CHAPTER 7

Rule based Part of Speech Tagging in Manipuri

7.1 Introduction

Rule based part of speech tagging use contextual information to assign tags to unknown or ambiguous words. These rules are often known as *context frame rules*. As an example, a context frame rule might say something like: "if an ambiguous or unknown word X is preceded by a determiner and followed by a noun, tag it as an adjective" [39].

det - X - n = X/adj

In addition to contextual information, the tagger uses many linguistics rules to aid in the disambiguation process.

It is found that statistical techniques were more successful than rule based methods, but their storage, improvement and adaptation cost was higher. In addition to that, the rule based tagger has many advantages over statistical tagger, including a vast reduction in stored information, the perspicuity of a small set of meaningful rules, ease of finding and implementing improvements to the tagger [10]. In this chapter we represent the rule based part of speech tagger of Manipuri by applying a set of hand written linguistics rules of Manipuri language. Nevertheless, it is very difficult to classify the lexical categories of Manipuri, an agglutinating Tibeto-Burman language of Northeast India. So, in this tagger we are using the affix stripping technique to segment the affixes from the root. As Manipuri has limited POS tagged corpus, the tagged output of this tagger will

be very helpful to analyze Manipuri part of speech by using many statistical models.

7.2 Related works

In 1963, Klein and Simmons introduced a computational approach for grammatical coding of English words. Their primary goal was to avoid the labour of constructing a very large dictionary. Their algorithm uses a set of 30 POS categories. It first seeks each word in dictionaries, then checks for suffixes and special characters as clues. Finally, the context frame tests are applied. This algorithm correctly and unambiguously tags about 90% of the words in several pages of the Golden Book Encyclopedia [49].

The next important tagger, *TAGGIT*, was developed by Greene and Rubin in 1971. The tag set used is very similar, but somewhat larger, at about 86 tags. The dictionary used is derived from the tagged Brown Corpus, rather than from the untagged version. This tagger correctly tags approximately 77% of the million words in the Brown Corpus [31].

In the year 1992 Eric Brill has been developed a rule based POS tagger with the accuracy rate of 95-99% [6]. POS tagging of some languages like Turkish [60], Czech [34] has been attempted using a combination of hand-crafted rules and statistical learning. Adopting rule based approach a POS tagger for Marathi has been developed in 2006 using a technique called SRR (suffix replacement rule) by Sachin Burange et al. [13].

As per the literature, there is a few works related to POS tagging in Manipuri and other Tibeto-Burman languages in the Indian Sub-

continent. In the year 2004, Sirajul Islam Choudhury, Leihaorambam Sarbajit Singh, Samir Borgohain, P.K. Das have designed and implemented a morphological analyzer for Manipuri language [18]. Besides, D. S. Thoudam et al. developed morphology driven Manipuri POS tagger in 2008 [71].

7.3 The Proposed Design for Manipuri Part of Speech Tagging

The proposed system design for rule-based Manipuri POS tagger is given below:

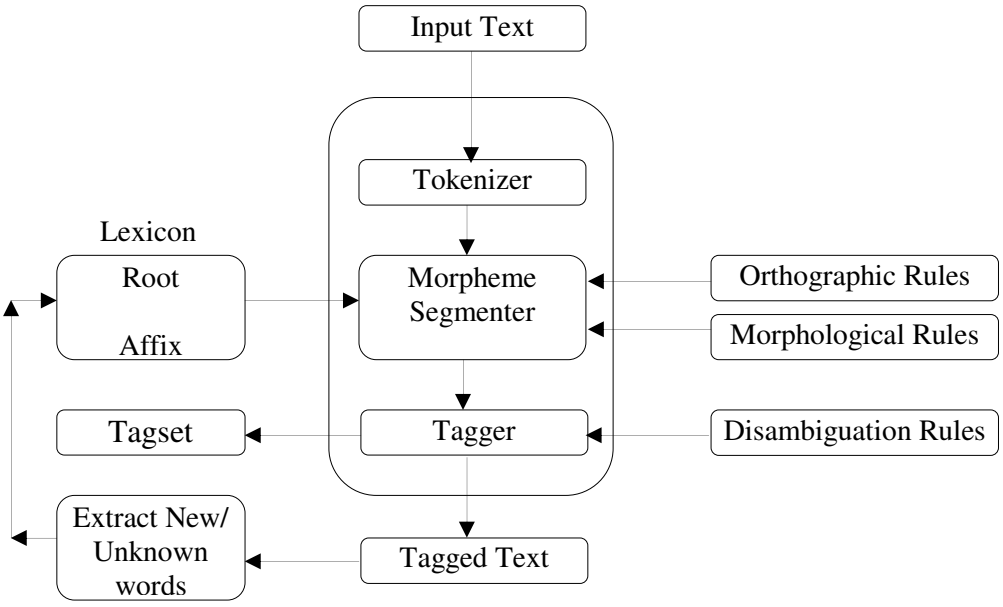

Figure 7.1 System design of proposed POS Tagger

The different modules involved in this architecture are explained as following:

Tokenizer: Tokenization is the first step in part of speech tagging of any natural language. It separates the words including punctuation marks and the symbols of the input text into tokens by using the whitespace between consecutive words. Manipuri has its own writing convention of keeping a whitespace between the words but there is no whitespace between a word and punctuation mark or

symbol. So for the proper tokenization of Manipuri text, we have developed a simple rule for such type of identification as follows:

If (word.endsWith ("punctuation mark" | |"symbol"))

then insert a whitespace between the word and punctuation mark or symbol;

An example of tokenization of a simple Manipuri sentence is given in Figure 5.3 below, here the filled box is word and blank box is whitespace.

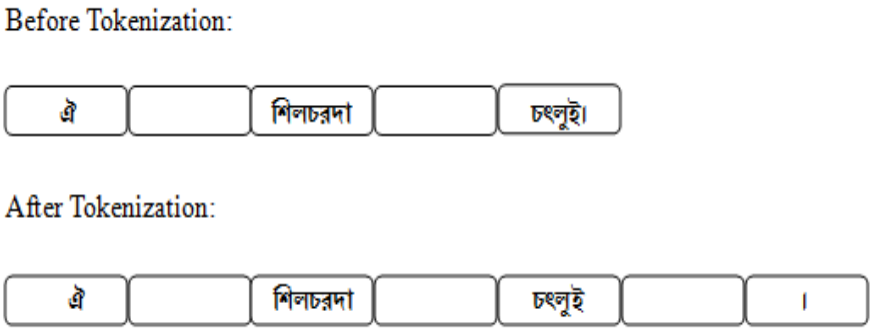

Figure 7.2 Tokenization of a simple Manipuri sentence

Morpheme Segmenter: It separates the affixes i.e., prefixes and suffixes from the stem or root word and analyzes the words according to the morphological rules and identifies the morphosyntactic categories of root and affixes including the relations between the morphemes. Morpheme Segmenter plays an important role to identify the affixes and stem in this architecture because affixation is one of the word formation processes of Manipuri language. It separates the suffixes starting from the right end of the word towards the left by using iterative suffix stripping technique. Figure 7.3 shows a diagrammatic view of segmenting a simple Manipuri word in to different individual morphemes:

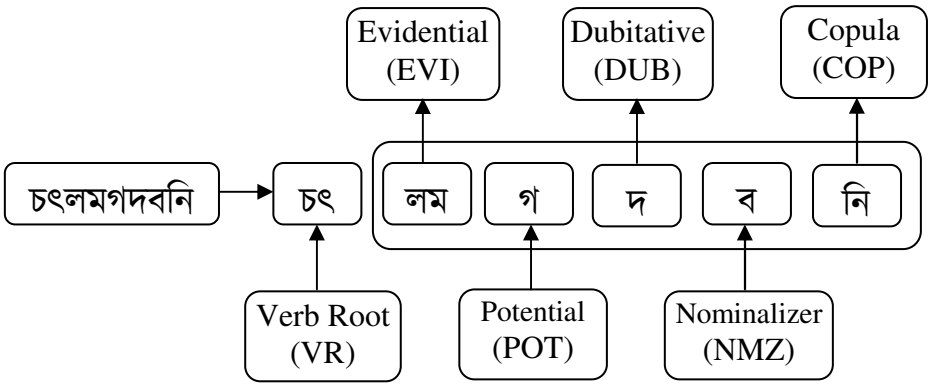

Figure 7.3 Segmentation of a simple Manipuri word

Lexicon: Lexicon consists of the list of roots and affixes with their corresponding part of speech tags which are in the tagset. Initially a tagged lexicon is developed manually by collecting limited words from Manipuri newspapers, books and dictionaries. The entry of lexical items in the lexicon is as follows:

Item	**Tag**
চৎ	VR
অফবা	MJ

Tagger: Tagger is used to resolve the ambiguity issues in tagging and tag the proper tag to a token. In some cases, a token has more than one tag but the tagger tagged only one tag to a token by applying hand written rules for those specific cases.

Extraction of new words: In this architecture, the tagger first tag the words which are in the lexicon and the words which are not available in the lexicon are also tagged by applying rules. If the rules failed to tag such words then the new words are given a specific tag as "UNK" i.e., unknown which can

be extracted from the tagged output text. The new words are then entered into the lexicon and new rules are created for the new words.

In this system design, three types of rules are applied for implementing this rule-based tagger. Different rules are formulated with example as shown below:

A. Orthographic Rules: There are orthographic variations in the spelling system of Manipuri having difficulties in formulating orthographic rules consistently. However, attempt has been made to formulate the rules of orthography in Manipuri experimentally. A simple example of such kind of rule for Manipuri is given below:

If any stem getting after stemming the suffixes and ends with "ব/ba" or "প/pa" then "ব/ba" or "প/pa" will be replaced by "বা/baa" or "পা/paa" respectively like

অঙাংবগী→অঙাংবা_MJ গী_GEN

angaangba-gi → angaangbaa_MJ gi_GEN

চৎপগী→চৎপা_NV গী_GEN

chatpa-gi → chatpaa_NV gi_GEN

B. Morphological Rules: The word formation in Manipuri is employed by three morphological processes called affixation, derivation and compounding. Some verb roots can be formed noun, verb, adjective and adverb by affixation as shown below with examples in the underlying representation.

Prefix +bound root + suffix → Adjective

অ/Prefix + চেন/VR + বা/NMZ → অচেনবা / MJ

অ/Prefix + চৎ/VR + পা/NMZ → অচৎপা / MJ

If a word starts with "অ" ***and ends with*** "বা" ***or*** "পা"

then tag the word as Adjective (MJ)

B. Disambiguation Rules: Any natural language has the ambiguity issues as the single word has different tags or categories. To overcome the ambiguity issues and assigning a proper tag to a word, disambiguation rules are required. In Manipuri there are plenty of words which have multiple tags. Consider the following examples in this regard.

First example: ঐ চা থকলি। / ei cha thakli. / I am taking tea.

Second example: ঐ কমলা চারি। / ei kamla chaari. / I am eating orange.

The word "চা" might be Common Noun (NC) in the first example or Verb Root (VR) in the second example. Now the rule of disambiguation is as

Given Input: "চা"

If (+1 is VR/MJ/NC) ***/* if next word is verb root, adjective or Common noun */***

Then assign NC tag

Else assign VR tag

7.3.1 The Proposed Algorithm

Algorithm used for this tagging is as follows:

Step 1: Input the Manipuri text.

Step 2: Tokenize the input text.

Step 3: If the word is in the form of affixation, derivation and compounding then feed the word to segmenter for splitting and checks the word with the lexicon for a match.

Step 4: If match is found the word is properly tagged by the tagger.

Step 5: If match is not found or multiple tags exists for a single word then tagger tagged the word by using rules.

Step 6: Repeat step 4 and 5 till the end of the input text.

Step 7: Returned the tagged output text.

Step 8: Extract those unknown new words from the tagged output.

Step 9: Make the new entry for the unknown new word to the lexicon.

Step 10: Add the new rules for newly entered words.

