Chapter 2

LITERATURE REVIEW

This section will discuss different part of speech tagging technologies and the analysis of their results. We also present brief review of the prior work in part of speech tagging in Indian languages as well as International status on POS tagging. This chapter presents review of the work in part of speech tagging as well as corpus in Nepali language.

2.1 A Review of POS Tagging Approaches

Part-of-speech tagging is an important research topic in Natural Language Processing [45]. Taggers are often pre-processors in NLP systems. Considerable amount of work has already been done in the field of POS tagging for English. Different approaches like the rule based approach, the stochastic approach and the transformation based learning approach along with modifications have been tried and implemented. However, if we look at the same scenario for South-Asian languages such as Bangla, Hindi and Nepali, we find that not much work has been done in this area of research work.

A large number of current language processing systems use a part-of-speech tagger for pre-processing. The tagger assigns a part-of-speech tag to each token in the input and

passes its output to the next processing level, usually a parser. For both the applications, a tagger with the highest possible accuracy is required. Recent comparisons of approaches that can be trained on corpora have shown that in most cases statistical approaches yield better results than finite-state, rule-based, or memory-based taggers.

Tagging can be seen as a prototypical problem in lexical ambiguity; advances in part-of-speech tagging could readily translate to progress in other areas of lexical and perhaps structural ambiguity, such as word sense disambiguation and prepositional phrase attachment disambiguation. Also, it is possible to cast a number of other useful problems as part-of-speech tagging problems, such as information retrieval, letter-to-sound translation and building pronunciation networks for speech recognition.

.

The work on automatic part of speech tagging started in early 1960s [21, 26]. Klein and Simmon's rule based POS tagger can be considered as the first automatic tagging system [25]. Since rule based approaches need more sophisticated rules to capture the language knowledge, later on the data driven approaches were developed [20] and recently machine learning approaches are being developed [24, 28,27]. In the following sections, some of the related taggers that have been implemented for English and other language with their performance are reported and subsequently, the tagger available for Nepali language is also mentioned.

When Part-of-speech tagging was initially explored in [14][46], people manually developed rules for tagging; sometimes it developed with the help of a corpus. Due to availability of large corpora, simple Markov-model based stochastic taggers that were

automatically trained could achieve high accuracy. Markov model [42] based taggers assign to a sentence the tag sequence that maximizes P (word |tag) * P (tag | previous n tags). These probabilities can be estimated directly from a manually tagged corpus. These stochastic taggers have more advantages compared to the manually built taggers. In manual built tagger there is need for laborious manual rule construction, and possibly capturing useful information that may not have been noticed by the human engineer. However, stochastic taggers have the disadvantage that here linguistic information is captured from large tables of statistics and indirectly it captured the information. All the recent work in developing automatically trained part-of-speech taggers has been on further explored by Markov model based tagging.

Statistical methods have also been used [16][46]and these provide the capability of resolving ambiguity problems on the basis of most likely interpretation. Hidden Markov Model [47] has been widely used that assumes that a word depends on probabilistically on just its part-of-speech category, which in turn depends solely on the categories of the preceding one word (bigram) or two words (in case of trigram).Two types of training (i.e. parameter estimation) have been used with this model. The first makes use of a tagged training corpus. Merialdo and Derouault used a bootstrap method for training [17][47] and initially a relatively a small amount of text is manually tagged and used to train a partially accurate model. The model was then introduced more text for tagging, and the tags were manually corrected with linguistic experts and then used to retrain the model. Church used the tagged Brown corpus for training [18]. The second method of training does not require a pre-tagged corpus. In this case, the Baum-Welch algorithm (forward-backward algorithm) can be used [11]. Under this system the model is called a Hidden Markov

Model (HMM), as state transitions (i.e., part-of-speech categories) are assumed to be unobservable. Jelinek has used the above mention technique for training a text tagger [19]. Parameter smoothing can be conveniently achieved using the method of ‘deleted interpolation’ in which weighted estimates are taken from second and first-order models and a uniform probability distribution. Kupiec used word equivalence classes (ambiguity classes) based on parts of speech, to pool data from individual words. The most common words are still represented individually, as sufficient data exist for robust estimation. All other words are represented according to the set of possible categories they can assume. In this way, the dictionary of 50,000 words in the Brown corpus can be reduced to approximately 400 distinct ambiguity classes [62]. To further reduce the number of parameters, a first-order model can be employed where word's category depends only on the immediately preceding word's category.

