**CHAPTER 1**

**INTRODUCTION**

This chapter provides the introduction of NLP (natural language processing) and architecture of POS tagging process. It presents features and applications of POS taggers. It also describes different methods of tagging.

**1.1 Context**

NLP (natural language processing) is the process that provides the facility of interaction between human and machine. It is a component of computer science, linguistics and artificial intelligence. It is difficult task to build NLP application because human speech is not always specific. The main objective of NLP is to develop such a system that can understand text and translate between human language and another. The work in area of Part-of-Speech (POS) tagging has begun in the early 1960s. Part of Speech tagging is an important tool for NLP. It is one of the simplest as well as statistical models for many NLP applications. POS Tagging is an initial step of information extraction, summarization, retrieval, machine translation, speech conversion [2].

With the advancement of technology, the demand of Natural Language Processing (NLP) is also increasing and it becomes very important to find out correct information from collection of huge data only on the basis of queries and keywords. Sometimes user tries to search data with help of query and get unimportant or irrelevant data instead of correct data. Due to complex structural effect, this problem occurs mostly with Indian languages as compared to others. To avoid this problem, POS tagging is the best application of NLP that assigns exact part of speech to each word of a text (Mohnot, K, 2014). It is the process of marking up a word in a corpus as corresponding to a particular part of speech use its definition, as well as its relation. POS tags are also known as word classes, morphological classes, or lexical tags to choose correct grammatical tag for word on the basis of linguistic feature. 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. Hybrid based Part of Speech tagger is combination of Rule based approach and Statistical approach. Part of Speech tagging is an important application of natural language processing. It is used in several Natural Languages processing based software implementation. Accuracy of all NLP tasks like grammar checker, phrase chunker, machine translation etc. depends upon the accuracy of the Part of Speech tagger. Tagger plays an important role in speech recognition, natural language parsing and information retrieval (Mehta, D. N, 2015).

POS tagging is the process of assigning the best grammar tag to each word of text like verb, noun, pronoun , adjective , adverb, conjunction , preposition etc. some unknown words exist in every language so it is very difficult task to assign the appropriate POS tag to each word in a sentence [3]. The mostly work that has been done for Indian languages was one of the rule based approaches and other empirical based POS tagging Approach. But the fact was that rule-based approach requires proper language knowledge and hand written rule. Due to morphological effect of Indian languages, researchers faced a great problem to write proper linguistic rules and many cases it was noticed that results were not good. Most of natural language processing work has been done for Hindi, Tamil, Malayalam and Marathi and several part-of-speech taggers have been applied for these languages. After this, researchers moved to stochastic based approach. However the stochastic methods requires large corpora to be effective, but still many successful POS were developed and used in various natural language processing tasks for Indian language.

The main issue after morphological richness of Indian Languages is Ambiguity. It is very time consuming process to assign a correct POS tag to different context words. Due to this reason, POS Tagging is becoming a challenging problem for study in the field of NLP [1].

The main objective of Natural Language Processing is to facilitate the interaction between human and machine. POS tagging is the process of attaching the best grammar tag like to each word of a sentence of some language. A word in a sentence can act as a verb, noun , pronoun , adjective , adverb, conjunction , preposition etc so POS is defined as the grammatical information of each word of a sentence. While assigning a POS tag it is necessary to determine the context of the word i.e. whether it is acting like a noun, adjective, verb etc. Sometime a word can act as a noun in one sentence and in another sentence it can give the sense of verb. So before selecting a POS tag for a word the exact context of the word must be clear. For Indian languages it is a difficult task to assign the correct POS tag to each word in a sentence because of some unknown words in Indian languages. The earlier work that has been done for Indian languages was based rule based approaches. But the rule-based approach needs proper language knowledge and hand written rule. Most of natural language processing work has been done for Hindi, Tamil, Malayalam and Marathi and several part-of-speech taggers have been applied for these languages. The set of tags assigned by a part of speech tagger may contain just a dozen tags so such a big tagset can arise the difficulty in the tagging process. POS tagging is helpful in various NLP tasks like Information Retrieval, Machine Translation , Information Extraction , Speech Recognition etc. For Indian languages researchers find difficulty in writing linguistic rules for rule based approaches because of morphological richness . The other main issue after morphological richness of Indian Languages is Ambiguity. It is very time consuming process to assign a POS tag to each word according to its context in sentence by hand and that is why POS Tagging is becoming a challenging problems for study in the field of NLP.

**1.2 Tagging**

Automatic assignment of descriptors to the given tokens is called tagging. The descriptor is called tag. The tag may indicate one of the part of speech, semantic information and soon. So tagging is kind of classification.

**1.3 Parts-of-speech Tagging**

The process of assigning one of the parts of speech to the given word is called Parts Of Speech tagging. It is commonly referred to as POS tagging. Parts of speech include nouns, verbs, adverbs, adjectives, pronouns, conjunction and their sub-categories.

**Example:**

Word: Paper,

Tag: Noun

Word: Go

Tag: Verb

Word: Famous, Tag: Adjective

Some words can have more than one tag associated with. For example, chair can be noun or verb depending on the context.

**1.3.1 Features for POS tagging**

The Following features have been found to be very useful in POS tagging:

**Suffixes:** The next word of Current token is used as feature.

**Prefixes:** The previous word of Current token is used as feature.

**Context Pattern based Features** Context patterns are helpful for POS tagging. Eg.. Word prefixes and suffix context patterns.

**Word length:** Length of particular word is useful feature.

**Static Word Feature:** The previous and next words of a particular word are used as features. **Presence of Special characters:** Presence Special characters surrounding the current word are used as features.

**1.4 Parts of Speech Tagger**

The broad utilization of internet for making search of information is difficult due to the search systems consist container of words which causes problem in retrieval due to synonyms. There is need to accept the word boundary between what kinds of query information are submitted by humans and what kinds further result get [5]. So for text indexing and retrieval uses POS information. POS tagging is used as an early stage of text analysis in many applications such as subcategory acquisition, text to speech synthesis and alignment of parallel corpora. POS tagging is a necessary pre-module and building block for various NLP tasks like Machine translation, Natural language text processing and summarization, User interfaces, Multilingual and cross language information retrieval, Speech recognition, Artificial intelligence, Parsing , Expert system and so on [2]. Parts of speech (POS) tagging are one of the most well studied problems in the field of Natural Language Processing (NLP). Different approaches have already been tried to automate the task for English and other western languages there are large numbers of POS tagger available for English language which has got satisfactory performance but cannot be applied to Marathi language. Part-of-speech tagging in Marathi language is a very complex task as Marathi is highly inflectional in nature & free word order language [2]. The process of assigning description to the given word is called Tagging. The descriptor is called tag. The tag may indicate one of the parts-of-speech like noun, pronoun, verb, adjective, adverb, preposition, conjunction, and interjection. The input (Raw Text) is tokenized and a corpus is used for detecting the corresponding part of speech of each token in the sentence. For correct POS tagging, training the tagger, corpus and a proper tagset is also important Disambiguation is the most difficult problem in tagging. The ambiguity which is identified in the tagging module is resolved using the grammar rules. Parts Of Speech tagger or POS tagger is a program that does this job. Taggers use several kinds of information: dictionaries, lexicons, rules, and so on. Dictionaries have category or categories of a particular word. That is a word may belong to more than one category. For example, run is both noun and verb. Taggers use probabilistic information to solve this ambiguity.

