

2008년 8월

석사학위 논문

네팔어를 위한 비감독 품사 태깅 방법

조선대학교 대학원

컴퓨터공학과

Gautam Dipesh

The Method of Unsupervised POS Tagging for Nepali Language Text

네팔어를 위한 비감독 품사 태깅 방법

2008년 8월

조선대학교 대학원

컴퓨터공학과

Gautam Dipesh

The Method of Unsupervised POS Tagging for Nepali Language Text

지도교수 김 판 구

이 논문을 공학석사학위신청 논문으로 제출함.

2008년 8월

조선대학교 대학원

컴퓨터공학과

Gautam Dipesh

디페스 거우팀의 공학석사논문을 인준함

위원장 조선대학교 교수 _____

위 원 조선대학교 교수 _____

위 원 조선대학교 교수 _____

2008년 8월

조선대학교 대학원

CONTENTS

ABSTRACT	ii
LIST OF FIGURES	iv
LIST OF TABLES	v
I. INTRODUCTION	1
II. BACKGROUND CONCEPTS	4
A. TEXT CORPUS	4
B. LEXICON	4
C. PARTS OF SPEECH TAGGING	6
D. WORD SENSE DISAMBIGUATION	6
E. PROXIMITY MEASURE AND CLUSTERS	6
1. <i>Similarity Measure</i>	7
2. <i>Dissimilarity Measure</i>	8
III. RESEARCHES ON NEPALI LANGUAGE	9
A. CURRENT STATE OF NEPALI LEXICON	9
B. NELRALEC PROJECT	9
C. THE NELRALEC TAGSET	11
D. CONTEMPORARY NEPALI DICTIONARY	12
E. DOBHASE (TRANSLATOR)	13
IV. TAGGING NEPALI LANGUAGE TEXT	14
A. OVERVIEW	14
B. TEXT MANIPULATION TOOLS	15
C. TEXT COLLECTION	17
D. WORD DICTIONARY	18
E. BIGRAM AND BIGRAM DICTIONARY	18
F. FEATURE GENERATION AND CLUSTERING	20
1. <i>Feature Generation</i>	20
2. <i>Pre-Classification of Pronouns, Particles and Verbs</i>	22
3. <i>Clustering Feature Vectors</i>	23
V. EXPERIMENTAL RESULTS	25
VI. CONCLUSION AND FUTURE WORKS	29
REFERENCES	31

ABSTRACT

The Method of Unsupervised POS Tagging For Nepali Language Text

Gautam Dipesh

Advisor: Prof. Pankoo Kim, Ph. D.

Department of Computer Science,

Graduate School of Chosun University

Parts of Speech (POS) tagging is also known as morphosyntactic categorization or syntactic word class tagging [1]. Most of the tagging tasks rely on large collection of training corpus. But availability of pre-tagged training corpus is the major constraint for tagging natural language text. There has been developed various corpora like American National Corpus [30], Bank of English [31], British National Corpus [32], Helsinki Corpus [33] etc. Even though most of the world's literature languages like Nepali has large collection of lexical dictionaries and encyclopedia, but still the electronic corpus are not available for many rich languages. So some method which helps in tagging from scratch should be proposed.

In this thesis we propose a method of POS tagging of Nepali language text. In the first step we manually constructed pronoun and particle lexicons as they are in a small number in Nepali text. We then use these lexicons to pre-classify

pronoun, particle and last occurring verb as every sentence ends either with verb or particle. In the next step we use co-occurring word statistics as the feature for words clustering as different study [14] [15] [16] on natural languages suggests that co-occurring words convey important information for the processing of the language. For feature generation several most frequent words from large collection of news paper article are selected as dimensions in vector space. The components of feature vector of each word are the number of times each dimension word occurs to the left and right of the word. These vectors are clustered to group the words in collection into several syntactic categories and the POS tags from NELRALEC [19] are assigned to each cluster. During tagging several corresponding honorific, gender or other agreement specific tags unlike suggested by NELRALEC are considered to be in same syntactic category for the purpose of this thesis thus reducing the number of POS tags. Finally the performance is evaluated by precision and recall value of the result. Despite the moderate performance as a result of several error sources, this research is noble in the POS tagging of Nepali language text as few researches in the area have been performed.

List of Figures

Figure 1. Contemporary Nepali Dictionary

Figure 2. Translator with translation example

Figure 3. System Architecture of Text Analysis Tool

Figure 4. Text Manipulation Tool

Figure 5. Sample Text Collection transformed to XML

Figure 6. Sample Word Dictionary

Figure 7. Sample Bigram

Figure 8. Bigram Dictionary

Figure 9. Word frequency and frequency plot

List of Tables

Table 1. Different forms of English and corresponding Nepali lexeme “*do*”

Table 2. Different Lexemes with same words

Table 3. Portion of NELRALEC tagset

Table 4. Gender specific Verbs in Nepali Language

Table 5. Bigram From Example Text

Table 6. Classification of words to POS Tags

Table 7. Precision and Recall of Tags

I. Introduction

The underlying mechanism for human to understand and interpret natural language is very much unknown if not quite unknown. Most dull human mind can understand and establish the semantics of spoken language. But computer is unable to understand even a very simple sentence without proper adjustment. Interpreting the language and deriving new conclusion from the input fed to it is quite impossible for a computer unless the use of proper algorithm.

Natural Language Processing is one of the most interested fields of artificial intelligence for researchers. Much work has been done in this field for many languages like English, French, German and Spanish etc. One of the most important task of NLP is Parts Of Speech (POS) tagging in which words in a text are categorized to the syntactic categories depending upon the context they are used. In the following chapter we will briefly discuss about POS tagging and some other aspects of NLP.

Different POS taggers exist to date. Taggers using bigram and trigram models[6][7] require large tagged training data. Brill[8] has introduced transformation-based tagging which also requires pre-tagged text as training data.

Hidden Markov Models [9][10] though require no pretagged text, require lexicon that specifies the possible parts of speech for each word.

Ratnaparkhi[11] proposed maximum entropy model for tagging new words not present in corpus.

Brill et al [12] infers grammatical categories of words from bigram statistics. Finch and Chater [13] uses vector models in which words are clustered according to the similarity of their close neighbors in a corpus.

All these taggers either require large corpora or lexicon as training data which is not always available for all languages.

A few researches in the filed of NLP of Nepali language text have been done to date. NELRALEC [19] has proposed tag sets for Nepali language text. Statistical text analyzer tool for Hindi which could be adapted to other languages like Nepali language was proposed by Sunita Arora, et al [17]. But these researches haven't released any corpus for Nepali language as they are in development phase.

Schutze[14][15] proposed the method of POS tagging using statistical distribution of co-occurring words which suggests more the two words share the set of words to the left and right the more they belong to the same syntactic category. Another method proposed by Chris Biemann[16] proposed unsupervised POS tagging using Chinese Whispers[18] graph clustering algorithm which was capable of finding number of classes in unguided way.

