NLP Assignment (2-3) Report : Developing a POS Tagger

- Santosh K (201150883), Vini Dixit (201205579), Lavanya Prahallad (201222631)

1. Introduction

1.1 What is POS tagging

Part of speech tagging (POS Tagging) is a process of assigning the grammatical categories or classes for every word in the given sentence according to the context. In other words, POS tagging is the identification of words as nouns, verbs, adjectives etc.. in their relationship with other words using NLP techniques. There are a number of approaches to implement part of speech tagger, i.e. Rule Based approach, Statistical approach and Hybrid approach. Rule-based tagger use linguistic rules to assign the correct tags to the words in the sentence or file. Statistical Part of Speech tagger is based on the probabilities of occurrences of words for a particular tag. HMM based approach fall under this category. Further HMM based approach is developed using either n-gram method or viterbi algorithm method. Hybrid based Part of Speech tagger is combination of Rule based approach and Statistical approach.

POS tagging can be used in various NLP applications like parsing, word sense disambiguation, information extraction, machine translation, question answering systems, text to speech systems, speech recognition, etc,.

In this assignment, we have developed the POS tagger for 4 Indian languages namely: Hindi, Kannada, Telugu and Marathi in both supervised and unsupervised approaches. Our tool accepts the raw text as input given in UTF-8 format and gives the tagged output. We implement the tagger using HMM model (Viterbi algorithm) and also using unsupervised techniques like word clustering. Stemming and handling of unseen words (words that are outside of trained corpus) is implemented to analyze the efficiency of the POS tagger. Though the developed POS tagger has been trained and tested on four languages mentioned above, the same can be ported to any new language by just replacing the training data..

1.2 Why do we need POS tagging

Automatic tagging of the text is highly efficient concept in natural language processing. Manual tagging though accurate is not scalable for large corpus where it contains millions of words. It is very expensive and time consuming. This brings us to the idea of developing methods and algorithms that rely on small set of training corpus (manually tagged dataset) which then, can run on a large corpus for tagging. The efficiency of such POS taggers depends on the amount and variability of the training data. POS tagger helps in creating the annotated corpus for large data and indeed help in major NLP applications and information processing tools.

Part of speech tagging for Indian languages is not considerably easy as compared to English. There have been many attempts and reasonably good research is going on in developing the POS taggers for Indian languages. Indian languages are morphologically rich, highly agglutinative and complex languages. Taggers have been developed using linguistics rules coupled with the approaches like HMM based, transformation rule based are used.

1.3 Issues in POS tagging

Part of speech tagging is not often a simple process as it has to deal with word sense disambiguation, which is identifying the right and intended meaning of the word used in the sentence to its context. For example, In kannada language there are few ambiguous words as given below in (a) and (b).

(a) ಮಗು, ನಿಮ್ಮ ಅಮ್ಮನ ಕರಿ/VM
boy, please call your mother

(b) ಈ ಕರಿ/NN ತುಂಬಾ ದೊಡ್ಡದಾಗಿ ಇದೆ
this elephant is gigantic

"kari (ಕರಿ)" is used in two ways in kannada, one is as *verb* (to call) and other as *noun* (elephant). POS tagging turns out to be inaccurate as per the context and usage in the sentence, if this issue is not addressed well. We have to use the abundant knowledge of the grammar of the language for which the POS tagger is developed as the rules of grammar is different for each language though some of them are similar in word order and structure.
Eg:

ಹೆಮ್ಮೆ	JJ	NN
proud	JJ	NN

The word ಹೆಮ್ಮೆ is a NN (noun), but based on the context the same word can be tagged as an JJ (adjective). The following example highlights different possibilities of POS tag for a same word in different contexts.
context 1:
(kannada): nanna putrana nodidare nanage ***hemme***
(translation): I feel proud of my son - where its is an example of noun
context 2:
(kannada): nannu obba ***hemme***ya yande
(translation): I am a proud father - where it is an example of adjective

Such cases can be handled using the contextual information which in turn depends on the size and variability of the training corpus.

Example in Telugu

a) ఈ కొట్టు/NN లో సరుకులు నాణ్యమైనవిగా ఉంటాయి
Groceries in this store are maintained quality

(b) తోటలో పామును కొట్టు /VB
Hit the snake

“kottu(కొట్టు)” is used in two ways in Telugu, one is as *verb* (to hit) and other as *noun* store/shop). For example, from the given dataset Kannada language has a of 2124 unique words, where in 65 words have more than one tag.

Example in Hindi:

a) सोना/NN और् चांदि बहुत् मेहंगा है
Gold and Silver are very costly

(b) मुझे सोना/VB है
I want to sleep

The tagset should be exhaustive. The tags should be designed such that it covers all possibilities of the variations in the grammars of the language. Predominantly all Indian languages have reduplication of some words when spoken. For example, in Hindi ***ghar ghar ko bata diya*** is phrase where in the words “ghar ghar” together deliver a different meaning. Tagging of such words is resolved coining new tag RDP. Other issue is handling *echo* words, which are common in most of the Indian languages. Example of echo words in Hindi is ***pyaar vyaar***, where the second word (vyaar) does not have any meaning in the dictionary but occurs in real world scenarios. ECH is the tag used for such words.
There is another category of recognizing and tagging unknown words or foreign words. The tag meant for that purpose is UNK. Right approach should be followed to use algorithm to get high accuracy of the pos tags to the words. The corpus should be linguistically correct and grammatically sound. An issue which always comes up while deciding tags for the annotation task is whether the tags should capture 'fine grained' linguistic knowledge or keep it 'coarse'. Telugu is morphologically rich language as one word contains lot of information in that. ‘raamuda’? It is a noun but it is a NNP and question particle is involved it in. A tagset should be able to handle these issues at POS tagging level or it can be left to the morphological analyses.

