2020-03-20

DEVELOPING PART OF SPEECH TAGGER FOR GURAGIGNA LANGUAGE

Gizachew, Fitsum

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

SCHOOL OF RESEARCH AND POSTGRADUATE STUDIES

FACULTY OF COMPUTING

DEVELOPING PART OF SPEECH TAGGER FOR GURAGIGNA LANGUAGE

Fitsum Gizachew Deriba

Bahir Dar, Ethiopia
November 3, 2017

BAHIR DAR UNIVERSITY
SCHOOL OF GRADUATE STUDIES
FACULTY OF COMPUTING
DEPARTMENT OF COMPUTER SCIENCE

DEVELOPING PART OF SPEECH TAGGER FOR GURAGIGNA LANGUAGE

Fitsum Gizachew Deriba

A thesis submitted to the school of Research and Graduate Studies of Bahir Dar Institute of Technology, BDU in partial fulfillment of the requirements for the degree of Master in the Computer Science in the Faculty of Computing.

Advisor Name: ***Prof. Bandaru R.K.Rao***

Bahir Dar, Ethiopia
November 3, 2017

Declaration

I, the undersigned, declare that the thesis comprises my own work. In compliance with internationally accepted practices, I have acknowledged and refereed all materials used in this work. I understand that non-adherence to the principles of academic honesty and integrity, misrepresentation/ fabrication of any idea/data/fact/source will constitute sufficient ground for disciplinary action by the University and can also evoke penal action from the sources which have not been properly cited or acknowledged.

Name of the student_____ Signature _____

Date of submission: _____

Place: Bahir Dar

This thesis has been submitted for examination with my approval as a university advisor.

Advisor Name: _____

Advisor's Signature: _____

Bahir Dar University

Bahir Dar Institute of Technology-

School of Research and Graduate Studies

Faculty of Computing

THESIS APPROVAL SHEET

Student:

Fitsum Gizachew — Signature — 03/11/2017 E.C

Name — Signature — Date

The following graduate faculty members certify that this student has successfully presented the necessary written final thesis and oral presentation for partial fulfillment of the thesis requirements for the Degree of Master of Science in computer science

Approved By:

Advisor:

Prof. Ramdas — Signature — 3/11/2017.

Name — Signature — Date

External Examiner:

Solomon Tegegne (PhD) — Signature — 05/11/2017

Name — Signature — Date

Internal Examiner:

Dr. K Srinivasa Reddy — Signature — 03/11/2017

Name — Signature — Date

Faculty Dean:

Dawud Nasir — Signature — 03/11/2017

Name — Signature — Date

Acknowledgment

First and foremost, I offer my deepest heartfelt thanks and glory to almighty GOD, who is the source of my strength and inspiration in the ups and downs of my life. He helps and gives me in all my ways. **Glory is to him! Amen!!!**

I am grateful to all those supported me to complete this research paper. I specially thank my advisor prof. Bandaru for providing me many valuable ideas and guidance. You constantly supported me through this work starting from title selection to end of this work. The Gurage zone culture and Tourism officer, Ato Bahiru lilaga, really I appreciate and you play great role in this work. You provide me Guragigna books, and helped me starting from data collection up to tagging the text and explain me the linguistic properties. Next, I would like to thank Mr. Ermiyas who is Waklite university instructor. Ermi thank you for your help, you spend your time with me by searching linguistic expert and encouraging me in difficulties. Finally, I want to thank walkite university computing staffs Getinet, Meskerem, Misir for their cooperative support. I don't think I quickly come up to understand POST Working principles if I didn't discuss with my friends Tsegaye and Abebaw you are really cooperative and enjoying friends.

My father Gizachew Deriba, I would not forget you what you did for me. Who passed away suffering a lot to bring a bright future for his child and GOD place your soul in heaven. My mother Aberash Keti directs me on the right way of my life and scarifies a lot to raise me. And thank you dear bro Daraje that you always ask and encouraging words "Indet Yetegefalih new? Berta!". I am very grateful to her.

TABLE OF CONTENTS

List of Figures

List of Tables

Acronyms and Abbreviations

HMM	Hidden Markov Model
NLP	Natural Language Processing
HMM	Hidden Markov Model
POS	Part of Speech
MCW	Multi Word Category
POST	Part of Speech Tagging
ANN	Artificial Neural Network
SBG	Sebat Bet Gurage
SOV	Subject Object Verb
WSD	Word Sense Disambiguation
UTF-8	Unicode Transformation Format: 8 bit Block of character
CRF	Conditional Random Field
TnT	Trigrams`n'Tags

ABSTRACT

The presence of Natural language processing (NLP) discipline allows computers to understand human language and process them. It provides basic role in different research tasks like part of speech tagger (POST), spelling correction and parsing, Machine translation, grammar checking, text summarization and so on. Among them POST is one of the foundation for other NLP tasks as this is used as preprocessing component. The task of POST is labeling each word to corresponding part of speech category so as to assign part of speech tags to words in a sentence.

Several parts of speech taggers were developed for local and foreign languages. However, these POS taggers can't be directly used for other language. As far as researcher's knowledge is concerned, there is no part of speech tagger developed for Guragigna language. So, the aim of this study is to develop part of speech tagger in Guragigna language. To do this first different literatures related to this work are reviewed to understand the nature and behavior of the language, and to identify possible tagsets. As a result, 17 tagsets are identified. In order to train and evaluate the performance of tagger 6,745 words are collected. The main source of our corpus is from Guragigna fiction and editorial category.

In order to develop the tagger, Hidden Markov model (HMM) approach and hybrid approach which is a combination of rule based and HMM based are used. Initially raw Guragigna text is tagged by HMM tagger based on the most probable path for given sentence of word. After that rule based tagger is used to correct HMM tagger based on predefined set of rules. The algorithm used for HMM is Viterbi. Additionally in our experiment we also use CRF approach.

For experiment analysis, we used 90% of the data for training and the rest 10% for testing. Different experiments are conducted for each tagger independently. Having tested on the same data the performance analyses of the taggers are 66.56, 74.46 and 78.42 for CRF, HMM tagger and Hybrid tagger respectively.

Increasing the size of training data and examining the tagger influences the result. Result from our experiment shows that adding of rule based tagger performs better result than HMM tagger alone.

Keywords: *Guragigna, Part of Speech Tagger, Hidden Markov Model, NLP, Hybrid Tagger, Viterbi algorithm*

CHAPTER ONE
INTRODUCTION

1.1 Background of the study

Language is a method of human communication, either spoken or written, consisting of words in structured and conventional way. It has a great role in our day to day activities. In its written form it provides a means to keep information and knowledge for long period of time and pass it to next generation. In its spoken form, it can be used as a way to cooperate our day to day activities with others [1]. Languages that can be learned from environment and used for communications by human beings are known as natural language such as English, Amharic, Afaan Oromo, and Guragigna. In contrast artificial languages are based on a set of prescribed rules and developed for a specific purpose such as programming language.

In order to understand natural language there are mainly four kinds of knowledge. The first one is morphological knowledge. That is concerns of word formation and it studies the patterns formation of words by the combination of sounds into minimal distinctive units of meaning called morphemes. So, morphological knowledge concerns how words are constructed from morphemes. The second one is syntactic knowledge that deals with how words are combined to form phrases, phrases are combined to form clauses and clauses join to make sentences or formation of sentence. So, this implies that description of the ways in which words must be ordered to make structurally acceptable sentences. The third one is semantic knowledge that concerns with the meanings of the words and sentences, and describes the ways in which words are related to the concepts. Lastly pragmatic knowledge that deals with the contextual aspects of meaning in particular situations and it concerns how sentences are used in different situations and how it affects the interpretation of the sentence.

Natural Language Processing (NLP) is a research discipline related to artificial intelligence, linguistics, philosophy, and psychology [2]. The aim of this discipline is building systems capable of understanding and interpreting the computational mechanisms of natural languages. There are several research attempts under investigation in NLP; for instance, machine translation, information extraction and retrieval using natural language, text-to-speech synthesis, automatic written text recognition, grammar checking, and part-of speech (POS) tagging [3].

POS sometimes called as word classes, morphological classes or lexical tags and it contains words that are divided into different classes. Traditionally grammars have few parts of speech (noun, verb, adjective, preposition, adverb, conjunction, etc.). For example, consider the following sentences that are tagged with English part of speech tagger.

Given sentence

I eat Kitfo at my lunch.

Tagged output sentence

I/PRP eat/VBR Kitfo/NN at/IN my/PRPS lunch/NN ./PUNC

PRP: -	Personal pronoun
VBR: -	Verb non-3rd person singular present form
NN:-	Noun, singular, common
IN:-	Preposition
PRPS$: -	Possessive Pronouns, singular
PUNC: -	Punctuation

Figure 1-1: English Part of Speech Tagger

Part of Speech tagging (POST) is one of the application areas of NLP and task of labeling each word in a sentence with its appropriate syntactic category. It is a very important preprocessing component for language processing activities and helps in doing in deep parsing of text and in developing information extraction systems, semantic processing, Information Retrieval and Machine translation.

Depending on the degree of automation used in the tagging process, the taggers can be classified as supervised or unsupervised [4]. Supervised taggers typically rely on pre-tagged corpus which serve as the basis for creating tools (dictionary, word/tag frequencies, tag sequence probabilities, rule set, etc.) to be used throughout the tagging process. Unsupervised models, on the other hand, are those which do not require a pre-tagged corpus, but instead use sophisticated computational methods to automatically induce word groupings (i.e. tag sets).

Based on these automatic groupings, to model and develop the tagger there are different approaches which are used to predict word category information to the words in a text. The most

know approaches are rule based approach, Stochastic approach or HMM (Hidden Markov model) approach, Artificial neural network and a combination of neural network and rule based or HMM and rule based [5]. Rule based approach uses a large database of hand-written disambiguation rules considering the morpheme ordering and contextual information. The stochastic approach uses a clearly tagged text to estimate the probabilities to select the most likely sequence. The detail description of tagging approach is described in section 2.2.

Researches in natural language processing have motivated by two main aims. Those are lead to better understanding of the structure and functions of human language and to support the construction of natural language interfaces. Thus is used to facilitate communication between humans and computers.

Recently researches in part of speech tagging are developed for international and local languages. For example tagger taggers developed for foreign language are POST for Chinese [6], Hindi [7], Arabic [8], Portuguese, Turkish [4], Bengali [9]. Locally for Amharic [10, 11], Afaan Oromo [12, 13] and Tigrigna languages [14, 15], taggers are developed.

However, in order to push Guragigna language toward the technology, researches made in Guragigna language are very limited researched. Particularly in Guragigna language part of speech tagger is not developed.

1.2 Statement of the problem

Guragigna is one of the widely used languages in Ethiopia, which is afro Asiatic language of southern Ethiopic branch of Semitic family that are spoken by Gurage people. Currently there are around 6.8 million native speakers of the language according to the 2007 national census. The language is used in medium instruction for primary school and used in different institute of the region. In addition to that, it is used in different sectors such as Walkite radio station. Magazine, educational books and fictions are published with the language. Due to this currently the number of users of the language is highly increasing from time to time.

Nowadays researches conducted with the Guragigna language are very limited. However, some tools are developed like keyboard for the language [16]. Still as far as researcher’s knowledge is concerned, very limited researches are conducted in the area of NLP for the language.

Particularly, there is no research conducted on development of POS tagger. Due to increasing of the number of language users and to do different researches, it is essential to develop part of speech tagger for the language.

The absence of Guragigna tagger is the main obstacle for researchers who do in NLP High level applications such as in the area of Machine translation, Spell checkers, Dictionary compilation, Automatic sentence parsing. The reason is that, all of these NLP applications use part of speech tagger as their preprocessing components for their task because of it provides additional information about corpora content in the form of word class category [15]. Additionally, there is problem in word categorization for the language.

At the end of this study the following research question are answered and investigated.

- ✓ How can we automatically identify the Guragigna part of speech with high accuracy?
- ✓ How to determine rule based tagger to improve the performance of hybrid based tagger?
- ✓ Which tagging approach among Hidden Markov Model, Hybrid model perform better for Guragigna language?
- ✓ How to prepare corpus that can contribute to determine the effectiveness and robustness of individual and hybrid tagger?

1.3 Objective of the study

The objective of this study is described as general objective and specific objectives.

1.3.1 General objective

The general objective of this study is to develop part of speech tagger for Guragigna Language text using HMM and hybrid approach.

1.3.2 Specific objectives

To achieve the general objective, the following specific objectives are addressed.

- ✓ To analyze word category construction and behaviors of Guragigna language.
- ✓ To design a POS Tagger for the Guragigna language.
- ✓ To develop a Guragigna POS Tagger Prototype.

- ✓ To test and evaluate the tagger performance of the selected POS tagger algorithm for Guragigna.
- ✓ To state the conclusions based on experimental results and to recommend research area in the future.

1.4 Scope and Limitation of the study

The scope of this study is limited to explore the hybrid based approach to design automatic POS tagger for Guragigna language. The study also focuses on categorizing part-of-speech of the language for word lexical categorizations. The probabilistic model which is used in this work is bigram.

Moreover, this study excludes some of word categories such as tense and gender. The main reason for this limitation is lack of data corpus and linguistic professional. Since tagging every word in corpus with word category is difficult due to lack of enough human professionals. So, it needs more time and effort [17]. The main limitation in conducting this study is the absence of readily available annotated corpus for Guragigna language.

1.5 Methodology

The following methods are used for the achievement of this study.

1.5.1 Literature Review

In order to develop foundation of idea on the study area, to be up-to-date in information and to avoid replication; different Books, journal article, conference paper, research reports published and unpublished materials are reviewed. Additionally, to understand the morphological property of Guragigna related works on Guragigna word classes are conducted. Works that are related with part of speech tagging relevant to this work is also reviewed.

1.5.2 Data Collection

In this thesis work, the data were collected from a book which exists in hard copy format and editorial category of soft copy format. There is no readymade data set for the language. The main source for this thesis work is fiction book which is the first fiction called የጫሙት ሽካ that is written in Guragigna language in 1960 by Assistance Professor H/Markos at AAU. And other

data is taken from Editorials of Gurage language, Gurage zone state communication website. In order to train and test the model totally we use dataset of 307 sentences that contain 6745 words.

1.5.3 Testing and Evaluation

To test the model, we used 90% data from entire corpus for training. And the rest 10% are used for testing. The training starts with 10% of data from the entire training corpus and measure the performance of the result. Then, adding of 10% training data to previous training data until desired training corpus. Finally, to test the performances of tagger on set of test data, we evaluate using confusion matrix for each tagger.

1.6 Application of the Study

Natural language systems are generally composed of a set of interconnected pipelined tasks. Each task is may or may not related to other pipelined tasks. However, among these tasks POST is one of them. Therefore, the development of POS tagger has a big impact for other pipelined tasks.

So, developing POST has different benefit for Gurage language community as well as research community.

- ✓ For the Guragigna language community: Help them to discover the word categories and grammar construction. In addition to that it has an advantage for research community.
- ✓ For research community: It contributes for researchers who do in higher level of NLP application of Guragigna language as preprocessing component.

1.7 Organization of the thesis

The rest of this chapter is organized as follows. Chapter two is review of literature and related work. This chapter mainly focuses on overview of part of speech tagging, different tagging approaches that are used in POS. Moreover, discusses on related work that are done using Hybrid approach. In chapter three linguistic property and tagsets of Guragigna language are presented. Basically, it explains about morphology, sentence structure and word classes of Gurage language, and at the end discusses on tagsets that are selected for the study. Detail design process, algorithms that are used in the study and architecture of the POS tagger for selected approaches on the language are presented in chapter four. Chapter five presents the details of corpus preparation for sake of the experimentation. Preprocessing component for taggers is also

discussed. Additionally, implementation and experiment of selected tagger, performance analysis and results obtained are also addressed in this chapter. Lastly, conclusions, contribution of the study and recommendations for future works are presented in chapter six.

CHAPTER TWO
LITERATURE REVIEW AND RELATED WORK

2.1 Overview

As discussed in section 1.1, part of speech tagging is the process of assigning each word in sentence with the proper POS tag in the context it appears. POS tagging is harder than just having a list of words and their parts-of-speech. Because some words can represents more than one part-of-speech at different times. Such words are called multi-category words (MCW) and are ambiguous in nature [18]. For example, consider the following two sentences which have the same word **water** but have more than one lexical category depending on the context that they appear in sentences.

Give me some ***water***.

They ***water*** the plants daily.

- ✓ In the first sentence, the word WATER names something. So, it is a **noun**.
- ✓ In the second sentence, the same word WATER expresses an action. It tells what they do. Here it is a **verb**.

Figure 2-1: One word having Part of Speech tagger

So, the challenging task in the POS tagging is to find the correct POS tags of new words and disambiguating multi-sense words.

To resolve ambiguous problem and unknown words during POS tagging, the lexical and the context information i.e. the relationship between adjacent and related words in phrase, sentence or paragraph has to be considered or necessary [19].

Generally, this chapter explores on different POST approaches that are conducted to develop POST. And the way how they work, and identify their strength and weakness. Additionally, the chapter deals about researches that are conducted on POST for local language as well as foreign languages. Finally discusses on works that related works to this study.

2.2 Approaches of Part of speech tagging

In order to develop POST there are different approaches. However, each approach has their pros and cons. So, this section focuses on brief description of different approaches that are used for part of speech tagging. Several algorithm and method are revised to deal with part of speech tagger. Among those approaches, the most known approaches are stochastic based [3, 20, 21], rule based [22, 23], neural network approach [24, 7], support vector machine [25, 11, 26], condition random field [11, 26], and hybrid based approaches [4, 27, 28]. The details of most known approaches are described as below.

2.2.1 Stochastic based part of speech tagging

The most popular approaches used nowadays are statistical or machine learning techniques. These approaches primarily consist of building a statistical model of the language. And using the model it can disambiguate a word sequence by assigning the most probable tag sequence given the sequence of words in a maximum likelihood approach [29].

The stochastic model is widely used in POS tagging for simplicity and language independence of the models. The intuition behind all stochastic taggers is a simple generalization of the 'pick the most-likely tag for this word' approach based on the Bayesian framework. Among stochastic models, bi-gram and tri-gram HMM are quite popular.

Probability is the basic principle behind HMM. [30]. Initially for a given sentence or a word sequence, HMM tagger chooses the tag sequence that maximizes:

$$P(word|Tag) * P(Tag|Previous\ n\ tags) \quad (1.1)$$

Where,

$P(word|Tag)$: The probability of a word being assigned a particular tag from the list of all possible tags for the word (most frequent tag).

$P(Tag|Previous\ n\ tags)$: The probability of one tag given by another n previous tag.

Generally, HMM tagger chooses a tag sequence for a given sentence rather than for a single word. Then it computes the most probable tag sequence of tags $T = (t_1, t_2 \ldots t_n)$ for a given sequence of words in the sentence $W = (w_1, w_2, \ldots, w_n)$:

$$\hat{T} = \underset{t\epsilon\,\mathrm{T}}{\mathrm{Argmax}}\, P(T|W) \tag{2.2}$$

Where,

$\hat{T} = \mathrm{Argmax}_{t\epsilon\,\mathrm{T}}\, P(T|W)$: The set of values of T for which P(T|W) attains its maximum value.