There are mainly two types of taggers: rule-based and stochastic. Rule-based Tagger use hand-written rules to distinguish the tag ambiguity. Stochastic taggers are either HMM based, choosing the tag sequence which maximizes the product of word likelihood and tag sequence probability, or cue-based, using decision trees or maximum entropy models to combine probabilistic features.

Ideally a typical tagger should be robust, efficient, accurate, tunable and reusable. In reality taggers either definitely identify the tag for the given word or make the best guess based on the available information. As the natural language is complex it is sometimes difficult for the taggers to make accurate decisions about tags. So occasional errors in tagging is not taken as a major roadblock to research.

**1.4.1 Architecture of POS tagger**

1) ***Tokenization:*** Tokenization is the process of separating tokens from raw text. Words are separated by white spaces or punctuation marks. The sentence is segmented by using white space because the occurrence of white space indicates the existence of a word boundary. There are various morphological problems where this approach fails. So by using this we can easily find out the tokens from the sentence. The given text is divided into tokens so that they can be used for further analysis. The tokens may be words, punctuation marks, and utterance boundaries (Bagul, P. et al., 2014).

***2) Ambiguity look-up:*** This is to use lexicon and a guesser for unknown words. While lexicon provides list of word forms and their likely parts of speech, guessers analyze unknown tokens. Compiler or interpreter, lexicon and guesser make what is known as lexical analyser [11].

***3) Ambiguity Resolution:*** This is also called disambiguation. Disambiguation is based on information about word such as the probability of the word. Disambiguation is also based on related information or word/tag sequences. For example, the model might prefer noun analyses over verb analyses if the preceding word is a preposition or article [11]. Disambiguation is the most difficult problem in tagging. The ambiguity which is identified in the tagging module is resolved using the Marathi grammar rules.

***4) WordNet:*** The main relation among words in WordNet is synonymy. WordNet is an electronic database which contains parts of speech of all the words which are stored in it. It is trained from the corpus for higher performance and efficiency (Bagul, P. et al., 2014). WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The majority of the WordNet’s relations connect words from the same part of speech (POS). Thus, WordNet really consists of four sub-nets, one each for nouns, verbs, adjectives and adverbs, with few cross-POS pointers. Cross-POS relations include the “morph semantic” links that hold among semantically similar words sharing a stem with the same meaning [10].

***5) Corpus:*** For correct POS tagging, training the tagger well is very important, which requires the use of well annotated corpora. Annotation of corpora can be done at various levels which include POS, phrase or clause level, dependency level etc. Corpus linguistics is the study of language as expressed in samples (corpora) of "real world" text. Corpus is a large collection of texts. It is a body of written or spoken material upon which a linguistic analysis is based. The plural form of corpus is corpora. Some popular corpora are British National Corpus (BNC), COBUILD/Birmingham Corpus, and IBM/Lancaster Spoken English Corpus (Bagul, P. et al., 2014).

***6) Tagset:*** Apart from corpora, a well-chosen tagset is also important. The language tagset represents parts of speech and consist on syntactic classes. According to contextual and morphological structure, natural languages are different from each other.In the top level the following categories are identified as universal categories for all ILs and hence these are obligatory for any tagset. Some common tags: [N] Nouns [V] Verbs, [PR] Pronouns, [JJ] Adjectives, [RB] Adverbs, [PP] Postpositions, [PL] Participles, [QT] Quantifiers, [RP] Particles, [PU] Punctuations (Mahar, J. A., & Memon, G. Q., 2010).

**1.4.2 Applications of POS tagger**

The POS tagger can be used as a preprocessor. Text indexing and retrieval uses POS information. Speech processing uses POS tags to decide the pronunciation. POS tagger is used for making tagged corpora.

**1.4.3 POS Tagging Techniques**

The POS tagger can be implemented by using either a supervised technique or an unsupervised technique.

Supervised POS taggers are based on pre-tagged corpora [6], which are used for training to learn information about the word-tag frequencies, rule and tag set, sets etc. The performance of the models generally increases with the increase in size of these corpora. Unsupervised POS tagging models do not require pretagged corpora. Instead, they use those methods through which automatically tags are assigned to words [6].

Advanced computational methods like the Baum-Welch algorithm to automatically include tag sets, transformation rules etc. Under these two categories different approaches have been used for the implementation of POS taggers such as: A. Rule Based Approach / Transformation Based The rule based POS tagging approach that uses a set of hand written rules. Rule base taggers depend on word list or lexicon or dictionary to assign appropriate tag to each word. The tagger divided into two stages. First, it search words in dictionary and second, it assigns a tag by removing disambiguity of words using linguistic features of word [6]. On the basis of level rule divided as lexical rules act in a word level, each sentence splits into small words called lexeme or token And, the context sensitive rules act in a sentence level, to check the grammar for the sentence [5]. The transformation based approach is similar to the rule based approach in the sense that it depends on a set of rules for tagging. The transformation based approaches use a pre-defined set of handcrafted rules as well as automatically induced rules that are generated during training [8]. The main drawback of rule based system is that it fails when the text is not present in lexicon. Therefore the rule based system cannot predict the appropriate tags. B. Statistical Approach / Stochastic Tagger A stochastic approach assign a tag to word using i frequency, probability or statistics. From the annotated training data it “selects the most likely tag for the word” and uses same information to tag that word in the unannotated text [1] [5]. Stochastic tagger as a simple generalization of the stochastic taggers generally resolves the ambiguity by computing the probability of a given word (or the tag).The drawbacks of this approach is that it can come up with sequences of tags for sentences that are not acceptable according to the grammar rules. So, it determines the best tag for a word by calculating the probability of previous tags on n value, where the value of n is set to 1, 2 or 3 are known as the Unigram, Bigram and Trigram models [5,8]. Hybrid approach Metaio The hybrid approach is a combination of Rule based approach and statistical approach, that assign most probable tag to the word using statistical after that, if disambiguity is found then by applying grammar rules tagger tries to change it.

***1) Unigram:*** It consider one word at a time and assigns each word to its most common tag.P (ti/wi) = freq (wi/ti)/freq (wi) .Here Probability of tag for current word is calculated by frequency count of word given tag divided by frequency count of that particular word[3].

***2) Bigram:*** It consider two tag the preceding tag and current tag into account.P (ti/wi) = P (wi/ti). P (ti/ti-1).Here P (wi/ti) is the probability of current word given current tag.P (ti/ti-1) is the probability of a current tag given the previous tag [3].

***3) Trigram:*** A model based approach uses prior knowledge of 3D objects in the environment along with their appearance [14]. It use current tag and based on previous two tags.P (ti/wi) = P (wi/ti). P (ti/ti-2, ti-1) where ti and wi indicate tag sequence and word sequence respectively. P (wi/ti) is the probability of current word given current tag. Here, P(ti|ti-2, ti-1)is the probability of a current tag given the previous two tags [3]. 4) Hidden Markov Model (HMM): It is called Hidden Markov model because We cannot determine the exact sequence of tags that generated and calculate usingt = argmax P(w, t) [8] and it is based on the Markovian assumption that the current tag depends only on the previous n tags.The HMM use a transition probability(i.e. forward tag and backward tags) to assign a tag.P (ti/wi) = P (ti/ti-1). P(ti+1/ti). P (wi/ti) P(ti/ti-1) is the probability of current tag given previous tag. P(ti+1/ti) is the probability of future tag given current tag. P (wi/ti) Probability of word given current tag [3].