1.4 Literature Survey

Though considerable research is going on in developing POS taggers for all Indian languages, till date only taggers are developed for Hindi, Tamil, Telugu, Kannada, Malayalam, Bengali, Punjabi, Marathi and Gujarati.
A number of POS taggers were developed in Hindi language using different approaches. In the year 2006, three different POS tagger systems were proposed based on handwritten rules (Smriti Singh) [1], morphology rules and analyzer (Agarwal Himanshu and Amni Anirudh) and maximum entropy approaches respectively (Aniket Dalal, Kumar Nagaraj, Uma Sawant and Sandeep Shelke). There are two attempts for POS tagger developments in 2008, both are based on HMM approaches and proposed by Manish Shrivastava and Pushpak Bhattacharyya [2]. Nidhi Mishra and Amit Mishra proposed a Part of Speech Tagging for Hindi Corpus in 2011 [3]. In an another attempt, a POS tagger algorithm for Hindi was proposed by Pradipta Ranjan Ray, Harish V., Sudeshna Sarkar and Anupam Basu.[4]

A substantial amount of work has already done in POS tagger developments for Bengali language using different approaches. In the year 2007, two stochastic based taggers (HMM and Maximum Entropy) were proposed by Sandipan Dandapat, Sudeshna Sarkar and Anupam Basu. Also Ekbal Asif developed a POS tagger for Bengali language using Conditional Random Fields (CRF) [5]. An Unsupervised Parts-of-Speech Tagger for the Bangla language was proposed by Hammad Ali in 2010. Debasri Chakrabarti of CDAC Pune proposed a Layered Parts of Speech Tagging for Bengali in 2011.[6]

There is only one publically available attempt proposed in POS tagger for Punjabi language using rule based approach by Singh Mandeep, Lehal Gurpreet, and Sharma Shiv, in 2008 [7]. Same is true for Gujarati where the work is published by Patel et. al. as proposed Part-Of-Speech Tagging for Gujarati Using Conditional Random Fields. Navanath Sarkar built a HMM based model using Viterbi algorithm for Assamese [8].

Some noticeable attempts were done in Dravidian languages like Tamil(CIIL, LTRC, Vasu Ranganathan, M.Selvan et al) [9] 10]. They used rule based approach, statistical methods.There are three noticeable POS tagger developments in Telugu, based on Rule-based, Transformation based learning and Maximum Entropy based approaches.The existing Telugu POS tagger accuracy was also improved by a voting algorithm by Rama Sree, R.J. and Kusuma Kumari P in 2007 [11-15].For Malayalam, HMM based tagger (Manju K., Soumya S. and Sumam Mary Idicula 2009), and SVM algorithms for malayalam and kannada are developed and is based on machine learning approach by Antony P.J, Santhanu P Mohan and Dr. Soman K.P [16-19]. Above all, LTRC, IIIT Hyderabad has done considerable amount of work in developing POS taggers for Indian languages using rule based, statistical approaches.

2. Dataset

The corpus contains 1000 sentences each, in four languages namely, Telugu, Kannada, Hindi and Marathi. All the 1000 sentences are used for training purpose and we have taken text from web for the testing. The data was not previously tagged.
Assumption: We cleaned the corpus by removing the punctuation marks, unknown characters, symbols etc..

3. Approach

The goal of this assignment is to build the POS tagger for the above mentioned languages in both supervised and unsupervised techniques. The unsupervised model does not require previously annotated data, instead this technique used algorithms to automatically tag the corpus and write the rules to develop a decently working POS tagger. This approach is very much useful to develop taggers for unknown languages and even when the grammar of a language is not known. Contrastly, the supervised technique needs a previously annotated corpus and this corpus is used for training to learn the word frequencies, tag frequencies and subsequently generate tags for any input data.

3.1. Unique words & Tagging

As explained in section 2, the corpus contains 1000 sentences from each language. To implement the supervised technique, we need to have the annotated data for training the corpus. We embraced the approach of taking the unique words from each language and we have tagged each word as nouns, pronouns, verbs, adjectives, adverbs or any other appropriate tags by using the LTRC shallow parser. We have removed all the punctuation marks and other unknown symbols for this assignment. After tagging each word, we have put these tags back in to the sentences and manually corrected each sentence for all 1000 to fit the tag as per the context of the sentence. Below sample in Telugu shows the words and their tags from (c) to (g) and (h) shows the sample sentence with tags in the format of (word_tag) which forms the training data.