By Bayes law and P (T|W) can be expressed as:

$$P(T|W) = \frac{P(T)P(W|T)}{P(W)} \tag{2.3}$$

So we choose the sequence of tags that gives

$$\hat{T} = \underset{t\epsilon\,\mathrm{T}}{\mathrm{Argmax}} \frac{P(T)P(W|T)}{P(W)} \tag{2.4}$$

Where as

$\mathrm{Argmax}_{t\epsilon\,\mathrm{T}} \frac{P(T)P(T|W)}{P(W)}$: The set of value of t which (P (T) P (T|W/P (W)) attains it's maximize value.

Since, we are looking for the most likely tag sequence for a sentence in given a particular word sequence, the probability of the word sequence *P* (*W*) will be same for each tag sequence and we can ignore it. So, we get

$$\hat{T} = \underset{t\epsilon\,\mathrm{T}}{\mathrm{Argmax}}\, P(T)P(W|T) \tag{2.5}$$

Where,

P (*T*) is the Prior probability and *P* (*W*|*T*) is the Likelihood probability.

Basically, HMM taggers work on word frequency approach called N-gram approach that calculates the probability of given tag sequence. So, the n^{th} word W is depending on the previous n-1 tag. Where, the value of n is set to 1, 2 or 3for practical purpose. They named as unigram, bigram, and trigram. However, the most widely used taggers are bigram [14] and trigram (TNT) [31].

Stochastic part of speech tagger has many advantages among them researchers may not need language professional, and the performance of tagger depends on the amount of

training set. However, it is relatively complex, not suitable for language that less annotated corpus and require vast amount of stored information [14, 32].

2.2.2 Rule based part of speech tagging

Rule based POS tagging is the earliest tagging approach in which set of rules is manually constructed and then applied to given text. This earliest algorithm is based on two stage architecture. The first stage uses a dictionary to assign each word a list of potential parts of speech and the second stage used large lists of hand-written disambiguation rules to examine a single part-of-speech for each word [30]. Among these taggers, the ENGTWOL tagger [33] is based on the same two stage architecture, although both the lexicon and the disambiguation rules are more sophisticated than the early algorithms.

The primary step towards development of a Rule Based Part-of-Speech tagger for any language demands in-depth understanding and analysis of that language [34]. This implies that it needs to know linguistic feature of specific language such as morphological, lexical and syntactic structure. These rules are developed by linguistic professionals. As an example, a context frame rule might say something like: "if an ambiguous/unknown word X is preceded by a determiner and followed by a noun, tag it as an adjective".

Det - X - n = X/Adj

Figure 2-2: Example for rule based POST

In addition to contextual information, many taggers use morphological information to aid in the disambiguation process. One such rule might be: "if an ambiguous/unknown word ends in an '-ing' and is preceded by a verb, label it a verb" (depending on language theory of the grammar).

Mainly there are two ways to develop rule based part of speech tagger. Those are, rule based approach which use contextual information [35]. This rule is called context frame rule and the second way is transformational rule in which the model tries to learn and store a sequence of rule using training data without manual construction this kind of work is known as brill transformation based approach [36, 27].

Even if rule based approach is more accurate, but it needs linguistic experts with high level of language knowledge. So, it is difficult to label every rule of the language.

The main drawbacks of rule based tagging are the laborious work of manually coding the rules and the requirement of linguistic background because of extracting rules for each sentence is difficult and requires time and effort.

However, it has many advantages over others taggers. For example, vast reduction in storing information the perspicuity of small set of meaning rules, ease of finding and implementing improvements to the tagger, and better portability for one tag set, corpus genre or language to other and so on [23].

2.2.3 Neural Network based Part of Speech Tagging

Neural networks are one of the most efficient techniques in machine learning approach that are used for scarce data. It consists of large number of simple processing units (neurons) which highly interconnected by direct weighted link; associated with each unit as an activation value. Through, this connection (activation) is propagated to other units. The interconnections of the neurons follow specific network architecture [37].

The main processing principle of neural nets is their capability to distribute activation patterns (learned from a training set) across the links via a learning algorithm. This is done in a way similar to the basic mechanism of the human brain. However, the similarity ends here. The human brain is fixed and deterministic fashion [38].

The most popular artificial neural network has three layers of units [37]. Namely a layer of input units, a layer of hidden units and a layer of output units as observed in Fig 2.3 below. Connections exist only between units in adjacent layers. The bottom layer is called input layer which is connected to the hidden layer that represents input of the network i.e. The raw information is fed to the network as an input. So, it can learn and adapt properties. Correspondingly, the middle layer, so called hidden layer, connected to the output layer that is determined by the activities of the input unit and the weights on the connections between the input and hidden unit. The output layer represents the result of the learning properties from the input layer and hidden layer. Its behavior depends on

activity of hidden units and the weight on the connection between the hidden and the output layer.

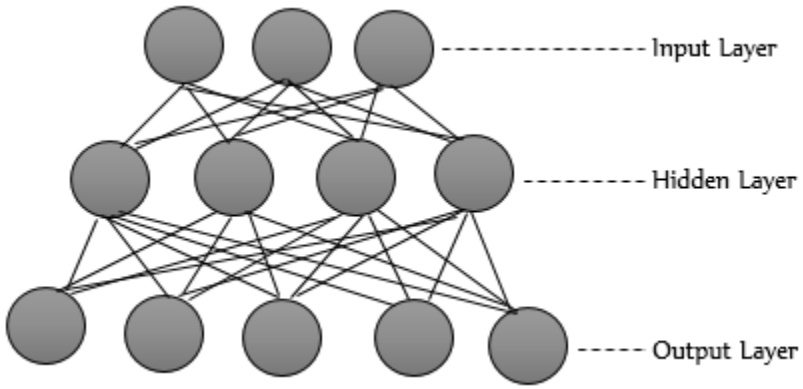

Figure 2-3: Three Perceptron Layer

In general, three entities are characterizing an Artificial Neural Network. The network topology or interconnection of neurons, characteristics of individual units or artificial neurons, and the strategy for pattern learning or training.

Based on the interconnection, ANN (Artificial Neural Network) is classified as feed-forward and feed-backward topologies. Feed forward allows signals to travel forward in one direction only; from input to output. There is no feedback or backward propagation, which means the output of any inner neuron (layer), does not affect that same layer. Feedback Artificial Neural Network can have input values traveling in both forward and backward [12]. In the forward pass, error is calculated from outputs and used to update output weights. In backward pass, error at hidden nodes is calculated by back propagating the error at the outputs through the new weights and hidden weights are updated.

The main drawback of this approach is having lower processing speed compared to stochastic approach and as number of tagsets increases, the performance of tagger not good, and selection and treatment of ambiguous word is deal with only considering corpus. Additionally, the POS of the word is uniquely determined by the word itself. For example, neural net tries to perform tagging based on the complete context. As a result,

even for the word on the left is the same, the tagging result will be different if the complete context is different. That is the neural tagger hardly acquires the rule with single inputs.

Furthermore, though lexical information is very important in tagging, it is difficult for neural net to use it because doing so would make the neural enormous [39].

However, it has benefits. That is the capabilities of expressing non-linear decision [12]. And also, it is suitable for language which has small number of tagsets and small amount of training set.

2.2.4 Conditional Random Field approach

Conditional random field (CRF) can be applied for variety of NLP tasks such as Named Entity Recognition, information extraction, text chunking, and POS concept tagging [11]. The main task of CRF in POST is segmenting or labeling sequence of data.

2.2.4 Hybrid based approach

This approach is a combination of different taggers or model to obtain better accuracy result. It was observed that different taggers have similar performances, although they usually produce different errors [40]. Based on this encouraging observation, it is essential to use more than one tagger by combining them. The combination may be rule based and stochastic [4, 28, 9], rule based and ANN approach [41] and so on.

The working principle of rule based and stochastic are firstly tagging begins by accept unannotated text into the stochastic tagger [14, 32]. Then the initial state annotator that is stochastic tagger tags all the words to its most likely tag based on the lexicon. Next the output of this tagger used as temporary corpus for the rule based tagger. The other possible tags act as the second tag if and only if the initial tag which are tagged by stochastic tagger are wrong, then the rules will be applied to change the initial tag (most likely tag) to one of the other possible tags.

Like other taggers hybrid approach has its own cons and pros. Among Advantages small set of rules that are sufficient for tagging is learned. As the learned rules are easy to understand, the development and debugging are made easier. Interlacing of machine-

learned and human-generated rules reduce the complexity in tagging. The rule based even can be ten times faster than the fastest Markov model tagger.

However, Training time is often intolerably long, especially on the large corpora which are very common in Natural Language Processing [42].

2.3 Related work

In this section, related research works are conducted. We can relate works based on different views. For example, the type of problem they address, using of the same methodology for similar problem, if our work is inspired by them and soon.

Therefore, in this work we review previous works of foreign and local language based methodology they used that are HMM and hybrid approach.

2.3.1 Previous work on foreign language

There is different POST conducted in foreign language. Among them let us see works that are related to our works.

According to Eric Brill [23, 27] **:**

They use corpus-based POST approach for English, called transformation-based error-driven learning a system that guesses the tag of each word, then goes back and fix the mistakes. The basic idea of the tagger is to assign each word within a given text it's most likely tag estimated by initial-state tagger that is trained on a large tagged corpus without regard to context [23]. Once the text passed through the initial state tagger, it compared with the reference text (manually tagged text). As a result, an ordered list of transformation rules is learned that can be applied to the output of the initial-state tagger to make it better resemble with the reference text.

Transformation-based part of speech tagging works as first the initial-state annotator assigns each word its most likely tag as indicated in the training corpus. An ordered list of transformations is then learned, to improve tagging accuracy based on contextual cues.

For testing the performance of the system, the researchers accompanied an experiment using 1.1 million words from Pen Tree-bank tagged Wall Street Journal corpus. From these total corpus, 950,000 words where used for training and 150,000 words where used for testing. Out of 950,000 words of the training corpus, 350,000 words where used to learn rules for tagging unknown words and 600,000 words where used to learn contextual rules. Generally, system has learned 243 rules for unknown words and 447 contextual tagging rules. Using the tagger without lexicalized rules, an accuracy of 96.3% and an unknown word accuracy of 82.0% is obtained. Finally, the overall performance of accuracy was 96.6% on the tested corpus.

According to Levent A., Zihni O. and Tunga G [4]**:**

In their work, the researchers use composite (rule based and statistical) part of speech tagging for Turkish. They used two additional features to increase the performance of the system. They used both word frequencies and n-gram (unigram, bigram and trigram) probabilities. In the first case, they incorporate a morphological analyzer which is used to obtain the part-of speech of words independent of the words with in the corpora. This enables the system to guess the tag of the word even if it does not exist within the corpus. In the second case, the researchers use another statistical approach which is related to the part-of-speech of words based on the position within the sentence i.e. making use of the word order property of the language. So, in order to increase the accuracy of the tagger, these heuristics are useful especially for fixed order languages like English, since the positions of the grammatical categories in sentences in these languages do not change.

The corpus used to train and test the tagger, the researchers totally used 7200 sentences. From the 7200 sentences, 6000 (85%) of the corpus used for training and the remaining 1200 (15%) of the sentences are used to testing purpose. The tag set used in this work is 13-word classes (POS) for the tagging process.

The system finds the tag of a word in three main steps. In the first step, the statistical analyzer module computes some statistical data from the training corpus. In the second step, the tag set finder finds possible part-of-speech for words to be tagged. Finally, the main modules of the system determine part-of-speech of words. To reach final decision

the tagger combines word frequencies, n-gram probabilities, heuristics data and data about candidate tags.

In order to test the performance of the system, they perform three experiments by using different parts of the corpus as training set and test set in each and addition to calculating the performance of the system, they calculated the performance when only the morphological analyzer is used (without any statistical data from the corpus). As a result, they have got an average of three experiment accuracy of 82.26% for system with statistical data and 66.73% for system without statistical data. In their work, statistical data greatly improves the performance when compared with the baseline (without statistical). The reason is stems from the fact that, being an agglutinative language, Turkish has a very complex derivational and inflectional morphology a word may change its part of speech freely by affixing different suffixes. These impose some difficulties for tagging of agglutinative languages in general and of Turkish in particular.

3.3.2 Previous work on local language

Many researches are conducted in POST for local language. Some of these are described below:

According to Mesfin [37]:

The researcher uses the Viterbi algorithm and bigram model to develop a POS tagger for Amharic language which is the first Amharic tagger. The prototype is implemented with Visual basic programming language and uses 290 Amharic words for training and testing purpose. The researcher evaluates the performance of the model in two ways. First conduct an evaluation with same dataset for training and testing and achieve 97% accuracy. The next evaluation is conduct with different datasets: 90% for training dataset and 10% for testing dataset and achieve 90% of model performance.

According to Getachew [44]**:**

Their study was conducted for part of speech tagging for Afaan Oromo language with HMM based approach. His work uses Unigram and Bigram models of Viterbi algorithm and with 159 sentences or 1621 distinct words. The researcher uses java programming language to implement the prototype. This work first evaluated the tagger with 20% of

untagged training set for correcting tagger errors through comparing with manual tagged portion of this training set. Finally, uses ten- fold cross validation, all corpuses are used for both training and testing purpose in different repeated ten phases and the accuracy rates are averaged. The performance evaluation is conducted for both unigram and bigram algorithms and achieves with 87.58% and 91.975% respectively. After measuring the performance researcher shows that bigram model of Viterbi algorithm is the promising approach in part of speech tagging for Afaan Oromo language.

According to Tekilay [14]:

To enhance the performance of tagger, researcher uses the hybrid approach (Rule based approach and Stochastic based approach) to develop a POS tagger for Tigrigna language, rather than the individual ones. The prototype is implemented with Python and uses 26, 000 words for training and testing purpose. The training set consists of 75% of the corpus and the testing set consists 25% of the corpus. In his work, different experiments are conducted for three types of taggers namely the HMM tagger, the rule based tagger and Hybrid tagger. Accordingly, 89.13 %, 91.8% and 95.88% performance for HMM, Rule based, Hybrid are obtained respectively.

As we have seen, most of the above related works were conducted using hybrid approach, in these languages conducting the research achieves better accuracy rather than using single either HMM or Rule based alone. So, using this approach performs better accuracy. Generally, the above related works are summarized as follows in the following table 2.1.

2.4 Summary of Literature Review and related Work

Table 2.1: Summary Literature review and related Work

Approaches	Used methodology	Corpus used	Implementation	Experiment result
Hybrid approach [14]	-Uses the hybrid approach (Rule based approach and Stochastic based approach) to develop a POS tagger for Tigrigna language	-Uses 26, 000 words for training (75%) and testing (25%) purpose.	-Python programming language	-HMM tagger: 89.13 % - The rule based tagger: 91.8% - Hybrid tagger: 95.88%
HMM [44]	-Part of speech tagging for Afaan Oromo language with HMM based approach.	-Uses Unigram and Bigram models of Viterbi algorithm and with 159 sentences or 1621 distinct words.	-Java programming language	-First evaluate the tagger with 20% of untagged training set for correcting tagger errors. -Finally, uses ten- fold cross validation and performance evaluation: unigram (87.58%) and bigram (91.975%)
A Composite Approach for POST Turkish [4]	-Use rule-based and statistical approach -It also use morphological and words position features.	-Used 7200 sentences. from the 7200 sentences, 6000 (85%) of the corpus used for training and the remaining 1200 (15%) of the sentences are used to testing	-Visual basic programming language	-They perform three experiments by using different parts of the corpus. -They have got an average of three experiment accuracy of 82.26% for system with statistical data and 66.73% for system without statistical data.

CHAPTER THREE
LINGUISTIC PROPERTY OF GURAGIGNA LANGUAGE AND TAGSETS PREPARATION

3.1 Overview

This chapter discusses about different variation of Guragigna language, phonology of Guragigna which is fundamental component of the language, the structure of Guragigna word classes, and sentences since each of these parts has its own impact on this study. Finally, at the end of this chapter we will discuss tagsets of the language that are used for this study.

The language spoken by Gurage people is known as Gurage language. The variations among these languages are used to group the Gurage people into three dialectically varied subgroups: Northern, Eastern and Western. The Gurage people speak six separate languages namely Sodo or Kistana, Zay, Inor, Mesmas, Masqan, SBG (Chaha), all belonging to the Southern subdivision of the Ethiopian Semitic languages that are listed in summarization form in appendix B. The languages are often referred to collectively as "GURAGINYA" by other Ethiopians. According to the 2007 national census, fluent Language speakers of Guragigna is 6.8 million people. This is 5.53 % of the total population of Ethiopia. The languages are transcribed with the Ge'ez script or Ethiopic writing system and subset of Guragigna language script has 40 independent glyphs which is described in appendix A. Currently the speakers of Gurage language is distributed through Ethiopia and the language becomes growing from time to time. Although, literally rate of the language still not grown well. But some works are done with the language. For example keyboard for Guragigna language [16].Textbooks for grade one up to four, Fiction books, new testimonial holy bible and some organizations at Gurage zone uses the language. Moreover, Walkite University is proposing to open Gurage language at degree level.

In this thesis work, SBG is selected to develop POST. The rationale behind choosing SBG (chaha) language is due to high rate of literacy compared to other category, large population or fluent speaker of the language, the growth rate of language and current usage of the language dominated at Gurage zonal organization, walkite radio station etc.

3.2 Guragigna morphology

Like other languages, Guragigna language has its own linguistic property.

In the following section, we will discuss consonant and vowels of the language and basics of phonological process which makes different from other Semitic languages. They play great role for the language.

3.2.1 Consonants and Vowels of SBG

SBG has a typical set of phonemes for an Ethiopian Semitic language. The language has a usual set of ejective consonants as well as plain voiceless and voiced consonants. However, the language also has a larger set of palatalized and labialized consonants than most other Ethiopian Semitic languages. In addition to the typical seven vowels of other Semitic languages, SBG has open- front and back vowels. Some of the dialects have both short and long vowel phonemes, and some have nasalized vowels. The Table 3.1 and Table 3.2 below shows that the details of SBG phonemes (consonant and vowels).

Table 3.1: SBG Consonants

Consonants									
		Labial		Dental	Post-alveolar	***Palatal***	Velar		Glottal
		Plain	round				plain	round	
Nasal		M	mʷ	n					
Plosive/ Affricate	Voiced	B	bʷ	d	ǧ	***gʸ***	g	gʷ	
	Voiceless	P	pʷ	t	č	***kʸ***	k	kʷ	
	Ejective			ṭ	č̣	***k̓ʸ***	k̓	k̓ʷ	
Fricative	Voiced			z	ž				
	Voiceless	F	fʷ	s	š	***xʸ***	x	xʷ	H

Table 3.2: SBG vowels

Vowels			
	Font	central	Back
High	I	ə /ɨ/	U
High-Mid	E		O
Low-Mid	ɛ	Ä	
Low		A	

All Semitic languages have complexity of morphology characteristic in the verb. Moreover, SBG exhibits another level of complexity because of the complex relationship between the set of consonants in the root of a verb. And how they realized in a particular form of that verb or a noun derived from that verb. For example, let us see the following table 3.3 which shows when the verb changed into perfective and impersonal of SBG.

Table 3.3: Derivation of Verb

Verb 'open'	Root word {*kft*}	Description
third person singular masculine of Chaha	*käfätä-m*. (he opened)	Perfective of Chaha
	'käf^{w}äč-i-m' (he was opened)	Impersonal of Chaha

3.2.2 SBG phonological processes

Gemination: In most Ethiopian Semitic languages, gemination (consonant lengthening) plays a role in distinguishing words from one another and in the grammar of verbs. For example, in Amharic, the second consonant of a three-consonant verb root is doubled in the perfective: {sdb} 'insult', sä**dd**äbä 'he insulted'. But in SBG as unlike Amharic the second consonant becomes t in the non-geminating dialects: sä**t**äβä-m 'he insulted'. This indicates that gemination is replaced by devoicing; only voiced consonants can be devoiced [45].

