymbol{X},\boldsymbol{i})$ is a transition feature function of the entire observation sequence and the labels at positions *i* and *i-1* in the label sequence; $\boldsymbol{Sk}(\boldsymbol{Y},\boldsymbol{X},\boldsymbol{i})$ is a state feature function of the label at position I and the observation sequence; and λj and μk are parameters to be estimated from training data.

$$\mathbf{Fj(Y, X) = \Sigma}\ \boldsymbol{fj}\ \mathbf{(Yi\text{-}1, Yi, X, i)\ \text{------}\ (6)}$$

where each *fj*(Yi-1,Yi,X,i) is either a state function *s*(Y*i-1*,Y*i*,X,i) or a transition function *t*(Y*i-1*,Y*i*,X,i). This allows the probability of a label sequence **Y** given an observation sequence **X** to be written as

$$\mathrm{P}\ (\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\lambda}) = (1/\mathrm{Z}(\mathbf{X}))\ \exp\ (\Sigma\boldsymbol{\lambda}j\ \mathrm{Fj}(\boldsymbol{Y}, \boldsymbol{X}))\ \text{------}\ (7)$$

Z(**X**) is a normalization factor.

A Conditional Random Fields (CRF) [16] based tagger was proposed by authors of [3] [22]. Hindi Morph Analyzer was used for the training of POS tagger and to get the root-word and possible POS tag for every word in the corpus. Other information like suffixes, word length indicator and presence of special characters is added to the training data. CRF++ was used to train the data [3][22].

For POS tagging authors started training with a basic template using a very local context of words over a window of 4 words as features. Several experiments with varying the feature frequency and the number of iterations showed that the system performed best with fitting value 5 and feature freq=3.

The baseline performance of the system was 77.48%. [3]

The authors have found during error analysis that lots of errors were being made for different forms of a root-word. They have tried morph analyzer to overcome these errors and also achieved better results as compared to previous results

2.3.4.1 Evaluation & Results

The corpus used for the training and testing purposes contains 1,50,000 words. The accuracy achieved by the authors with CRF using CRF ++ was 82.67% [3] and 78.66 % [22] with training data of 21,470 words and test data of 4924 words.

2.4 Punjabi

Punjabi language is a member of the Indo-Aryan family of languages, also known as Indic languages. Other members of this family are Hindi, Bengali, Gujarati, and Marathi etc. Indo-Aryan languages form a subgroup of the Indo-Iranian group of languages, which in turn belongs to Indo-European family of languages. Punjabi is spoken in India, Pakistan, USA, Canada, England, and other countries with Punjabi immigrants. It is the official language of the state of Punjab in India. Punjabi is written in 'Gurmukhi' script in eastern Punjab (India), and in 'Shahmukhi' script in western Punjab (Pakistan) [32] [33].

2.4.1 Tagging Approach Used

A rule based part-of-speech tagging approach was used for Punjabi, which is further used in grammar checking system for Punjabi [28]. This is the only tagger available for Punjabi Language. A part-of-speech tagging scheme based entirely on the grammatical categories taking part in various kinds of agreement in Punjabi sentences has been proposed and applied successfully for the grammar checking of Punjabi [28]. This tagger uses hand-written linguistic rules to disambiguate the part-of-speech information, which is possible for a given word, based on the context information. A tagset for use in this part-of-speech tagger has also been devised to incorporate all the grammatical properties that will be helpful in the later stages of grammar checking based on these tags. This part-of-speech tagger can be used for rapid development of annotated corpora for Punjabi. The part-of-speech tagging design used is as follows:

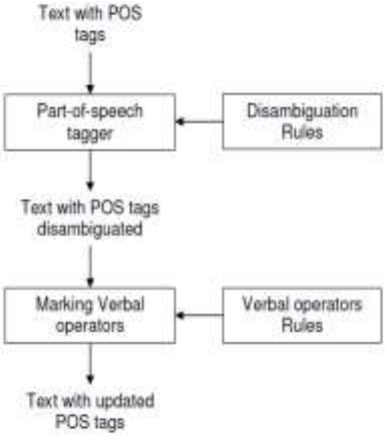


Figure 6: Part of Speech Tagging Design [12]