(c) రాష్ట్రంలో NN

(d) ఒకే JJ

(e) కొనసాగాడు VM

(f) ఆయన్ను PRP

(g) కాకుండా NEG

(h) ఆ_DEM తరువాత_RB హైదరాబాదుకు_NNP బదిలీ_NN అయ్యాడు_VM

Similar procedure is used to tag the corpus for all the four languages. Statistics are as shown below:

Language	No. of Unique words
Telugu	2312
Kannada	2124
Marathi	2095
Hindi	2140

Table 1: Language Vs Unique words

Handling Unknown words

Assumption: We had assumed to take the unknown words with the tag of “NNP” because taking this has highest frequency in the corpus and also its highest probable that a new word occurring should be a proper noun like any organization name etc.

3.2. Tagset

We have used common tagset to tag all the four languages and also to scale up to any new

language with just the raw corpus. To achieve this goal, we have used LTRC tagset to tag the words in the data. Below are the list of the tags that are used to tag commonly to tag each word in the data for the all four languages.

POS Tag Set for Indian Languages (Nov 2006, IIIT Hyderabad)

Sl No.	Category	Tag name	Example
1.1	Noun	NN	
1.2	NLoc	NST	
2.	Proper Noun	NNP	
3.1	Pronoun	PRP	
3.2	Demonstrative	DEM	
4	Verb-finite	VM	
5	Verb Aux	VAUX	
6	Adjective	JJ	
7	Adverb	RB	*Only manner adverb
8	Post position	PSP	
9	Particles	RP	bhI, to, hI, jI, hA.N, na,
10	Conjuncts	CC	bole (Bangla)
11	Question Words	WQ	
12.1	Quantifiers	QF	bahut, tho.DA, kam (Hindi)
12.2	Cardinal	QC	
12.3	Ordinal	QO	
12.4	Classifier	CL	
13	Intensifier	INTF	
14	Interjection	INJ	
15	Negation	NEG	
16	Quotative	UT	ani (Telugu), endru (Tamil), bole/mAne (Bangla), mhaNaje (Marathi), mAne (Hindi)
17	Sym	SYM	
18	Compounds	*C	
19	Reduplicative	RDP	
20	Echo	ECH	
21	Unknown	UNK	

Table 2: LTRC Tagset

3.3. Hidden Markov Models (HMM): Viterbi Algorithm

POS tagger based on HMM assigns the best tag to the sequence of words. Generally the most probable tag is used to assign to the word implementing Viterbi algorithm. A hidden markov model helps us handling the hidden events like parts of speech tags for the words given for the input sentence. For a given sentence or word sequence, HMM taggers choose the tag sequence that maximizes the following formula given by:

$$P(word|tag) \times P(tag \mid previous\ n\ tags)$$

HMM taggers generally choose a tag sequence for a whole sentence rather than for a single word based on how the word is used in the context. The algorithm chooses the tag t_I for word w_i that is most probable given the previous tag t_{i-1} and the current word w_i:

$$t_i = argmax_j P(t_j|t_{i-1}, w_i)$$

We can restate the above Equation to give the basic HMM equation for a single tag by using markov assumptions as follows:

$$t_i = argmax_j P(t_j|t_{i-1})P(w_i|t_j)$$

The parameters of a hidden Markov model are of two types, *transition probabilities* and *emission probabilities* (also known as *output probabilities*). The probability of a tag sequence given a word sequence is given by the product of emission and transition probabilities. Viterbi algorithm is mainly of two steps: Forward step, calculate the best path to a node and Find the path to each node with the lowest negative log probability. Second is Backward step, reproduce the path by backtracing the highest probable tag.

$$P(t|w) \propto \prod_{i=1}^{N} P(w_i|t_i) \cdot P(t_i|t_{i-1})$$

3.3.1. Emission Matrix

Based on the previously tagged text in the training corpus, the maximum likelihood estimate of the word and its tag is calculated from the total number of tags.

$P(w_i \mid t_j) = C(t_j, w_i)/C(t_j)$

where $C(t_i, w_i)$ is the co-occurrences of word w_i and tag t_j

and $C(t_j)$ is total number of occurrences of tag t_j

Emission probability matrix is a wXt matrix, where w is the word taken in x axis and t is for tag taken in y axis. The unique words are taken for the x axis and the total types of tags that are assigned to the words are taken as y axis. The simplest way to calculate the emission probability matrix is to assign a count of one to every entry to unique word of the corpus and tag occurring together or in the sequence. This number is divided by the total number of times that the tag has appeared in the distribution. This gives to the calculation of C(w|t). Similar procedure is applied to find out the emission probabilities for the unique words and tags for all the four languages. The only difference in calculating Hindi and Marathi matrix is that the unique words are brought out from the corpus after performing the stemming procedure, which was not used at this level for Telugu and Kannada. Below shows the emission

3.3.2. Transition Matrix

Based on the training corpus of previously tagged text, the maximum likelihood estimate can be counted from the counts of the observed tags.

$P(t_i \mid t_{i-1}) = C(t_{i-1}, t_i)/C(t_{i-1})$

where $C(t_{i-1}, t_i)$ is the number of times tag t_{i-1} is followed by tag t_i

and $C(t_{i-1})$ is total number of occurrences of tag t_{i-1}

Emission probability matrix is a tag vs tag matrix, where tags of the unique words are taken in x axis and total types of tags are taken in y axis. The simplest way to calculate the transition probability matrix is to assign a count of one to every entry to tag of the word and type of tag occurring together or in the sequence. This number is divided by the total number of times that the type of tag has appeared in the distribution. This gives to the calculation of C(ti|ti-1).