The following figure 3.1 shows that how the Guragigna voiced consonant can be devoiced or geninationed.

$b/\beta \rightarrow p$, $d \rightarrow t$, $g \rightarrow k$, $b^{w} \rightarrow p^{w}$, $\check{g} \rightarrow \check{c}$, $g^{y} \rightarrow k^{y}$, $g^{w} \rightarrow k^{w}$, $z \rightarrow s$, $\check{z} \rightarrow \check{s}$.

Figure 3-1: SBG Gemination

Labialization: Several morphological processes cause consonants to be labialized (rounded). For example, in SBG verb {gkr} 'be straight', there is the derived adjective gʷəkʷər 'straight'. Labial and velar consonants can be labialized [46, 47].

The following figure 3.2 shows that how consonants can be rounded or labialized.

$p \rightarrow p^{w}$, $b \rightarrow b^{w}$, $\beta \rightarrow w$, $f \rightarrow f^{w}$, $k \rightarrow k^{w}$, $\underline{k} \rightarrow \underline{k}^{w}$, $g \rightarrow g^{w}$, $x \rightarrow x^{w}$.

Figure3-2: SBG Labialization

Palatalization: like labialization, several morphological processes cause consonants to be palatalized (sound change). For example, the second-person feminine singular form of verbs in SBG that are similar with Amharic: {*kft*} 'open', *təkäft* 'you (M.) open', *təkäfč* 'you (F.) open'. Dental and velar consonants can be palatalized [47, 48].
The following figure 3.3 shows that how the consonants can be palatalized.

$t \rightarrow \check{c}$, $\underline{t} \rightarrow \underline{\check{c}}$, $d \rightarrow \check{g}$, $s \rightarrow \check{s}$, $z \rightarrow \check{z}$, $k \rightarrow k^{y}$, $\underline{k} \rightarrow \underline{k}^{y}$, $g \rightarrow g^{y}$, $x \rightarrow x^{y}$.

Figure 3-3: SBG palatalization

To represent the palatalized consonants not found in Ge'ez, Amharic, or Tigrinya, modified characters were introduced to the script, such as using wedges on the tops. The original use of this was done in the New Testament published by the Ethiopian Bible Society, then for the entire Bible; it has now become generally adopted.

3.3 Guragigna Sentence Structure

Linguistics defined sentence as; a textual unit consisting of one or more words that are grammatically linked. When viewed from structural point of view, it is a result of the combination of two phrases NP and a VP as its immediate constituents. They can also include words grouped meaningfully to express statement, questions, exclamation, request, command or suggestion. Moreover, the meaning of sentence is analyzed form the meaning of individual words and the way they arranged.

Various languages classify sentence depending on their purpose, structure and so on. In Guragigna language sentence can be classified based on the number of verbs they contain in sentences. Those are simple sentence and complex sentences.

- ✓ A simple Guragigna sentence: Those are sentence types that consist of subject and one verb. For example, ሁት ወሐኸነም/ he come. In this example, the verb is ወሐኸነም (come) and the subject going to express the verb is ሁት (he).
- ✓ Complex sentence is a sentence contains two or more clauses and at a least one clause is made dependent by subordination.

3.4 Word Class of Guragigna

Every language has its own forms of sentence construction system and rules. This implies that arrangement of words with sentences can be varied from language to language; even if the required information from the provided sentences was the same. However, similar to most of other Semitic languages, Guragigna sentence have word order form of SOV (Subject-Object-verb).

We put words into categories or lexical groups, according to how they work within phrase, clause or sentence. Word class of Guragigna language can be divided into two broad categories. Those are closed class type and open class type. Closed classes have relatively fixed membership for example pronoun, preposition, and conjunction and so on. Generally, they are functional words or grammatical words and they serve to link up open class words in longer meaning structure. They are very short, occur frequently, and play an important role in grammar [30]. By contrast open class is a type that larger numbers of words are belongs, and new words are continually coined or borrowed from

other languages. Noun, Verb, Adjective and adverb are some examples of open class category. The main criteria to determine category of given word are the meaning of the word, the form of the word and environment of the word in a sentence [10].

Generally, in this study five open classes are identified that are Noun, Verb, Adjective, Adverb and Numbers. And five closed class are identified. These are Pronoun, determiner, Preposition, Conjunction, and Interjection.

In terms of group, we group them into ten common or main word categories and six sub word categories that are described below.

3.4.1 Noun and Its Subclass

Noun is the name given to the lexical class in which the words for most people, places, or things occur. But since lexical classes like noun are defined functionally rather than semantically, so some words for people, places, and things may not be nouns and conversely some nouns may not be words for people, places, or things [30]. Guragigna nouns, like English are words used to identify classes of people (for example Fitsum), or idea (for example love), things (for example Book), place (for example Bahir Dar).We can use nouns in different ways or functionality for example አፈጠረን ይሮጥን (fast runner).In this example, the noun is ይሮጥን (runner) because it refers to person.

Traditionally nouns are classified into proper nouns and common nouns [30]. Proper nouns are names of specific persons or entities like ሃና/Hana, ክትፎ /Kitfo, ማርካቶ /Markato etc. And common nouns which are used to call group of things that share common properties however it is not used to all specific things. They are also divided into count nouns and mass nouns. Count nouns are those that allow grammatical enumeration; that is, they can occur in number both the singular and plural (ጤ/ ‘goat ‘/goats‘ጣይ’) and they can be counted (አትጤ/ ‘one goat’). Mass nouns describes composites or substances and used when something is conceptualized as a homogeneous group. For instance, አሾ(salt), ስኔ(wheat), ዝራቦ(rain) and so on.

In this thesis work, we identify general one noun class and two subclasses of noun which are described below.

- ✓ Nouns such as common nouns, concrete nouns, abstract nouns etc. That can be classified or tagged commonly as NN. Examples are: ማዔበ (food, Common noun), ኻሪኹታ (arawit, Concrete noun), etc.
- ✓ Nouns that can be attached with preposition and can't be separate from the noun. Such nouns are classified as noun with position that can be tagged as NPREP. Example: በምሳሌ.
- ✓ Nouns that characterize the name of a specific person, place, thing, organizations, etc. are tagged commonly as NP. Example ቋንቋንዳ/ our language.

3.4.2 Verb and Its Subclasses

The Verb is a word that tells us the state of doing or being. A sentence without verb cannot give a complete meaning. So, Verbs are the most important part of any text almost in any language. A lot of words with other POS are derived primarily from verbs. They represent action, command or assertions.

According to [14], there are two major approaches to identify verbs from other word categories: syntactical and morphological approach. In the former case, verbs function as predicates in a simple sentence and they are found at the end of a sentence. In the latter case, they reflect grammatical categories such as aspect, mood and agreement.

In this study, we identified one general verb (V) class and one subclasses of verb (VP) which are described below.

- ✓ VP is subclass of verb that usually expresses time, location, and manner are tagged as VP. Example በሰብኛኽ (by people).
- ✓ Verbs that cannot classified under the above class are generally considered or tagged as V. For example, ትቢዮ (you said).

3.4.3 Adverb

An adverb is a part of speech, it is any word that modifies any other part of language: verb, adjective (including numbers), clause, sentence and other adverbs except nouns; modifiers of nouns are primarily determiners and adjectives. It specifies the location or direction (for example here, Directional or locative), describe extent of some action (for

example extremely, Degree), about manner of action (for example slowly, Manner), describes about event or some action took place (for example yesterday, Temporal).

Like English, in Guragigna language modifiers of verbs or verb phrase usually expressed by giving information: how the action occurs (manner); where the action occurs (time); how many times the action occurs (frequency).
For more clarification, let as look examples of Guragigna adverbs that are exemplified in table 3.4 below.

Types of adverb	Manner adverb	Frequency adverb	Place adverb	Time adverb
Example	ሂት **አነችም** ዝርኬችም ፣ የዝረኬችን ይናሰጣነም። She spoke very **loudly**. We could all hear what she was saying.	**አታትጊ** ይያ ምሸት የጨነችወ ከጊ ይትረሴ። I **sometimes** forget my wife's birthday.	የረ ሰብ ትያ **መዬ** ቅመም። There was somebody standing **nearby.**	ሃና **ሸኳ** አሹሽናም። Have you seen hana **today**?

Table 3.4: Guragigna adverbs

In this thesis work, generally, we identify adverbs in one common adverb class that are described below.

- ✓ Any form of adverb that appears in the sentence are tagged or considered as ADV.

4.4.4 Adjective and Its subclass

The word that describes or clarifies a noun in a sentence is known as adjective. Its main purpose is to give clear explanation for noun and give us more information about people, things or animal represented by noun or pronoun. It includes many terms that describe the properties or qualities. Many languages have adjectives for the concept of color, age, value and so on.

Like other languages Guragigna is usually describes nouns by giving some information about an object's behavior (for example ወኼ/good', ጣፎ/bad), weight (for example ጀንጅር/fat, ስሰ/thin, ንቅየ/big, እስየ/small), height (for example ጌፍ/long, አጨሩ/short) size, shape, color (for example ጻድ/white, ብሻ/red) of the noun. For more clarification let us

look the following example: ሀም በምርካሳ ቤት ይረብሮ which means “They live in a beautiful house”. In this sentence the word ምርካሳ is used as adjective that describes the house.

For tagging purpose of our work, we identify one general adjective class and one sub-class that are explained below.

- ✓ An adjective that are tagged with preposition are considered or tagged as ADJPREP. For example, የጉራጌ
- ✓ An adjective that are tagged with pronoun are considered or tagged as ADJPRON. For example, ኧኧመታ
- ✓ Any form of adjective that are not classified under the above class are generally considered or tagged as ADJ. For example, የቀንጥኽ (reviled).

3.4.5 Prepositions

Prepositions are words that don’t convey any meaning alone unless they are attached with other basic classes like noun, adjective, adverb, etc. In constituent of structures, their position may precede (preposition) or follow (postposition) the word category in which they form a syntactic unit. Their role is expressing relationship or association among things, person, event or others.

In Guragigna language prepositions are placed together with verb, noun, or pronoun. Examples of preposition and postposition in this language includes: ኽማ (as), የ (to), ቦሰጥ (below), ተፎሬ (above) በ (by) and so on.

In this thesis work we identify preposition as only general class that are described below:

- ✓ Any form of preposition that appears in the sentence is generally tagged as PREP.

4.3.6 Pronoun and Its Subclasses

As linguistics defines, pronoun is a word that substitutes for noun or noun phrase. It is used to create inference for a given text to refer the preceding sentence. Additionally, when name of thing is not known explicitly, pronouns are used to convey message.

There are different categories of pronoun, that includes personal pronoun (for example he, I, we, she, they), demonstrative pronoun (for example this, that, these, those),

possessive pronoun (for example mine, our, your, his), reflexive pronoun (myself, oneself). Like other languages Guragigna also share some of the above-mentioned characteristics. Let us see the following Guragigna personal pronouns listed below in table 3.5.

Table 3.5: Guragigna pronoun classes

Person	**Singular**	**Plural**
1st	እያ(I), ትሁ(me)	ይና(we), የናቱ (us)
2nd	ህም (you)	ህም (you)
3rd	ሁት(he,she), ሂት(she/her, it)	ሁነህኖ/They, them

With absence of pronoun we have to repeat nouns in sentence or communication but that make our speech or text unstructured. Most pronouns are very short. Consider the following example shown below in table 3.6 which is pronoun replace noun in the sentence.

Table 3.6: pronoun replaces nouns

Noun	*Pronoun*
ፍጹም የኮምቲዉተር ሳይንስ ተማሪዉ። **Fitsum** is computer science student.	**ሁት** የኮምቲዉተር ሳይንስ ተማሪዉ። **He** is computer science student.

In this the above in table 3.6 shows that pronoun which is ሁት reference as the nouns which they replace ፍጹም. Generally, in this study, we identify pronoun into one general class that are tagged as PREP.

3.4.7 Numerals

Numeral includes a word that refers to number or quantity of something. They can be classified as cardinal number and ordinal numbers. Cardinal number refers to the counting numbers because they show quantity. For example, አት (one), ኼት (two), ሶስት (three), አርበት (four) are cardinal numbers. On other side Ordinal number tells us the order of things and their rank. Examples of ordinal number are አታነ (first), ኼታነ (second), ሶስተነ (third)አርነተነ(fourth) and so on. As seen from this example Guragigna ordinal number adds suffix -ነ at the end of word.

- ✓ In this thesis work, we identify two sub classes class of numeral that is represented with ORDNUM and CARDNUM that means ordinary number and cardinary number respectively.

3.4.8 Punctuation

The Guragigna writing system uses some homegrown punctuation. For example, arat netib(።), dirib serez (፤), netela serez (፥) and foreign or borrowed punctuation mark such exclamation, question mark etc. The punctuation that is used in this work described below in table 3.7.

Table 3.7: Guragigna Punctuation

No.	Punctuation Mark	Symbol	Purpose
1.	Arat netib (full stop)	።	Marks end of a word and at the same time the end of a sentence.
2.	Dirib serez (colon)	፤	Used to connect two sentences, clauses.
3.	Hulet netib (comma)	፥	Separate individual words in a sentence.
4.	Question Mark (Tiyake milikit)	?	Marks the end of an interrogative sentence.
5.	Exclamation (Kale agano)	!	End of emphatic declaration or command.
6.	Quotation mark (Timhirte tiks)	“ “ ‘ ’	Used at the beginning and end of words that is being quoted.
7.	Preface colon	፡-	Introduces speech from a descriptive prefix or beginning of list mark.

- ✓ In this thesis work all punctuation words and other unique symbols are assigned tag are PUNC.

4.4.9 Interjection Class

Like English, Guragigna has many words or phrases that are used to express sad emotions or strong feeling like sudden surprise, pleasure, annoyance and so on. Words

that are used for such purposes are called interjections. For example, words like ወ! ወኬቀ! (goš! ‘Good job!’), so on are common interjections in the Guragigna language. They can stand alone or can appear anywhere in a sentence.

- ✓ In this work; we identify interjection as a one general class that is tagged with INT.

3.5 Guragigna tagsets

The detail description of word categorization in Guragigna was described in the previous section. However, in this section the actual tagsets[1] which is used in this thesis work are determined and discussed. As far as the researchers’ knowledge is concerned, there is no publicly available tagsets for Guragigna language. In order to identify and develop tagsets, the researchers have made interview questions that are show in appendix E and discussion with Guragigna language professional Ato Bahiru Lilaga (from Gurage zone linguistic and culture communication officer) and Ato Getinet who is an instructor in Walkite University.

In this work, we classified the tagsets as basic classes and sub-classes. The basic classes are noun, verb, adjective, pronoun, adverb, preposition, Numeral, Conjunction, Interjection and punctuation is considered that are described below. In addition to these different sub classes of Guragigna language are included. Generally, we summarize the overall tags that are used in this work described in the following table 3.8.

[1]tagsets refers to a collection of tags which is used to mark word classes of a text corpus

3.6 Summary of Guragigna tagsets

Table 3.8: Summary of the Guragigna tagsets

T.N	Basic Class	Derived Class	Description	Examples
1.	Noun (NN)	N	Noun	መርካቶ(Merkato), ክትፎ(Kitfo)
2.		NPREP	Noun with preposition	በምሳሌ(with example)
3.		NP	Noun Phrase	ጥብንቹ
4.	Pronoun(PRON)	PRON	Pronoun	እያ(I), ይና(we), ህም(you)etc.
5.	Verb(V)	V	Verb	ጀገሮቹ
6.		VP	Verb phrase	ወረወረም
7.	Adjective(ADJ)	ADJ	Adjective	ወኬ (good), ጣፎ (bad), ጌፍ (long)
8.		ADJPREP	Adjective with preposition	የጉራጌ
9.		ADJPRON	Adjective with pronouns	ኧኧመታ
10.	Adverb(ADV)	ADV	Adverb	አነችም (loudly), አታትኒ (sometimes), መዬ (nearby), ኧካ (today).
11.	Preposition(PREP)	PREP	Preposition	ኸማ (as), የ (to), ቦሰጥ (below), ተፎሬ(above),በ(by)
12.	Numerals(NUM)	NUMCR	Cardinal Number	አት (One), ኼት (Two), ሶስት (Three).
13.		NUMOR	Ordinal Number	አታነ (first), ኼታነ (second), ሶስተነ (third).
14.	Conjunction(CONJ)	CONJ	Conjunction	ቄም
15.	Interjection(INT)	INT	Interjection	ወ! (gosh!)
16.	Punctuation(PUNC)	PUNC	Punctuation marks	፣፤ ,፡፡ ,?,፥,፤
17.	Unknown(UNK)	UNK	Unknown word	

CHAPTER FOUR
DESIGN OF GURAGIGNA POS TAGGER

4.1 Overview

Part of speech tagging is often used as a prerequisite for more complex NLP applications such as information extraction, syntactic parsing, machine translation or semantic field annotation etc. [13]. Guragigna POS tagging is a method of assigning a specific Guragigna part of speech tag to each word in a sentence to disambiguate the function of a word in the specific context. As discussed in section 2.2, in this work hybrid based part of speech tagger is used which uses lexical and contextual rule to assign given POS. This POST at the start uses statistical technique to extract information from training corpus that use program to learn rules which reduces the fault that introduced by the statistical mistake [23].

Generally, in this chapter, we will discuss the detail description of design issues and techniques of Guragigna POST and the algorithm that are used to design for HMM tagger. The rule based tagger used in this work also discussed. Finally, the architecture, algorithm used for Hybrid taggers are described. And at the end of this section summary of the chapter will be described.

4.2 Approach and Techniques

During assigning part of speech for the words, there are problems that might occur in tagging of POS. These problems can be solved using different approaches. For example, using HMM approach, ANN approach, Rule Based approach and hybrid based approach.

As discussed earlier in section 2.2.1, 2.2.2 and 2.2.3using of separate approach either of HMM, ANN or rule based approach may have their own drawback. However, it is possible to improve the performance of tagger by combining of two approaches [14, 12]. The reason is that they gain shared benefits from individual approaches.

The main aim of this thesis work is overcoming the problem mentioned above by combining two approaches. In this study, HMM and hybrid approach are used and will show the performance of combining tagger is better than single HMM tagger.

During the process of tagging with statistical approach, extracting the statistical property at training phases are done in order to label word with their part of speech tag. One of the most widely used statistical approach for POST is HMM approach.

4.3 Design of HMM tagger

A Hidden Markov Model is a probabilistic or extension of finite state machine that used in speech and language processing. Markov chain is useful when we need to compute probability for sequence of event that we may observe in the word. The events we are interested in may not be directly observable in the world. For example, in part-of speech tagging we do not observe part of speech tags in the world; what we know are the words, and had to deduce the correct tags from the word sequence. So, the hidden part is called as part-of speech tags because they are not observed. The HMM allows both observed model events (like words that we see in the input) and hidden events (like part-of-speech tags) that we think of as causal factors in our probabilistic model [49].

States in HMM can be represented by the set $S = \{S_1, S_2, ..., S_n\}$ which are invisible part. So, what visible is a set of Observations $O = \{O_1, O_2, ..., O_m\}$ that are the result of the machine moving from one state to the other. The probabilities of the machine starting in one of the states S_i is specified by the one-dimensional matrix Π of size n. The probabilities of the machine moving from one state to another is specified by a two-dimensional matrix A of size n*n. Finally, the probability of an observation being observed when the machine is in a certain state is given by the two-dimensional matrix B of size n*m. Formally HMM can be characterized by [46]:

$H = f(\Pi, A, B)$

Where:

- ✓ H = The HMM
- ✓ Π = Start probabilities. The element Π_i represents the probability that the HMM starts a sequence in State S_i, where i in (0...n-1).