7.3.2 Graphical User Interface Tool

A Graphical User Interface tool named "POSTIM" has been developed by using NetBeans IDE 7.3, JDK 6 and JRE 6. The front-end of the tool has been implemented in java and its interface is connected with a text file of Manipuri lexical items called "lexicon" as the back-end. The selection of textual database is for simplicity and to extend support for multiple platforms without the need of the installation of any DBMS server like MYSQL etc. by the end user. Each lexical item entry in "lexicon" file has two fields:

ITEM: Manipuri lexical item like "লাইরিক" or "Lairik" i.e., "book".

CATEGORY: The morphosyntactic category of the lexical item like NC (Common Noun).

The Manipuri words are entered into the lexicon using Kalpurush font of Bengali-Script. A screenshot view of the tool is shown as below.

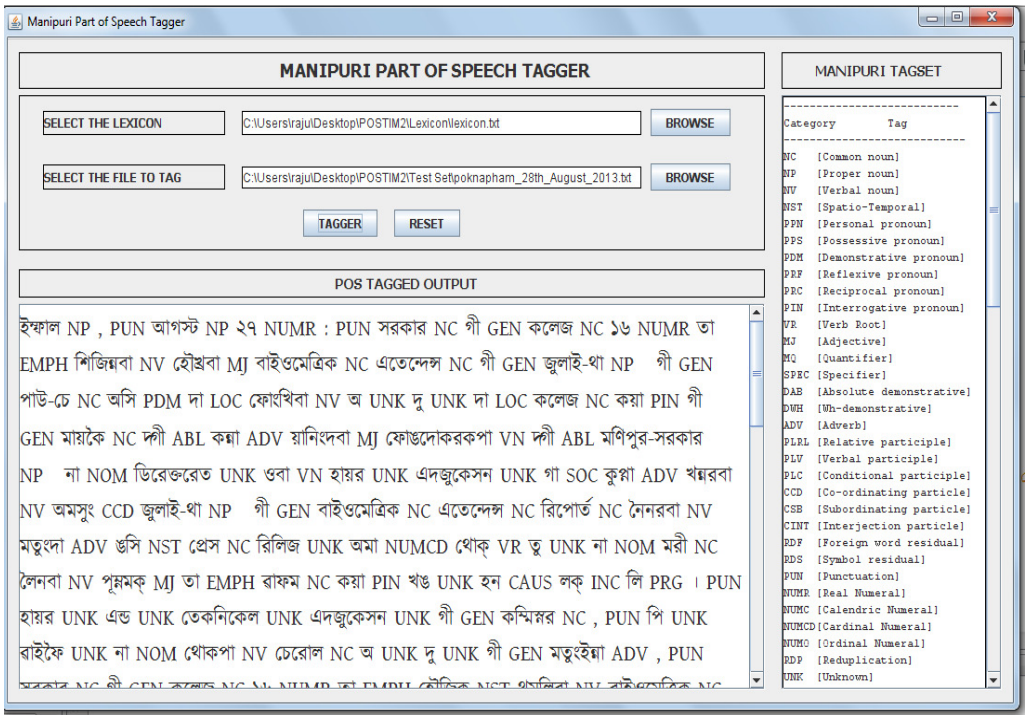

Figure 7.4 Manipuri POS Tagger GUI tool "POSTIM"

7.3.3 Experimental Results and Discussions

In order to measure the performance of the system three lexicons are prepared based on three domains viz., government/politics, sports and tourism which are the main contents of the daily news paper. Each lexicon contains 300 morphemes with their part of speech categories which are collected from English-Manipuri, Manipuri-English bilingual dictionaries which are available in hard copy format. Accuracy is tested by using three test sets of three different domains i.e., government/politics, sports and tourism. Each of which consisting of 2500 words collected from two Manipuri leading daily news papers viz., "POKNAPHAM" (www.poknapham.in) and "HUEIYEN LANPAO" (www.hueiyenlanpao.com). Accuracy percentage of the tagger is calculated using the formula given below:

$$\text{Accuracy Percentage} = \frac{\text{Correctly tagged words}}{\text{No. of words in evaluation set}} \text{ X } 100$$