In this thesis we propose a method of tagging Nepali language text where the pre-tagged text is not available. Because Nepali language has few pronouns and the particles, we constructed pronoun and particle lexicons manually which previous works haven't considered. Before the text collection is subjected to vector clustering algorithm, the pronouns, particles and verbs are pre-classified by lexicon lookup, which is described in the following chapter. As we pre-classify some words, we expect high precision of particle and high recall of pronoun, particle and verb which is evident from the precision and recall value in the performance evaluation (verb recall = 0.72, pronoun recall = 0.74, particle precision = 0.73 and particle recall = 0.80).

We collected large collection of news paper article from two online news papers "Kantipur Daily" and "Himal Khabar" for the purpose of experiment. The Unicode text is extracted from the webpage by the method of HTML tag filtering using regular expressions. The filtered text is tokenized and converted

to XML document in the subsequent phases. Some most frequent words in the text collection forms the *dimension words* for the vector dimension space and the frequency count of bigrams constructed from the sentences are used to obtain the feature vectors and these feature vectors are subjected to clustering to obtain the word cluster. The method of feature vector construction and clustering is described in the following chapters. Several corresponding honorific, gender or other agreement specific tags are considered to be in same syntactic category for the purpose of this thesis thus reducing the number of POS tags.

II. Background Concepts

Before going through the detail of theory, we will present some of the fundamental concepts of natural language processing. Understanding text refers multi-aspect of text processing like measurement of semantic relationship between words or the syntactic structure of the sentence. The following subsection discuss about the aspect of semantic relationship measurement and syntactic analysis of natural language text.

A. Text Corpus

A corpus is a large and structured set of texts that are generally stored electronically and processed. The corpus are used for statistical analysis, or mining linguistic rules on specific domain. Corpus may be the texts of single language in case of monolingual corpus or in multiple languages in case of multilingual corpus. Special multilingual corpora formatted for one to one comparison of each word is called aligned parallel corpora. Corpora is not just a mere structured text, rather the process known as annotation by which the text's properties are defined is one of the important aspect of corpus to make it useful for linguistic research. POS tagging is one of such annotation. This research paper considers annotation of Nepali language text from scratch as no corpus is not available to use for the research.

B. Lexicon

Lexicon in linguistic is a vocabulary of a language which includes lexemes [21]. Lexemes are the most fundamental form of language which actualizes

words. In other words a lexeme corresponds to set of words that are different forms of same word.

Table 1. Different forms of English and corresponding Nepali lexeme “do”

Words	English Meaning	Pronunciation
गर्नु	do	garnu
गर्छ	does	garcha
गरेको	done	gareko cha
गर्‍यो	did	garyo
गर्दै छ	doing	gardai cha

Same lexeme may constitute different strings of words or different lexemes may constitute same word, which is shown in Table 1 and Table 2.

Table 2. Different Lexemes with same words

words	English Meaning	Pronunciation
धार	stream	dhaar
	edge	
कर	tax	kar
	compel	
कर्म	work	karma
	fate	
सुन	listen	soon
	gold	

Lexicon organizes language vocabulary according to certain principles (for instance, all verbs of motion may be linked in a lexical network) and a generative device produces new words according to certain lexical rules. For example, the prefix and suffix added to words [21].

C. Parts of Speech Tagging

Parts of Speech tagging is also known as morphosyntactic categorization or syntactic word class tagging [1]. The basic task of POS analysis is to assign each word of a text to the appropriate morphosyntactic category viz. noun, verb, adjective, and adverb etc. The tagging of corpus is manual task by the experts. These manually tagged corpus provide basic knowledge base for automatic tagger which serves as the most important tools for tagging texts. Different POS taggers have been developed to date. However Tagged corpus for several literarily rich languages is not available, in such case some statistical properties of language could serve for bootstrapping method.

D. Word Sense Disambiguation

Ambiguity occurs in natural language when a word represents more than one sense. From the viewpoint of meaning the sense represents different meaning and from the viewpoint of POS tag, different POS is represented by the word. Automatic disambiguation of ambiguous words is an important part of natural language text analysis. Several disambiguation algorithms exist depending on the context of disambiguation [4] [5].

E. Proximity Measure and Clusters

Clusters are the groups of similar entities. More intuitively, if vectors representing entities are viewed as points in l -dimensional space, then clusters are described as “continuous regions of this space containing a relatively high

density of points, separated from other high density regions by regions of relatively low density of points [2].

The Proximity between two vectors evaluates how close two vectors are located in the l -dimensional space. Broadly speaking, there are two types of proximity measures viz. Dissimilarity Measure (DM) and Similarity Measure (SM) [2].

1. Similarity Measure

A Similarity Measure (SM) s on X is a function

$$s : X \times X \rightarrow R$$

Where R is a set of real number, such that

$$\exists s_0 \in R : -\infty < s(x, y) \leq s_0 < +\infty, \quad \forall x, y \in X$$

$$s(x, x) = s_0, \quad \forall x \in X$$

and $s(x, y) = s(y, x), \quad \forall x, y \in X$

If in addition $s(x, y) = s_0, \quad \text{if and only if } x = y$

and $s(x, y)s(y, z) \leq [s(x, y) + s(y, z)]s(x, z), \quad \forall x, y, z \in X$

2. Dissimilarity Measure

A Dissimilarity Measure (DM) d on X is a function

$$d : X \times X \rightarrow R$$

where R is a set of real number, such that

$$\exists d_0 \in R : -\infty < d_0 \leq d(x, y) < +\infty, \quad \forall x, y \in X$$

$$d(x, x) = d_0, \quad \forall x \in X$$

and $d(x, y) = d(y, x), \quad \forall x, y \in X$

If in addition $d(x, y) = d_0, \quad \text{if and only if } x = y$

and $d(x, z) \leq d(x, y) + d(y, z), \quad \forall x, y, z \in X$

We use Euclidean distance dissimilarity [2] given by (1) for l -dimensional vector clustering.

$$d_2(x, y) = \sqrt{\sum_{i=1}^l (x_i - y_i)^2} \quad (1)$$

III. Researches on Nepali Language

Few years back some pioneer organizations have been involving in the linguistic research of Nepali language. To the date small taggers, parsers, language translators have been developed. But these works are in the beginning phase so not available for commercial or for research purposes. In this chapter we will briefly discuss on researches status of Nepali Language.

A. Current State of Nepali Lexicon

Nepali language is the descendent of world's oldest literary language [22]. Sanskrit has rich tradition of poetry, drama, scientific, technical, philosophical and Hindu religious texts. As being the descendent of Sanskrit, Nepali language is rich in literature, drama, technical and philosophical texts. At present there are a large collection of paper based dictionaries for Nepali language, but electronic version of Nepali lexicon doesn't exist till date [20]. However some pioneer work has been done in digitization and annotation of Nepali language text.