3.4 Stemming

Two types of stemmers are used, one is a based on linguistic rules and the other is based on statistics. The rule based stemmer is similar to porter stemmer where in common suffixes are manually identified. The statistical approach is based on the frequency of occurring of suffixes in the training corpus. Rule based stemmer is applied for Hindi and Marathi, whereas statistical stemmer is applied for Telugu and Kannada during POS tagging.

4. Clustering

As explained in section 2, the corpus contains 1000 sentences from each language. To implement the un-supervised technique, we need not have the annotated data for training the corpus.

4.1. Unique words

To implement the unsupervised technique, as the first step we calculated the unique words for each of 1000 sentences in each language corpus.

Language	No. of Unique words
Telugu	2312
Kannada	2124
Marathi	2095
Hindi	2140

Table 3: Language Vs Unique words for unsupervised

4.2. Approach

4.2.1. Unsupervised method of POS tagging

Unsupervised technique of tagging a text with appropriate POS tags is used for the languages where its not labeled and typically to a foreign language. In this method there is no need of the training data or the rules to tag the word in a sentence of the corpus. In clustering approach we make use of the distributional properties and co-occurrence patterns of the text. The key characteristics to be considered here are how the context vectors are defined, size of the context vectors, how many clusters to consider and how the word classes are induced on the clusters. Word clusters are usually induced based on a distributional-similarity criteria: words are clustered based on the words that tend to occur before or after them that occur in similar context and behave in similar manner.

To implement the unsupervised method of POS tagging, as the first step unique words and their frequencies are calculated for the given corpus in all the four languages. Let the number of unique words be m. In each language, among the unique words and the frequencies, top t $(t = 50)$ words are ignored as they are usually stop words or function words. Next n, $(n = 200)$

words are taken as feature words.

The dimension of the feature vector is twice the number of the feature words. Every unique word in corpus is represented in terms of the $2n\,(2n = 400)$ dimensional feature vector, where each dimension is a word from the feature words. The feature vector is updated based on the frequency of the co-occurrence of the unique word with every feature word. The left co-occurrence and right co-occurrence values are updated independently, hence the dimension of the feature vector is twice that of the number of feature words. The frequency of co-occurrence depends on the width of the context $w, (w = 2)$. The ensemble of all the feature vectors is our feature space, where each word in the corpus is represented by a vector on $2n$ dimensions. The feature space thus formed is sparse and is non-negative as the co-occurrences cannot be negative.

For clustering the words similarity between the vectors is computed in two different ways.

1. Cosine similarity
2. k-means clustering

The cosine similarity is computed between each pair of word vectors. The cosine similarity between any two vectors a, b is given by:

$$cos\,\theta = \frac{\{a.b\}}{|a|.|b|}$$

A similarity matrix of dimension $m \times m$ is formed with diagonal elements being 1, representing the similarity between the self vector. Clusters are formed by taking the high similar vectors for each cluster.

The k-means clustering cannot be applied directly as the feature space is sparse and the magnitude of each vector varies a lot. We wanted to use the nature of cosine similarity (angular distance between vectors, rather than euclidean distance) in k-means. So each vector in $2n$ dimension is unit length normalized, which means all the vectors will lie on the unit hypersphere. Now, the result of computing cosine similarity will be same as computing euclidean distance between any two vectors. The clusters formed are imbalanced, i.e., one cluster has majority of the data points, whereas the remaining have only few data points. So, mean subtraction is done across each dimension as a pre-processing normalization step, followed by variance normalization. This resulted in clusters that are decently distributed the data points. Further more, k-means is applied on huge clusters to get smaller clusters.

The following list presents the top 10 words in each cluster. Smaller clusters (data points < 10) are ignored. For larger clusters, top 10 words that are closer to centroid are reported below. Words that are closer to centroid doesn't mean that the words are similar. It means that these words occur more number of times in the context of feature words.

Results for Test Data Sets

Hindi :
1. हिंदी_NN के_PSP पहले_JJ प्रगतिशील_JJ लेखक_NN थे_VAUX प्रेमचंद_NNP
2. काका_NNP साहब_NNP की_PSP मातृभाषा_NN मराठी_NN थी_VAUX
3. यह_DEM लेख_NN सूरा_NNP के_PSP बारे_PSP में_PRP है_VAUX
4. बाबा_NN के_PSP नाम_NN हैं_VAUX कई_QF सम्मान_NN
5. कालिदास_NNP को_PSP लोकप्रिय_JJ बनाने_VM में_PRP दक्षिण_JJ भारतीय_JJ भाषाओं_NN के_PSP फिल्मों_NN का_PSP भी_RP काफी_QF योगदान_NN है_VAUX

error% = 2*100/41 = 4.87% (2 words out of 41 words are not upto the expectation of tags)
accuracy% = 96% (for above sample of 5 sentences)

Observation :

Viterbi rules are working good with Hindi except for few ambiguous words. For example, में is IN but its treating as PRP.
Also stemming is working well here.