The performance of the tagger is tested by using a lexicon with fixed number of words and different rules. A summary of tests and results of the tagger with graph is given below:

Table 7.1: Experimental Results of Government/Politics data set

Size of Lexicon (in words)	No. of rules applied	No. of words in test set	No. of correctly tagged words	Accuracy
300	10	2500	516	16%
300	15	2500	725	29%
300	20	2500	945	37.8%
300	25	2500	1150	46%
300	30	2500	1350	54%
300	35	2500	1680	67.2%
300	40	2500	1950	78%
300	45	2500	2325	93%

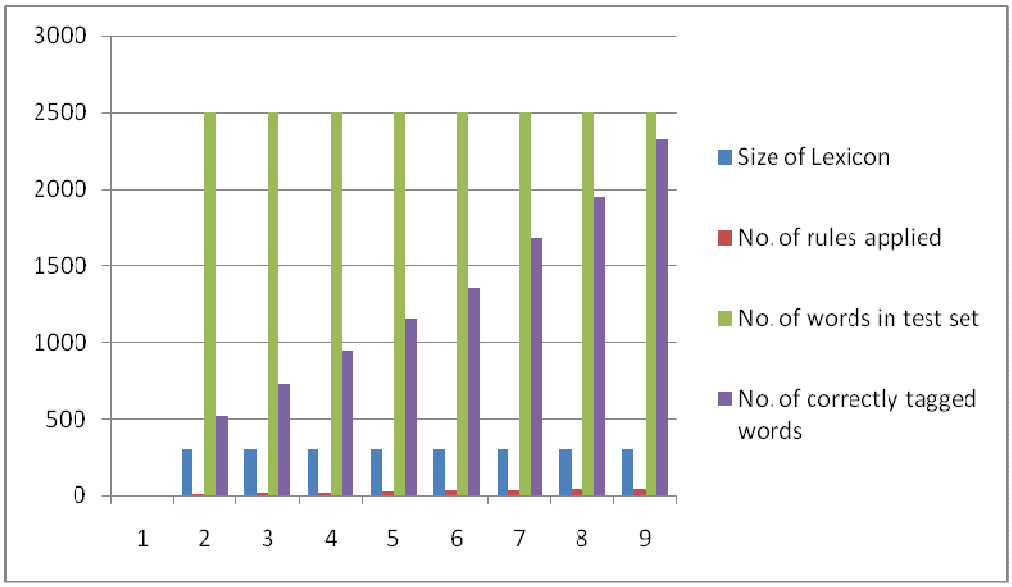

Figure 7.5 Chart View Results of Government/Politics data set

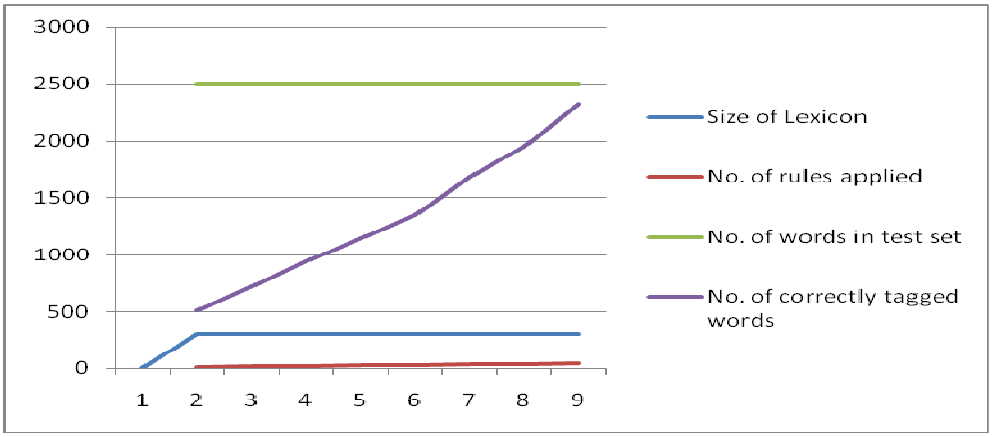

Figure 7.6 Line View Results of Government/Politics data set

Table 7.2: Experimental Results of Sports data set

Size of Lexicon (in words)	No. of rules applied	No. of words in test set	No. of correctly tagged words	Accuracy
300	10	2500	475	19%
300	15	2500	712	28.48%
300	20	2500	912	36.48%
300	25	2500	1100	44%
300	30	2500	1450	58%
300	35	2500	1690	67.6%
300	40	2500	1995	79.8%
300	45	2500	2275	91%