B. NELRALEC Project

The NELRALEC project (Nepali Language Resources and Localization for Education and Communication) is a three-year research project funded by the EU Asia IT&C committee, known in Nepali as Bhasha Sanchar [23]. The project seeks to address a variety of needs in terms of computational support for the Nepali language, ranging from text-to-speech software and a localised operating system to educational structures and language resources to support

the development of corpus and computational linguistics in Nepal, through the implementation of new corpus-based lexicography techniques in a new, empirical Nepali dictionary. Various partners like Madan Puraskar Pustakalaya, Central Department of Linguistics at Tribhuvan University from Nepal, and Lancaster University UK¹[24], Göteborg University Sweden and the European Languages Resource Association (ELRA) France from Europe [25] have been working.

At present NELRALEC has proposed tagset consisting of 112 tags for manual and automated analysis of morphosyntactic units of Nepali [19].

Madan Puraskar Pustakalaya [26] has been developing Nepali national corpus. With the help of this national corpus, frequency count of Nepali words in different context, collocation and concordance of word analysis will be possible [20]. During the initial development of Corpus, the lexical entries are selected manually from the dictionaries. At present about 11,000 entries has been updated in the lexicon [20].

¹ As a partner in the Nelralec project, Lancaster has contributed expertise in the areas of corpus design, encoding, annotation and analysis. Part of our ongoing co-operation with the partners in Nepal has been to develop a framework for part-of-speech tagging in Nepali.

C. The NELRALEC Tagset

The NELRALEC tagset was developed by a team of linguists from Tribhuvan University and Lancaster University. Subset of tags from 112 tags proposed by NELRALEC is shown in table 3.

Table 3. Portion of NELRALEC Tagset

Category Defined	Examples (Latin)	Examples (Devanagari)	Tag
Common Noun	keTo, keTaa, kalam	केटो, केटा, कलम	NN
Proper Noun	raam	राम	NP
Masculine Adjective	moTo, raamro	मोटो, राम्रो	JM
Feminine Adjective	moTii, raamrii	मोटी, राम्री	JF
Feminine third person medial-honorific singular verb	garina, garii, che, garthii	गरिन, गरी, छे, गर्थी	VVYN1F

D. Contemporary Nepali Dictionary

Recently Bhasha Sanchar has released online “Contemporary Nepali Dictionary” with 8000 words [27].

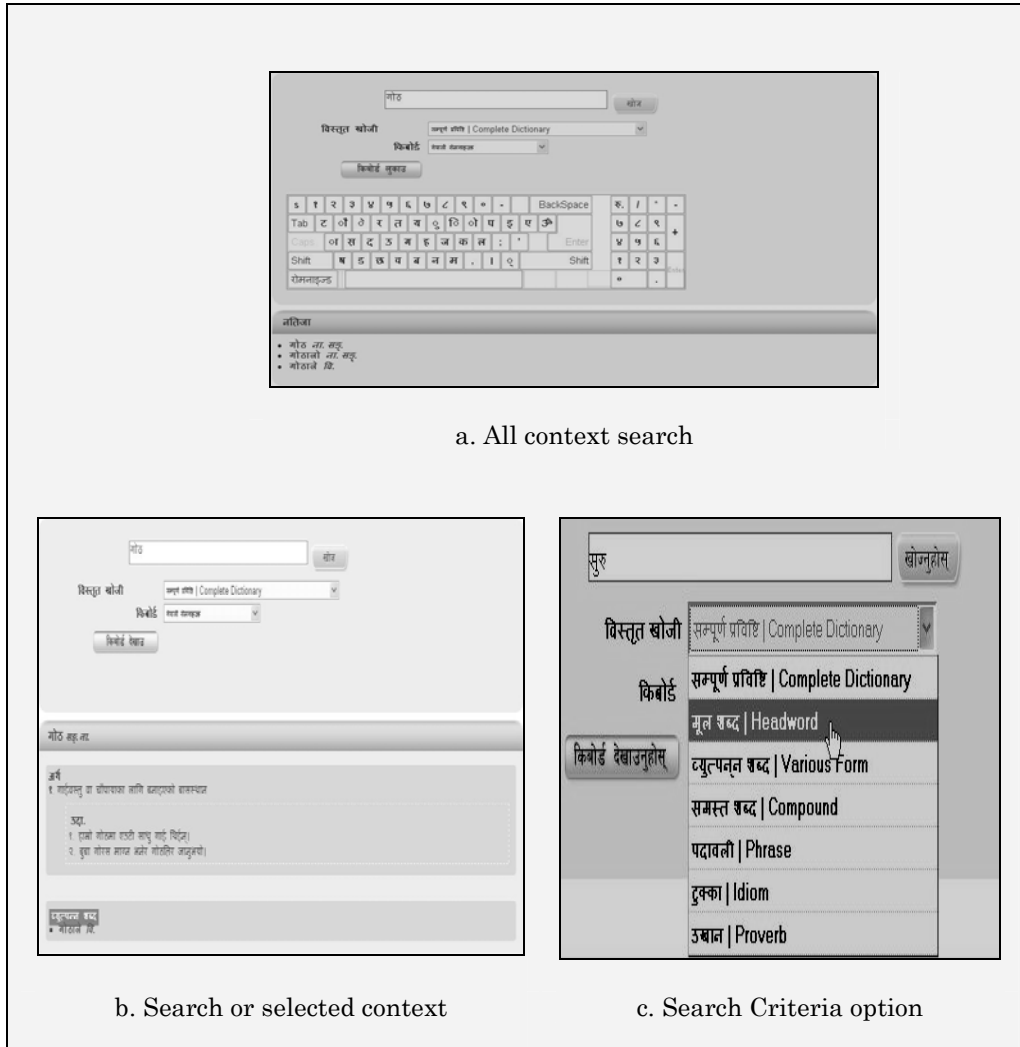


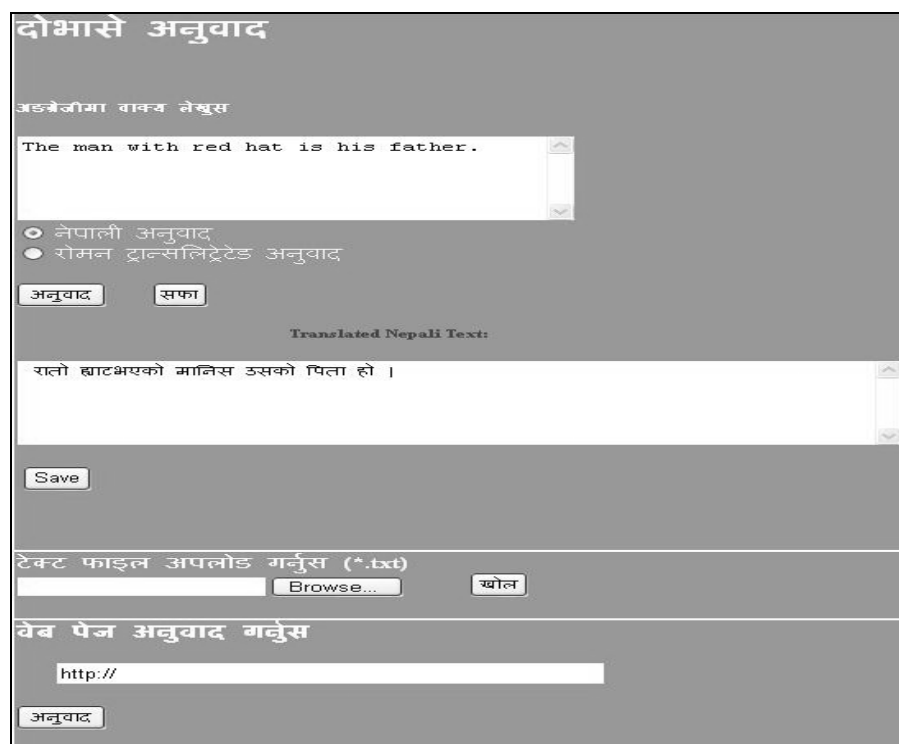
Figure 1. Contemporary Nepali Dictionary