Part-of-speech tagging (POS) itself is a useful tool in lexical disambiguation; for example, knowing that "Plant" is being used as a verb rather than as a noun indicates the word's appropriate meaning. Many words have different meanings even while occupying the same part of speech. The approaches for part of speech tagging can be classified into two main classes depending on the tendencies followed for establishing the Language Model(LM): the linguistic approach, based on hand-coded linguistic rules and the learning approach derived from a corpora.

Much research has been done to improve tagging accuracy using several different models and methods. Most NLP applications demand at initial stages shallow linguistic information (e.g., part–of–speech tagging, base phrase chunking, named entity recognition).This information may be predicted fully automatically (at the cost of some

errors) by means of sequential tagging over unannotated raw text. Generally, tagging is required to be as accurate as possible, and as efficient as possible. But, certainly, there is a trade-off between these two desirable properties. This is so because obtaining a higher accuracy relies on processing more and more information, digging deeper and deeper into it. However, sometimes, depending on the kind of application, a loss inefficiency may be acceptable in order to obtain more precise results. Or the other way around, a slight loss in accuracy may be tolerated in favour of tagging speed.

Some languages have a richer morphology than others, requiring the tagger to have into account a bigger set of feature patterns. Also the tagset size and ambiguity rate may vary from language to language and from problem to problem. Besides, if few data are available for training, the proportion of unknown words may be huge.

Sometimes, morphological analyzers could be utilized to reduce the degree of ambiguity when facing unknown words. Thus, a sequential tagger should be flexible with respect to the amount of information utilized and context shape. Another very interesting property for sequential taggers is their portability. Multilingual information is a key ingredient in NLP tasks such as Machine Translation, Information Retrieval, Information Extraction, Question Answering and Word Sense Disambiguation, just to name a few. Therefore, having a tagger that works equally well for several languages is crucial for the system robustness. Besides, quite often for some languages lexical resources are hard to obtain. Therefore, ideally a tagger should be capable for learning with fewer (or even none)annotated data. The symbol [21] is intended to comply with all the requirements of modern NLP technology, by combining simplicity, flexibility, robustness, portability and efficiency with state–of–the–art accuracy. This is achieved by working in the Support

Vector Machines (SVM) learning framework [48], and by offering NLP researchers a highly customizable sequential tagger generator.

In the recent literature, several approaches to POS tagging based on statistical and machine learning techniques are applied, including among many others like Hidden Markov Models [13], Transformation–based learning, Maximum Entropy taggers Decision Trees, Memory–based learning and Support Vector Machines [16]. Most of the previous taggers have been evaluated using the Penn Treebank set of POS categories on the English WSJ corpus, and a lexicon constructed directly from the annotated corpus. Although the evaluations were performed with slight variations, there was a wide consensus in the late ‘90s that the state–of-the–art accuracy for English POS tagging was between 96.4% and 96.7%[49]. TnT is an example of a really practical tagger for NLP applications. It is available to anybody, simple and easy to use, considerably accurate, and extremely efficient, allowing training from 1 million word corpora in just a few seconds and tagging thousands of words per second. Many natural language tasks is to developed accurate Part-Of-Speech (POS) to tag unseen text. Due to the availability of large corpora which have been manually annotated with POS information, many taggers use annotated text to "learn" either rules or probability distributions and use them to automatically assign POS tags to unseen text.

2.2 Indian Language Taggers

Looking at the scenario for Indian languages, it was found that very little work has been done on POS tagging of Hindi or any other Indian language. In this section, the work done in Indian language related with POS tagging and NLP are reviewed.

The part-of-speech tagging problem was solved as an essential requirement for local word grouping. Lexical sequence constraints were used to assign the correct POS labels for Hindi and a POS disambiguation was attempted by Ray [10].