POS Tagging

Supervised

Unsupervised

Rule Based

Stochastic

Hybrid

Rule Based

Stochastic

Hybrid

Unigram

Bigram

Trigram

HMM

**Fig. 1 Classification of POS tagging models**

**1.5 Organization of Thesis**

The structure of the rest of the Thesis is as follows:

Chapter 2 presents the background of various POS tagging approaches for various languages and it covers the detail about. It also includes literature review of study.

Chapter 3 Tells about the present work, methodology in detail. It explains the algorithm and flowchart of present study.

Chapter 4 presents the results of study and compares this with existing techniques on the basis of different output parameters.

Chapter 5 contains the conclusion and future work. In the end references are marked.

.

**CHAPTER 2**

**LITERATURE SURVEY**

This chapter provides the information of POS taggers for different languges and review of various techniques used for tagging.

In this paper [1] Antony P J and Dr. Soman had presented a survey on developments of different POS tagger systems as well as POS tagsets for Indian languages and the existing approaches that have been used to develop POS tagger tools . They concluded that almost all existing Indian language POS tagging systems are based on statistical and hybrid approach.

This Paper [2] specifies A CRF (Conditional Random Fields) based part of speech tagger and chunker for Hindi had been used by Aggarwal Himashu and Amni Anirudh. After evaluation they found that the strength of Conditional Random Fields can be seen on large training data and CRF performs better for chunking rather than for POS tagging with the training on same sized data. With training on 21000 words with the best feature set, the CRF based POS tagger is 82.67% accurate, while the chunker performs at 90.89% when evaluated with evaluation script from conll 2000.

In this paper [8] Navneet Garg, Vishal Goyal, Suman Preet used Rule Based Hindi Part of Speech Tagger for Hindi. The System is evaluated over a corpus of 26,149 words with 30 different standard part of speech tags for Hindi. The evaluation of the system is done on the different domains of Hindi Corpus. These domains include news, essay, and short storie and system achieved the accuracy of 87.55%.

This paper [7] specfies A Comparison of Unigram, Bigram, HMM and Brill‟s POS Tagging Approaches for some South Asian Languages has been done by Fahim Muhammad Hasan compared the performance of n-grams, HMM or transformation based POS Taggers on three South Asian Languages, Bangla, Hindi and Telegu. And we found that the HMM based tagger might perform better for English, but for South Asian languages, using corpora of different sizes, the transformation based Brill‟s approach performs significantly better than any other approach when using a 26-tags tagset and pre-annotated training corpora consisting of a maximum of 25426, 26148 and 27511 tokens for Bangla, Hindi and Telegu respectively.

In this paper [9] Manjit Kaur , Mehak Aggerwal and Sanjeev Kumar Sharma introduced an improving Punjabi Part of Speech Tagger by Using Reduced Tag Set. They Effort to improve the accuracy of HMM based Punjabi POS tagger has been done by reducing the tagset. The tagset has been reduced from more than 630 tags to 36 tags. We observed a significant improvement in the accuracy of tagging. Their proposed tagger shows an accuracy of 92-95% whereas the existing HMM based POS tagger was reported to give an accuracy of 85-87%.

In this paper [10] Nisheeth Joshi1, Hemant Darbari and Iti Mathur described efforts to build a Hidden Markov Model based Part of Speech Tagger. They used IL POS tag set for the development of tagger. HMM based statistical technique was used to train POS tagger for Hindi. They disambiguated correct word-tag combinations using the contextual information was available in the text and attained the accuracy of 92.13% on test data.

In this paper [12], Pallavi Bagul et al. proposed a rule based pos tagger for Marathi language. The input sentence sent to tokenized function, the one which tokenizes the string into tokens and then comparing tokens with the Word Net. Tagging module assigned a tag to tokenized word and search for ambiguous word and pronoun. The ambiguous words were those words which can act as a noun and adjective in certain context, or act as an adjective and adverb in certain context. Then their ambiguity is resolved using Marathi grammar rules. Author used a corpus which is based on tourism domain called annotated corpus and 3 grammar rules are used for the experiment to resolve ambiguous word which acts a noun and adjective in certain context, or act as an adjective and adverb in certain context.

In this paper [13], H.B. Patil et.al proposed a Partof-Speech Tagger for Marathi Language using Limited Training Corpora. It is also a rule based technique. Here sentence taken as an input generated tokens. Once token generated apply the stemming process to remove all possible affix and reduce the word to stem. SRR used to convert stem word to root word. They developed 25 SRR rule. The root-words that are identified are then given to morphological analyzer. The morphological analysis is carried out by dictionary lookup and morpheme analysis rules. Disambiguation is removed by the use of rule-base model or Hidden Markov Model. Based on the corpus they have identified 11 disambiguation rules that are used to remove the ambiguity. Stemming process removes all possible affixes, it change the meaning of stem word like (Anischit-Nischit).The size of the corpus is increased then more Rules can be discovered which will help to reduce the error rate.

In this paper [14], Jyoti Singh Nisheeth & Joshi Iti Mathur Proposed a Development of Marathi Part of Speech Tagger Using Statistical Approach. They used statistical tagger using Unigram, Bigram, Trigram and HMM Methods. To achieve higher accuracy they use set of Hand coded rules, it include frequency and probability. They train and test their model by calculating frequency and probability of words of given corpus. In unigram technique find out how many time each word occur in corpus and assign each word to most common tag. Bigram tagger makes tag suggestion based on preceding tag i.e. it take two tags previous and current tag. In Trigram provides the transition between the tags and helps to capture the context of the sentence. The probability of a sequence is just the product of conditional probabilities of its trigrams. Basic idea of HMM is assigns the best tag to a word by calculating the forward and backward probabilities of tags along with the sequence provided as an input. Powerful feature of HMM is context description. The POS taggers described here is very simple and efficient for automatic tagging, but it is difficult for Marathi as it is morphological rich language.

In this paper [15], Nidhi Mishra & Amit Mishra proposed Part of Speech Tagging for Hindi Corpus. The system scans the Hindi (Unicode) corpus and then extracts the Sentences and words from the given Hindi corpus. Finally Display the tag of each Hindi word like noun tag, adjective tag, number tag, verb tag etc. and search tag pattern from database. The proposed model for Hindi language is apprehensible, but need to training data to increase accuracy. The efficiency of system judge on the basis of parameter of used need.

In this paper [16], Namrata Tapaswi Suresh Jain proposed a Treebank Based Deep Grammar Acquisition and Part-Of-Speech Tagging for Sanskrit Sentences. In the Sanskrit morphology meaning of the word is remain same. When affixes are added to the stem, words are differentiated at data base level directly. The input is one sentence per line, split the sentence in to words called lexeme. Read each word to find longest suffix, and eliminated the suffix until the word length is 2. Apply the lexical rules and assign the tag. Remove the disambiguity using context sensitive rules. For experimental result Author taken set of 100 words and manually evaluated, The system gives 90% correct tags for each word. The evaluation was done in two stages. Firstly by applying the lexical rules and secondly, after applying the context sensitive rule. The POS taggers described here is very efficient for Sanskrit but it is difficult for Marathi as affix is attached to root word so the meaning of word get change.

In this paper [2010] Ghulam Qadir MEMON proposed a system for “Rule Based Part of Speech Tagging of Sindhi Language”. Take input text, and generate token. Once token generated search and compare selected word from lexicon (SWL) .If word is found one or more times, then store associated tag and if not found add that word into lexicon by generating linguistic rule for new word. The tagset contains 67 tags. A lexicon named SWL is developed having entries of 26366 words. Author also found the frequency for tag. For this purpose, set of 186 disambiguation rules are used for SPOS tagging system. The contextual information is used for rule-based approach and manually assigns a part of speech tag to a word. Accuracy of 96.28% was achieved from SPOS. When more words will be tagged and rules will be added then accuracy will be increased.