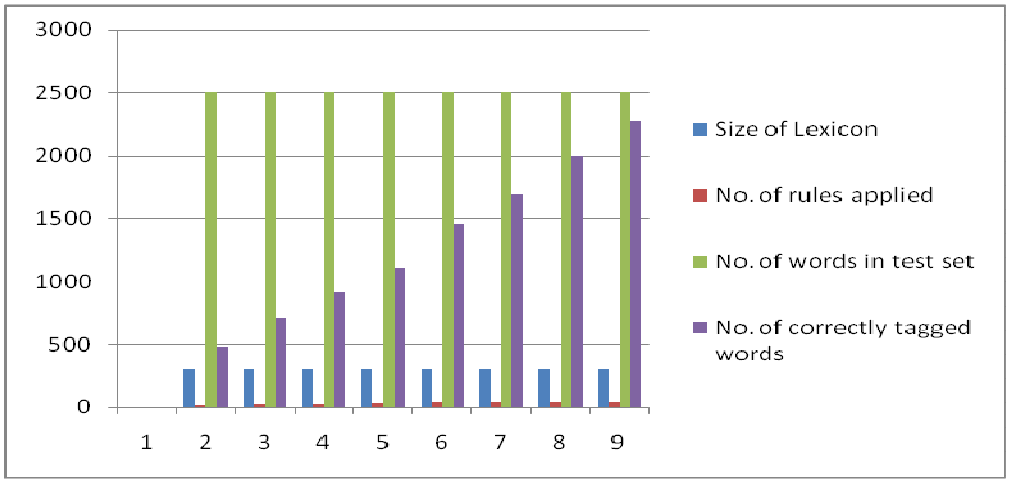

Figure 7.7 Chart View Results of Sports data set

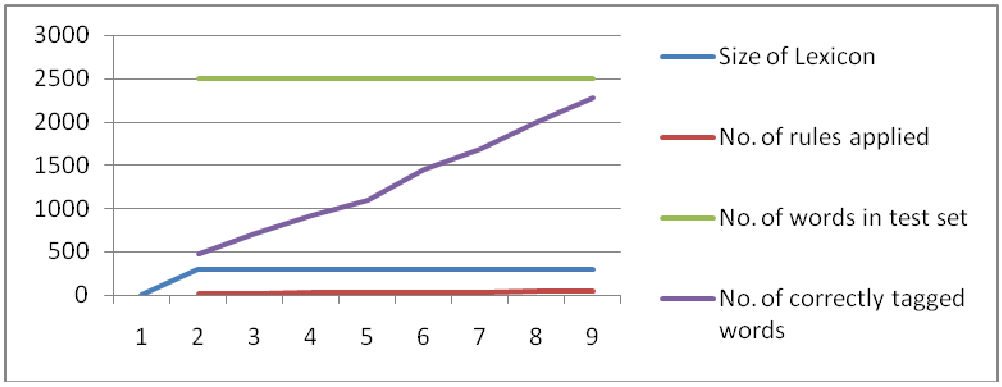

Figure 7.8 Line View Results of Sports data set

Table 7.3: Experimental Results of Tourism data set

Size of Lexicon (in words)	No. of rules applied	No. of words in test set	No. of correctly tagged words	Accuracy
300	10	2500	490	19.6%
300	15	2500	650	26%
300	20	2500	820	32.8%
300	25	2500	1250	50%
300	30	2500	1460	58.4%
300	35	2500	1650	66%
300	40	2500	1985	79.4%
300	45	2500	2300	92%