This online dictionary has ability to search word’s meaning with the context as noun, verb, adjective etc. It also provides the options of searching criteria for

complete dictionary, various forms, and compound words and so on. The sample search criteria option is shown in Fig 1 c.

E. Dobhase (Translator)

It is another noble project in machine translation for Nepali Language developed by Language Processing Research Unit (LPRU) [28], Kathmandu University [29]. The grammar of the system consists of 22, 000 words. It is online translation system which translates English sentences to Nepali sentences. It has capability to translate website and also the text file. Figure 2. shows the web interface of the translator.



The screenshot displays the web interface of the Dobhase Translator. At the top, the title "दोभासे अनुवाद" (Dobhase Anuvad) is visible. Below it, the text "अङ्ग्रेजीमा वाक्य लेख्नुस" (Write sentence in English) is shown. A text input field contains the English sentence: "The man with red hat is his father." Below the input field, there are two radio buttons: "नेपाली अनुवाद" (Nepali translation) and "रोमन ट्रान्सलिटेरेड अनुवाद" (Roman transliterated translation). The "नेपाली अनुवाद" option is selected. There are two buttons: "अनुवाद" (Translate) and "सफा" (Clear). Below these buttons, the text "Translated Nepali Text:" is displayed. A text output field shows the translated Nepali sentence: "रातो ह्याटभएको मानिस उसको पिता हो ।" Below the output field, there is a "Save" button. At the bottom of the interface, there are two sections: "टेक्ट फाइल अपलोड गर्नुस (*.txt)" (Upload text file) and "वेब पेज अनुवाद गर्नुस" (Translate web page). The "टेक्ट फाइल अपलोड गर्नुस" section has a text input field and a "Browse..." button. The "वेब पेज अनुवाद गर्नुस" section has a text input field with "http://" and a "अनुवाद" (Translate) button.

Figure 2. Translator with translation example

IV. Tagging Nepali Language Text

A. Overview

The goal of this thesis is to identify the parts of speech tag of each word in a text collection. The lack of pre-tagged text in Nepali Language text motivated towards the research for different approach which could be employed to tag the text from scratch. So we devised statistical distributional approach to cluster each token. NELRALEC [19] has proposed POS tag sets consisting of 112 tags for Nepali Language.

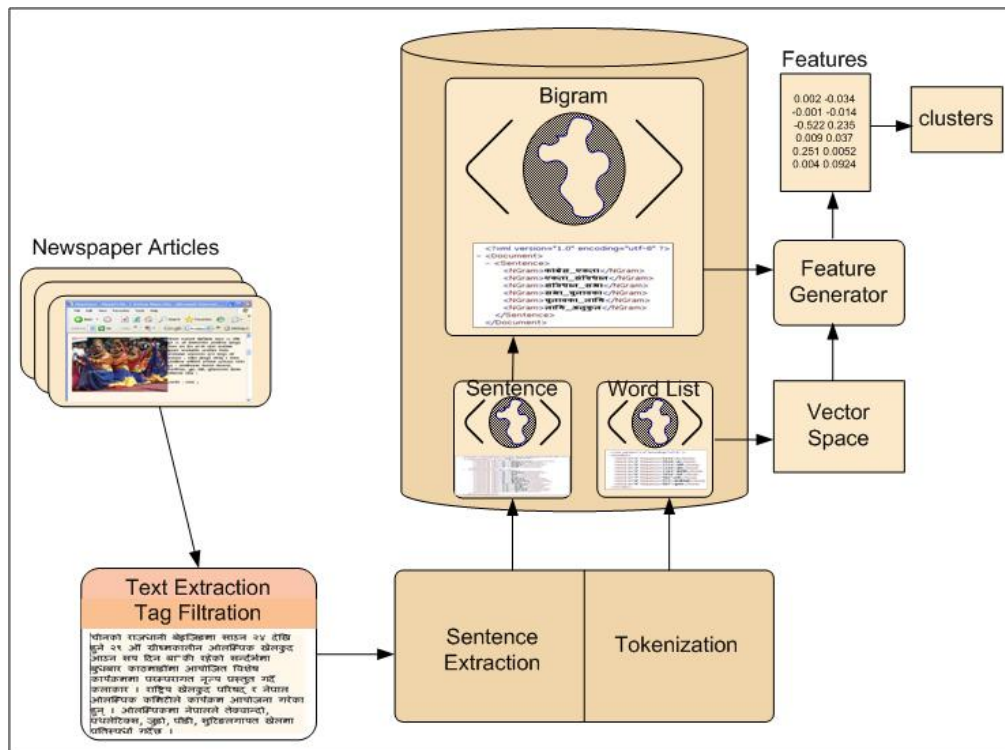


Figure 3. System Architecture of Text Analysis Tool

Verbs, adjectives, pronouns or nouns in Nepali language text are gender specific or they have some inflections. Some lexicon should be constructed beforehand to handle such situation. However instead of considering such

words as separate group, we group them together without considering gender. The Table 4 shows the example of gender specific verbs.

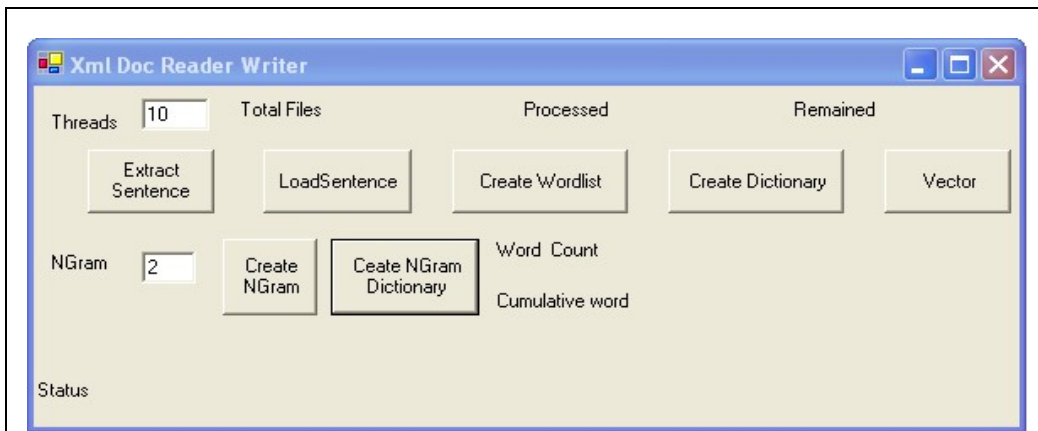
Table 4. Gender specific Verbs in Nepali Language