In 2012, Kamal Sarkar, Vivekananda Gayen proposed “A Practical Part-of-Speech Tagger for Bengali”. The system has two major phases: training phase and testing phase. In the training phase, the system is trained on a handful of POS tagged Bengali sentences by computing tag transition probabilities and word likelihoods or observation probabilities. In the testing phase, untagged Bengali sentences are submitted to the system for tagging. Viterbi algorithm is used for finding the most likely tag sequence for each sentence in the input document. Author implemented a supervised Bengali trigram POS Tagger from the scratch using a statistical machine learning technique that uses the second order Hidden Markov Model (HMM).The performance of the POS tagger can be improved by introducing more accurate method for unknown word handling.

In 2012, N. Garg et al. have presented paper on Rule Based Part-of-Speech Tagger for Hindi language. Their System is tested over dataset of 26,149 words with 30 different POS tags for Hindi. They have evaluated their system on the different domain of Hindi Corpus including news, essay, and short stories. The system achieved the accuracy of 87.55%.The system mainly works in two steps- firstly the input words are found in the database; if it is present then it is tagged. Secondly if it is not present then various rules are applied. If a sentence consists of 12 words out of which 8 words are unknown, then system fails to tag them. It is hard to decide which rules should be handled first because word tagging resolution is based on neighbor’s word and hence it fails.

In 2008, M. Shrivastava and P. Bhattacharyya present a paper in which they have presented a simple Hidden markov model based POS tagger. As a pre-processor they have employed longest suffix matching stemmer and claim to achieve accuracy of 93.12%. The core idea of their approach is to “explode” the input in order to increase the length of the input and to reduce the number of unique types they encountered at the time of learning. This orderly increases the probability score of the correct choice while simultaneously decreasing the ambiguity of the choices at each stage.

In 2006, H. Agrawal and A. Mani have presented paper on “Part of Speech Tagging and Chunking with Conditional Random Fields” for Hindi. In which training is performed using a morph analyzer and CRF++ to provide extra information like root word and possible Part of speech tags for training. With training on 21000 words and best feature set, they have claimed to have achieved 82.67% accuracy.

In 2013, POS tagger using hidden markov model has been created by N. Joshi, H. Darbari and I. Mathur [23]. They have used 15,200 sentences (3, 58,288 words) from tourism domain to train the system. They have used two special tags ~~and~~ to denote starting and ending of the sentence which was added to all the sentences of the training corpus. The accuracy of 92.13% on test data was attained. Another Part of speech tagger by A. Ekbal, S. Mondal, and S. Bandyopadhyay Has been developed. Their tagger had been at the beginning trained on a Bengali training set of 20396 tokens. Then tested on the Bengali development test set consisting of 5022 tokens and demonstrated 90.9% accuracy. After that their POS tagger is trained on the Hindi training set consisting of 21470 tokens and Telegu training sets consisting of 21415 tokens later. The POS tagger had been tested with Hindi and Telegu development sets, consisting of 5681 and 6098 tokens and it had demonstrated 82.05% and 63.93% accuracies respectively [24].

S. Singh et al. have developed a methodology of POS tagging which the languages having lack of corpora can make use of. Their methodology makes use of locally annotated relatively small-sized corpora of around 15,562 words, fully comprehensive morphological analysis supported by high coverage terminology and a decision tree based learning algorithm [25]. The performance evaluation of their system was done with 4-fold cross validation on the dataset of news domain and they claim to have 93.45% accuracy.

Sandipan Dandapat et al. presented a paper and described about a model that uses composition of supervised and unsupervised learning techniques using a Hidden Markov Model [26]. They have made use of small tagged corpus and also large untagged corpus. They also make use of Morphological Analyzer that takes a word as input and gives all possible POS tags for the word. They took 1003 words from CIIL corpus and tagged it manually. They have obtained an overall accuracy of 95%.

Asif Ekbal et al. made the tagger for Bengali using Maximum Entropy, which makes use of the different circumstantial information of the words with the variety of features that are helpful in predicting the various Part of speech classes [27]. Their tagger has been trained with a training corpus of 72, 341 word forms and it uses a tag set of 26 different POS tags, defined for the Indian languages. Tagger has demonstrated an accuracy of 88.2% for a test set of 20K word forms.

One more tagger presented by S. Dandapat, and S. Sarkar on Bengali that makes use contextual information of words. They have used a structural or morphological analyzer to make better tagging accuracy of the tagger. Further, they have made use of semi-supervised learning by increasing the small labeled training dataset provided with a larger unlabeled training dataset (1,00,000 words). The tagger had an accuracy of about 89% on the test data provided. The supervised model parameters are estimated from the annotated training data including 3085 sentences. Hidden Markov Model learned through supervised training is treated as the basic model for unsupervised learning [28]. The model parameters are than reestimated using the Baum-Welch algorithm by training provided over a fixed set of 11,000 untagged sentences taken from CIIL (Consortium in Indian Languages) corpus.

S. Bandyopadhyay, R. Haque and A. Ekbal et al. Have developed a POS Tagger using Conditional Random Field. They have used tag set of 26 POS tags, which are defined for the Indian languages [29]. The POS tagger they developed has been trained and tested with the 72,341words and 20k word forms, respectively. Their experimental results show that the CRF based tagger achieves accuracy of 90.3%.

J. Singh et al. have developed POS tagger for Marathi. They have used Trigram Method using statistical approach. The concept mainly used here is to explore the most likely POS tag for a current word based on given knowledge of previous two tags by calculating probabilities to determine which is the best sequence of tag . For testing the performance of the system, they have developed a test corpus of 2000 sentences. They claim to have got an accuracy of 91.63%. [31].

Another Marathi Rule Based POS tagger has been developed by H.B.Patil et al. In this approach they have demonstrated Part-of-Speech tagger for Marathi Language which is based on rules [32]. The hand–constructed rules are learnt from corpus and some manual additions after studying the grammar of Marathi language were added. They have tried to disambiguate tagging by analyzing the grammatical feature of the current word, its antecedent word, its succeeding word, etc. After testing their tagger, they claimed to have an accuracy of about 78.82% on three different types of data sets. P. Bagul, A. Mishra et al. Presented a paper on “Rule based POS tagger for Marathi Text”. They describe their system as the one which tokenizes the sentences into tokens and then comparing tokens with the WordNet to assign their particular tags. They had resolved the ambiguity of the words using Marathi grammar rules [33].

K. R. Singha et al. have presented a paper of an attempt to develop a lexicon based POS Tagger for Manipuri. They have applied a set of hand written language specific rules of Manipuri language [34]. In this paper they have designed a 3- tier tag set for Manipuri. This tag set consists of 97 tags including generic attributes and language specific attribute values.

One more tagger by B.S. Purkayastha et al. has been developed for Manipuri using stochastic approach namely Hidden Markov Model. Manipuri rule-based tagger gives tagged output that is used as corpus for training. In order to measure performance of tagger they have used manually annotated test set data that consist of 97 category of Manipuri language. They had claimed to have achieved the accuracy of 92% [35].