Marathi :
1. महरषी_NNP यांनी_PRP केलेले_VM योगदान_NN पाहा_VM
2. शरावण_NNP आणि_NNP भादरपद_NNP या_NNP महिनयात_NNP शरद_NN ऋतू_NN असतो_VAUX
3. मराठी_NN माणसे_NN किंवा_CC मराठे_NNP हे_NNP महाराषटरीय_NNP महणूनही_NNP ओळखले_VM
4. आणि_NNP लगन_NNP झालयावर_NNP परथम_NNP ती_PRP मला_PRP या_DEM भागात_NN फार_INTF कमी_JJ लोक_NNP राहतात्_NNP
5. राषटरीय_NNP परेम_NNP दिन_NN चेक_NNP परजासतताक्_NNP काही_NNP संदरभ_NNP मिळाले_NN तर_CC पाहतो_NN

error% = 6*100/60 = 10% (6 words out of 60 words are not upto the expectation of tags)
accuracy% = 90% (for above sample of 5 sentences)

Observation :

This language is not working that correctly as Hindi with viterbi rules. As it has lots of variations for a particular root word to various forms. And as we are using stemming, those meanings are changing and hence its final tag.

Telugu
1. వరి_NN మరియు_CC ప్రతి_NN ఇక్కడ_NST పండించే_VM ప్రధాన_JJ పంటలు_NN

2. తరువాత_RB ఎన్నో_QF మంచి_JJ చిత్రాలకు_NN కథ_NN మాటలు_NN పాటలు_NNS రాశారు_VM
3. ఒకరోజు_NN ఇంద్రుడు_NNP సభ_NN తీర్చి_VM దేవతల_NN గురువు_NN అయిన_VM
4. బృహస్పతి_NNP అక్కడకు_NST వస్తాడు_VM
5. ఇక్కడ_NST సుబ్రహ్మణ్య_NNP స్వామి_NN ఆలయము_NN కలదు_VM
6. పెద్దేముల్_NN ఆంధ్ర_NNP ప్రదేశ్_NN రాష్ట్రములోని_NN రంగారెడ్డి_NNP జిల్లాకు_NN చెందిన_VM
7. ఒక_JJ మండలము_NN
8. చిత్తూరు_NNP జిల్లా_NN ఆంధ్రప్రదేశ్_NNP జిల్లా_NN

Error % : 0% (0 out of above 8 sentences are incorrect)
Accuracy% : 100% (for above sample data)

Kannada :
1. ದಕ್ಷಿಣ_NN ಕನ್ನಡ_NN ದ_NN ಒಂದು_QC ತಾಲೂಕು_NN
2. ನಂತರ_RB ಈ_NNP ದೇವಾಲಯದ_NNP ಪುನರ್_NNP ನಿರ್ಮಾಣವಾಯಿತು_NNP
3. ಪ್ರಯೋಗ_NN ಟೆಂಪ್ಲೇಟುಗಳು_NN ಪದವನ್ನು_NN ಬೇರೆ_NN ಲೇಖನಗಳಲ್ಲಿ_NN ಹುಡುಕಿ_VM
4. ಕನ್ನಡ_NN ಚಿತ್ರರಂಗದ_NN ಪ್ರಸಿದ್ಧ_JJ ಹಿನ್ನೆಲೆ_NN ಗಾಯಕಿ_NN
5. ಅವರು_PRP ತಮ್ಮ_NN ಪದವಿಯನ್ನು_NNP ಬಾಂಬೆ_NNP ವಿಶ್ವವಿದ್ಯಾಲಯದಲ್ಲಿ_NN ಪಡೆದರು_VM

Error% : 5*100 / 30 = 16.6% (5 out of 30 words are correct)
Accuracy% : 84% (for above sample data)

Observation :

This language has more error rate as the algorithm developed is not handling the varying word order of the languages. We had assumed to take the unknown words with the tag of “NNP” and taking this has highest frequency in the corpus. Based on the context the tag that come with the context NNP is sometimes being incorrect.

Telugu clusters:

రంగారావు, ప్రాంతానికి, సంవత్సరంలో, జాబితాలో, తెలుగువారి, భారతదేశంలో, నాగిరెడ్డి, రఘు, వీటి, వహించిన,
---
ఉన్నత, వున్నవి, ఎనిమిది, వరకూ, బహుశా, చెయ్యాలి, పూర్వం, అత్యధిక, ఉండొచ్చు, పౌర్ణమి,
---
హిందీ, వాడుకలో, పేరే, సాంప్రదాయం, పేర్లూ, దేశంలో, చిత్రాలకు, రాష్ట్ర, రాష్ట్రానికి, మహారాష్ట్ర,
---
ద్రవిడ, గ్రీకు, ప్రధానమైన, మతాలకు, దాని, పాఠం, నీ, అధికార, వ్యవసాయము, వాళ్ళకు,
---