There are around 630 tags in this fine-grained tagset. This tagset includes all the tags for the various word classes, word specific tags, and tags for punctuations. During tagging process with proposed tagger, 503 tags out of proposed 630 tags were found in 8-million words corpus of Punjabi, which was collected from online sources. For disambiguation of POS tags rule-based approach was used. A database was designed to store the rules, which is used by rule based disambiguation approach. The texts with disambiguated POS tags are than passed for marking verbal operators. Four operator categories have been established to make the structure of verb phrase more understandable. During this step the verbal operators are marked based on their position in the verb phrase and the forms of their proceeding words [12]. A separate database was maintained for marking verbal operator.

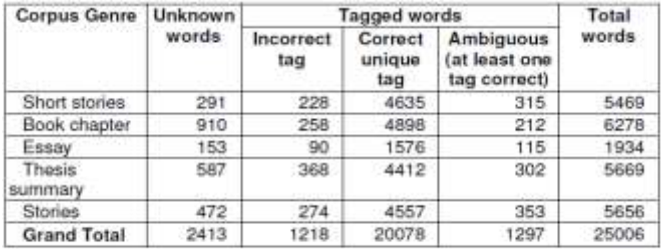
2.4.2 Results Analysis

The accuracy of any Part of Speech tagger is measured in terms of the accuracy i.e. the percentage of words which are accurately tagged by the tagger. This is defined as belows:

$$\textbf{Accuracy} = \frac{\textit{Total Number of words having correct tags}}{\textit{Total number of number tagged}} \text{ ------ (8)}$$

For evaluation of the proposed tagger, a corpus having texts from different genres were used. The outcome was manually evaluated to mark the correct and incorrect tag assignments. 25,006 words collected randomly from an 8 million corpus of Punjabi were manually evaluated and are grouped into five genres. Table 4 are based on the present state of our POS tagger having around 40 handwritten disambiguation rules and the tagset having around 630 tags. Total 503 tags of the possible 630 tags were found at least once in the 8 million words corpus of Punjabi

Table 5: Result of Part-of-speech tagging [12]



Corpus Genre	Unknown words	Tagged words			Total words
		Incorrect tag	Correct unique tag	Ambiguous (at least one tag correct)	
Short stories	291	228	4635	315	5469
Book chapter	910	258	4898	212	6278
Essay	153	90	1576	115	1934
Thesis summary	587	368	4412	302	5669
Stories	472	274	4557	353	5656
Grand Total	2413	1218	20078	1297	25006

Based on the data presented in table 4, the following different accuracy measures were calculated:

$$\textbf{Accuracy 1} = \frac{\textbf{Total words having correct tags}}{\textbf{Total words}} \text{ ------ (9)}$$

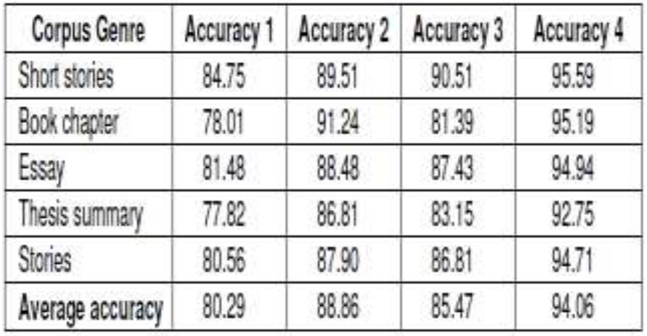
$$\textbf{Accuracy 2} = \frac{\textbf{Total words having correct tags}}{\textbf{Total words} - \textbf{Unknown words}} \text{ ------ (10)}$$

$$\textbf{Accuracy 3} = \frac{\textbf{Total words having at least one correct tag}}{\textbf{Total words}} \text{ ------ (11)}$$

$$\textbf{Accuracy 4} = \frac{\textbf{Total words having at least one correct tag}}{\textbf{Total words}-\textbf{Unknown words}} \text{ ------ (12)}$$