Shambhavi.B.R et al. has developed POS Tagger for Kannada language. In this POS tagging task of Kannada language they have chosen Second order Hidden Markov Model and Conditional Random Fields. Their training data consists of 51,269 tokens and test data set incorporate around 2932 tokens. Both data set are taken from EMILLE corpus. Corpus was partitioned into 95% for training and 5% for testing. Their experimental result shows the accuracy of the tools which is based on HMM is 79.9% and CRF is 84.58% [36].

Another tagger for Kannada has been developed by P.J Antony and Soman.K.P. They have developed their own tag set which consist of 30 tags and developed part of speech Tagger using a machine learning algorithm namely Support Vector Machine for Kannada Language. A corpus was collected from Kannada news papers and books, and it is manually morphologically analyzed and tagged using their developed tag set. After the performance of the system is evaluated they found that more efficient and accurate results were obtained as compared with preceding methods for Kannada POS tagging [37].

T.N. Vikram and Shalini R. have also presented their work on Kannada in [38]. Prototype morphological analyzer for south Indian language Kannada has been presented in this paper. The analyzer can simultaneously work as stemmer, spell checker and POS tagger and hence it is very efficient tool.

A Sinhala language based Part of Speech (POS) Tagger using lexical semantics and HMM has been presented by A. Jayaweera and N.G.J Dias. They have used a statistical based approach, in which the tag identification process is done by calculating the probability of tag sequence and the probability of word-likelihood from the given dataset, where the language specific knowledge is axiomatically extracted from the annotated dataset. In this research they used the Beta version of the UCSC Corpus which contains around 6,50, 000 words and from which definite words are 70000, For known words they claim to have an accuracy of more than 90% [39].

In this paper [40] POS Tagging and chunking using CRF and Transformation Based Learning for Telugu has been described. They showed the use of Conditional Random Fields with the help of morphological information and the transformation rules in POS tagging and Chunking. How to train CRFs to achieve good accuracy over other machine learning method is shown. They have also shown how improved training Methods based on the morphological information, contextual and the lexical rules were difficult in achieving good results. They claimed to achieve an accuracy of about 77.37% for Telugu, 78.66% for Hindi, and 76.08% for Bengali using CRF and TBL based POS tagger.

**CHAPTER 3**

**PRESENT WORK**

This chapter provides statement of the problem undertaken for the research work carried out with the objective on which this research study is based upon justify the need for the proposed methodology. This chapter also includes steps taken for implementing the research methodology and flow chart depicting the research process.

**3.1 Problem Formulation**

In Natural Language Processing (NLP), many researchers have done work on Part-of-Speech (POS) taggers for different languages texts using various methods. Taggers with more than 95% word-level accuracy have been developed for English, German and other European Languages, for which large labeled data is available. Due to lack of a large annotated corpus limited work has been done for Indian languages. Very little work has been done previously on POS tagging of Punjabi. As 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 Punjab (India), and in “Shahmukhi” script in western Punjab (Pakistan). On the basis of previous study only seven taggers are studied, which are very less as compared to foreign languages. POS taggers for the official language of the state of Punjab in India. Punjabi is written in “Gurmukhi” script in eastern Punjabi language need to be increased on the basis of limited labeled data.

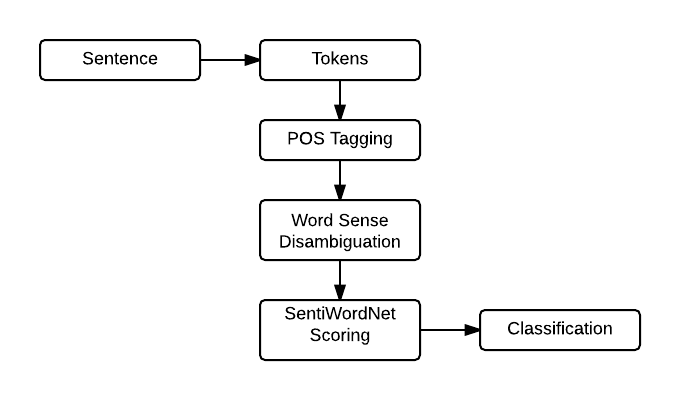
**3.2 Objectives**

The main objectives of this study are following:

* To study and analyze POS taggers for different languages.
* To increase POS taggers for Punjabi language similar to foreign language.
* To implement such taggers on different platform.
* To compare the results of existing and new developed POS taggers.

**3.3 Methodology**

The first step in devising the POS tagset is to understand the word classes and the grammatical information (or features) that will be required for the words of these word classes. The choice of features that are required for various words depends directly on the NLP application in which one has to use the tagset. In grammar checking, or any other similar natural language processing application, these features will be the formal grammatical categories for which the words show inflection and mark various grammatical relations in a typical sentence. However, some other application in which one’s purpose may be just to know the word class of a word will not require these features. As stated earlier, our aim is to use this tagset in grammar checking application. Therefore, we will highlight only those features that can be helpful in checking the grammaticality of Punjabi sentences, thus ignoring any semantic features. When considering grammatical features ou,r only on the features resulting from inflectional morphology and not derivational morphology will be studied.



**Fig. 3.1 Flowchart of POS tagging for Punjabi Language**

Figure 3.1 describes that firstly a sentence is chosen and tokens are assigned to each word of sentence. After that POS tagging of tokens starts by assigning automatic descriptors to tokens. The next step of this process is to identify the sense of words which are tokenized. Sentiwordnet assigns scores to each synset of WordNet three sentiment scores: positivity, negativity, objectivity. According to scores words are classified into taggers.

**CHAPTER 4**

**EXPERIMENTAL RESULTS**

This chapter provides the information of simulation platform and experimental results of study. The results of different techniques are compared on the basis of output parameters.

**4.1 Simulation Platform**

Linux

**4.2 Introduction to Linux Operating system**

As biological data sets have grown larger and biological problems have become more complex, the requirements for computing power have also grown. Computers that can provide this power generally use the Linux operating system

* Linux is a command line interface, used by most large, powerful computers.
* It is very popular, and very easy to find information and get help.
* Linux is very stable - computers running Linux almost never crash.
* Linux is very efficient which can smoothly manage extremely huge amounts of data.
* Most new bioinformatics software is created for Linux - it’s easy for the programmers

**4.3 Architecture of the Linux Operating System**

**Kernel**

The Linux kernel includes device driver support for a large number of PC hardware devices (graphics cards, network cards, hard disks etc.), advanced processor and memory management features, and support for many different types of filesystems (including DOS floppies and the ISO9660 standard for CDROMs). The kernel (in raw binary form that is loaded directly into memory at system startup time) is typically found in the file /boot/vmlinuz, while the source files can usually be found in /usr/src/linux.The latest version of the Linux kernel sources can be downloaded from <http://www.kernel.org>.

**Shells and GUIs**

Linux supports two forms of command input: through textual command line shells similar to those found on most Linux systems (e.g. sh - the Bourne shell, bash - the Bourne again shell and csh - the C shell) and through graphical interfaces (GUIs) such as the KDE and GNOME window managers. If you are connecting remotely to a server your access will typically be through a command line shell.