స్టేషన్, ఏదో, వరలక్ష్మి, కన్యాశుల్కం, జ్ఞానపీఠ, సరే, విగ్రహం, కనిపించడం, వ్యవసాయ, పాత్ర,
---
పాఠం, భారతీయ, పేజి, సినిమాలలో, కొద్ది, స్వాగతం, లింకులు, సందేహాలుంటే, గ్రామాలకు, వస్తుంది,
---
నాలుగవ, ఐదవ, పదవ, క్రితం, రాసిన, పౌరాణిక, నారాయణరావు, మొదటి, ఊరిలో, గ్రామంలో,
---
వరకే, పుట్టిన, పోయినా, లేని, యీ, ప్రక్క, మా, కన్నడ, నిర్మాత, రెడ్డి,
---
మాత్రం, నెలలు, కృతజ్ఞతలు, ఉందని, తప్పు, కథ, ముందు, పడి, వీలుగా, అందంగా,
---
అభిప్రాయాలు, నిర్వాహకులకు, జవాబు, ఉన్నపుడు, ప్రతిపాదన, ఇవి, సిద్ధాంతము, అవగాహన, సభ్యులకు, సారాంశం,
---
నగర్, గుంటూరు, అనంతపురం, ఆదిలాబాదు, మెదక్, వరంగల్, కి, వికీపీడియాకు, ప్రజలకు, బళ్ళారి,
---
ఆచంట, జంతువు, మూసల, వీరి, వ్యాపారానికి, కానీ, తిరుపతి, దేశం, ఊరు, ద్వారా,
---

Hindi clusters:

थी, विकि, मंगल, जल्दी, युवा, हूं, मनोवैज्ञानिक, द्विज, लगाव, प्राकृतिक,
---
राज, उर्दू, ब्रज, उच्च, शृंगार, बांग्ला, प्रोग्रामिंग, वाली, हर, कुमाऊँ,
---
किताब, रचनाओं, नियम, द्वितीय, आपने, वक्त, वर्तमान, संदर्भ, विद्वान, पत्र,
---
चलता, दीप, रह, प्रयास, मिल, प्रारंभ, स्तर, दे, उचित, जरूरी,

गए, रहे, आये, आते, संबद्ध, बेटे, बोलते, प्रवर्तक, मौजूद, भाई,
---
वोल्डेमॉर्ट, जायें, रॉन, किरदार, ग्रंथ, सभ्यता, पुराण, यानि, इसकी, पंजाब,
---
विकास, जिसका, तथ्य, भाषापरिवार, इंडोनेशिया, छन्द, अंचल, विश्व, एशिया, वाले,
---
सनातन, पारसी, सिक्ख, बौद्ध, इस्लाम, यहूदी, रोमन, सच्चे, गाया, इकाई,
---
कारण, तक, नही, चर्चा, शब्दों, उर्दू, होती, समस्याओं, सुनहु, स्थापना,
---
संबंधित, व्यक्तिगत, माया, बोली, महाभारत, जोरि, विद्वान, सड़क, सिक्ख, दोनों,

---
तत्व, धार, हटा, प्रीति, काली, मुख्यालय, रचना, पसंद, आभार, देखने,
---
संग्रह, आवश्यक, संघ, मन्त्रालय, खास, सामान्य, उद्देश्य, वकील, कला, रेल,
---

Kannada clusters:

ಮುಖ್ಯ, ಮಾಡಿತು, ಪ್ರಶಸ್ತಿಗಳು, ವಿರಳ, ಇದೊಂದು, ನೇಪಾಳ, ಮಾಡಿದ್ದು, ವಿಶಿಷ್ಟ, ಬೇಕಿದ್ದರೆ, ಉಪಯೋಗಿಸುವುದು,
---
ಮಾಡುವುದು, ಮಾಡಿದರೆ, ಬರೆದರೆ, ಇದಕ್ಕಿಂತ, ಮಕ್ಕಳ, ಈ, ಸಮುದಾಯ, ಶ್ರೀರಂಗಪಟ್ಟಣ, ಬ್ರಿಟಿಷ್, ವಿಜಯನಗರ,
---
ಚೀನಾ, ಪುಟದ, ಪದರದ, ವಿಷಯಗಳ, ಮುಗಿದ, ಅಲೆಗಳ, ಮಾನವನ, ಶಾಸ್ತ್ರದ, ಪ್ರಮಾಣದ, ಸಿಂಹಾಸನದ,
---
ಕಂಡು, ಮುಖ್ಯ, ಪೋಷಕ, ಪ್ರವಾಸಿ, ಜರ್ಮನಿ, ಚಿಕ್ಕಮಗಳೂರು, ಮಂತ್ರಿ, ಓದಿ, ತಾಯಿ, ಕುವೆಂಪು,
---
ದ್ವೀಪರಾಷ್ಟ್ರ, ಸಾಹಿತಿಗಳಲ್ಲೊಬ್ಬರು, ಲ್ಯಾಟಿನ್, ನೇಪಾಳ, ಕೆನಡಾ, ಕಲಾವಿದೆ, ಮುಖ್ಯವಾಗಿ, ಪತ್ರಿಕೋದ್ಯಮ, ವರ್ಗಗಳು, ಜೀವನ,
---
ಸಂಕ್ರಾಂತಿ, ತುಂಗಭದ್ರಾ, ಸಂಧಿಯಾಗಿ, ಸಂಪರ್ಕಗಳು, ಇಂಡಿಯಾ, ವಿಚಿತ್ರ, ಹೆಸರಿನಲ್ಲಿ, ದಾಸ, ಪ್ರಕಾಶ್, ಸಾಹಿತಿಗಳು,
---
ಅಕಾಡೆಮಿ, ಇವರಿಗೆ, ಶಿವಮೊಗ್ಗ, ವಿಶ್ವವಿದ್ಯಾಲಯ, ಅದೇ, ಕರ್ನಾಟಕದ, ಸಾಹಿತಿ, ಲೇಖನಗಳು, ಸೇವೆ, ಆಧಾರಿತ,
---
ಸಾಕಷ್ಟು, ಸರ್ಕಾರದ, ಮತ್ತಷ್ಟು, ಮಾಡಲು, ಆತ್ಮಚರಿತ್ರೆಗಾಗಿ, ಬಗೆಗಿನ, ಹೆಚ್ಚಿನ, ಚರ್ಚೆ, ಚಲನಚಿತ್ರದ, ಜ್ಞಾನೇಶ್ವರ,
---
ಮಾತ್ರ, ಕಾರ್ಯಕ್ರಮಗಳು, ವೇಳೆ, ಮಾಹಿತಿಯನ್ನು, ಮಳೆ, ಎಂಬುದನ್ನು, ಸ್ಥಳದಲ್ಲಿ, ಸಂಭಾಷಣೆಗಳು, ಒತ್ತು, ಎಸ್.ಪಿ.ಬಾಲಸುಬ್ರಹ್ಮಣ್ಯಂ,
---
ಕುರಿತಾದ, ಸಂಬಂಧಿತ, ಕುರಿತ, ಪ್ರದೇಶದ, ಒಳಗೊಂಡ, ಲೇಖನಗಳ, ಸಂಬಂಧಪಟ್ಟ, ಗಳ, ಧರ್ಮ, ಬರೆದ,
---
ವಿಶ್ವವಿದ್ಯಾಲಯದಿಂದ, ಸ್ಥಳ, ತೀರ್ಥಹಳ್ಳಿ, ಸಂಘ, ಭೌತಶಾಸ್ತ್ರ, ವೈದ್ಯಕೀಯ, ವಿಷಯದಲ್ಲಿ, ಚಿಕ್ಕಮಗಳೂರು, ಸಾಗರ, ಹುಬ್ಬಳ್ಳಿ,
---

Marathi clusters:

हातभार, घोषित, जवळील, जागतिक, निर्णय, झाल्या, कौलपानावर, कृषी, त्याच्या, सिंगापूर,

---
सियांग, गोवा, सिंधुदुर्ग, कृष्णा, पाली, रत्नागिरी, पुरुलिया, दौसा, नागौर, उदयपुर,
---
आवश्यक, उत्तरेकडे, अध्यक्ष, वास्तव्यास, पंतप्रधान, स्वरूपाचे, सांगत, प्रदेशात, आले, प्रकाशित,
---
कुवैत, पेरू, पोलंड, इजिप्त, फ्लोरिडा, इराण, हिस्पॅनियोला, इस्रायल, स्मारक, फ्रान्स,
---
गुजरात, व्यक्तिरेखा, आत्तापर्यंत, भारताची, जपानची, मोठे, शैक्षणिक, नेदरलँड्स, डेन्मार्क, भारताच्या,
---
येत, असेल, वापरून, गांधी, केंद्र, ओळखला, साम्राज्य, काढून, नवे, कंपनी,
---
नावाचा, पुण्याला, अजमेर, कायम, महत्त्वाचे, भाग, याबाबत, पुरुलिया, किंवा, पक्षाचे,
---
दुवा, पाली, पाटील, साच्याचा, ऐतिहासिक, मेलबर्न, युरोपमधील, पात्र, नागौर, अत्यंत,
---
फळ, किल्ला, प्रांत, खंड, चर्चापानावर, कि, बोलीभाषा, गाव, पैठण, अजमेर,
---
घोषित, सुरु, वर्णन, बेचिराख, समाविष्ट, सुरक्षित, सुरक्षीत, द्वारे, दुर्लक्ष, दिग्दर्शन,
---
लिखाण, शकेन, स्पर्धक, असावा, वसाहत, संबंध, होतील, वाक्य, प्राण, करीता,
---
रशिया, तेंडुलकर, उल्लेख, दिवसांत, रंगलाल, सत्य, सैन्यासह, नंतर, आदेश, मंत्री,
---
जिल्ह्याचे, याचे, टिचकी, याबद्दल, पश्चिम, केन्द्र, शब्दाच्या, अभय, म्हणजे, जे,
---
होऊन, असल्याने, विकिवर, बऱ्याचदा, पर्वतरांग, पानावर, तसे, त्यांनी, कृपया, पुढे,
---

5. References

[1] Smriti Singh, Kuhoo Gupta, Manish Shrivastava, and Pushpak Bhattacharyya. (2006). Morphological Richness Offsets Resource Demand-Experiences in Constructing a POS Tagger for Hindi, In Proceedings of Coling/ACL 2006, Sydney, Australia, July, pp.779-786.