Accuracy achieved by the proposed tagger based on the Table 4 for these accuracy measures are:

Table 6: Accuracy of Part of Speech Tagger [12]



Corpus Genre	Accuracy 1	Accuracy 2	Accuracy 3	Accuracy 4
Short stories	84.75	89.51	90.51	95.59
Book chapter	78.01	91.24	81.39	95.19
Essay	81.48	88.48	87.43	94.94
Thesis summary	77.82	86.81	83.15	92.75
Stories	80.56	87.90	86.81	94.71
Average accuracy	80.29	88.86	85.47	94.06

From the results it is found that the accuracy of 80.29% including unknown words and 88.86% excluding unknown words was achieved by the proposed tagger.

2.5 Telugu

Telugu is classified as a Dravidian language with heavy Indo-Aryan influence. It is the official language of Andhra Pradesh. Telugu grammatical rule is deduced from a Sanskrit canon. Telugu uses many morphological processes to join words together, forming complex words [34].

2.5.1 Tagging Approach Used

For Telugu, three POS taggers have been proposed by using different POS tagging approaches ways viz., (1) Rule-based approach, (2) using Transformation based learning (TBL) approach of Erich Brill (3) using Maximum Entropy Model, a machine learning technique [24]. For transformation based learning and Maximum Entropy model an annotated corpus of 12000 words was constructed to train the taggers.

2.5.1.1 Rule based tagging

There are various functional modules which works together to give tagged Telugu text. The pre-edited Telugu text is given as input to Tokenizer which separates input text into separate sentences and each sentence to words for doing tokenization. These words are than given to MA for analysis.

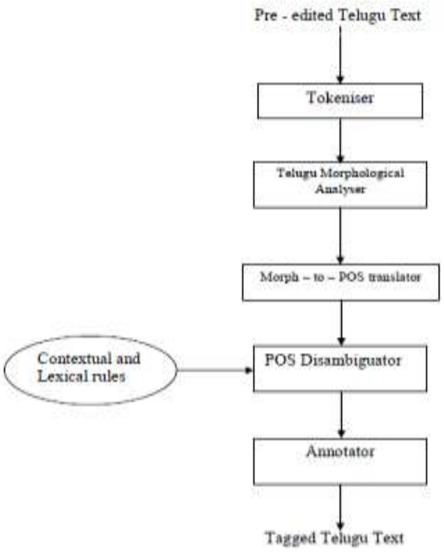

Pre - edited Telugu Text
Tokeniser
Telugu Morphological Analyser
Morph – to – POS translator
Contextual and Lexical rules
POS Disambiguator
Annotator
Tagged Telugu Text

Figure 7: Rule based POS tagger [24]

The Morph-to-POS translator than converts morphological analysis into their corresponding tags using pattern rules. The disambiguation problem is handled by the POS disambiguator which reduces the problem of POS ambiguity. This ambiguity is reduced by unigram and bigram rules. Finally, the tagged text is produced by Annotator.

2.5.1.2 Brill's and Maximum Entropy based approaches

Brill transformation rule based Learning (TBL) was also used to build a POS tagger for Telugu. For any language there are three phases of Brill tagger. These phases are: (i) Training Phase (ii) Verification Phase (iii) Testing Phase.

For Maximum Entropy based POS tagger, Maximum Entropy Modeling toolkit [MxEnTk] was used which is freely available on the Internet.

2.5.1.3 Results Analysis

The results obtained from the three proposed taggers are summarized in the following Table:

Table 7: Comparison of POS tagger Accuracy [24]

	Rule Based	Brill's Tagger	Maximum Entropy
Accuracy	98 %	90 %	81.78 %

The authors have used simple voting algorithm which gives one vote to each tagger output to improve the accuracy of POS tagging. The overall error rate reduces by 3% for machine learning tagger and 0.75% for Rule-base Telugu Tagger [24]