**System Utilities**

Virtually every system utility that you would expect to find on standard implementations of UNIX has been ported to Linux. This includes commands such as ls, cp, grep, awk, sed, bc, wc, more, and so on. These system utilities are designed to be powerful tools that do a single task extremely well (e.g. grep finds text inside files while wc counts the number of words, lines and bytes inside a file). Users can often solve problems by interconnecting these tools instead of writing a large monolithic application program. Like other UNIX flavours, Linux's system utilities also include server programs called daemons which provide remote network and administration services (e.g.telnetd and sshd provide remote login facilities, lpd provides printing services, httpd serves web pages, crond runs regular system administration tasks automatically). A daemon (probably derived from the Latin word which refers to a beneficient spirit who watches over someone, or perhaps short for "Disk And Execution MONitor") is usually spawned automatically at system startup and spends most of its time lying dormant waiting for some event to occur.

**Application programs**

Linux distributions typically come with several useful application programs as standard. Examples include the emacs editor, xv (an image viewer), gcc (a C compiler),g++ (a C++ compiler), xfig (a drawing package), latex (a powerful typesetting language) and soffice (StarOffice, which is an MS-Office style clone that can read and write Word, Excel and PowerPoint files).

**4.4 Performance Analysis**

The previous work contributes in the investigation of POS taggers for different Indian languages only with use of limited labeled data. The previous methods worked to improve the accuracy with use of limited POS taggers. In this research, new platform is used to introduce new taggers for Punjabi language and accuracy of work is also increased. The experimental setup for Punjabi POS taggers is as following in figure 4.1.

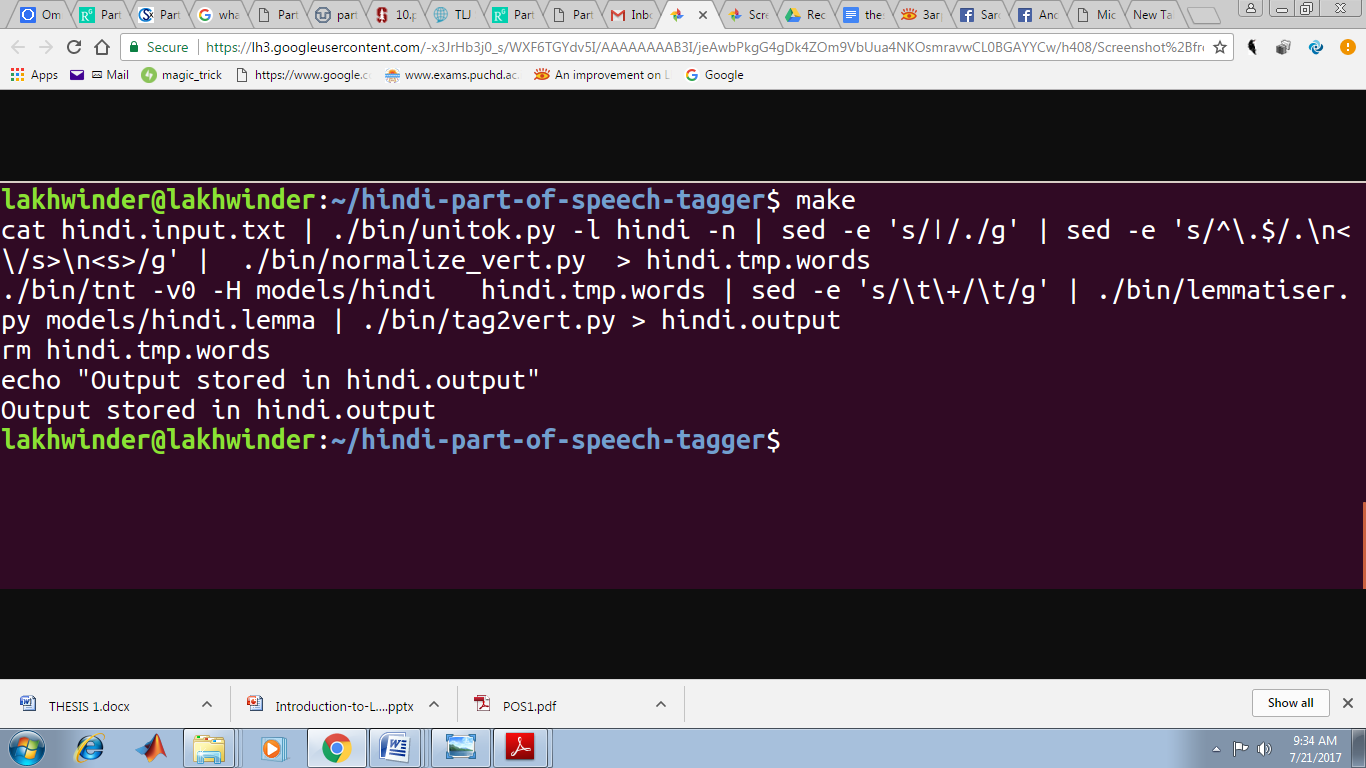


Fig. 4.1 Experimental setup of Linux for Punjabi POS taggers

**4.5 POS tags Chosen for the Current Scheme**

This section gives the rationale behind each tag that has been chosen in this tag set.

**5.1.1 NN**

Noun

The tag NN for nouns has been adopted from Penn tags as such. The Penn tag set makes a distinction between noun singular (NN) and noun plural (NNS).

**5.1.2 NST**

Noun denoting spatial and temporal expressions

**5.2 NNP**

Proper Nouns

**5.3.1 PRP**

Pronoun

**5.3.2 DEM Demonstratives**

**5.4 VM**

Verb Main

**5.5 VAUX**

Verb Auxiliary

**5.6 JJ**

Adjective

**5.7 RB**

Adverb

**5.8 PSP Postposition**

**5.9 RP**

Particle

**5.10 CC**

Conjuncts(co-ordinating and subordinating)

5.11 WQ Question Words

5.12.1 QF

Quantifiers

**5.12.2 QC Cardinals**

**5.12.3 QO Ordinals**

**5.12.4 CL**

Classifiers

**5.13 INTF Intensifier**

**5.14 INJ**

Interjection

**5.15 NEG**

Negative

**5.16 UT**

Quotative

A quotative introduces a quote. Typically, it is a verb. Many Indian languages use quotatives.

**5.17 SYM Special Symbol**

All those words which cannot be classified in any of the other tags will be tagged as SYM

**5.18 \*C**

Compounds (Make it XC – where X is a variable of the type of the compound of which the current word is a member of)

**5.19 RDP**

Reduplication

In this phenomenon of Indian languages, the same word is written twice for various purposes such as indicating emphasis, deriving a category from another category