- ✓ A = Transition probabilities. The element $A_{i,\ j}$ represents the probability of a transition from State S_i to State S_j, where i and j in (0...n-1).
- ✓ B = Emission probabilities. The element $B_{i,\ j}$ represents the probability of an Observation O_j occurring while the machine is in State S_i, where i in (0...n-1) and j in (0...m-1).
- ✓ n = Number of states (tags in our case).
- ✓ m = Number of unique observations.

As stated in section 1.1 the main objective of POS tagging is to tag each word of a sentence with its appropriate part-of-speech tag. However, some words can be unambiguously tagged. The fact that the word exists in the sentence is known. Though, the POS for the word is unknown. Therefore, the HMM built for POS tagging that models the words as visible observations and the set of possible POS as the hidden states.

As far as POS tagging is concerned, the main problems that can be solved by HMM are finding the most likely state sequence for a given observation sequence. This is useful for word sense disambiguation (WSD). So, we can tell the most likely POS that a particular word in a sentence belongs to. These problems are solved using the Viterbi algorithm that will be briefly discussed in section 4.3.2. Before Viterbi algorithm let us discuss how the HMM are trained.

4.3.1 Training HMM Tagger

The training phase of the tagger starts first by running the tagged corpus. Then splitting tagged corpus take place. In order to separate out the word, tags and preparing for probability calculation tokenization module performed. So, given a sentence by using of words w_1, w_2, w_3...w_n the tagger need to finds the set of tags. So, this is can be performed using equation 4.1 given below.

$$P(word|Tag) \tag{4.1}$$

Where,

- ✓ P (word | Tag): Is the probability of a word being assigned a particular tag from the list of all possible tags for the word (most frequent tag).

Tokenization at word level is done for the purpose of estimating transition (contextual) and emission (lexical) probability for preparation of probability tag sequence. The transition probability is calculated using the following equation 4.2.

$$P(ti|ti-1) = \frac{C(ti-1, ti)}{C(ti-1)} \quad (4.2)$$

Where,

- ✓ $P(ti|ti-1)$: Probability of tag given previous tag.
- ✓ $C(ti|ti-1, ti)$: Count of tag sequence t_i-1, t_i in the corpus.

And the lexical probability is calculated using the following formula.

$$P(wi|ti) = \frac{C(wi, ti)}{C(ti)} \quad (4.3)$$

Where,

- ✓ $P(wi|ti)$: Probability of tag given previous tag.
- ✓ $C(wi, ti)$: The count of word i $C(w_i)$ is assigned to tag $C(t_i)$ in corpus

After transition probability and likelihood probability calculated, it given to viterbi matrix calculator to calculate the probability of the most-probable path of tags sequence in a given corpus that described in the next section.

4.3.2 The Viterbi algorithm

The optimal sequence of part of speech tags for a given sequence of words in an input sentence to be tagged can be found using the Viterbi algorithm [34, 8, 17]. The Viterbi algorithm is a dynamic programming algorithm that finds the optimal path in the tagging process. In simple terms, the Viterbi algorithm calculates the probability of all possible paths of the word tag pairs in the input sentence. Afterwards, it selects the path of the word tag pair with the highest probability to be the best path [34, 17]. It uses the lexical and contextual probabilities obtained from the lexical and contextual model to find the best path.

As stated by [30], Viterbi algorithm performs tagging processes in three steps which are initialization, iteration step and sequence step. The following algorithm shows how the

three steps are performed in Viterbi to return best path state sequence. The Viterbi algorithm used for HMM tagger is described below in figure 4.1.

Function VITERBI (observations of len T,state-graph of len N) **returns** *best-path*

create a path probability matrix Viterbi[N+2, T]

for each state s **from** 1 **to** N **do** // Initialization step

viterbi $[s,1] \leftarrow a_{0,s} * b_s(O_1)$

backpointer $[s,1] \leftarrow 0$

for each time step t **from** 2 **to** T **do** // Recursion step

for each state s **from** 1 **to** N **do**

viterbi $[s, t] \leftarrow max_{s'=1}^{N}$ Viterbi $[s' t-1] * a_{s',s} * b_s(o_t)$

backpointer$[s, t] \leftarrow max_{s'=1}^{N}$ Viterbi $[s', t-1] * a_{s',s}$

Viterbi $[q_F,T] \leftarrow max_{s=1}^{N}$ Viterbi$[s,T] * a_{s,q_F}$ // Termination step

Backpointer$[qF,T] \leftarrow argmaz_{s=1}^{N}$ Viterbi $[s,T] * a_{s,q_F}$ // Termination step

return the backtrace path by following backpointer to states back in time from *Backpointer*$[q_F, T]$

Figure 4-1: Viterbi algorithm for finding optimal sequence of tags

In general, the working principle or architecture of HMM tagger is described below in figure 4.2 using diagram with its detail description.

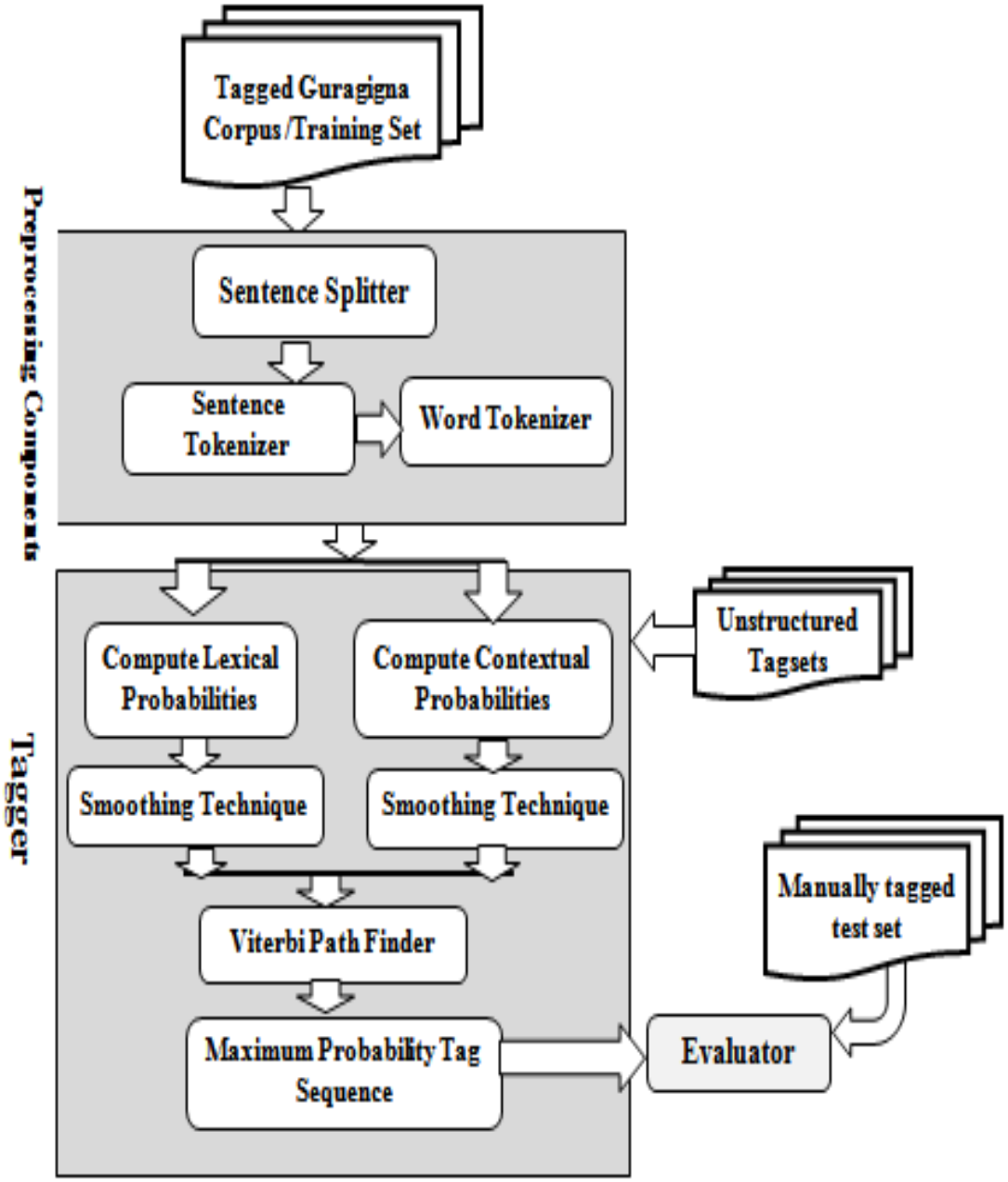

Figure 4-2: HMM Tagger Model

In the above figure 4.2 the training corpus that is tagged with Guragigna text is an input to the model. Next, the tagged Guragigna corpus is given to sentence splitter module in order to prepare it for the training at sentence level. Afterward, the segmented sentence is given to tokenizer for splitting each sentence to token level. After each sentence is tokenized in to word, the lexical and contextual probabilities are computed.

Before training the tagger, the computed probability must be smoothed in order to address the poor estimation due to variability in small datasets, words which are not in vocabulary, the designed system required cautious handling of such small numbers and zero probabilities at various points. First, propagating and multiplying partial probabilities in the induction step of Viterbi lead to number under flows. In order to eliminate this problem, the Laplace smoothing is being used after computing the transition and lexical probabilities.

It is smoothing technique which adds one to all counts before computing the lexical and contextual probabilities. Later, maximum probability of tagged sentence computed. And finally, the annotated sentences that are trained in the tagger can be evaluated with manually tagged test set or reference with the help of evaluator.

4.5 Design of Hybrid Tagger

As discussed previously in section 4.1 for this study hybrid tagger is used. It consists of HMM tagger and the Rule based tagger. At the starting HMM tagger acts as an initial tagger for hybrid tagger. In order to train, the tagger accepts tagged training set and unstructured tagsets. In addition to this unlabeled Guragigna test sets are an input.

From unstructured tagsets (containing of tags, their description, examples) we need only tagsets for sake of training. This can be done by jumping the line before equal to sign. Since tagsets are written in form of tags then equal sign and followed by their description and example in a line.

The tagsets are used as unique column for contextual model. Each individual tagsets placed horizontally and vertically in the form of matrix. Then it cheeks which tags with whom it occurs and how many times. So, the view of the HMM tagger is considered as

an application that takes annotated raw corpus in addition to unstructured tagsets file. But first the HMM tagger split the tagged training set using space. After annotated row corpus processed by the sentence splitter module and tokenizer module in preprocessing components as input it gives to contextual model. Similarly, structured tagsets are pass to lexical model in order to train for the sake tagging process for later use as an input during the tagging of new text. Before passing initial parameters to Viterbi algorithm, lexical model is generated based on unlabeled test set and tagsets from HMM tagger module. Based on the output of contextual and lexical model the Viterbi algorithm gives us tagged Guragigna text. This tagged text is in form of tag sequence and can be input for rule based tagger. The rule based tagger accepts another input which is manually tagged. Next it cross checks tag sequence with manually tagged text. If there result not match with specified condition compared to manually tagged test set, then the rule will be applied. Finally, the rule based tagger produce optimal tag sequence that are again checked with manually tagged test set with help of evaluator.

The high-level view architecture of the hybrid tagger used for this work is given below in figure 4.6.

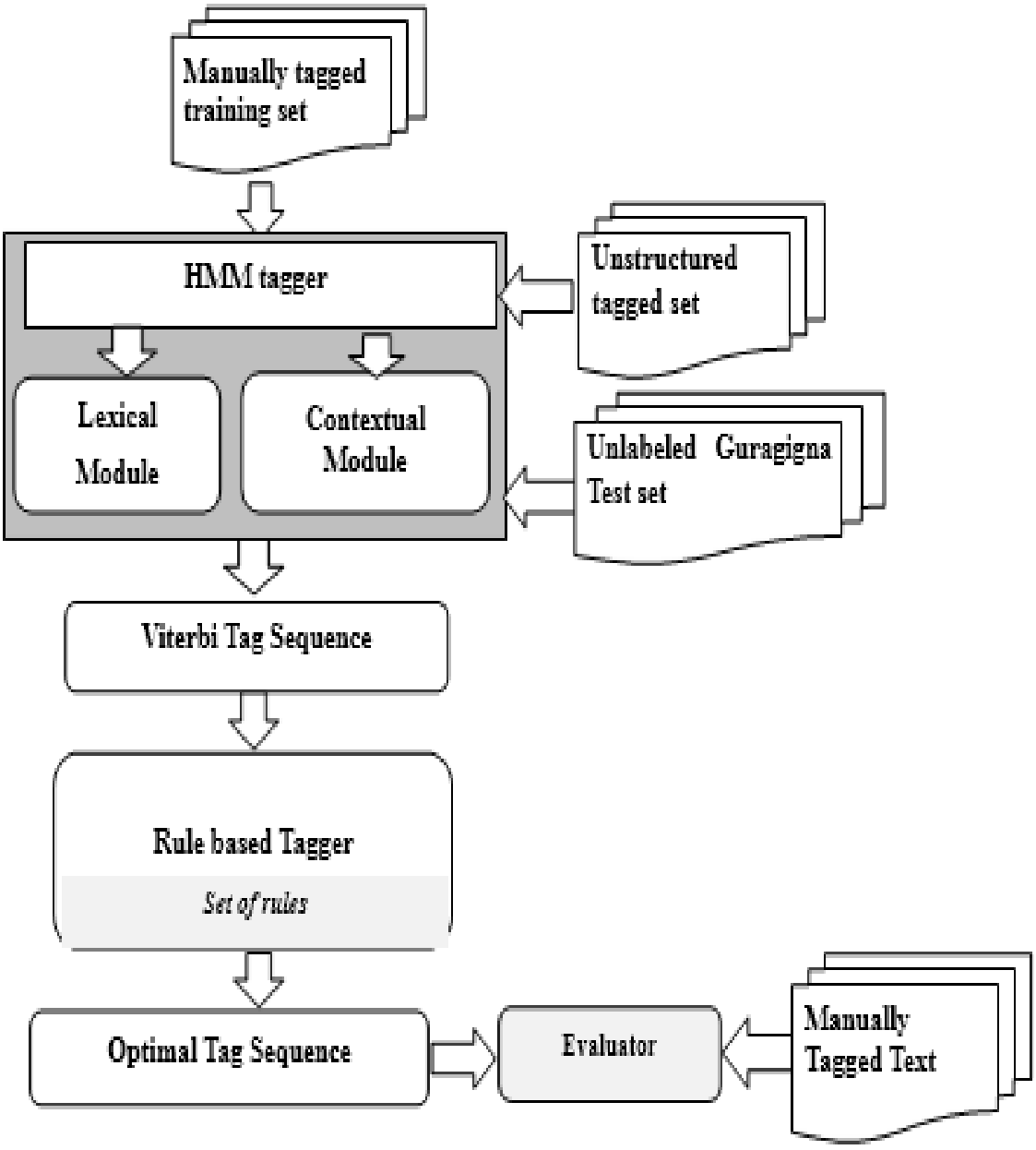

Figure 4-3: Architecture of the hybrid tagger

Algorithm for Hybrid tagger are shown below in figure 4.7

1. Get HMM Model Tagger
2. Read raw text
3. Tag the raw text using HMM tagger
4. Get HMM word to tag sequence
5. Split sequence to word/tag pair
6. For each word/tag pair Loop
    6.1. If apply Rule sequence (model-word-tag-pair,)
        6.1.1. Predict possible Category (Word, Tag)
    6.2. Else
        6.2.1. Continue;
7. End Loop
8. Generate Optimal Tag sequence

Figure 4-4: Algorithm of Hybrid Tagger for Guragigna Language

4.6 Summary

POST is the process of marking up a word in a text (corpus) with corresponding particular part of speech. All through during assigning part of speech there are problems that occurs in tagging with their appropriate POS, these problems can be solved by using different approaches such as HMM approach, Rule Based approach and hybrid based approach. However, each approach alone has pros and cons. For sake of part of speech tagger with possibly a higher performance of tagging is desired, there is a need to take the advantages from two or more different approaches thereby remedying the shortcomings of the approaches. In our work, we use statistical and rule based approaches.

The statistical approach obtains the statistical properties of words in the training phase to label words with their correct part of speech. One of the most widely used models in the statistical approach for POST is the Hidden Markov Model. The main goal of this model

is to assign an optimal sequence of part of speech tags to a sequence of words in a given sentence. The problem of finding the optimal sequence of part of speech tags to a sequence of words can be done using Viterbi algorithm. Therefore, the Viterbi algorithm is adapted for this Hidden Markov Model constituent of the tagger.

Rule based tagger uses a set of rules as a basic component of the tagger. Rules can be constructed either manually using linguistic professionals, or using data automatically learning by the machine i.e. the rule is induced directly from the training corpus without human intervention or expert knowledge. Constructing rules manually are time consuming, labor intensive and it requires linguistic expert that has deep knowledge of the concerned language. However, deducing rules automatically is simple and easy but it may result rules that don't represent morphological and syntactic property of the language. In this thesis work we use rules that are constructed using linguistic professional.

A hybrid approach combines the best properties of these two approaches to get better accuracy. The HMM based tagger first tags the Guragigna sentence and tag words based on the most probable path for a given sequence of word. Then the HMM tagged word sequences are given for the rule based tagger for correction based on predefined test of rules.

CHAPTER FIVE
IMPLEMENTATION AND PERFORMANCE ANALYSIS

5.1 Overview

In this chapter, we will present the detail implementations of Guragigna POST and experiments conducted in this study. The sections are organized based on the flow of task described in the architecture. First, we will give a brief explanation regarding of corpus used. Next, we will discuss how the component in each phase implemented starting from preprocessing phase to tagging. Finally, the performance evaluation will be discussed based on the experiment analyzed.

For sake of implementation, Java programming language is used for implementation of HMM and Hybrid tagger. Rationale behind choosing of java programming language is, relatively richness with natural language processing in built model than other programming language like C++ and .NET. The main benefit of java is that it can be used to create platform independent application. In addition to that any computer or mobile device can able to run java virtual machine with application, freely available on internet and has a lot of libraries [38, 48]. And CRF++ are used for implementation of CRF tagger. The reason to use this tool is a simple, customizable, and open source toolkit of conditional random field for segmenting or labeling sequence of data [52].

5.2 Corpus preparation

According to different scholar's corpus can be defined in various ways. However, as linguistics defined it is a collection of linguistic data, either compiled as written text or as transcription of recorded speech. The main purpose of corpus is to verify a hypothesis about language. For example, to determine how the usage of a particular sound, word, or syntactic construction varies [53].

Specifically, corpus having additional linguistic information is called as annotated or tagged corpus that can have part of speech information, sentiment information that specify the word class category and corpus without linguistic information are called

unannotated corpus [49]. Annotated corpus can be used in many NLP applications such as part of speech tagger, parsing, sentiment analysis and so on.

Also we can classify corpora into two based on their genre. Those are balanced corpus and category specific corpus. A balanced corpus must contain or represents all domain of the language such as text of news category, fiction category, editorial category, scientific category, academic category. Developing of balanced corpus increases the performance of tagger but getting this all different domains areas needs time, effort/skills of language experts and money. Category specific corpus as its name implies specific category of domain [14, 32].