Pradip Ranjan, Harish V, Sudeshna Sarkar and Anupam Basu[20] of IIT, Kharagpur developed an algorithm for local word grouping and other algorithm for POS tagging. They used Morphological Analyser (MA) on each word of the sentence and after getting the output from MA (all possible lexical information about the root) the word is input to the tagger. The Possible Parts of Speech (PPOS) associated with the surface form of the word is obtained from the MA.

Sandipan Dandapat, Sudeshna Sarkar and Anupam Basu [23] of IIT, Kharagpur developed a hybrid model of POS tagging by combining both the supervised and unsupervised stochastic technique of POS tagging. In the training phase, they used five hundred tagged sentences for supervised learning and fifty thousand words as raw data for re-estimating parameter. In the first stage, they processed the tagged data by supervised learning and in the subsequent iterations the untagged data were processed and updated the probabilities (i.e. transition and emission). Untagged data were used to re-estimate the probabilities (transition and emission). For re-estimating the probabilities Baum-Welch algorithm was used. And in the decoding phase Viterbi Algorithm was used to determine the best probable sequence of tags. They have tested three different approaches of POS tagging.

- Method I: POS tagging was tested using only supervised learning.
- Method II: POS tagging was tested using a partially supervised learning and decoding the best tag sequence using Morphological Analyzer restriction.

• Method III: POS tagging using a partially supervised learning and de- coding the best tag sequence without using Morphological Analyzer restriction.

In these three cases, accuracies were 64.31%, 67.6% and 96.28% respectively. In 2007 Avinesh PVS, Kartik G[6] of IIIT Hyderabad used a technique for part of speech tagging using conditional Random Fields (CRF) and transformation based learning.

Smriti Singh, Puspak Bhattacharyya, Manish Shrivastava, and Kuhoo Gupta[24] of IIT, Bombay developed a POS tagger for a morphologically rich language like Hindi. They established a methodology of POS tagging which was very effective for the resource disadvantaged languages (lacking annotated corpora). They used locally annotated modestly sized corpora (15562 words), exhaustive morphological analysis backed by high coverage lexicon and a decision tree based learning algorithm. The heart of the system was the detailed linguistic analysis of morph syntactic phenomena, adroit handling of suffixes, accurate verb group identification and learning of disambiguation rules.

Gautam Kumar Saha [22] of C-DAC Kolkata developed a system for machine assisted part of speech tagging of Bangla words in Bangla corpora. His work aims to provide an integrated user-friendly software interface to the user to annotate the Bangla word set selected from various electronic Bangla corpuses. Himanshu Agrawal [71] of IIIT Hyderabad developed a tagger for South Asian languages. He used Conditional Random Fields to train the system on the corpus made available by the SPSAL workshop at ICJAI 2007. Basically they have worked on improving the machine's learning without using any machine specific tools like dictionaries, morphological analyzer etc. The overall performance for the three languages was 79.13%.

Aniket Dalal, Pushpak Bhattacharyya, Uma Sawant, Sandeep Shelke and Kumar Nagraj [74] of IIT Bombay developed a tagging system for a morphologically rich language: Hindi. They employed maximum entropy Markov model with a rich set of features capturing the lexical and morphological characteristics of the language. The system was evaluated over a corpus of 15562 words (75% training set and 25% test set) developed at IIT Bombay. Their system achieved the best accuracy of 94.89% and an average accuracy of 94.38%. Sathish Chandra Pammi [19] of IIIT Hyderabad and Kishore Prahallad of Carnegie Mellon University, USA developed a POS tagger and chunk tagger using Decision Forests. This work focused on the investigation towards exploring different methods for part of speech tagging of Indian languages using sub-words as units. Generally most of the POS taggers use morphological analyzer as a module in their tagger [10].

Team	Language	Affiliation	POS Tagging Accuracy (%)		
			Prec.	Recall	Precision
Mla	Bengali	IIT-Kgp	84.32	84.36	84.34
Indians	Telugu	IIIT-Hyd	81.59	81.59	81.59
Iitmcsa	Hindi	IIT-M	80.72	80.72	80.72
Tilda	Hindi	IIIT-Hyd	80.46	80.46	80.46
ju_cse_beng	Bengali	JU,Kolkata	79.12	79.15	79.13

Table 2.2: POS tagging accuracy in the NLPAI machine learning contest

But building morphological analyzer to a particular Indian language is a very difficult task. They tried to capture similar information in an indirect way by splitting the word to be tagged into sub words.