**5.21 UNK Unknown**

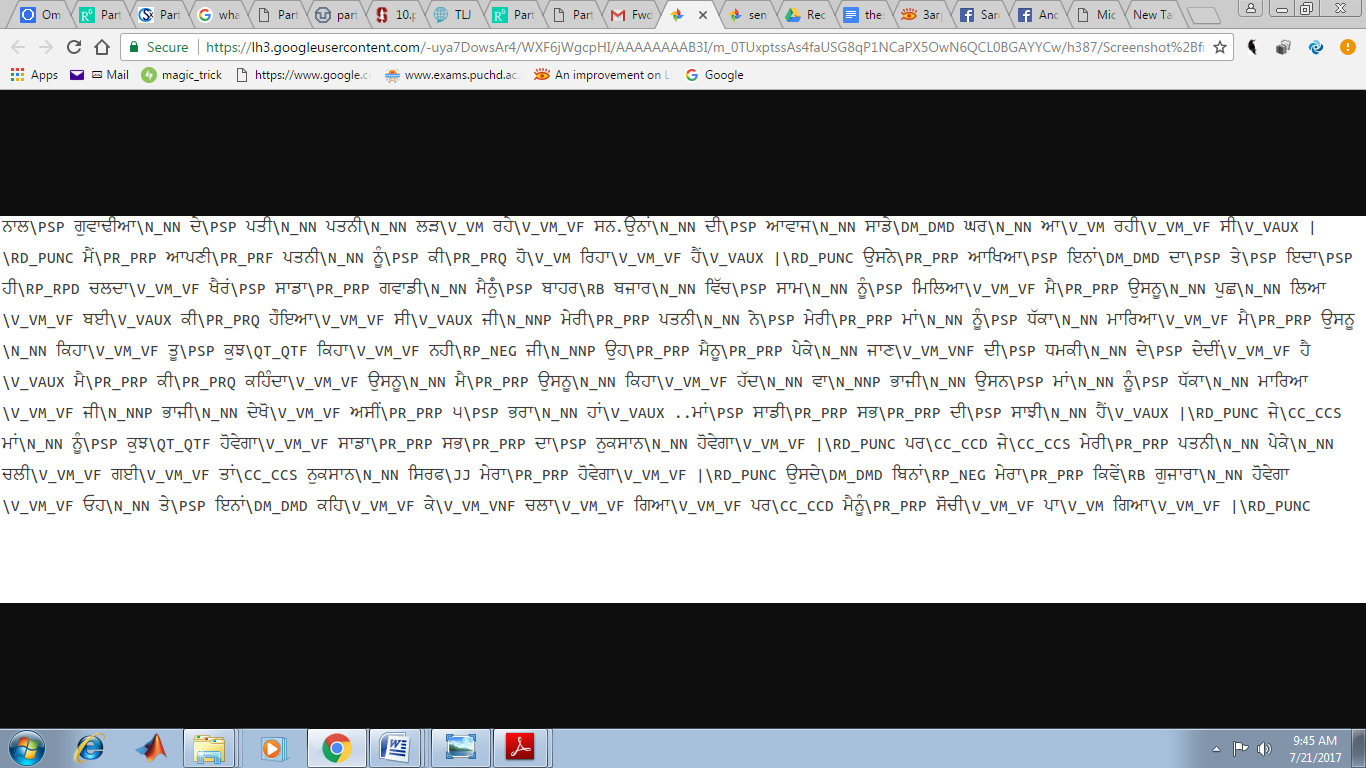
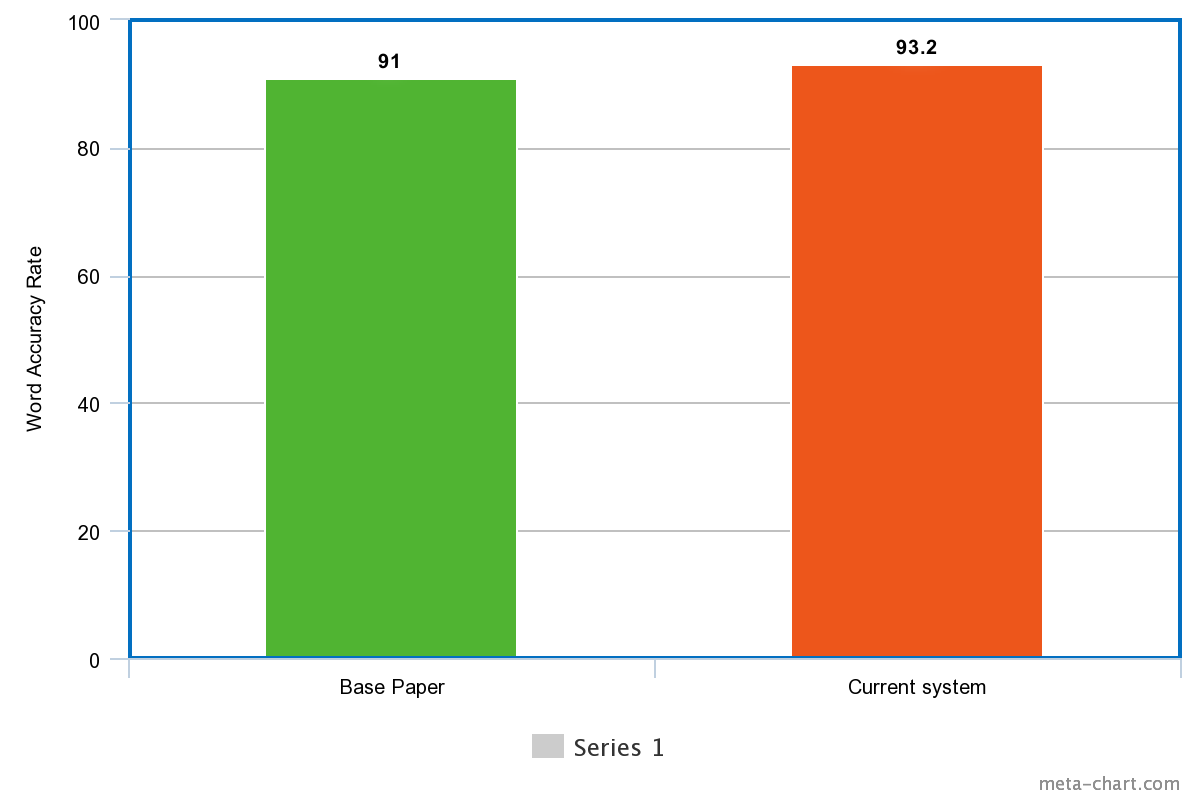


Fig. 4.2 Results of Punjabi POS Taggers

Figure 4.2 shows the results of newly introduced POS taggers. These are increased from seven to twenty five. These taggers also affected the accuracy of tagging and improved efficiency of limited data for Punjabi language.

**4.6 Comparison**



**Fig. 4.4 Comparison of obtained results with existing results**

Figure 4.4 indicates that in previous study the word accuracy rate was just 91%. But, as the new POS taggers introduced and linux platform used, it increased the accuracy rate of tagging by 2.2%.

**CHAPTER 5**

**CONCLUSION AND FUTURE SCOPE**

This chapter provides the conclusion of research study and its future scope.

**5.1 Conclusion**

In this research work, we developed a corpus to tag real time punjabi text. This system has many applications, it can be used by one to learn the Punjabi grammar easily. Part of speech tagging area comes under NLP. In this research work, we use different methods to make it an efficient system. We first use Hidden markovs model POS-tagging algorithms to build on rule-based technique. The concept not only helps the student to learn the grammar for the online source but also helps the teacher to deliver a lecture in a smart classroom. This research work gives clear approach to Punjabi learning step by step procedure. The user first create input text according to its need, and after that type the make command in the terminal. This will produce the output file for the tagger.

**5.2 Future Scope**

The current system only works with the Linux based systems. Our future goal is to make the system that works on all the platforms like windows and the mobile platforms like android and ios.