As far as researcher knowledge is concerned there is no corpus developed for the Guragigna language. In this thesis work, dues to considering the above difficulties, we collect only from two domains. First, from Guragigna fiction called የቈሙኑት ሽካ [54] and secondly from editorial domain category. For this study, we used 4613 words collected from fiction and 2132 words from editorial domain. All data are tagged manually with help of linguistic professional. Totally 6745 words or 307 sentences are collected for training and testing.

To prepare corpus for the study, the collected data are given to linguistic professional for sake of tagging texts manually. During tagging process seventeen (17) tagsets are identified as discussed in section 3.5. Then, finally these manually tagged corpus are used for training and testing.

After tagging the text manually, for the experimentation, the entire corpus is divided into 90% for training that is consist 276 sentences and 10% that is 31sentences used for testing purpose. The reason to choose 90:10 is relatively having of a small data set. Sample corpus tagged and untagged data are shown in appendix D.

5.3 Preprocessing component

As discussed in section 4.3.2 preprocessing components have mainly two modules. Namely sentence tokenizer and sentence splitter modules.

Before starting preprocessing, tagged corpus is needed as an input. The corpus may or may not transcribe in Latin script. But in this work no need of transcribing the corpus into Latin script. The reason is that the corpus is readable by machine. Since the machine supports UTF-8.

Initially the corpus reader java class reads the tagged file from specified location for purpose of processing by splitter and tokenizer module. After reading the file it gives to sentence splitter.

Then the sentence splitter takes corpus and split it based on Guragigna end marker that are one of from the characters '።', '?'Or '!'. Another task of sentence splitter is accepting unstructured tagsets which contains tag, description, and example and extracts tags from the file. The purpose of extracting tagsets is for the sake tagging process for later use as an input during the tagging of new text.

For language processing, we need to break up the corpus into word and punctuation level. So, this task is performed by tokenizer module. Tagged word or punctuation was tagged in the form of word or token and its corresponding tags during training phase. Here the word and its POS are separated using forward slash character (/). This provides the tagger to compute statistical information for both word and part of speech tags. Then it extracts each line from corpus by looping through all files in the given folder since each line is stored as a sentence. This preprocessing component (tokenization) is done during training as well as the testing phase.

5.4 Implementation of HMM Tagger

The probabilities used by the tagger are calculated from the tagged training corpus. There are two probabilities that are used by the tagger. These are lexical (emission probability) and transition (contextual probability).

Lexical probability indicates that p (wi/ti) . Thisrepresents the probability of given a tag that will be associated with a given word i.e. the probability of a word being assigned particular tag from the list of all possible tags. It can be estimated by computing the relative frequencies of every word per category from the training annotated corpus. All

statistical information, that enables to develop probabilities, are derived automatically from a hand annotated corpus (the lexicon).

For example, to find probability of ዝህ to be determiner: P(ዝህ/DET)

C (ዝህ, DET)=11,

$C(\text{DET}) = 150$ so,

$p(\text{ዝህ}, \text{DET}) = \frac{count(\text{ዝህ/DET})}{count(\text{DET})} = \frac{11}{150} = 0.073$

Where, C and P are count and Probability Respectively.

The following table 5.1 shows that sample word taken from corpus. This describes given probability of word with respect to their correspondence and computed lexical probability of word for specific part of speech category.

Table 5.1: Sample Lexical Probability

Word with given probability	**Probability**
P(ዝህ/DET)	0.073
P(ጾትሁንም/V)	0.000532
P(ኧዉንም/ADJ)	0.002041
P(ሰብ/N)	0.05423
P(ኤነ/V)	0.015949

The transitional or contextual probabilities are the second probability that is used by the tagger. It is computed as p (t/t-1). This is the probability of one tag or information of one part-of speech category followed by other categories (n-gram). Where; t is part of speech category. It is developed from training lexicon corpus. For this study, we use bigram model. Let us see the following examples for more clarification that finds the probability of verb given a noun:

$C(V)$=1881,

$C(N, V)$=1383,

$P(V|N) = \frac{C(N,V)}{C(V)} = \frac{1383}{1881} = 0.735$

The following table 5.2 shows the probability of bigram category tag with given previous tag and with corresponding of computed transitional probability.

Table 5.2: Sample Transition Probability

Bigram category	Probability
$P(V\|N)$	0.735
$P(ADJ\|N)$	0.354
$P(DET\|ADJ)$	0.261
$P(ADV\|N)$	0.363
$P(PRN\|ADV)$	0.357

5.5 Implementation of Hybrid Tagger

As discussed in Design section 4.5, during implementation of hybrid tagger, the tagger is consisting of the HMM tagger and the rule based tagger. The HMM tagger acts as an initial tagger for the raw text of Guragigna to be tagged and gives corresponding tagged output. The output of the initial state tagger is an input for the learning phase for rule based tagger. During learning, if result is not matched with specific condition compared to manually tagged test set, then rule based tagger corrects the output of the HMM tagger by applying rules defined. Finally, testing is done. At testing, new unseen Guragigna texts can be tagged in the trained tagger and cross check with the reference text (i.e. manually tagged of the test corpus) for performance evaluation.

5.6 Experiment with HMM Tagger

In order to see goodness of training set ten different experiments are conducted. The training of HMM tagger start by dividing the entire training set into ten equal sizes. Each size is 10% of the total training set. The training starts with 10% of the entire training set and measure the performance on test set. Next adding 10% of data into previous data until training corpus is desired or 100%. For each training set the performance is measured. The experiment is conducted with different portion of training set with their performance measured. That is shown in table 5.3 below.

Table 5.3: Experiment on HMM tagger with different portion of training set

Training set (%)	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Performance (%)	54.83	58.04	60.7	61.21	62.9	67.42	69.33	72.8	73.86	74.46
Difference Prev value-cur value	0	3.21	2.66	0.51	1.69	4.62	1.81	3.47	1.06	0.7

The experimental analysis of HMM tagger performance curve shows that during starting of training with training set of 10% it gets the accuracy of 54.83 accuracy. Continue by incrementing of the training data and measuring the performance. As we see from experiment analysis in table 5.3 when the training set increases the performance also increase. Finally, when the training data reached to desired the performance it gets accuracy of 74.46.

Figure 5.1 below shows the performance curve analysis of HMM tagger. So, as we see training set and performance are direct proportional. That means when training set increase the performance of tagger also increases.

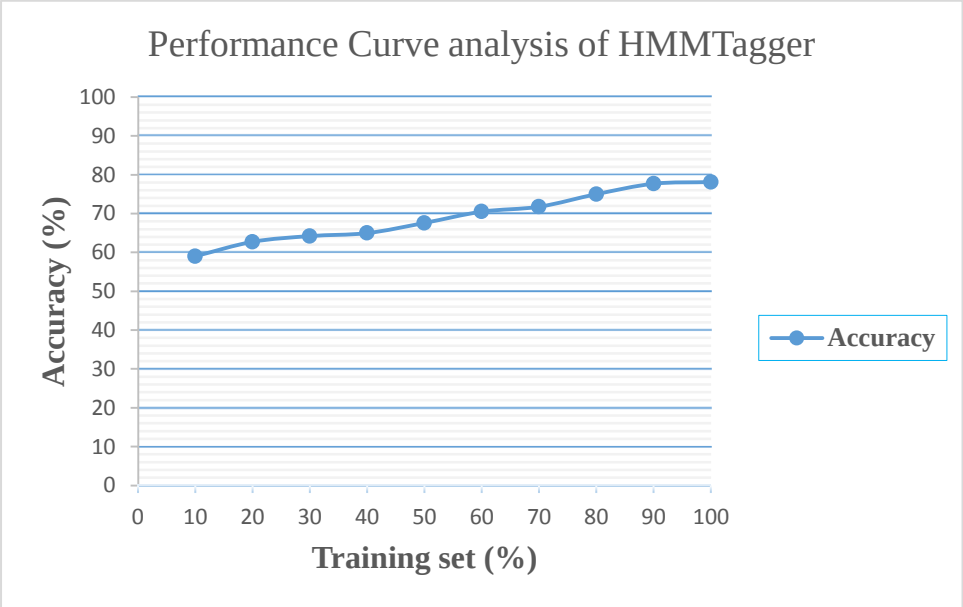

Figure 5-1: performance curve analysis of HMM tagger

5.7 Experiments with CRF

To train and test our data first we transcribe the geez script (Guragigna text) into Romanization or Latin script. This is done only for experiment of conditional random field. As discussed previously in section 5.1 the experiment is done using CRF++ tool. In

this experiment we follow the same procedure during experimenting the only change is tool used in this approach which is free and customizable as described in section 5.1. The experiment conducted with CRF is shown below in table 5.4.

Table 5.4: Experiment on CRF with different portion of training set

Training set	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Accuracy (%)	47.98	49.66	52.01	54.15	57	58.74	61.33	62.14	65.07	66.56
Differences (Prev value-cur value)	0	1.68	2.35	2.14	2.85	1.74	2.59	0.81	2.93	1.49

As shown in the above table 5.4 experimental analyses the result is decreased when compared with HMM approach. The reason is that during translating the geez script into Romanization or Latin script the characters which are found in the language doesn't recognized shown in appendix E and appendix F. So the tool only recognizes the Latin script and ignores other four additional scripts.

The performance curve analysis generated for conditional random field is shown below in figure 5.2.

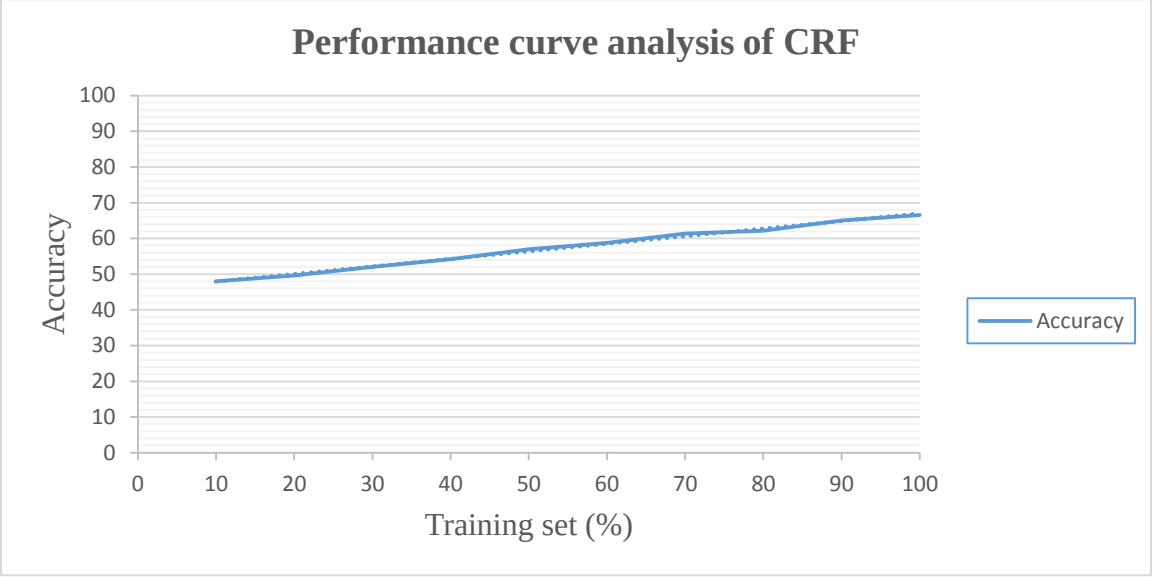

Figure 5-2: Performance curve analysis of CRF

5.8 Experiment with Hybrid Tagger

In order to train hybrid tagger similar experiment with HMM tagger is conducted. The experiment conducted with different portion of training set with respect to performance measured in hybrid tagger is shown in table 5.5.

Table 5.5: Experiment on Hybrid tagger with different portion of training set

Training set	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Performance (%)	59.03	62.74	64.21	64.98	67.6	70.5	71.77	75.01	77.69	78.12
Differences (Prev value-cur value)	0	3.71	1.47	0.77	2.62	2.9	1.27	3.24	2.68	0.43

As describes in section 5.4 the experimental analysis Hybrid tagger uses the output of HMM tagger as input for Hybrid tagger. However, in hybrid tagger there is a little change on performance when comparing with HMM. The reason is that we use common rules in order to correct wrongly tagged in the HMM tagger.

So, Figure 5.3 shows that starting training with 10% of training set and gets the accuracy of 59.03 and when increasing training set to desired level we get the accuracy of 78.12.

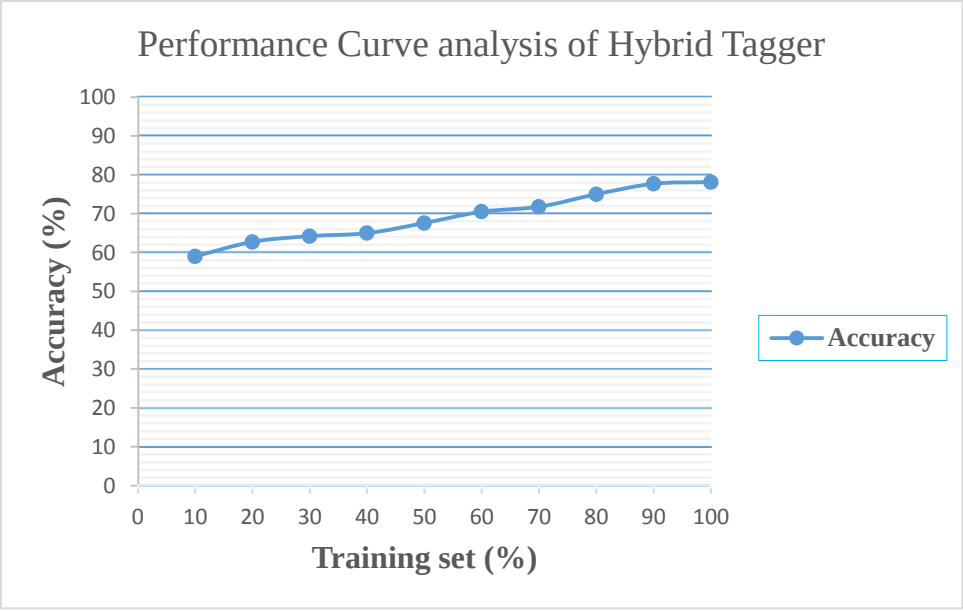

Figure5-3: Performance curve analysis of Hybrid Tagger

- **Comparison of HMM and Hybrid Tagger**

Comparison of HMM and Hybrid tagger using performance curve analysis are described below in figure 5.3.

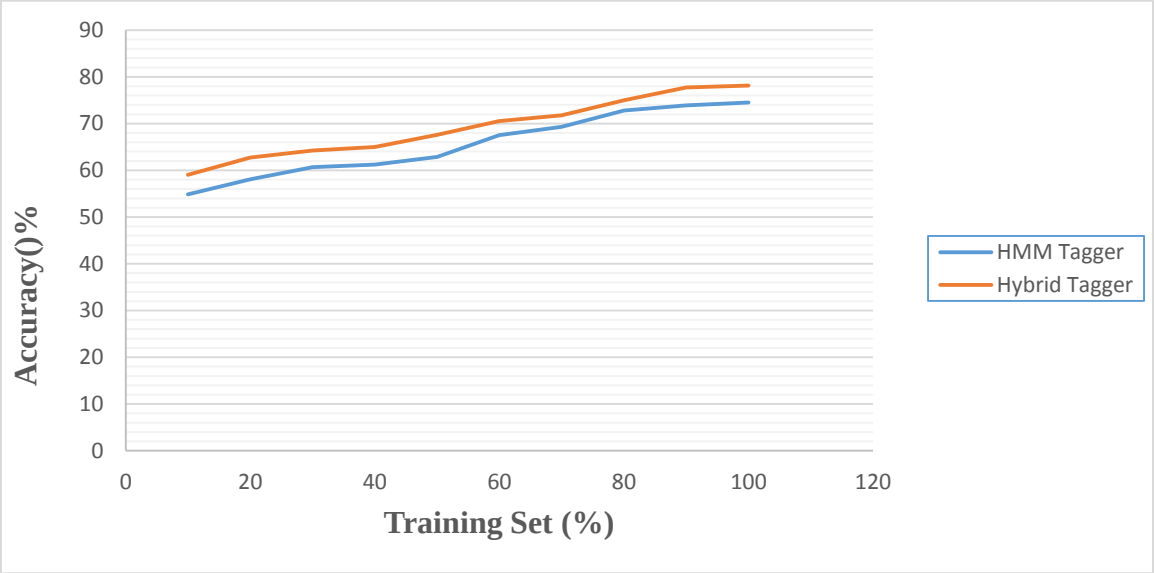

Figure 5-4: Comparison of performance curve analysis on HMM and Hybrid tagger

- **Comparison of HMM,CRF and Hybrid Tagger**

As shown below in figure 5.5 the comparison of three taggers on the test set of data are shown.

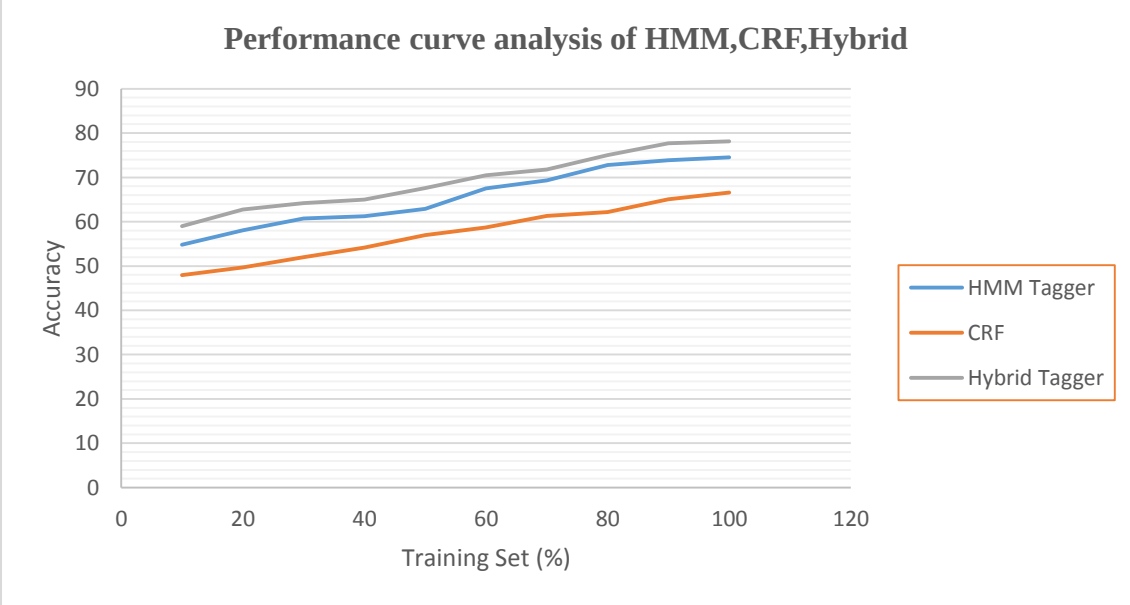

Figure 5-5: Comparison of performance curve analysis on three taggers on training set

5.9 Performance Analysis

The experimental analysis of HMM tagger and Hybrid tagger developed in this work are analyzed using frequency of tagsets. The frequency of tagsets that found in the total corpus, training set and testing set are taken for analyzing the performance of Guragigna tagger. Totally 17 tags are identified for the experiment. Depending on frequency of tags, we divide tagsets into two groups. The most frequent tags that are 12 tags and the rest as others.