2.3 Nepali Language Taggers

After giving short description of previous work on POS tagging for English and other resource rich language, in this section, the work done in Nepali language related with POS tagging and NLP is reviewed.

The Unitag1 has been developed or customized for Nepali language and was used for semi automatic tagging of Nepali National Corpus under the NERLAC project and tagset used is NERLAC project with 112 tags. Originally, Unitag was developed for Urdu language by Hardie etal [62].It consists of lexical analysis, a powerful morphological system and twin disambiguation modules, hand-written rules and the other using a probabilistic system based on a Hidden Markov model. After tagging, the corpus was manually reviewed and then correction was done. Since the tagset used was very large, it showed more error in tagging. Later the TnT tagger has been used as POS tagger with the 43 tags and training corpus of medium size as one of the pipelined modules for computational grammar analyzer.

2.4 Corpus Review

A corpus is a valuable resource in Natural Language Processing. Their existence in correct form makes the NLP a more fruitful process. The most well known corpora for English are probably the Brown Corpus and the Penn Tree Bank corpus. The Brown Corpus contains over a million words of American English and it was tagged in 1979 using the TAGGIT [23] tagger. Nowadays, corpora tend to be much larger, and are compiled mainly through

projects and initiatives such as the Linguistic Data Consortium (LDC), the Consortium for Lexical Research (CLR), etc. These associations provide corpora as the Wall Street Journal (WSJ, 300 million words of American English), the Hansard Corpus (bilingual corpus containing 6 years of Canadian Parliament sessions), The Lancaster Spoken English Corpus (SEC) etc. Although most corpora limit their annotation level to part of speech tags, some other higher level annotations constitute an important source of knowledge for those researching in NLP. For instance, Penn Tree bank corpus was an example of syntactically analyzed corpora (called Treebanks), which contains 3 million words from the WSJ corpus. Until few years ago, the existing corpora were all of the English language. Nevertheless, the success and applicability of corpus in linguistics as well as in NLP, has raised a wide interest and caused its quick extension to other languages. The following list(not exhaustive) provides some examples of available corpora of languages other than English.

2.5 Nepali Corpus

The Nepali National Corpus (NNC) from NELRALEC (Nepali Language Resources and Localization for Education and Communication) project, which contain 14 million Nepali words. It consists of speech corpus, spoken corpus, core sample (CS), general collection, and parallel data.

- The core corpus is a collection of Nepali written texts that concur as far as possible with the date, number and genres of the international FLOB and FROWN corpora consisting of 500 texts of 15 different genres with 2000 words each published between 1990 and 1992.

- The spoken corpora, designed on the basis of Goteborg Spoken Language corpus (GSLC), has been collected from 17 social activities in their natural settings and contain about 2,60,000 words. These texts are audio-video recordings of the activities with their phonological transcriptions and annotations about the participants but their audio-visual materials can later be transcribed and used for analyzing their paralinguistic and extra linguistic features. Each activity is stored in three files (recording as such in .mpeg, transcription in .txt and recording information in .doc). The main purpose of this collection is to compare it with the written texts and identify their differences.
- The parallel corpora consist of two collections in two genres, computing and national development. Computing texts in both Nepali and English, is about 3 million words, whereas national development text is about 966, 203 words. These corpora can be used as useful resources in developing machine translation system for translating Nepali texts into English and vice versa. They can also be helpful in preparing a Nepali-English/English-Nepali bilingual dictionary, contrastive studies and devising teaching materials for language teaching.

- The speech corpus [72] is a specialized text-to-speech (TTS) corpus for use in creating the software to speak Nepali from written texts. It consists of 1,880 sentences and 6053 words, extracted from the core corpus and later recorded in both male and female voices.

It was first manually tagged in some part (One hundred and sixty texts from the NNC‑CS were annotated manually using this tagset with 112 tags). This data then served as the basis for the training of an automatic tagger. The Nepali English parallel corpus annotated

with 43 pos tags developed at Madan Puraskar Pustakalaya (MPP) contains nearly 88000 words [29].

.