References

1. Kumar, D., & Josan, G. S. (2010). Part of speech taggers for morphologically rich indian languages: a survey. *International Journal of Computer Applications*, *6*(5), 32-41.
2. Antony, P. J., & Soman, K. P. (2011). Parts of speech tagging for Indian languages: a literature survey. *International Journal of Computer Applications (0975-8887)*, *34*(8), 22-29.
3. Joshi, N., Darbari, H., & Mathur, I. (2013). HMM based POS tagger for Hindi. In *Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing (AISC-2013)*
4. Antony, P. J., & Soman, K. P. (2011). Parts of speech tagging for Indian languages: a literature survey. *International Journal of Computer Applications (0975-8887)*, *34*(8), 22-29.
5. Agrawal, H., & Mani, A. (2006). Part of Speech Tagging and Chunking with Conditional Random Fields. In *the Proceedings of NWAI workshop*.
6. Garg, N., Goyal, V., & Preet, S. (2012, December). Rule Based Hindi Part of Speech Tagger. In *COLING (Demos)*, 163-174.
7. Hasan, F. M., UzZaman, N., & Khan, M. (2007). Comparison of different POS Tagging Techniques (N-Gram, HMM and Brill’s tagger) for Bangla. In*Advances and Innovations in Systems, Computing Sciences and Software Engineering* Springer Netherlands, 121-126.
8. Kaur, M., Aggerwal, M., & Sharma, S. K. (2014). Improving Punjabi Part of Speech Tagger by Using Reduced Tag Set. *International Journal of Computer Applications & Information Technology*, *7*(2), 142.
9. Mohnot, K., Bansal, N., Singh, S. P., & Kumar, A.(2014). Hybrid approach for Part of Speech Tagger for Hindi language*. International Journal of Computer Technology and Electronics Engineering,* 4(1), 25-30.
10. Mahar, J. A., & Memon, G. Q. (2010, February). Rule Based Part of Speech Tagging of Sindhi Language. In *Signal Acquisition and Processing, 2010. ICSAP'10. International Conference on* (pp. 101-106). IEEE.
11. Voutilainen, A. (2003). Part-of-speech tagging. The Oxford handbook of computational linguistics, 219-232.
12. Mohnot, K., Bansal, N., Singh, S. P., & Kumar, A.(2014). Hybrid approach for Part of Speech Tagger for Hindi language*. International Journal of Computer Technology and Electronics Engineering,* 4(1), 25-30.
13. Mehta, D. N., & Desai, N. P. (2015). A Survey on Part-Of-Speech Tagging of Indian Languages. *History*, *43*(198), 125-131.
14. Raju, S. B., Chandrasekhar, P. V., & Prasad, M. K. (2002, December). Application of multilayer perceptron network for tagging parts-of-speech. In*Language Engineering Conference, 2002. Proceedings* (pp. 57-63). IEEE.
15. Bagul, P., Mishra, A., Mahajan, P., Kulkarni, M., & Dhopavkar, G. (2014). Rule Based POS Tagger for Marathi Text. *Proc. Int. J. Comput. Sci. Inf. Technol.(IJCSIT)*, *5*(2), 1322-1326.
16. Patil, H. B., Patil, A. S., & Pawar, B. V. (2014). Part-of-Speech Tagger for Marathi Language using Limited Training Corpora. *International Journal of Computer Applications*, 0975-8887.
17. Singh, J., Joshi, N., & Mathur, I. (2013, August). Development of Marathi part of speech tagger using statistical approach. In *Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on*(pp. 1554-1559). IEEE.
18. Mishra, N., & Mishra, A. (2011, June). Part of Speech Tagging for Hindi Corpus. In *Communication Systems and Network Technologies (CSNT), 2011 International Conference on* (pp. 554-558). IEEE.
19. Tapaswi, N., & Jain, S. (2012, September). Treebank based deep grammar acquisition and Part-Of-Speech Tagging for Sanskrit sentences. In *Software Engineering (CONSEG), 2012 CSI Sixth International Conference on* (pp. 1-4). IEEE.
20. Mahar, J. A., & Memon, G. Q. (2010, February). Rule Based Part of Speech Tagging of Sindhi Language. In *Signal Acquisition and Processing, 2010. ICSAP'10. International Conference on* (pp. 101-106). IEEE.
21. Sarkar, K., & Gayen, V. (2012, November). A practical part-of-speech tagger for Bengali. In *Emerging Applications of Information Technology (EAIT), 2012 Third International Conference on* (pp. 36-40). IEEE.
22. http://wordnet.princeton.edu/
23. <http://language.worldofcomputing.net/pos-tagging/parts-of-speechtagging.htm>
24. Garg, N., Goyal, V., & Preet, S. (2012, December). Rule Based Hindi Part of Speech Tagger. In *COLING (Demos)* (pp. 163-174).
25. Shrivastava, M., & Bhattacharyya, P. (2008, December). Hindi pos tagger using naive stemming: Harnessing morphological information without extensive linguistic knowledge. In *International Conference on NLP (ICON08), Pune, India*.
26. Agarwal, H., & Mani, A. (2006). Part of speech tagging and chunking with conditional random fields. In *the Proceedings of NWAI workshop*.
27. Joshi, N., Darbari, H., & Mathur, I. (2013). HMM based POS tagger for Hindi. In *Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing (AISC-2013)*.
28. Singh, S., Gupta, K., Shrivastava, M., & Bhattacharyya, P. (2006, July). Morphological richness offsets resource demand-experiences in constructing a POS tagger for Hindi. In *Proceedings of the COLING/ACL on Main conference poster sessions* (pp. 779-786). Association for Computational Linguistics.
29. Dandapat, S., Sarkar, S., & Basu, A. (2004, December). A Hybrid Model for Part-of-Speech Tagging and its Application to Bengali. In *International conference on computational intelligence* (pp. 169-172).
30. Dandapat, S., & Sarkar, S. (2006). Part of speech tagging for bengali with hidden markov model. *Proceeding of the NLPAI Machine Learning Competition*.
31. Ekbal, A., Haque, R., & Bandyopadhyay, S. (2007, December). Bengali part of speech tagging using conditional random field. In *Proceedings of Seventh International Symposium on Natural Language Processing (SNLP2007)* (pp. 131-136).
32. Singh, J., Joshi, N., & Mathur, I. (2013). Part of speech tagging of Marathi text using trigram method. *arXiv preprint arXiv:1307.4299*.
33. Dey, A., Paul, A., & Purkayastha, B. S. (2014). Named Entity Recognition for Nepali language: A Semi Hybrid Approach. *International Journal of Engineering and Innovative Technology (IJEIT) Volume*, *3*, 21-25.
34. Patil, H. B., Patil, A. S., & Pawar, B. V. (2014). Part-of-Speech Tagger for Marathi Language using Limited Training Corpora. In *IJCA Proceedings on National Conference on Recent Advances in Information Technology NCRAIT (4)* (pp. 33-37).
35. Singha, K. R., Purkayastha, B. S., & Singha, K. D. (2012). Part of Speech Tagging in Manipuri with Hidden Markov Model. *International Journal of Computer Science Issues (IJCSI)*, *9*(6).
36. Singha, K. R., Purkayastha, B. S., & Singha, K. D. (2012). Part of Speech Tagging in Manipuri with Hidden Markov Model. *International Journal of Computer Science Issues(IJCSI)*, *9*(6).
37. Antony, P. J., & Soman, K. P. (2010, July). Kernel based part of speech tagger for kannada. In *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on* (Vol. 4, pp. 2139-2144). IEEE.
38. Vikram, T. N., & Urs, S. R. (2007, December). Development of prototype morphological analyzer for he south indian language of kannada. In *International Conference on Asian Digital Libraries* (pp. 109-116). Springer Berlin Heidelberg.
39. Jayaweera, A. J. P. M. P., & Dias, N. G. J. (2014). Hidden Markov Model Based Part of Speech Tagger for Sinhala Language. *arXiv preprint arXiv:1407.2989*.
40. PVS, A., & Karthik, G. (2007). Part-of-speech tagging and chunking using conditional random fields and transformation based learning. *Shallow Parsing for South Asian Languages*, *21*.