ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

AMHARIC SENTENCE TO ETHIOPIAN SIGN LANGUAGE TRANSLATOR

BY

DANIEL ZEGEYE ZELEKE

A THESIS SUBMITTED
TO THE SCHOOL OF GRADUATE STUDIES OF ADDIS ABABA UNIVERSITY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTERS OF SCIENCE IN COMPUTER SCIENCE

June, 2014

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
COLLEGE OF NATURAL SCIENCES
DEPARTMENT OF COMPUTER SCIENCE

AMHARIC SENTENCE TO ETHIOPIAN SIGN LANGUAGE TRANSLATOR

BY

DANIEL ZEGEYE ZELEKE

ADVISOR:

Sebsibe Hailemariam (PhD)

APPROVED BY:

EXAMINING BOARD:

1. Dr. Sebsibe Hailemariam, Advisor ____________________
2. ____________________ ____________________
3. ____________________ ____________________

ACKNOWLEDGMENT

First and foremost, I would like to thank the almighty God for making me strong and for being with me in all the way. Next I would like to express my deepest appreciation and thanks to Dr. Sebsibe Hailemariam, for his great and continuous advice and encouragement.

Moreover, this work would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to Ato Siefu Yilma (sign language instructor and Sign language student at AAU, master's program), Ato Ephrem Girma (English lecturer), W/ro Aynalem Desta (translator), W/t Himanot (translator), Ato Habtamu Kebede (manager of Ethiopian national association of the deaf), Ato Kifle (sign language lecturer), and Mekanissa School of the Deaf.

Finally I would like to express my special gratitude and thanks to my whole family for their moral support and encouragement.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS AND ABBREVIATIONS

EthSL	-	Ethiopian Sign Language
ASL	-	American Sign Language
SIL	-	Summer Institute of Linguistics
ENAD	-	Ethiopian National Association of the Deaf
TVET	-	Technical & Vocational Education and Training
NLP	-	Natural Language processing
eSIGN	-	Essential Sign Language Information on Government Networks
3D	-	3 Dimensions
SiGML	-	Signing Gesture Markup Language
XML	-	EXtensible Markup Language
NP	-	Noun Phrase
VP	-	Verb Phrase
Adj	-	Adjuctive
N	-	Noun
V	-	Verb
Prep	-	Preposition
AAU	-	Addis Ababa University
HTML	-	Hyper Text Markup Language
CSS	-	Cascade Style Sheet
PHP	-	Hypertext Preprocessor
IE6	-	Internet Explorer Version 6
POS	-	Part of Speech
NLTK	-	Natural Language Toolkit
HamNoSys	-	Hamburg Notation System
MT	-	Machine translation

IR	-	Internal Representation
TL	-	Target Language
RBMT	-	Rule Base Machine Translation
SMT	-	Statistical MT
CBMT	-	Corpus Based MT
EBMT	-	Example-Based MT
VGuido	-	Virtual Guido
AmESL-T	-	Amharic to EthSL Translator
ASCII	-	American Standard Code for Information Interchange
ESL	-	Ethiopian Sign Language
DBMS	-	Database Management System
UML	-	Unified Modeling Language
CGS	-	Candidate Gesture Selection
HMTD	-	Hand Movement Trajectory Determination
MHD	-	Modified Hausdorff Distance
EMA	-	Ethiopian Manual Alphabet
SER	-	Sign Error Rate
BLEU	-	Bilingual Evaluation Understudy
DGS	-	German Sign Language
HHD	-	Hidden Marcov Model
mWER	-	Word Error Rate
mPER	-	Position-independent Word Error rate
DRS	-	Discourse
IST	-	Information Society Technologies
NGO	-	Non-Governmental Organization

ABSTRACT

Sign languages that exist around the world are usually identified by the country where they are used such as Ethiopian sign language. Mostly, the communication among the hearing impaired people involves signs that stand for words by themselves. However, to make a sign language complete as a spoken language, the hearing impaired community around the world use manual alphabets for names, technical terms, and sometimes for emphasis. As there are different alphabets for different spoken languages such as Amharic, there are manual alphabets or finger spellings used by the deaf people. Therefore sign language in general is a tool that deaf communities use to communicate with each other.

There is no problem when the communication is limited between the deaf, but they struggle to communicate with hearing people due to the language barrier. Using translators was the solution for filling the communication gap especially in Ethiopia, even if it has its own draw backs with respect to economy or privacy issue. Consequently, developing software which fills the communication gap between the deaf and hearing people is a best solution.

This thesis contributes on the development of a model and system for Amharic sentence to Ethiopian sign language translator which accepts Amharic sentences, letters, or numbers, and outputs 3D animation of Ethiopian sign language based on the pre-lingual deaf grammar. The model bases on rule based machine translation approaches and the developed system has three basic components; the interface component, the back-end component, and the database component. The first component (front-end) acts as a bridge between the users and the back-end component. The back-end component has three modules; Amharic text analysis, natural language processing (NLP), and text-to-sign mapping. Amharic text analysis modules analyze Amharic sentence and pass Romanized sentence to the NLP module. The NLP module accepts the Romanized Amharic sentence and performs all language processing and return sentence in EthSL with including of morphological information. Then the final module (text-to-sign mapping) maps each word with the SiGML (sign script) and send to the interface component and the 3D avatar animation display the sign. In addition to enhance the quality of the translator we use a POS tagging which combine the previous work (naïve Byes classifier) and the new created one; using a brill tagging approach.