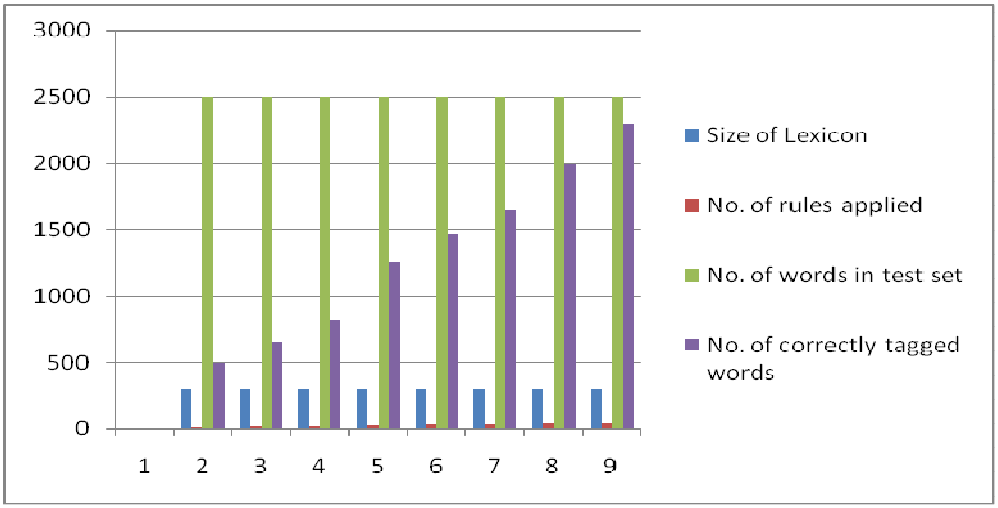

Figure 7.9 Chart View Results of Tourism data

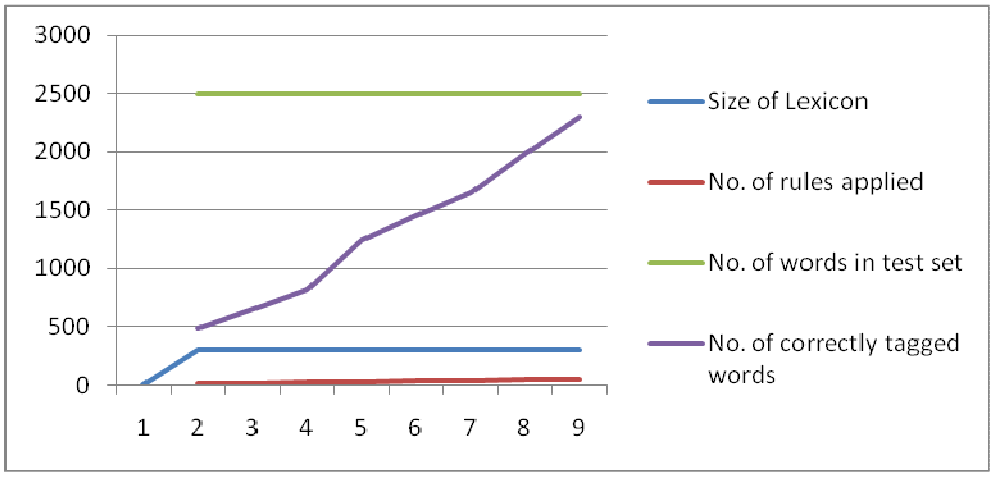

Figure 7.10 Line View Results of Tourism data set

It gives the average accuracy of 92% and it is clear that accuracy percentage is increased with the increment of the number of rules applied.

7.4 Chapter Summary

This chapter presents an overview of rule-based part of speech tagging and its related works. It then presents the proposed system design of rule based part of speech tagger of Manipuri and its different modules. It also presents the proposed algorithm of rule based part of speech tagger of Manipuri and features of the graphical user interface POS tagger tool "POSTIM" which is developed by applying Manipuri linguistics rules. Further, the chapter presents experimental results generated by POSTIM using three test sets of three different domains viz., government/politics, sports and tourism.