Masculine	Feminine	Meaning
गऱ्यो, garyo	गऱी, gari	did
हिँड्यो, hidyo	हिँडी, hidi	walked

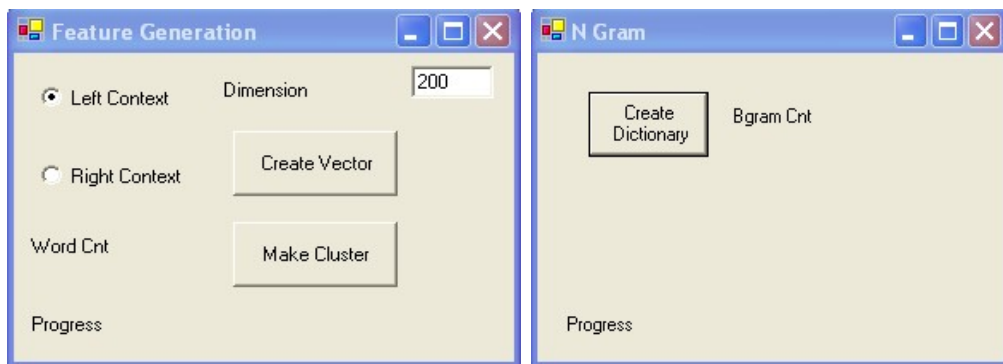
For the purpose of research we collected large collection of Nepali language text from two online newspapers “Kantipur Daily” and “Himal Khabar.” Sentences are extracted from these texts and then tokenized and finally bigrams are extracted from the sentences. Feature vectors of each token are constructed by counting co-occurring words in left and right of each token. Then these tokens are clustered using their feature vectors. Figure 3 shows the system architecture. The detail of feature vectors construction and clustering are described in following sections.

B. Text Manipulation Tools

As the source of Nepali language text is web document, the plain text should be extracted beforehand of further processing. we developed a special tool for text manipulation which extracts plain text sentences from the web document and tokenizes and constructs bigram dictionary. The tool is developed in C#.NET. The choice of C# is obvious for the reason that HTML tag filtering is easier by using regular expressions. Further, implementation of thread is comparatively easier. Another major advantage of .NET is it provides efficient library for manipulation of XML documents. The snapshot of tool is shown in Figure 4.



a. Control Panel



b. Feature Generator

b. Bigram Dictionary Creator

Figure 4. Text Manipulation Tool

C. Text Collection

we selected two daily newspapers for the source of large collection of text. “Kantipur Daily”, one of the Nepal’s a large volume daily newspaper provided 64013 long and short documents. Similarly “Himal Khabar” an online newspaper provided 707 documents.

```
<?xml version="1.0" encoding="utf-8" ?>
- <Document id="465.html">
  - <Sentence id="1">
    <Word id="1">सातदलने</Word>
    <Word id="2">सात</Word>
    <Word id="3">ठाउँमा</Word>
    <Word id="4">सभा</Word>
    <Word id="5">गर्ने</Word>
  </Sentence>
  - <Sentence id="2">
    <Word id="1">सात</Word>
    <Word id="2">दलको</Word>
    <Word id="3">पुष</Word>
    <Word id="4">२०</Word>
    <Word id="5">गते</Word>
    <Word id="6">बसेको</Word>
    <Word id="7">बैठकले</Word>
    <Word id="8">मुलुकको</Word>
    <Word id="9">सात</Word>
    <Word id="10">ठाउँमा</Word>
    <Word id="11">सभा</Word>
    <Word id="12">गर्ने</Word>
    <Word id="13">निर्णय</Word>
    <Word id="14">गरेको</Word>
    <Word id="15">छ</Word>
  </Sentence>
</Document>
```

Figure 5. Sample Text Collection transformed to XML

The text manipulation tool first extracts sentences from the web documents. The extracted sentence is then transformed into XML document with individual tokens separated within each sentence element. In Figure 5, sample transformed document is displayed. The root node of XML document is “Document” with filename as “id” attribute. Each “Sentence” node has child nodes to represent word tokens.

D. Word Dictionary

Words from all the documents are grouped in a single XML document. The frequency of occurrence of each word is represented by “*frequency*” attribute of “*Word*” node. The tokens are arranged in the order of decreasing frequency. The sample dictionary is shown in Figure 6.

```
<?xml version="1.0" encoding="utf-8" ?>
- <Wordlist>
  <Word id="1" frequency="2141">र</Word>
  <Word id="2" frequency="2010">रु</Word>
  <Word id="3" frequency="1714">पनि</Word>
  <Word id="4" frequency="1133">रुन</Word>
  <Word id="5" frequency="1104">भएको</Word>
  <Word id="6" frequency="1045">गते</Word>
  <Word id="7" frequency="984">तथा</Word>
  <Word id="8" frequency="972">माओवादी</Word>
  <Word id="9" frequency="887">चुनाव</Word>
</Wordlist>
```

Figure 6. Sample Word Dictionary

E. Bigram and Bigram Dictionary

```
<?xml version="1.0" encoding="utf-8" ?>
- <Document>
  - <Sentence>
    <NGram>कांग्रेस_एकता</NGram>
    <NGram>एकता_संविधान</NGram>
    <NGram>संविधान_सभा</NGram>
    <NGram>सभा_चुनावका</NGram>
    <NGram>चुनावका_लागि</NGram>
    <NGram>लागि_अनुकूल</NGram>
  </Sentence>
</Document>
```

Figure 7. Sample Bigram

we extracted bigram from each sentence and transformed to XML file. We again created bigram dictionary with the frequency count of each bigram.

```
<?xml version="1.0" encoding="utf-8" ?>
- <NGramList>
  <NGram id="1" frequency="353">नेपाली_कांग्रेस</NGram>
  <NGram id="2" frequency="305">बताएका_छन्</NGram>
  <NGram id="3" frequency="264">सात_दलको</NGram>
  <NGram id="4" frequency="255">भएको_छ</NGram>
  <NGram id="5" frequency="251">गरेको_छ</NGram>
  <NGram id="6" frequency="243">संविधानसभा_चुनाव</NGram>
  <NGram id="7" frequency="187">उल्लेख_गरे</NGram>
  <NGram id="8" frequency="145">प्रधानमन्त्री_गिरिजाप्रसाद</NGram>
  <NGram id="9" frequency="144">माओवादी_अध्यक्ष</NGram>
  <NGram id="10" frequency="134">भनाइ_गियो</NGram>
  <NGram id="11" frequency="132">गरेका_छन्</NGram>
  <NGram id="12" frequency="131">संविधान_सभाको</NGram>
  <NGram id="13" frequency="119">जानकारी_दिए</NGram>
</NGramList>
```

Figure 8. Bigram Dictionary

Bigram Dictionary is also sorted in ascending order of occurring frequency. Figure 7 and 8 show sample bigram and bigram dictionary respectively.