The details frequency distribution of tags with total corpus, training set and testing set are shown below in table 5.6.

Table 5.6: Tag Frequency

N.T	Tags		Total corpus		Training set		Testing set	
			Tag frequency	In %	Tag frequency	In %	Tag frequency	In %
1	**V**		2013	**29.84%**	1880	**27.77%**	133	**2.07%**
2	**N**		1542	**22.87%**	1383	**20.43%**	159	**2.44%**
3	**PUNC**		954	**14.14%**	855	**12.63%**	99	**1.52%**
4	**ADV**		593	**8.79%**	504	**7.45%**	89	**1.27%**
5	**ADJ**		548	**8.12%**	495	**7.31%**	53	**0.81%**
6	**INT**		357	**5.29%**	329	**4.86%**	28	**0.43%**
7	**CON**		232	**3.44%**	115	**3.18%**	17	**0.26%**
8	**PRON**		192	**2.85%**	182	**2.69%**	10	**0.15%**
9	**DET**		150	**2.22%**	128	**1.89%**	22	**0.34%**
10	**NP**		49	**0.73%**	34	**0.5%**	15	**0.23%**
11	**NUMOR**		49	**0.73%**	42	**0.64%**	6	**0.09%**
12	**PREP**		31	**0.46%**	26	**0.39%**	5	**0.08%**
13	**UNK**	**Others**	14	**0.21%**	10	**0.15%**	4	**0.06%**
14	**VP**		11	**0.16%**	6	**0.89%**	5	**0.08%**
15	**ADJPREP**		6	**0.09%**	1	**0.01%**	5	**0.08%**
16	**NUMCD**		1	**0.01%**	1	**0.01%**	0	**0.0%**
17	**NPREP**		3	**0.04%**	0	**0.0%**	3	**0.05%**
	Total		6745	**100%**	6092	**90%**	653	**10%**

In order to measure the performance of model on set of test data we use confusion matrix. It is table form of matrix that contains information about test data and desired tags. To perform the analysis, it requires two parameters namely; reference tag that are actual tag of each word supposed to be which are tagged manually and predicted tag that are basically what the tagger generated using input of test data. For both HMM Tagger, CRF and Hybrid tagger confusion matrix are shown below in table 5.7, 5.8 and table 5.8 respectively with their description.

Table 5.7: Confusion matrix for HMM Tagger

		Predicted test Tags												**Total**	**Performance (%)**
		V	**N**	**PUNC**	**ADV**	**ADJ**	**INT**	**CON**	**PRON**	**DET**	**NP**	**NUMOR**	**PREP**	**Others**		
Reference Tags	**V**	**119**	7		4			2	1						**133**	**89.47**
	N	18	**137**			3								1	**159**	**86.16**
	PUNC			**99**											**99**	**100.00**
	ADV	3	5		**79**								2		**89**	**88.76**
	ADJ		9			**44**									**53**	**83.02**
	INT	1	7				**20**								**28**	**71.43**
	CON	4						**11**	2						**17**	**64.71**
	PRON				2				**8**						**10**	**80.00**
	DET		6					1		**15**					**22**	**68.18**
	NP		3								**11**		1		**15**	**73.33**
	NUMOR	1										**5**			**6**	**83.33**
	PREP									1			**4**		**5**	**80.00**
	OTHERS	6	1											**10**	**17**	**58.82**
	Total	**152**	**175**	**99**	**85**	**47**	**20**	**14**	**11**	**16**	**11**	**5**	**7**	**11**	**653**	**74.46**

In the above performance analysis, the confusion matrix of HMM tagger shows that total tagged of 653 test sets 562 are correctly tagged. The remaining 91 are wrongly tagged. The main reason for wrongly tagged is lack of balanced corpus, size of training and testing data, incorrect labeling of words during preparing corpus.

For example, in the experiment analysis among 133 Verbs 199 are tagged correctly as V and wrongly assigned 7 as N. Those are ባረችም, በናፊጂ, ይመስር, ይኸሬ, ተንተ, ትፈጢሎ, አጂ. 4 wrongly assigned as ADV. Those are ያነ, የረፐሮ, አዝራሹም,ተረፐረ. 2 wrongly assigned as CON and those are ዩድ, ቆማም. 1 wrongly assigned as VP and that are ይጫወጅዮ. The confusion matrix of HMM tagger shows that the order of performance of the tags as PUNC, V, ADV, N, NUMOR, ADJ, PRON, NP, INT, DET, CON, PREP, OTHERS in ascending order. Generally, the diagonal numbers show that test set that is correctly tagged.

The confusion matrix Conditional random field is shown in table 5.7 below with their description

Table 5.8: Confusion matrix for CRF

		Predicted test Tags													**Total**	**Performance (%)**
		V	**N**	**PUNC**	**ADV**	**ADJ**	**INT**	**CON**	**PRON**	**DET**	**NP**	**NUMOR**	**PREP**	**Others**		
Reference Tags	**V**	**91**	39		1	2									**133**	**68.42**
	N	7	**151**								1			1	**159**	**94.97**
	PUNC			**99**											**99**	**100.00**
	ADV	3	17		**65**	2								2	**89**	**73.03**
	ADJ		11		2	**37**								3	**53**	**69.81**
	INT		4				**24**								**28**	**85.71**
	CON		2					**14**					1		**17**	**82.35**
	PRON		2						**8**						**10**	**80**
	DET		5							**16**				1	**22**	**72.73**
	NP	1	3								**11**				**15**	**73.33**
	NUMOR											**6**			**6**	**100.00**
	PREP		1										**4**		**5**	**80.00**
	OTHERS		5											**12**	**17**	**70.58**
	Total	**102**	**235**	**99**	**68**	**41**	**24**	**14**	**8**	**16**	**12**	**6**	**5**	**19**	**653**	**66.56**

The above table 5.8 shows the confusion matrix of CRF. As we see from above table 115 are wrongly tagged and 538 are correctly tagged. The wrongly tags are mostly words that are not transcribed in Latin scripts or having one letter that are not transcribed to Romanization in the word leads to wrongly tagged. For this case wrongly tagged as default nouns word class category. For example, yažə□, yə□rə□, yatəwanä□, yətəšə□rə□, etc. are wrongly tagged as default noun. The box in the word indicates that scripts that are not readable or not correctly transcribed because of the tool doesn't recognize this special characters of the language. This experiment biases to the wrongly tagged words to noun.

Table 5.9: Confusion matrix for Hybrid tagger

		Predicted test Tags													Total	Performance (%)
		V	N	PUNC	ADV	ADJ	INT	CON	PRON	DET	NP	NUMOR	PREP	Others		
Reference Tags	**V**	**121**	8		1			2	1						**133**	**90.98**
	N	12	**140**		2	3					1			1	**159**	**88.05**
	PUNC			**99**											**99**	**100.00**
	ADV	3	1		**81**	4									**89**	**91.01**
	ADJ		5		2	**43**								3	**53**	**81.13**
	INT	1	7				**20**								**28**	**71.43**
	CON	4						**11**	**2**						**17**	**64.71**
	PRON		1					1	**9**						**10**	**90.00**
	DET		6							**15**					**22**	**68.18**
	NP		2								**12**		1		**15**	**80.00**
	NUMOR	1										**5**			**6**	**83.33**
	PREP									1			**4**		**5**	**80.00**
	OTHERS	4												**13**	**17**	**76.47**
	Total	**146**	**170**	**99**	**86**	**50**	**20**	**14**	**12**	**16**	**13**	**5**	**5**	**17**	**653**	**78.12**

As we can see in the above table 5.9 confusion matrix of hybrid tagger out of 653 testing set 573 are correctly tagged and the remaining 80 are tagged wrongly. Here in Hybrid tagger some tags improve their performance when we compare with HMM tagger. For instance, the V tagging accurately increased from 119 to 121 out of 133 test sets in hybrid tagger. The performance of tags result was PUNC, ADV, V, PRON, N, NUMOR, ADJ, PREP, NP, INT, DET CON, Others, in ascending order.

5.10 Summary

In this chapter, we presented the experimentation aspects of HMM tagger and Hybrid tagger. We use confusion matrix to measure the performance of each tag that are confused to other part of speech. We obtained the accuracy of 74.46 and 78.12 for HMM and Hybrid tagger respectively. The performance comparison of each Tags for HMM tagger and Hybrid tagger are listed below table 5.9 with their difference.

Table 5.10: Comparison of each tags for HMM tagger and Hybrid Tagger

Tags	HMM Tagger	Hybrid Tagger	Difference
V	89.47	90.98	**1.51%**
N	86.16	88.05	**1.89%**
PUNC	100.00	100.00	**0.00%**
ADV	88.76	91.01	**2.25%**
ADJ	83.02	81.13	**-1.89%**
INT	71.43	71.43	**0.00%**
CON	64.71	64.71	**0.00%**
PRON	80.00	90.00	**10.00%**
DET	68.18	68.18	**0.00%**
NP	73.33	80.00	**6.67%**
NUMOR	83.33	83.33	**0.00%**
PREP	80.00	80.00	**0.00%**
OTHERS	58.82	76.47	**17.65%**

As we can see from comparison table the hybrid tagger has no effect on some tags such as PUNC, INT, CON, DET, NUMOR and PREP. The reason is that we use only common rules of the language not include all tags. And one tag namely ADJ have negative effect on hybrid tagger this implies reduce the performance of hybrid tagger. Generally, tags those are V, N, ADV, PRON, NP and other tags are improved their performance in Hybrid tagger.

- **Comparison of HMM and Hybrid tagger on Test set**

The following figure shows the comparison of performance curve analysis using bar char with given of test sets.

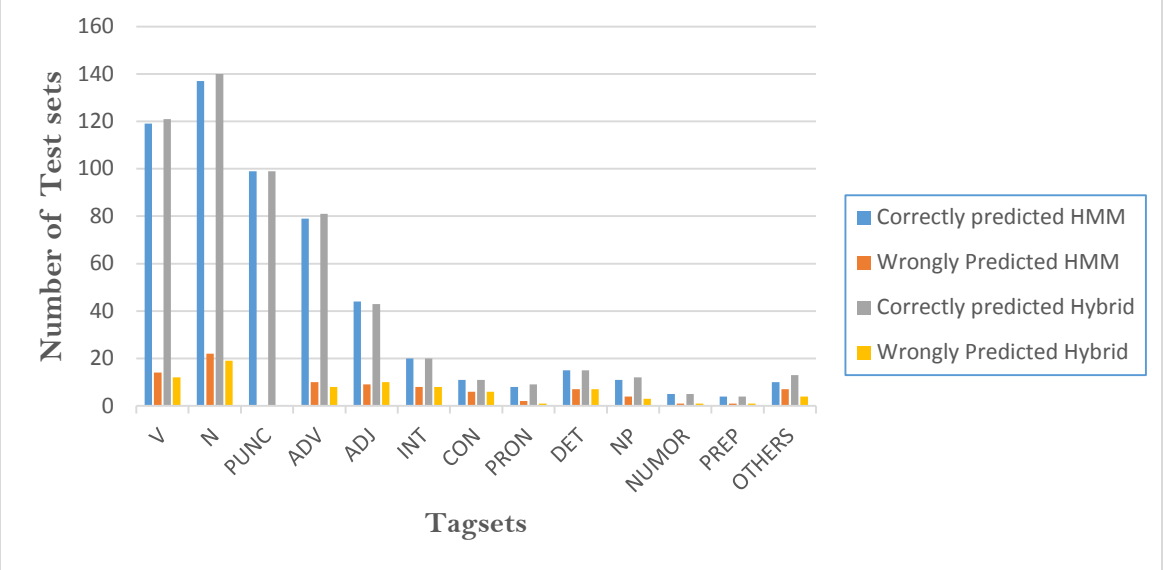

Figure 5-5: Comparison of HMM and Hybrid tagger performance analysis on test sets

- **Comparison of HMM,CFR and Hybrid on Test set**

The following figure 5.5 shows the comparison of performance curve analysis of HMM, CRF and Hybrid tagger using bar chart with given test set.

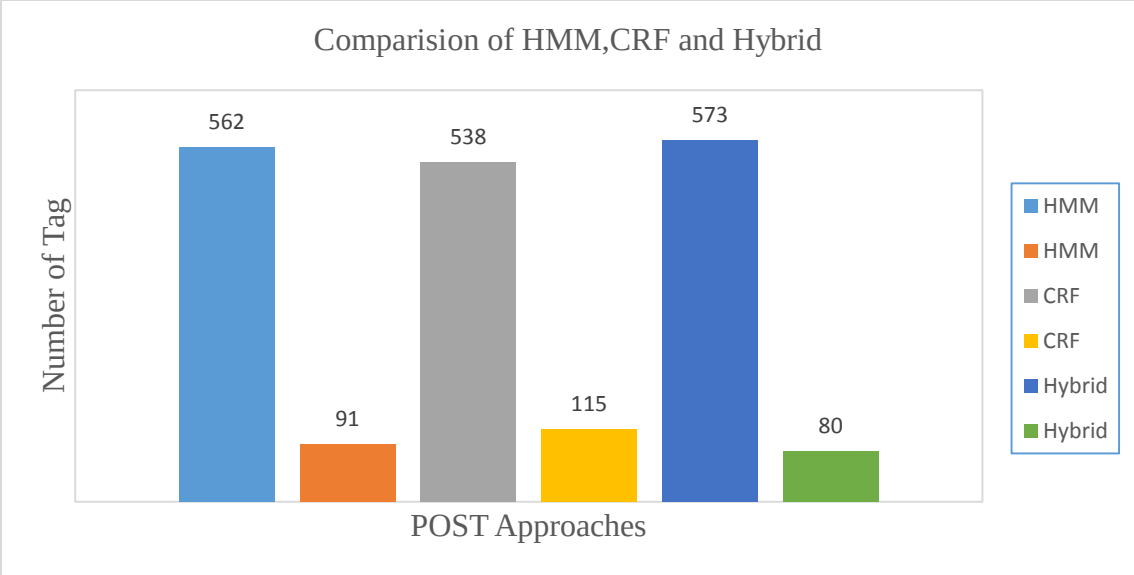

Figure 5-6: Comparison of three taggers on test set using bar chart.

5.11 User interface Design

In this work an interactive user interface is designed. The interface allows user to set location of files and process tagging. The interface is designed using java GUI programming techniques. So, that our interfaces are interactive and also platform independent.

The following figure 5.3shows snapshot of the interface at the configuration of tagger. The configuration of tagger contains training corpus location, tagsets location, dictionary location of the training data and annotated tagsets location.

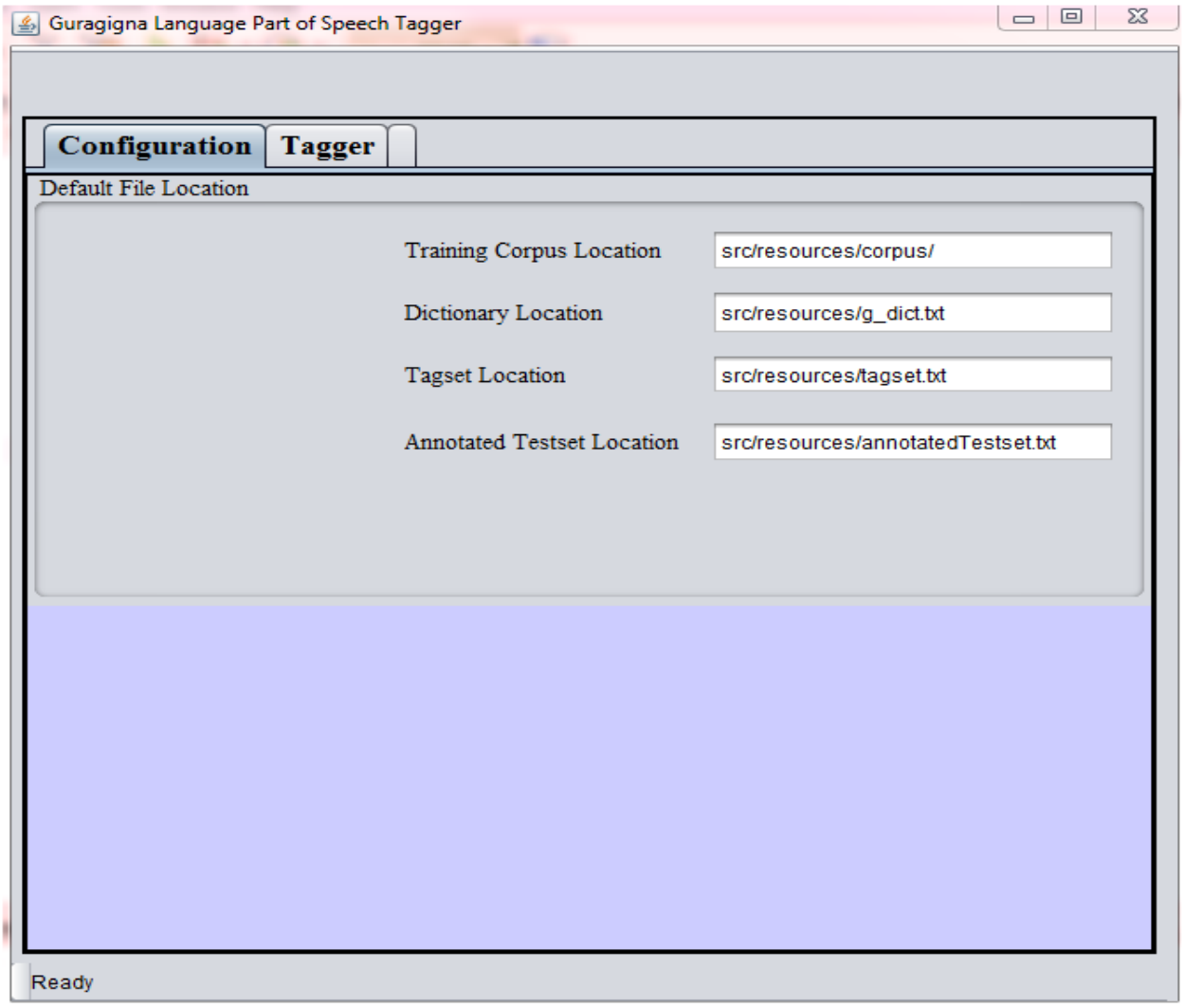

Figure 5-7: Snapshot of configuration of the tagger

Figure 5.8 shows that snapshot of HMM tagger prototype that takes test sentences from user and when the action button (tag sentence) performed it shows the tagged result of tagged sentence with their sentence probability.