3. Conclusions

At last we conclude that Part of Speech tagging is the most important activity of any Natural Language based applications. The accuracy of any NLP tool is dependent on the accuracy of POS tagger. Different approaches have been used by authors for the development of part of speech tagger for Indian Languages. They are broadly categorized into Supervised and Unsupervised Models [19]. In case of Malayalam HMM based and SVM based Part of speech taggers have been used. The accuracy achieved by the proposed taggers is 90 % and 94 % respectively. The POS tagger proposed with machine learning approach i.e. SVM based performs better as compared to HMM based approach. For Bengali language, four POS taggers have been proposed. These taggers are based on Hidden Markov Model (HMM), Maximum Entropy (ME), Support Vector Machine (SVM) and Conditional Random Field (CRF) approaches. Different variations of HMM & ME based approaches were proposed by the authors. Supervised, Semi Supervised and Semi Supervised with Morphological Analyzer were proposed for both HMM & ME based approaches. To further improve the proposed model, suffix information was also taken into consideration by the authors for both HMM & ME based approaches. The accuracy achieved by Supervised HMM with MA and Suffix Information (HMM-S+Suf+MA), Semi supervised HMM with MA and Suffix Information (HMM-SS+Suf+MA) and ME with MA and Suffix Information is 88.75 %, 87.95 % and 88.41 % resp. On the other hand the accuracy achieved by SVM & CRF based model is 86.94 % and 90.3 %.

For Hindi, four taggers have been proposed based on HMM, ME, CRF and a morphology driven approach. The average accuracy as reported by different authors is 93.05%, 89.34%, 82.67% and 93.45% resp. A rule-based POS tagger was proposed for Punjabi. This is the only tagger available for Punjabi. The accuracy of 80.29% including unknown word and 88.86% excluding unknown words was achieved by the proposed tagger. In case of Telugu, rule based, Brill's tagger based and ME based approaches were used for the development of tagger. The accuracy achieved by all these taggers is 98%, 90%, 81.78% respectively. From this study, it is found that the Indian Languages are morphologically rich languages. So, morphological analyzer plays a vital role in developing a POS tagger. Further, machine learning based approaches gives somewhat better results as compared to other approaches. Very limited work has been done on Indian Languages for Part of speech tagging. So, different approaches can be used for the development of efficient tagger.

4. REFERENCES

[1] Aniket Dalal, Kumar Nagaraj, Uma Sawant and Sandeep Shelke, "Hindi Part-of-Speech Tagging and Chunking: A Maximum Entropy Approach", In Proceeding of the NLPAI Machine Learning Competition, 2006.

[2] Antony P.J, Santhanu P Mohan, Soman K.P,"SVM Based Part of Speech Tagger for Malayalam", IEEE International Conference on Recent Trends in Information, Telecommunication and Computing, pp. 339-341, 2010

[3] Agarwal Himashu, Amni Anirudh," Part of Speech Tagging and Chunking with Conditional Random Fields" in the proceedings of NLPAI Contest, 2006

[4] Brants, TnT – A statistical part-of-speech tagger. In Proc. of the 6th Applied NLP Conference, pp. 224-231, 2000

[5] Cutting, J. Kupiec, J. Pederson and P. Sibun, A practical part-of-speech tagger. In Proc. of the 3rd Conference on Applied NLP, pp. 133-140, 1992

[6] Dermatas and K. George, Automatic stochastic tagging of natural language texts. Computational Linguistics, 21(2): 137-163, 1995

[7] Ekbal, Asif, and S. Bandyopadhyay,"Lexicon Development and POS tagging using a Tagged Bengali News Corpus", In *Proc. of FLAIRS-2007*, Florida, 261-263, 2007

[8] Ekbal, Asif, Haque, R. and S. Bandyopadhyay, "Named Entity Recognition in Bengali: A Conditional Random Field Approach", In *Proc. of 3rd IJCNLP*, 51-55, 2008

[9] Ekbal, A. Bandyopadhyay, S., "Part of Speech Tagging in Bengali Using Support Vector Machine", ICIT- 08, IEEE International Conference on Information Technology, pp. 106-111, 2008