F. Feature Generation and Clustering

1. Feature Generation

The co-occurring frequency of words to the left and right of token is the basis of feature generation in this research. Suppose for example the document consists of 4 sentences as shown by following lines. The Capitalized bold “D’s” emphasizes the dimension words of vector space (in our collection, 200 most frequent words are considered as dimension words).

abc**D**₄de**D**₂fg.

pqr**D**₂se**D**₂ft.

mno**D**₁xe**D**₃yz.

aqa**D**₄aa**D**₄sv.

Table 5. Bigram From Example Text

Bigram	Frequency	Bigram	Frequency	Bigram	Frequency	Bigram	Frequency
a_b	1	p_q	1	m_n	1	a_q	1
b_c	1	q_r	1	n_o	1	q_a	1
c_ D ₄	1	r_ D ₂	1	o_ D ₁	1	a_ D ₄	1
D ₄ _d	1	D ₂ _s	1	D ₁ _x	1	D ₄ _a	1
d_e	1	s_e	1	x_e	1	a_a	1
e_ D ₂	1	e_ D ₂	1	e_ D ₃	1	a_ D ₄	1
D ₂ _f	1	D ₂ _f	1	D ₃ _y	1	D ₄ _s	1
f_g	1	f_t	1	y_z	1	s_v	1

Thus following the intuition suggested by Schutze [14], the vector dimension for the example document is as follows

$$[\mathbf{D}_1 \mathbf{D}_2 \mathbf{D}_3 \mathbf{D}_4]$$

From the bigram in Table 5, the number of occurrence of each bigram containing the dimension words (shaded in the table) gives the vector

component in each dimension. Let's consider the left context vectors of words "a" and "b" in the document as

$$\mathbf{a}: [0 \ 0 \ 0 \ 1]$$

$$\mathbf{b}: [0 \ 0 \ 0 \ 0]$$

Similarly right context vectors for "a" and "b" are

$$\mathbf{a}: [0 \ 0 \ 0 \ 2]$$

$$\mathbf{b}: [0 \ 0 \ 0 \ 0]$$

Concatenating the left context vector and right context vector [14] [15] and arranging in a matrix, gives the feature matrix of the document whose rows represent the feature vectors of words. As a result we can obtain following feature matrix of the document.

$$A = \begin{matrix} a \\ b \\ \text{---} \\ \text{---} \\ \text{---} \end{matrix} \begin{bmatrix} 00010002 \\ 00000000 \\ \text{-----} \\ \text{-----} \\ \text{-----} \end{bmatrix}$$

One of the drawbacks of such feature matrix is its sparseness, i.e. most of the entries are 0's. The Singular Value Decomposition (SVD) of the feature matrix avoids this drawback [3] by concentrating the effect of many dimensions vector space to few dimensions. Thus SVD has two fold of advantages. First it generalizes the sparse matrix to some few dimensions (concept used in Latent Semantic Indexing in Information Retrieval), and second by reducing higher dimension matrix to approximate lower dimension thus avoiding the ill effect of sparse entries. The feature matrix **A** after SVD re represented as three factor matrix as in (2).

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T \quad (2)$$

Where orthogonal matrix \mathbf{U} represents each word, diagonal matrix \mathbf{S} represents the singular values and orthogonal matrix \mathbf{V} represents the Dimensions. For clustering matrix \mathbf{U} is selected as feature matrix as it is approximation to the feature matrix of words. In our experiment after concatenating 200 dimension context vectors, the 400 dimension context vectors after SVD is reduced to 200 dimensions as the rank of the matrix obtained to be 200.

2. Pre-Classification of Pronouns, Particles and Verbs

One of the useful properties of Nepali language is a few numbers of pronoun and particles. In our experiment, we created pronoun lexicon with 30 pronouns and particle lexicon with 6 particles. Another important property we considered during pre-clustering is that every Nepali language sentence ends either with verb or particle. So before the feature vectors are subjected to clustering, we pre-classify pronoun, particles and verbs using lexicon lookup. The pseudo-code for pre-classification is as follows.

Pre-Classification

```

For all words  $w$  in collection
  If  $w$  is not last word in sentence then
    If  $w \in$  pronoun lexicon
      Classify  $w$  as pronoun
    End If
  Else
    If  $w \in$  particle lexicon
      Classify  $w$  as particle
    Else
      Classify  $w$  as verb
    End If
  End If

```

3. Clustering Feature Vectors

After applying Singular Value Decomposition, the feature matrix represented by U is applied to clustering algorithm. In this thesis, we have implemented Modified Basic Sequential Algorithm Scheme (MBSAS) [2] for clustering the feature vectors. In MBSAS scheme the dissimilarity measure defined in previous section is used as proximity measure between the clusters. Selection of Different threshold values results in different clustering. The following listing is the implementation of MBSAS in matlab.

MBSAS Clustering implemented in MATLAB

```
function [clust, ci] = MBSAS(X, Dt, q)
    %X = matrix, Dt = threshold distance, q = max number of cluster
    % cluster determination
    m = 1;
    [N,P] = size(X);
    c(m).vect = X(1,:); % first vector in first cluster c(m=1,:)
    c(m).mean = c(m).vect; % initialize cluster mean as the single vector in
                          %the cluster
    cind(1) = m; % assign cluster index to input vector
    unassigned = [];
    for i=2:N
        for j=1:m
            dst(j) = pdist([c(j).mean;X(i,:)]); % calculate distance
        end
        [d index] = min(dst);
        if(d>Dt && m<q) % q is max number of clusters
            m=m+1;
            c(m).vect = X(i,:); % create new cluster and add x to new cluster
            c(m).mean = c(m).vect; % assign mean for new cluster
            cind(i) = m; %assign cluster index to input vector
        else
            unassigned = [unassigned;i];
        end
    end
    %pattern classification
    [N, P] = size(unassigned);
    for i=1:N
        ua = unassigned(i);
        for j=1:m
            dst(j) = pdist([c(j).mean;X(ua,:)]); % calculate distance
        end
        [d index] = min(dst);
        c(index).vect = [c(index).vect;X(ua,:)]; % adds vector to appropriate
                                                %cluster
        cind(ua) = index; %assign cluster index for remaining
    end
    clust = c;
    ci = cind;
end
```

V. Experimental Results

In experimental setup, the tokens are sorted in descending order of their occurrence frequency. When the frequency distribution of tokens in the text collection is plotted with frequency in vertical axis and tokens in horizontal axis, it showed some important aspects of language. For the purpose of simplicity, we avoided the punctuations in the sentences. From the plot it is noticed that most of the coordinating conjunctions occur at high frequency region. Usually the verbs and the noun or pronouns also occur at high frequency region. Adjectives and adverbs occur comparatively at low frequency region.