[2] Manish Shrivastava and Pushpak Bhattacharyya (2008), “Hindi POS Tagger Using Naive Stemming : Harnessing Morphological Information Without Extensive Linguistic Knowledge”, Department of Computer Science and Engineering, Indian Institute of Technology, Bombay. Proceeding of the ICON 2008.

[3] Nidhi Mishra and Amit Mishra. (2011). Part of Speech Tagging for Hindi

Corpus, In the proceedings of 2011 International Conference on Communication systems and Network.
[4] Pradipta Ranjan Ray, Harish V., Sudeshna Sarkar and Anupam Basu, “Part of Speech Tagging and Local Word Grouping Techniques for Natural Language Parsing in Hindi”,Department of Computer Science & Engineering, Indian Institute of Technology, Kharagpur, INDIA 721302. www.mla.iitkgp.ernet.in/papers/hindipostagging.pdf.
[5] Asif Ekbal, Samiran Mandal and Sivaji Bandyopadhyay (2007), “POS Tagging Using HMM and Rule-based Chunking”, Workshop on shallow parsing in South Asian languages
[6] Debasri Chakrabarti (2011), “Layered Parts of Speech Tagging for Bangla”, Language in India www.languageinindia.com, M a y 2 0 1 1, Special Volume:Problems of Parsing in Indian Languages.
[7] Dinesh Kumar and Gurpreet Singh Josan,(2010), “Part of Speech Taggers for Morphologically Rich Indian Languages: A Survey”, International Journal of Computer Applications (0975 – 8887) Volume6–No.5, September, 2010, www.ijcaonline.org/ volume6/number5 /pxc3871409 .pdf..
[8] Navanath Saharia, Das, Utpal Sharma (2009)"Part of speech tagging for Assamese Text"; Proceedings of ACL-IJCNLP 2009 Conference short papers; Suntec. Singapore. pp. 33-36
[9]Rajendran (2006); “Parsing in Tamil”, LANGUAGE IN INDIA "S. ;www.languageinindia.com Volume 6: 8 August, 2006. Technologies, pp.554-558.
[10] M. Selvam, A.M. Natarajan (2009),“Improvement of Rule Based Morphological Analysis and POS Tagging in Tamil Language via Projection and Induction Techniques” , International Journal of Computers, Issue 4, Volume 3, 2009.
[11] A Part of Speech Tagger for Indian Languages (POS tagger), Tagset developed at IIIT - Hyderabad after consultations with several institutions through two workshops, 2007.
[12] G.M. Ravi Sastry , Sourish Chaudhuri and P. Nagender Reddy, “An HMM based Part-Of-Speech tagger and statistical chunker for 3 Indian languages”, www.cs.cmu.edu/~schaudhu/publications.html.
[13] Sathish Chandra Pammi and Kishore Prahallad (2007), “POS Tagging and Chunking using Decision Forests”, Workshop on shallow parsing in South Asian languages, 2007. shiva.iiit.ac.in/SPSAL2007/proceedings.php.
[14] Mona Parakh, Rajesha N. and Ramya M (2011), “Sentence Boundary 1] Akshar Bharathi and Prashanth R. Mannem (2007), “Introduction to the Shallow Parsing Contest for South Asian Languages”, Language Technologies Research Center, International Institute of Information Technology, Hyderabad, India 500032.
[15] Vijayalaxmi .F. Patil (2010), “Designing POS Tagset for Kannada, Linguistic Data Consortium for Indian Languages (LDC-IL), Organized by Central Institute

of Indian Languages, Department of Higher Education Ministry of Human Resource Development, Government of India, March 2010..
[16] Disambiguation in Kannada Texts", Language in India www.languageinindia.c o m 1 1 : 5 M a y 2 0 1 1 Special Volume:Problems of Parsing in Indian Languages, Pages 17-19.
[17] Delip Rao and David Yarowsky (2007), "Part of Speech Tagging and Shallow Parsing of Indian Languages", Department of Computer Science, Johns Hopkins University, USA, 2007. The proceedings of the wworkshop on "Shallow Parsing in South Asian Languages"
[18]Antony P J, Santhanu P Mohan, Soman K P (2010);"SVM based part of speech tagger for malayalam";; Proceedings of 2010 International Conference on Recent trends in IEEE.
[19] Shambavi B R and Dr. Ramakanth Kumar P;"Current state of the ART POS tagging for Indian langages - A study"; IJCET, May-June(2010). pp 250-260
[20]Jyoti Singh, Nisheeth Joshi, Iti Mathur; *PART OF SPEECH TAGGING OF MARATHI TEXT : USING TRIGRAM METHOD*; International Journal of Advanced Information Technology (IJAIT) Vol. 3, No.2, April 2013
[21] Antony P J, Dr. Soman K P; "*Parts Of Speech Tagging for Indian Languages: A Literature Survey"*; International Journal of Computer Applications (0975 – 8887), Volume 34– No.8, November 2011
[22] http://en.wikipedia.org/wiki/Part_of_speech_tagging
[23] Jurafsky and Martini: Speech Language Processing
[24] Manning and Schutze: Foundations of Statistical Natural Language Processing
[25] http://en.wikipedia.org/wiki/Hidden_Markov_model
[26] http://en.wikipedia.org/wiki/Viterbi_algorithm