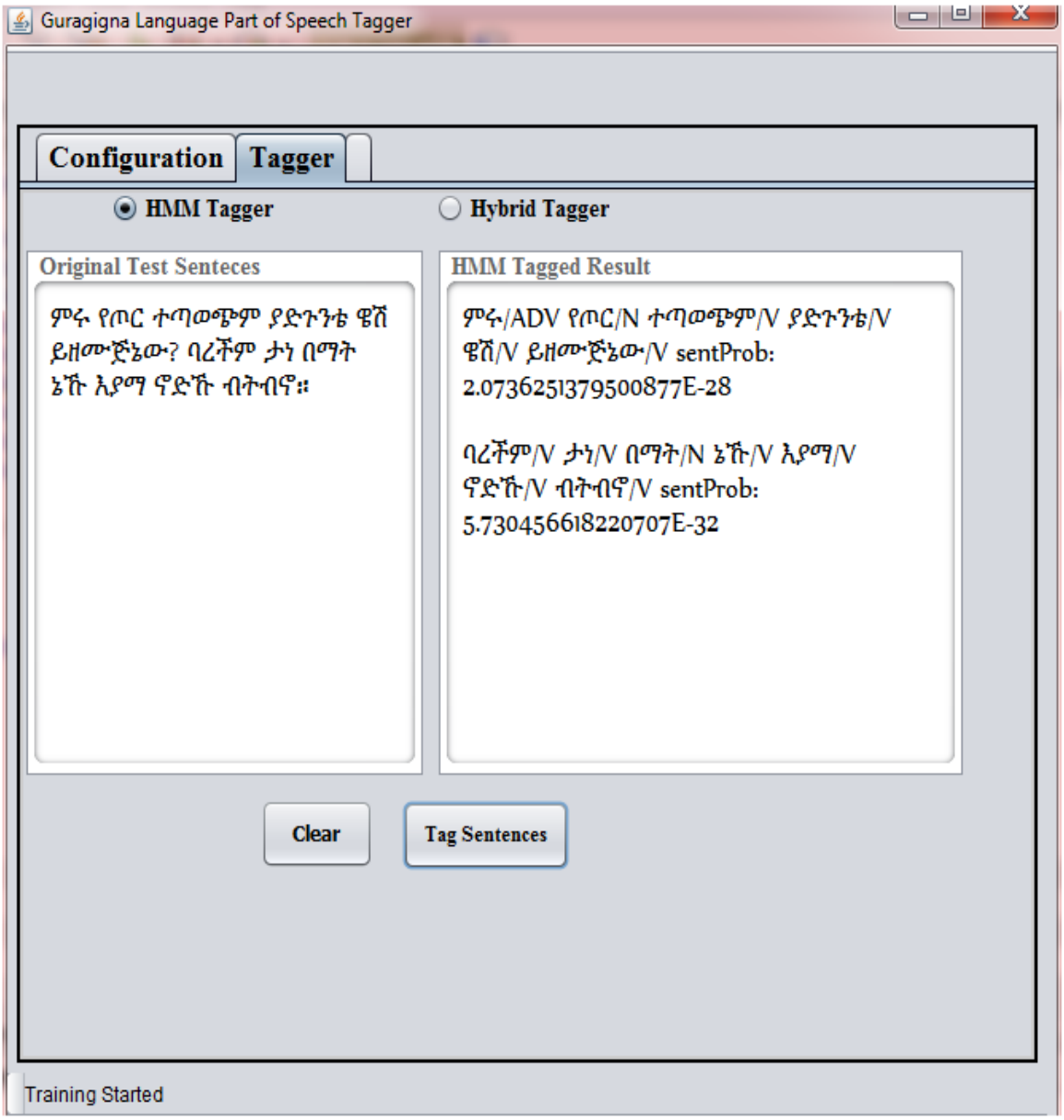

Figure 5-8: Snapshot of HMM tagger interface

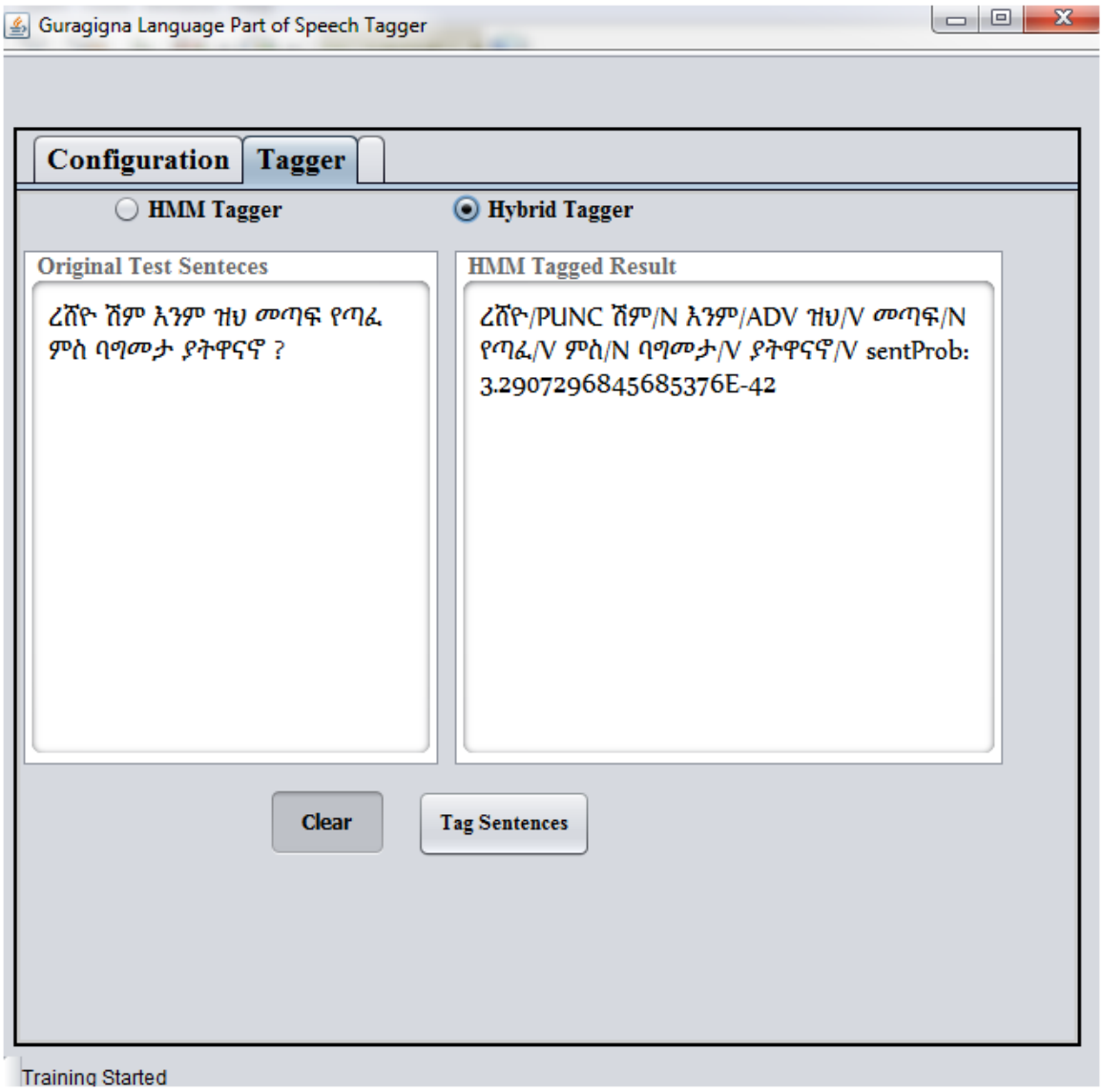

Figure 5-9: Snapshot of Hybrid Tagger interface

Like the HMM tagger, the Hybrid tagger works in the same manner. But the difference is seen when only we test large amount of data. Else with sample test set the result may similar with HMM one. Because, we use common Guragigna rules for Hybrid tagger.

CHAPTER SIX

CONCLUSION AND RECOMMENDATION

The presence of NLP discipline allowed computers to understand human language and process them. It plays basic role in different research tasks like POST, Spelling correction and parsing, Machine translation, Grammar checking, Text summarization and so on. Among these tasks POST is one of the foundation tasks for the other NLP tasks as this is used as preprocessing component. The task of POST is labeling each word to corresponding part of speech category so as to assign part of speech tags to words in a sentence.

In this thesis work we developed POST tagger for Guragigna language. Before developing the tagger, we have studied some POST developed for local and foreign language. And POST approaches have also studied to select the approach that can give better performance of Guragigna language. The nature, word category construction and behavior of Guragigna language have also studied before developing the tagger.

6.1 Conclusion

This study proposed and designed Part of speech tagging for Guragigna language. This POST can be developed through different approaches such as ANN, rule based, HMM based and hybrid. In our study, the tagger is developed based on HMM approach and Hybrid approach and each of them is implemented independently. The HMM is implemented using Viterbi algorithm as well as the hybrid is implemented by combining rule based and HMM.

For training the model, we developed Guragigna corpus of around 307 sentences. Seventeen (17) tagsets have been identified and dealt in this research work. Here the total corpus is divided into training and test sets. In our study 90% of total corpus has been used for training model and 10% have been also used for testing.

The implemented tagger involves three stages: preprocessing component, tagging and evaluation. After input is taken; tokenization and splitting take place followed by tagging

sentences. During tagging stage computing lexical and contextual probability is performed to get optimal tag sequence of tagged text. Finally, the tagged text by tagger is evaluated with manually tagged text.

After evaluation, the test result shows encouraging outcome on test data of 6745 tokens and it obtains 66.56, 74.46 and 78.12 for CRF, HMM tagger and Hybrid tagger respectively.

6.2 Contribution of the work

In this study, the mains contribution work is:

- ✓ Identifying the nature and word category of Guragigna language and how they can be processed to be understood by the machine.
- ✓ Contribution of preprocessing component for other researcher working on high level NLP applications.
- ✓ Extracting rules of Guragigna using hybrid tagger.

6.3 Recommendations

There are works that should be considered by future researchers in the area of natural language processing in general and part of speech tagging particular for Guragigna language.

- ✓ Since accuracy and effectiveness of language model depends on corpus, developing balanced standard corpus should be considered for future work.
- ✓ Numbers of tagsets considered are limited to 17 in this work. This category doesn't include all word categories of the language. So, the researcher can work by including those into sub categorization of word classes.
- ✓ Hybrid approach used in this work is not rich on rules. Only common rules are used. So, the researcher can advance the rule based part of speech approach to get better result.
- ✓ There are also other approaches that are different from statistical and rule based which can be applied for the same problem. So, the researcher can develop the same work for the language using neural network to compare the result.

Reference

[1] L. K. Wudinesh, "Design and Development of Automatic Morphological Synthesis for Amharic Perfective verb. Addis Ababa:," Addis Ababa, Unpublished Master thesis (Addis Ababa University), 2002.

[2] B. W. Erik Cambria, "Jumping NLP Curves: A Review of Natural Language Processing Research," *IEEE*, pp. 48-56, May 2014.

[3] G. M. Meshesha, "Part-of-Speech Tagging for Afaan Oromo," *Proceeding International Journal of Advanced Computer Science and applications*, vol. Vol.1, no. special issue on Artificial Intelligence, pp. 1-5, 2011.

[4] L. Altunyurt, "A Composite Approach for Part of Speech Tagging in Turkish.," in *International Scientific Conference Computer Science*, Istanbul, Turkey,Bo_aziçi University, Computer Engineering Dept, 2006.

[5] *Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging.", Computational Linguistics,* Brill (1995).

[6] X. W. Weiwei Sun∗, "Towards Accurate and Efficient Chinese part of speech tagging," *Association for Computational Linguistics*, vol. 42 no 2, 2016.

[7] V. S. S. C. Ravi Narayan, "Quantum Neural Natwork Based Part of Speech Tagging for Hindi," *International Journal of Advancment in Technology*, Vols. 5,No 2, pp. 137-152, July2014.

[8] Y. O. M. Elhadj, "Statistical Part-of-Speech Tagger for Traditional Arabic Texts,," *Journal of Computer Science, Science Publications,*, vol. 5, pp. 794-800, 2009.

[9] S. S. A. B. Sandipan Dandapat, "A hybrid model for Part-of-Speech tagging and its application to Bengali," in *Department of Computer Science and Engineering Indian Institute of Technology Kharagpu*, Kharagpu, 2007.

[10] S. Asres, *Automatic Amharic Part of Speech Tagging using Hybrid Approach (Neural Network and Rule based),* Ethiopia: Addis Ababa University (Master thesis), 2008.

[11] M. Y. T. a. S. T. A. a. L. Besacier, "Part-of-Speech Tagging for Under-Resourced and Morphologically Rich Language- the case of Amharic," in *Conference on*

Human Language Technology for Development, Alexandria, Egypt, May 2011.

[12] M. H. Abubaker, *Part of speech tagger for Afaan Oromo Language using transformational error driven learning approach(TEL),* Ethiopia: Addis ababa university(Unlpublished master thesis), February 2010.

[13] A. G. Ayana, *Improving Brill's Tagger Lexical and transformational rule for Afaan Oromo Language,* Ethiopia: Addis Ababa University (unpublished Master Thesis), 2013.

[14] T. G/egziabher, *Part of speech tagger for Tigrigna Language,* Ethiopia: Addis Ababa University(Master Thesis), 2010.

[15] M. A. Sium, *Automatoc part of speech tagger for Tigrigna language using Hybrid Approach*, Addis Ababa: Unpublished master Thesis , 2016.

[16] T. a. Fedlu, "designing and developing Keyboard for guragigna Language," *Walkitie University, School of Informatics and computing, Unpublished,* 2016.

[17] G. M. Mesfin Getachew, *Automatic part-of-speech tagging for Amharic language an experiment using stochastic Hidden Markov Approach*, Ethiopia: School of Graduate Studies, Addis Ababa University,MSc Thesis, 2001.

[18] A. L., "Part-of-Speech Tagging with Evolutionary Algorithms.," in *In International Conference on Intelligent Text Processing and Computational linguistics*, Springer Verlag, 2002.

[19] W. Daelemans, "Memory Based Part of Speech Tagger," no. Computitional Linguistic and AI, pp. 14-25, Jul 11, 1996.

[20] K. K. Zin, "Hidden Markov Model with rule based Approach for part of speech tagging of Mymar Language," in *Proceding of the 3rd International Conference on Communications and Information Technology*, florida, december,2009.

[21] B. M. B. Morteza Okhovvat, "A Hidden Markov Model for Persian Part-of-Speech Tagging," in *Procedia Computer Science*, Iran University of Science and Technology,Iran, 2010.

[22] K. R. S. D. S. Bipul Syam Purkayastha, "Part of Speech Tagging in Manipuri: A Rule-based Approach," *International Journal of Computer Applictions*, Vols.

51,No.14, pp. 31-36, 2012.

[23] E. Brill, "Simple Rule based Part of speech tagging," in *In Preciding of third conference on applied natural Language Processing*, Trento, 1992.

[24] R. N. a. S. Chakraverty, "Neural Network based Parts of Speech Tagger for Hindi," in *Third International Conference on Advances in Control and Optimization of Dynamical Systems*, Kanpur, India, March 13-15, 2014. .

[25] S. S. C. S. P. S. P. Bishwa Ranjan Dasa, "Part of speech tagging in odia using support vector machine," in *International Conference on Intelligent Computing, Communication & Convergence*, Odisha, India, 2015.

[26] B. G. Gebre, *Part of Speech Tagging for Amharic,* Wolverhampton University, UK: Research Institute in Information and Language Processing.

[27] E. Brill, "Transformation-Based Error-Driven learning and natural language processing: case stuty in part of speech tagging," *Association for computitional Linguistic*, Vols. 21, Number 4, pp. 544-565, 1995.

[28] L. Q. Z. Marylyn Alex, "Kadazan Part of Speech Tagging using Transformation-Based Approach," in *The 4th International Conference on Electrical Engineering and Informatics*, Malaysia, 2013.

[29] Y. M. Michal Ptaszynski, "Part-of-speech tagger for Ainu language based on higher order Hidden markov model," *Science Direct,* vol. Volume 39, no. Issue 14, p. Pages 11576–11582, 15 October 2012.

[30] D. J. a. J. H. Martin, Speech and Language Processing, singapore: Person Education Inc Pte. Ltd, 2005.

[31] B. T, "TnT – A statistical part-of-sppech tagger.," in *In Proceedings of the 6th Applied NLP Conference, 224-231*, 2000.

[32] Z. Mekuria, *Design and Development of Part of speech tagger for Kafi-Noonoo Language,* Ethiopia: Addis Ababa University(Unpublished Master Thesis), 2010.

[33] V. A., "Constraint Grammar: A LanguageIndependent System for Parsing Unrestricted Text," vol. I, no. Berlin, p. 165–284, 1995.

[34] J. a. Martin, Speech and Language processing, Printice Hall, 2000.

[35] N. U. a. M. K. Fahim Muhammad Hasan, "Comparison of Different POS Tagging Techniques (n-grams, HMM and Brill‟s Tagger) for Bangla," in *Center for Research on Bangla Language Processing*, BRAC University, Bangladesh, December 2006.

[36] M. Hepple, "Independence and Commitment: Assumptions for Rapid Training and Execution of Rule-based POS Taggers," in *In: proceeding of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-200)*, Hon kong, October 2000.

[37] S. Helmud, "Part of speech tagging with neural network," *Institute of computitional Linguistics*, vol. I, pp. 172-176, 1994.

[38] N. Marques, "Neural Networks, Part-of-Speech Tagging and Lexicons*," *International Association of Journal and advancment technology*, pp. 53-58, June 2011.

[39] M. M. K. U. H. I. Qing Ma, *Hybrid Neuro and Rule-Based Part of Speech Taggers*, Japan: Ministry of posts and communication.

[40] N. I. a. F. J. Demarawu, HANDBOOK OF NATURAL LANGUAGE PROCESSING SECOND EDITION, Cambridge, UK: Microsoft Research Ltd., 2010.

[41] M. M. K. U. ,. I. Qing Ma, "Hybrid Neuro and Rule Based Part of Speech Taggers," in *Communication research Laboratory Minstry of POSTS and Communication*, Japan, 2003.

[42] B. S. P. Bipul Roy, "A Study on Different Part of Speech (POS) tagging approach in assame Language," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 5, pp. 934-938, March 2016.

[43] M. Getachew, "Mesfin Getachew.(2001) Automatic part-of-speech tagging for Amharic language an experiment using stochastic Hidden Markov Approach.," *MSc. Thesis. School of Graduate Studies, Addis Ababa University*, 2001.

[44] G. Mamo, *Part-of-Speech Tagging for Afaan Oromo Language.*, Ethiopia: Master’s thesis, Addis Ababa University., 2009.

[45] T. O. a. S. Rose, "Segmental effects on (de)gemination in Western Gurage," in *27th*

Annual Meeting of the Berkeley Linguistics Society,Special Session on Afroasiatic Languages, University of California, San Diego, 2000.

[46] G. k. Stefen Weninger, The Semetic Language: International Hand Book, Germeny: Wruyter GymH, 2012.

[47] D. Banksira, *Chaha labialization and palatalization as coalescence: The segmental phonology of Ethiopian semitic languages course,* Murphy,Leipzig: University of Leipzig, Nov 20, 2014 .

[48] S. Rose, *Chaha (Gurage) Morphology,* San Diego: University of California , 2013.

[49] G. Emiru, *Developing part of speech tagger using hybrid approach,* Ethiopia: Addis ababa University(Master thesis), 2016.

[50] "Salmon Run," 08 November 2008. [Online]. Available: http://sujitpal.blogspot.com/2008/11/ir-math-in-java-hmm-based-pos.html. [Accessed 14 May 2017].

[51] G. G, *Natural language processing. Annual Review ofinformation science and Technology,* Unitied Kingdom: University of Strathclyde Glasgow, 2003.

[52] A. M. P. John Lafferty, "Conditional Random Fields: Probabilistic model for segmenting and labeling sequence data," in *In Proceedings of the 18th International Conference on machine learning*, 2001.

[53] D. Crystal, *Dictinaries of language and languages,* London, United Kingdom: Penguin Books Ltd, 1994.

[54] A. P. H/Markos, *የሜሞሙ· ሸካ,* ethiopia, Addis Ababa, 1060.

[55] M. S. W. Bahiru Lilaga, *YEGURAGIGNA FIDEL GEBETA*, Walkite,Ethiopia: Unpublished, 2006.