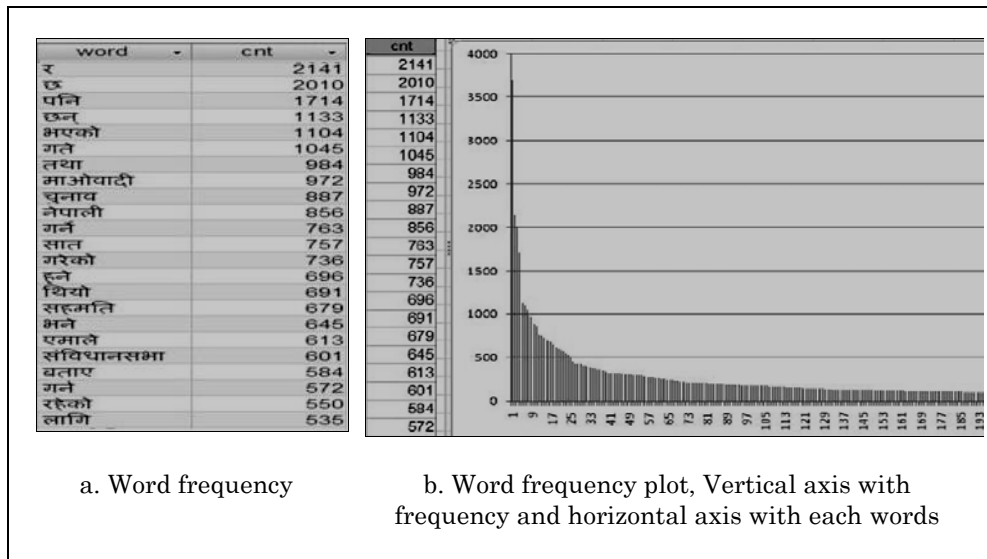


Figure 9. Word frequency and frequency plot

Figure 9 a and 9 b. shows the word frequency and the plot of it. From the figure, it is seen that the words like र (ra), छ (chha), पनि (pani) etc occur most frequently. These words are in the class of verbs or conjunctions.

Table 6. Classification of words to POS Tags

Tags	Devanagari	pronunciation	English Meaning
NN (noun)	घटना, दंगा, कपिलबस्तु, समूह, हिजो, देखि, शक्ति, रूप, भदौ, घटनाहरु, आज	ghatana, kapilbastu, samuha,hijo, dekhi, shakti, roop, bhadau, ghatanaharu , aaja	incident, kapilbastu(place name), group, yesterday, from, force, form, bhadau(month name), incidents, today
JJ (adjective)	साम्प्रदायीक, नयाँ, प्रभावशाली, सम्बद्ध, पहाडे	saampradayik, naya, prabhavshali, sambadda, pahade	communal, new, influential, related, hilly origin
VV (Verb)	होइन, चल्दै, आएको, हो, फैलिएको, गर्न, खोजिरहेका, छन्, के, भने	hoina, chaldai, ayeko, phailiyeko, garna, khojeraheko, chan, ke, bhane	is not, continuing, come, is, spread, to do, trying to, are, what,...
PP (Pronoun)	आफ्नो, आफूहरु, आफ्नै	aafno, aafuharu, aafnai	own, ownself, ownself
DD (Determiner)	त्यो, त्यस्तै, यो, उनका, उस्तै, एकअर्का, ती	tyo, tyestai, yo, ustai, ekarka, tee	that, similar as that, this, his, similar as this, among, those
RR (Adverb)	अहिले, बाहिर, अझै, त्यहाँ, त्यसैले, लगत्तै	ahile, bahira, ajhai, tyaha, tyesaile, lagattai	now, outside, again, there, that's why, immediately
TT (Particle)	पनि, नै, मात्र, त, समेत,	pani, nai, matra, ta, samet	also, ..., only,, also
CC (Conjunction)	र, तथा, या, तर, अर्थात,	ra, tatha, ya, tara, arthat	and, including, or, but, or

Also in Nepali language sentence, there are very few pronouns like म(ma), तपाईं (tapai) etc. and very few particles like मात्र(matra), समेत(samet) etc. we constructed pronoun lexicon of 30 pronouns and particle lexicons of 6 particles. Another useful aspect of Nepali language is that almost every Nepali language sentence ends with verbs and very rarely with particles. So prior of applying clustering algorithm, we clustered every last occurrence of sentence as either verb or particle depending upon particle lexicon entry. In the same way, the occurrence of pronoun is identified through pronoun lexicon lookup. However some inflections of pronouns are the source of error for pronoun identification.

After tagging last occurring verb, pronouns and particles, the remained untagged tokens are presented to MBSAS clustering algorithm for clustering. In the clustering process, the tags specifying gender, honorific or other agreement specific tags as specified by NELRALEC [19] are not distinguished. In a similar way, postpositions also occur embedded with a token, so postpositions are also not considered. Upon these constraints we identified altogether 8 tags. Table 6 shows some representative words obtained after clustering of words to respective tags. The misclassified tokens are shaded in the table.

Table 7. Precision and Recall of Tags

Tag	Precision	Recall
NN	0.68	0.70
JJ	0.46	0.60
VV	0.58	0.72
PP	0.58	0.74
DD	0.35	0.54
RR	0.60	0.62
TT	0.73	0.80
CC	0.57	0.88
Average	0.56	0.70

As the standard tagged corpus is not available and also some unsupervised method performance is evaluated for other languages rather than Nepali language, it is almost impossible to evaluate performance in terms of previous experiments. To overcome this difficulty, we used Nepali POS tagger binary supplied by [24] to get standard tagged text. This tagged text serves as the evaluation basis of the result. The tagged text is classified to eight groups of the experimental tag set. Table 7 summarizes the precision and recall obtained for the text clusters. From the table it is seen that the precision value of particle is high as there are few number of particles and we incorporated particle lexicon. However in case of pronoun, though we used lexicon, the precision not comparatively high as some words were misclassified as pronoun. But the use of lexicon resulted high recall of pronoun.

The experimental result indicates moderate performance. Several sources of errors like misspelled words, poor statistical distribution of words due to use of limited domain documents, lack of identification of base words of inflected words are the reasons for the degraded performance. Despite the performance result, the method could be employed as pre-tagging beforehand of applying manual or automatic tagging with bootstrapping methods.

VI. Conclusion and Future Works

In this thesis work we haven't considered the ambiguous words or the ambiguous tags. The performance is expected to be improved when disambiguation is employed before tagging. Another limitation of this work is we grouped all the respective gender specific, honorific or other agreement specific tags to single tag. It is sited that these tags should be considered to improve accuracy. Further to employ such tags, word inflection structure of Nepali language should be taken into account as several compound words embed more than one word in a single word or some compound verbs like “गर्दै छ (gardai chha)” have two verbs together. It is not too much to expect that stemming of words highly improves the performance. So this work motivated towards further work in the mentioned area in the future.

This thesis proposed the method of tagging Nepali Language Text. The proposed method is valuable in the sense that it is very much useful for automatic tagging of Nepali language text where the availability of pre-tagged text is very rare.