[10] E. Dermatas and K. George, Automatic stochastic tagging of Natural language texts, Computational Linguistics, 21(2): 137-163, 1995

[11] Ekbal Asif, et.al, "Bengali Part of Speech Tagging using Conditional Random Field" in Proceedings of the 7th International Symposium of Natural Language Processing (SNLP-2007), Pattaya, Thailand, 13-15 December 2007, pp.131-136

[12] Gurpreet Singh, "Development of Punjabi Grammar Checker, Phd. Dissertation, 2008

[13] Jurafsky D and Marting J H, Speech and Language Processing An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Pearson Education Series 2002

[14] James Allen, Natural Language Understanding, Benjamin/ Cummings Publishing Company, 1995

[15] Jes´us Gim´enez and Llu´ıs M`arquez., *SVMTtool:Technical manual v1.3*, August 2006

[16] John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conf. on Machine Learning*, pages 282–289.Morgan Kaufmann, San Francisco, CA.

[17] Kudo, T and Matsumoto, "Chunking with Support Vector Machines", In Proc. of NAACL, 192-199, 2001.

[18] Lafferty, J., McCallum, A., and Pereira, F., "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data", In Proc. of the 18th ICML'01, 282- 289, 2001.

[19] Linda Van Guilder (1995) Automated Part of Speech Tagging: A Brief Overview Handout for LING361, Fall 1995 Georgetown University

[20] Manju K., Soumya S., Sumam Mary Idicula, "Development of a POS Tagger for Malayalam - An Experience," artcom, pp.709-713, 2009 International Conference on Advances in Recent Technologies in Communication and Computing, 2009

[21] Manish Shrivastava and Pushpak Bhattacharyya, Hindi POS Tagger Using Naive Stemming: Harnessing Morphological Information Without Extensive Linguistic Knowledge, International Conference on NLP (ICON08), Pune, India, December, 2008 Also accessible from http://ltrc.iiit.ac.in/proceedings/ICON-2008

[22] PVS Avinesh, G Karthik, "Part-Of-Speech Tagging and Chunking using Conditional Random Fields and Transformation Based Learning" in the proceedings of NLPAI Contest, 2006

[23] Ratnaparkhi, A., "A Maximum Entropy Part of Speech Tagger", In Proc. of the EMNLP Conference, 133-142, 1996

[24] RamaSree, R.J, Kusuma Kumari, P., "Combining Pos Taggers For Improved Accuracy To Create Telugu Annotated Texts For Information Retrieval", 2007, Available at http://www.ulib.org/conference/2007/RamaSree.pdf

[25] Sumam Mary Idicula and Peter S David, A Morphological processor for Malayalam Language, South Asia Research, SAGE Publications, 2007

[26] Sandipan Dandapat, Sudeshna Sarkar, Anupam Basu," Automatic Part-of-Speech Tagging for Bengali: An Approach for Morphologically Rich Languages in a Poor Resource Scenario", Proceedings of the Association for Computational Linguistic, pp 221-224, 2007

[27] S. Singh , K. Gupta , M. Shrivastava and P. Bhattacharya, "Morphological Richness Offsets Resource Demand-Experiences in Constructing a POS Tagger for Hindi", In Proc. of COLING/ACL, 779-786, 2006

[28] Singh Mandeep, Lehal Gurpreet, and Sharma Shiv, 2008. "A Part-of-Speech Tagset for Grammar Checking of Punjabi", published in The Linguistic Journal, Vol 4, Issue 1, pp 6-22

[29] Smriti Singh, et.al," Morphological Richness Offsets Resource Demand- Experiences in Constructing a POS Tagger for Hindi", in the proceedings of COLING/ACL, pp. 779-786, 2006

[30] http://en.wikipedia.org/wiki/Malayalam

[31] http://www.bangla-online.info/PromotionalSite/Bangla Language/IntroductionOfBanglaLanguage.htm

[32] http://en.wikipedia.org/wiki/Punjabi_grammar

[33] http://en.wikipedia.org/wiki/Punjabi_language

[34] http://en.wikipedia.org/wiki/Telugu_language