Appendix A: Ye Guragigna Fidel Gebeta [55]

	Ä [ə]	U	I	a	e	ə [ɨ]	o	^{w}ä [^{w}ə]	^{w}i	^{w}a		^{w}e	^{w}ə [^{w}ɨ]
X	ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ	ኼኈ	ኼኍ	ኼኊ		ኼዄ	ኼዅ
x^{y}	ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ						
L	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ						
M	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ	ሙኈ	ሚኍ	ሚኊ		ሜዄ	ምኈ
R	ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ						
S	ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ						
Š	ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ						
ḳ	ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ	ቈ	ቀኍ	ቂኊ		ቄዄ	ቀኍ
$ḳ^{y}$	ቅ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ						
B	በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ	ቡኈ	ቡኍ	ቢኊ		ቤዄ	ብኈ
B	ቨ	ቩ	ቪ	ቫ	ቬ	ብ	ቮ						
T	ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ						
Č	ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ						
N	ነ	ኑ	ኒ	ና	ኔ	ን	ኖ						
Ñ	ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ						
ʾ	አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ						
K	ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ	ኩኈ	ኩኍ	ኪኊ		ኬዄ	ኩኈ
k^{y}	ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ						
W	ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ						
Z	ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ						
Ž	ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ						
Y	የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ						
D	ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ						
Ǧ	ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ						
G	ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ	ጉኈ	ጉኍ	ጊኊ		ጌዄ	ጉኈ
g^{y}	ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ						
ṭ	ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ						
Č̣	ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ						
P̣	ጰ	ጱ	ጲ	ጳ	ጴ	ጵ	ጶ						
Ṣ	ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ						
F	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ	ፈኈ	ፉኍ	ፋኊ		ፌዄ	ፍኈ
P	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ	ፑኈ	ፑኍ	ፒኊ		ፔዄ	ፑኈ

: - Scripts used in Guragigna Language

Appendix B: Summary of Guragigna Subgroup Languages

Information on the language	**Soddo (Kistane)**	**Zay**	**Inor**	**Mesmes**	**Mesqan**	**Sebat Bet Gurage (Chaha)**
Native speakers according to Ethnologue in 2015	260,000	4,900	280,000.	none who speak the language	227,135	1,440,000.
Word order	SOV	SOV	-	-	-	SOV
Language use	Use Amharic and Guragigna	Speakers are multilingual in Amharic, Oromo and Gurage language	-	Shifted to speaking Hadiyisa language		Also use Amharic
Language status	Vigorous	Threatened	Vigorous	Extinct	-	Developing
Language development	Literacy rate: 22%	Grammar level. But exact literacy rate is unknown	-	-	Not used in school and administrative purpose	Literacy rate: 25.3%.
Writing system used	Ethiopic script	Unwritten language		Has no script		Ethiopic script
Other	Not used in media. 2nd largest in literacy	Not used in media and currently decreasing due to migration and become Oromo speakers.		-	-	Used in media. 1st larger in literacy

Appendix C: Sample Rule for Guragigna Language

Appendix: lexical rules used

ADJ→N if the Adjective is followed by suffix“ ኑት”

ADV→V if the adverbis followed by suffix”ባራም”

N→ NP/if the Noun is preceded prefix“ታ”

V→N if the Verb is followed by the suffix“አት”

Appendix: Contextual rule used

V →ADJ: if the tag of preceding word is adjective and tag of following word is adjective.

ADJ→ ADJ: if the tag of preceding word is adjective and tag of following word is adjective.

N→ ADJ: if the tag of preceding word is Noun and tag of following word is adjective.

ADJ → N: if the tag of preceding word is Determiner and tag of following word is Noun.

ADV→ ADJ: if the tag of preceding word is Noun and tag of following word is adjective.

Appendix D: Tagged POST sample

ህ/DET ጀፐረችም/V ኦዶችናኽም/V ይና/PRON ንብረት/N ዝመታው/N አኽ/PRON ጭን/CON በምር/ADV ኤነት/N የረፐርኽ/V ኸማ/UNK ገፔም/V ።/PUNC ተዛ/DET ምሳኽ/N ጋመ/ADV አኸ/PRON ቆጦ/INT አኽ/PRON ቆጦ/INT ተበበርኹም/V የረፐርኹ/V ቃር/N ኤመስሬ/N ባረም/V ቢያዥና/ADJ ይና/PRON አቻም/ADJ ሰብ/N አነበነ/V አቸ/INT ሜተና/V ያኸሮ/ADJ ቃር/N ባነቦ/V ምስ/N አቻም/ADV ተኸነ/V ዝክም/CON ያኹ/PRON ኸማ/UNK ይፍት/ADJ ጨዋዳ/N ብትብር/V ፦/PUNC እወ/INT ድርድግ/INT ይፍት/ADV ይፍት/ADV ባንትጨወጂ/V ሃንዝ/N ተቅቧቦርም/V ይዝረጉቄ/V ?/PUNC የምስ/ADJ እንዝር/N ኤስማ/V ባረችም/V ዝርጓት/N ።/PUNKባይ/INT የምሸቴ/ADJ እንዝር/N አንስማ/V የበር/V ቢብር/V ናማጋ/N የግራኸ/INT አሚንሸ/INT እያ/PRON ጠንቋላ/N ንህር/V በዘንጋ/N የረገድከ/V ቃር/N መሠረናኸም/V ?/PUNC አዟሴ/V ትምብሩሸ/V ደጃና/N ዝሼ/INT ባትም/ADJ ቃር/N አንትራገድነ/V ኸማ/UNK የጫሙት/N ኡድነያም/V ባረም/V አሸናም/V ታነ/CON ቢደቅ/V እንም/ADV ዳቆም/V ።/PUNC ዝርጓት/N ሙሸርቅ/V ባረችም/V ታነ/CON ኧኳ/ADV ጫሙትኸሸ/NP በኬርታዊ/N ጭቃራ/N ትትመርግዉ/V ተሰማኸ/V ባረችም/V ባታዥና/V ወ/INT አጂ//INT መምር/ADV ጭቃራ/N መነግህወም/V ?/PUNC ሸም/N አጥፎት/V ትኽረማዌ/V አህማ/PRON ?/PUNC አቸ/INT በርደፈረ/N ይቀርጤ/V ኸማው/ADV ?/PUNC ባረችም/V ቢጠርቅና/V አኽ/PRON ንበራኔ/ADJ ቃር/N ትታችን/ADV ዳር/ADV የኹትም/ADJ ዘንጋ/N ባነ/V ንጫወድኔ/V የረፐረ/V ።/PUNC የኬርታዊ/ADJ ምስነትበ/N ትኼሬ/V ትትብር/V ጥረነቀችንደም/V ባንኸረሼ/CON መላ/ADV ታቴላን/V አንቸች/V ባረም/V ቅንበታ/N ቢገፉ/V አትከረ/ADV የኸረ/INT በትክባበሴ/V አዣችንም/V ባነ/V ይህሩ/UNK ዛህ/DET ባዣችወ/ADJ ኤን/N ትትቄርጥን/V ነበረችም/V ታነ/CON ኧኳ/ADV ዬብሶ/ADJ ላሙ/N መሠረም/V በዝ/DET ገባም/ ብታዥን/V ኽኖኽታ/N ጠርጠር/V ቴብር/V አንቸን/V ታትፈጢም/V አጂ/V የበርደፈረ/N ባረችም/V ብታዥና/V ጫሙት/N ታትዝረኪም/V ናማጋ/N ተቆፐረም/V ታነ/CON እያም/PRON መሰሬምሸ/V ንቃር/ADV ሰቀረችም/V አወንደችንም/V አንኸረዌ/V ባረም/V ።/PUNC ሰቦድረድግ/INT እያ/PRON ምር/ADV ባኹም/V ሰቀርኹንም/V ?/PUNC የሰቀርኹኢ/CON አኹ/INT ብትብር/V ኦ/INT አኽሸ/PRON ባወንድኺም/V ኸማ/UNK ነርኺ/V ?/PUNC ባረም/V አሸናም/V።/PUNC ጫሙትም/N የስቅሮት/N ያውርዶት/N የምር/ADV ጠማር/N ጠማር/N ይጠብጤ/V ?/PUNC ባረችም/V አሸችንም/V በስቀርኹንም/V አነኸውን/V በሮታኺ/V ?/PUNC ዌሸ/CON መምሩ/ADV ?/PUNC ባረም/V አሸናም/V ታነ/CON ኔኽ/V አት/NUMOR ዘንጋ/N ኖድኸ/V ምርቱዴዬው/ADV ብትውን/V አኽ/PRON በሸኺም/V አትረኽቢ/V ኸማ/UNK ሁት/PRON ጭን/CON በሸናኽ/V ኤበኽርኽ/V ኸማ/UNK ምሩ/ADV የጦር/N ተጣወጭም/V ያድጉንቴ/V ዌሸ/CON ይዘሙጅኔው/V ?/PUNC ባረችም/V ታነ/CON በማት/N ኔኹ/V እያማ/PRON ኖድኹ/V ብትብኖ/V ፦/PUNC እንዴ//INT ባረናም/V ናማጋ/N ዝርጓት/N ቁስ/ADV መነችም/V የኼችም/ADV ያጋጋ/ADV ታዥኖ/V ጫሙት/N ባወጣች/V ኸማ/UNK አኹ/PRON ዝ/DET ትትብረዊ/V ኩርታዊአሁ/N ትትራከቦ/V ጫሙት/N ውረንያ/N ባትኸርኸዌ/V ?/PUNC ቡንም/V ታነ/CON ኧኒ/INT በባረ/V እያ/PRON ኤኸር/V ባምባሁን/V አሁ/PRON ቅርጦን/V አትኸሬ/V አናገባና/V በባረም/V ያገባይ/V ኸማ/UNK ብምቧት/N ባምቡሰድኹን/V ዛም/DET ቅርጦን/V ባረችም/V ብታዥኖ/V ዘንጋ/N ነረብኽ/V የጦር/N ተጣበጥኹም/V ታድኒኡው/V ዌሸ/CON ትዘምጂቴ/V ?/PUNC ባረም/V

Appendix E: untagged transcribed (Latin script) POST sample of training set

bäzə mät'afə dänə yägurage säbə nəbərätäta mərə yəmäsərə häma ፧ badəbabe ፧ boԋre ፧ bähanəᎮta bänəgudə ʾənəgudəmə mädärə yanänə qwabärə bäməsale ʾämänomə ʾätəᎮnäwuꝶmə . yäräšäyo šəmə ʾənəmə zəhə mät'afə yät'afä məsə bagəmäta yatəwanano yägurage šəməro banəhäre bämurəmə yanäbo ፧ wemə yäᵐᵐtäbo säbə yəräsə šəmə ʾänəhäro ፧ bäzəka šəmə yät'änäyo säbə zəhäta ʾebärəyä RON yəmäsərə betäbo ʾənətə/ ğägäroꝶ . yäč'äwadata yəšərəwe tiwurə ʾätatə t'əbə näšəyomə banəhäre mät'afäta ʾätə/NUMOR t'əbə tənəgudə t'əbə yarəsə ፧ wemə yərəqə barämə yatəwana qarə ʾänəhärə . ʾätatə t'əbə yäräšəyo yätaräkäta yəšərəwe tiwurə bägänə yəč'awäğəyo č'äwada yänəsotə ʾähəru . nəgusə tärəgusə yäwarəyo yät'əbə yäsädašəmə ፧ ʾäzəčəmačə tazəmačə tobärə yäšəꝶčəyo šəməro banəhäre tənəgudə bariqə yərəqə danätə yanäno säbə ʾänəhäro ፧ bähäräməč'ənə t'əbähuna babäno yäsädašəmə yəfəte yaᵐᵐrəyo yät'əbə bariqə wemə ʾäbäqat'əro . bämät'afəta burə zänəga tənəgäbanä ፧ zəhə mät'afə yäč'amätə tonəžätarəbə ԋrətawi tarikə yudə mät'afə . ʾäğəyätə zərəgätə wänəžätarəbə korətawi yäč'amutə yäbetə ʾäbə yəhäre bämäč'ämə tanäčəhäma ʾätəkärä qərärä zahə č'amutə yağəyäte bäčänäče zänəgata bətəräsaba mərəyahərə ʾätəranäsäčəmə yäžäpäräčəna häma tiyatəyäšə zərəgätə täč'amutə bähutəmə č'äwada tanäma banädäᵓ gučä bädənəgätə wänəžätarəbə korətawi čänämə yägäpahäma ፧ yäzahə yäqonət'äčənə məšətə bač'ə ʾämänämə bägura ʾenə ʾäšəmə yäʾäğəyätə zərəgätəmə bätəᵑᵑzotə ʾäzəraᵑᵑmə tiwät'a tähə kärä Ꮁnäšəmə yäč'amutə ᵑᵑnə yätəsäqärä häma ፧ yanəᎮ qərət'ənaᵑᵑta yanämädärə yähärähäma benäᵑᵑta ʾäšəčəmə tanä hutə bäqänət'änamə bäğäᵑᵑta tagäwane batəwanačənə qet'ə gägämäᵑᵑta bəməᑫtə yoräčə häma yäč'amutə šəka zəhə mät'afə ፧ wäräwärämə gaᵐᵐ REP yägurage qwanəqwa yäbetätä bähärä zata bät'aᎯᎮ batə humə žät'ä zäbärə täsədəsa yäräpäneꝶ nəbərätə yəᵑᵑrəᎱ ፧ batəwähätəma qwanəqwanəda yäᎮt'ä wemə ʾäčə yäqänä bähärä gurage dərä yägägäta qwanəqwa ᑫnänə säbə ʾənəmə bamarina nəzänəᵌ barämə tanä guragina täräsamə tawimə ᑫrəmə bägəzəyäta yəräbərə täräpärä zänəga yažəᎱ yəᵑᵑrəᎱ ፧ yatəwanäᎱ ፧ yətəšəᵐᵐrəᎱ təməbərə t'afəhunəmə . mät'afäta yanäbunə wemə yəsämanə säbə ʾənəmə yat'äᎯ qaru betəhu tehärəmə yäzanəᎮ yət'äfo säbə ʾema nəkəfətə təməbəru

Appendix F: tagged transcribed (Latin script) POST sample of testing set

yäč'äwadata/NPREP yəšərəwe/V tiwurə/V ʾätatə/N t'əbə/N näšəyomə/V banəhäre/CON mät'afäta/N ʾätə/NUMOR t'əbə/N tənəgudə/ADJ t'əbə/N yarəsə/ADJ ፧/PUNC wemə/CON yərəqə/ADV barämə/V yatəwana/V qarə/N ʾänəhärə/V ./PUNC ʾätatə/ADV t'əbə/N yäräšəyo/V yätaräkäta/N yəšərəwe/V tiwurə/V bägänə/N yəč'awäğəyo/V č'äwada/N yänəsotə/N ʾähəru/INT ./PUNC nəgusə/N tärəgusə/N yäwarəyo/V yät'əbə/NP yäsädašəmə/NP ፧/PUNC ʾäzəčəmačə/N tazəmačə/N tobärə/V yäšəʧəyo/ADJ šəməro/N banəhäre/INT tənəgudə/ADJ bariqə/N yərəqə/ADV danätə/N yanäno/V säbə/N ʾänəhäro/V ፧/PUNC bähärämək'ənə/CON t'əbähuna/N babäno/V yäsädašəmə/NP yəfəte/ADV yaʷᵊəyo/V yät'əbə/ADJ bariqə/N wemə/CON ʾäbäqat'əro/N ./PUNC bämät'afəta/N burə/ADJ zänəga/N tənəgäbanä/N ፧/PUNC zəhə/DET mät'afə/N yäč'amätə/ADJ tonəžätarəbə/ADJ ẖʷrətawi/ADJ tarikə/N yudə/V mät'afə/N ./PUNC ʾäğəyätə/ADJ zərəgätə/N wänəžätarəbə/ADJ korətawi/N yäč'amutə/N yäbetə/ADJ ʾäbə/N yəhäre/V bämäč'ämə/N tanäčəhäma/ADJ ʾätəkärä/ADV qərärä/N zahə/DET č'amutə/N yağəyäte/N bäčänäče/V zänəgata/N bətəräsaba/V mərəyahərə/ADV ʾätəranäsäčəmə/V yäžäpäräčəna/V häma/ADV tiyatəyäšə/V zərəgätə/N täč'amutə/NP bähutəmə/DET č'äwada/N tanäma/V banädäʯ/V gučä/N bädənəgätə/ADV wänəžätarəbə/ADJ korətawi/N čänämə/V yägäpahäma/ADV ፧/PUNC yäzahə/DET yäqonət'äčənə/V məšətə/N bač'ə/INT ʾämänämə/V bägura/ADJ ʾenə/N ʾäšəmə/V yäʾäğəyätə/ADJ zərəgätəmə/N bätəᎇzotə/ADV ʾäzəraʧʷmə/V tiwät'a/ADV tähə/DET kärä/ADV ቁnäšəmə/V 6/NUMOR barämə/V tiwät'a/CON ‹‹/PUNC yät'äfäʧʷ/INT :-/PUNC zə/INT məsəšə/N yäqänət'əʧ/ADJ bətəʾəna/V č'amutəmə/N forə/N gäpanamə/V tanä/V ፧/PUNC bašahəmo/V ʾəya/PRON yägämäya/ADJ zänəga/N wägə/N yəhärəbi/V ፧/PUNC ʾätəkärä/ADV təyažəyoma/V ʾäčä/INT wät'amə/V yəwädəqo/V !/PUNC ᎎ/ADV zamə/DET ẖʷrətawiwə/N zəməsə/N yätänəbi/INT bətəbəräme/V darə/CON ʾəyamə/PRON wäbärə/V čämə/V kerə/N yätänəʧʷ/V yəbərə/V ?/PUNC ʾəkawəšə/INT !/PUNC ʾäčä/INT məsə ቕt'o/INT !/PUNC ››/PUNC baräčəmə/V yağəyätə/ADJ zərəgätə/N ʾäšäčənamə/V ./PUNC 7/NUMOR ʾəka/INT ʧʷnaʧʷ/N ʾezägərə/V !/PUNC baräčəmə/V daqäčəmə/V ʾäğəyätə/ADJ zərəgätə/N .PUNC ‹‹/PUNC de/INT ᎎğahəmašə/INT zäkäräməwe/V ʾäčä/INT yäᵑməbe/ADV hämawəšə/INT !/PUNC ››/PUNC baräčəmə/V č'amutə/N sərəmə/ADV tatəfät'ərə/V ./PUNC‹‹/PUNC bifätaʧʷ/N

Appendix G: Interview Questions that are asked to Guragigna language professional.

1. ስለ ጉራግኛ ቋንቋ የተወሰነ ቢያብራሩልኝ? የጉራግኛ ቋንቋ ከሌላ ቋንቋዎች በምን ለየት ያደርገዋል?

* [illegible] labialization (round lips), palatalization (sound change), gemination (consonant lengthening or double) [illegible]

* [illegible]

2. ቋንቋዉ ስንት የፊደል ገበታዎች አሉት?

* [illegible]

3. በጉራግኛ የተጻፉ መጻሕፍት አሉ? ከየት ማግኘት ችላለሁ?

* [illegible] fiction [illegible] infant [illegible]

4. ከዚህ ቀደም በቋንቋዉ ዙርያ ምን የተሰሩ ስራዎች (ሪሰርች ወይም ፕሮጀክት) አሉ?

* [illegible] unpublished [illegible] (morphology) [illegible]

5. ወረዊ ወይም በተወሰነ ግዜያት የሚታተም በጉራግኛ ቋንቋ የተጻፉ መጽሄት ወይም ጋዜጦች አሉ?

* [illegible] seminar, symposium, [illegible]