The previous works for tagging are based on training corpus, or in some work they are not dependent on training corpus. However, these works are done for other languages rather than Nepali Language text. In this thesis, we implemented unsupervised method, where the raw text itself serves as training data for tagging. The important aspect of this research is that we used pronoun and article lexicon. And in the beginning stage of clustering, the last words of the sentence are identified as verbs or particles. This led to high precision of particles and high recall of pronoun, particle and verb.

The result showed moderate performance. The most important reason of moderate performance is the method itself as no pre-tagged training data is

used for experiment. Other sources or errors are the misspelled words, the poor statistical distribution of words, use of limited domain text and ignorance of the base form and inflected form words. Another source of error is the use of very few tags as we just used eight tags for classification. The few tags resulted unrelated word to forcefully classify to unrelated clusters.

This method though easier to implement, it might results high error rate. However this could be of important for initial tagging of text before manual tagging despite its accuracy. And it is expected that the implementation of disambiguated tags and words further improves the performance.

References

- [1] Halteren, H Van(Ed.), *Syntactic wordclass tagging*, Kluwer Academic Publishers, 1999.
- [2] Sergios Theodoridis and Konstantinos Koutroumbas, *Pattern Recognition*, 3rd Ed., pp.488 - 491, 527- 528, Academic Press, 2006.
- [3] David A. Grossman, Ophir Frieder, *Information Retrieval Algorithms and Heuristics*, 2nd Ed., p.70, Springer, 2004.
- [4] Mike Lesk, “*Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone*,” ACM Special Interest Group for Design and Communication Proceedings of 5th annual international conference on Systems Documentation, pp.24 – 26, 1986.
- [5] Satanjeev Banerjee and Ted Pedersen, “*An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet*,” Lecture Notes In Computer Science, Vol.2276, pp.136 – 145, 2002.
- [6] Kenneth W. Church, “*A stochastic parts program and noun phrase parser for unrestricted text*”, ICASSP, pp. 136-143, 1989.
- [7] Euene Charniak, Curtis Hendrickson, Neil Jacobson, and Mike Perkowitz, “*Equations of parts-of-speech tagging*,” Eleventh National Conference on Artificial Intelligence, pp.784-789, 1993.
- [8] Eric Brill, “*Automatic grammar induction and parsing free text: A transformation-based approach*”, ACL 31, 1993.
- [9] Doug Cutting, Julian Kupiec, Jan Pedersen, and Penelope Sibun, “*A practical part of-speech tagger*”, The 3rd Conference on Applied Natural Language Processing, pp.133-140, 1991.
- [10] Michele Banko, Robert C. Moore, “*Part of speech tagging in context*”, The 20th international conference on Computational Linguistics, pp.56-61, 2004.
- [11] Ratnaparkhi, “*A Maximum Entropy Model for Part-Of-Speech Tagging*”, Proceedings of EMNLP, pp.133-142, 1996.
- [12] Eric Brill, David Magerman, Mitch Marcus, and Beatrice Santorini, “*Deducing linguistic structure from the statistics of large corpora*,” DARPA Speech and Natural Language Workshop, pp.275-282, 1990.

- [13] Steven Finch and Nick Chater, “*Bootstrapping syntactic categories using statistical methods.*” W. Daelemans & D. Powers (Ed.), Proc. 1st SHOE Workshop, Tilburg University. Institute for Language Technology and AI, pp.229-235, 1992.
- [14] Hinrich Schutze, “*Parts-Of-Speech Induction From Scratch*”, Proceedings of ACL 31, Ohio State University, pp.251-258, 1993.
- [15] Hinrich Schutze, “*Distributional Part-of-Speech Tagging*”, Proceedings of the seventh conference on European chapter of the Association for Computational Linguistics, pp.141-148, 1995.
- [16] Chris Biemann, “*Unsupervised Part-of-Speech Tagging Employing Efficient Graph Clustering*”, COLING/ACL, pp.7-12, 2006.
- [17] Sunita Arora, Karunesh Kr. Arora, S.S.Agarwal, “*Vishleshika: Statistical Text Analyzer For Hindi and Other Indian Languages*”, Workshop on Spoken Language, ISCA, 2003.
- [18] Chris Biemann, “*Chinese Whispers - An Efficient Graph Clustering Algorithm And Its Application To Natural Language Processing Problems*”, the HLT-NAACL Workshop on Textgraphs, pp.73-80, 2006.
- [19] Andrew Hardie, Ram Lohani, Bhim Regmi and Yogendra Yadava, “*NELRALEC/ Bhasha Sanchar Working Paper 2 Categorisation for automated morphosyntactic analysis of Nepali: introducing the Nelralec Tagset (NT-01)*”, 2005.
- [20] Laxmi Prasad Khatiwada, Srishtee Gurung, “*Nepali Lexicon*”
- [21] <http://en.wikipedia.org/wiki/Lexicon>
- [22] <http://en.wikipedia.org/wiki/sanskrit>
- [23] <http://www.bhashasanchar.org/>
- [24] <http://www.lancs.ac.uk/staff/hardiea/nepali/>
- [25] <http://www.elra.info/>
- [26] www.mpp.org.np
- [27] <http://www.nepalisabdakos.com/>
- [28] <http://nlp.ku.edu.np/cgi-bin/dobhase>
- [29] <http://www.ku.edu.np/>
- [30] <http://americannationalcorpus.org/>
- [31] <http://www.collins.co.uk/books.aspx?group=154>
- [32] <http://www.natcorp.ox.ac.uk/>
- [33] <http://khnt.hit.uib.no/icame/manuals/HC/INDEX.HTM>

Acknowledgements

It is my great pleasure to thank all those who helped me in every step of my academic career. I am so grateful to my professor Pankoo Kim, who provided me great opportunity to work under his guidance throughout the term of my master degree in Chosun University. So many thanks to you professor for your kind support and precious academic instructions you provided to me. My special gratitude goes to Prof. Il-Young, Chung who supervised me from the very beginning of this thesis work. Thank you professor for the precious guidelines to my work. Abundant gratitude to my other two advisors, Prof. Boem-Joon, Cho and Prof. Choong-Won, Kim. Your valuable advises helped to improve my work during the course of my research. I would like to appreciate prof. Sangman Moh, Prof. Seokjoo Sin and Prof. Sang-Woong Lee for their valuable advises and lectures at various walk of my study.

Special thanks goes to my mother, father, brother and sister for their love and encouragement in every walk of my life. Though I am far from my home, my mother and father always blesses me. Many thanks to my parent, beloved sister and brother.

I would like to thank my lab mates, and colleagues without whom it would be impossible to stay in Korea, far from home especially in the absence of family. Thank you all for your support in every difficulty I faced due to language, and sometimes food. Thanks to Korean Kimchi for its delicious taste.

Last but not the least; I would like to thank Korean Government which granted me the scholarship through the ministry of Information and Technology.

Thanks to all, Thanks to Korea.