The translator performance evaluated into three classes; at sentences level, letter level, and number level and the result ranked into three categories; number of correctly translated sentences, number of understandable sentences, and number of wrong translations. All results without any errors were considered as correctly translated sentences. The results that conveyed meaning but not clear sense were considered as understandable sentences. But the results that did not covey meaning as well as sense were considered as wrong translations. Finally the system gave an accuracy of 58.77%, 75.76%, and 84% at sentence, letter, and number level respectively.

CHAPTER ONE

INTRODUCTION

1.1. Introduction

Hearing impairment, deafness, or hearing loss refers to the inability to hear things, either totally or partially. Deafness has many causes and can occur at any age or at birth. People can go deaf suddenly as a complication of a virus, or lose their hearing over time because of disease, nerve damage, or injury caused by noise [27].

Deafness in Ethiopia certainly brings a special set of challenges, and nationwide, there are very few services available to the deaf community [30]. Being deaf in Ethiopia has its own troubles to access basic information or services, receive an education, communicate with the rest of the world, hold a meaningful job or trade, participate in basic community activities, and so on [30].

Deaf communities (the hearing impaired) have many distinctive cultural characteristics, some of which are shared across different countries. Sign language is at the center of these communities and it is the single most unifying characteristic [35]. The deaf acquire language in different ways, depending on their home environment. Language development plays an important role in literacy learning. Children who are deaf but born to hearing parents generally start learning language later, and with less consistent and less useful experiences. Such children do not share a native language with their family. Their hearing loss, on average, is not identified until their first birthday. These children are exposed to less linguistically rich environments than deaf children of deaf parents or hearing children of hearing parents [35]. Because of these differences in language exposure, children who are deaf start their language learning at a later age than their peers.

In families where parents are learning a new language, such as EthSL or signed Amharic, with which to communicate with their child, children have a tendency to acquire inconsistent or incorrect linguistic input [14]. People who are profoundly deaf can hear nothing at all. In order to communicate with people, they are totally reliant on lip-reading and/or sign language. People who are born deaf find lip-reading much harder to learn compared to those who became hearing impaired after they had learnt to communicate orally or with sounds.

Deaf education is the education of students with various hearing levels in a way that addresses the students' individual differences and needs [28]. Deaf education in Sub-Saharan Africa originated in the 19 century by European missionary for small number of deaf population, the project extended in 31 schools in the continent by American and other countries missionaries [18]. The acceptance of sign language has started growing in schools, work environments, and communities within the region, and now it is a major communication tool in deaf communities. Despite other languages there is no common standard or representation for sign language. Every country has its sign language representation, even there is difference in sign representation within a country, and this has an impact on the development of the language. Even if there are ongoing researches to eliminate the problem, further researches need to be conducted.

It is believed that the Ethiopian sign language has originated from American Sign Language (ASL). According to SIL Ethiopia pilot survey (2005) from 249 signs, 25% are brought from ASL by performing a little enhancement from cultural point of view [14]. The Ethiopian statistics agency's census 1994 indicated that there were about 190, 220 individuals with hearing loss [24]. The number could be twice when considering the difficulty of obtaining a reliable census.

There are two aspects to deafness; born deaf and late deaf. People who are born hearing and become deaf late in life, are physically deaf, but culturally hearing [29]. They grew up speaking a spoken language, using the telephone, the TV, the radio, and so on. They speak, read, write and base their opinions on the world they knew before they became deaf and can describe their idea in a better way than born deaf. People who are born into the deaf community, and whose first native language is a sign language, not a spoken one, are culturally deaf. Most of them are physically deaf as well. Some of them are born-deaf or became deaf at a very young age. They range from hearing people born into all-deaf families, and even though they can hear, even though they speak a spoken language, their first language was a signed language, not a spoken language. They base their view of the world from the deaf perspective. They are physically hearing but culturally-deaf [29, 6].

Both born and late deaf have their own way of communication or language structure. Late deaf use direct translation to the verbal language (signed Amharic), for example ቀዩ መኪና ሄደ is translated as ቀዩ መኪና ሄደ respectively, and it exactly follows Amharic language grammar, because their native language is Amharic. In contrast of late deaf, the born deaf native language is singed language and they have their own grammatical rules and do not follow any spoken language grammar. For example the previous sentence can be expressed as መኪና ቀይ መሄድ.

Even if both parties have their own way of ideal explanation depending on their level of understanding, as a medium of communication they use signed language. Signed language can be understood by both parties (born and late deaf) but it is difficult to communicate using the late deaf way of communication because the late deaf follows Amharic grammar (signed Amharic) which is completely not understandable by born deaf due to their previous language experience. Late deaf can understand the spoken language using lip reading. In addition they can explain their idea in a better way than the born ones on the world they knew before. Ethiopian sign language follows born deaf way of communication (sign language) and it is used as a common communication tool for both born and late deaf. Therefore, this paper's work bases on the born deaf grammar.

To increase the deaf participation in the country's economic growth, the Ethiopian national association of the deaf (ENAD) has been making efforts through supporting the united nation educational, scientific, and cultural organization's principle of "education for all" by letting the deaf to get equal educational opportunity. In this regard ENAD has opened few special schools and many special units in mainstream [6]. Additionally, the Ethiopian higher institution strategies give equal opportunity to attend their higher level education; for example in the TVET program the deaf people are motivated to attend their class by preparing translator persons for filling the gap between the trainer and deaf students.

Although in Ethiopia using translator person is a current solution for filling the gap between hearing and non-hearing people, it has its own draw back according to privacy and economic factors. Due to lack of tools to translate the languages, it is assumed as a best solution especially in Ethiopia. Therefore developing such tools have a positive role on the development of the language and incur the deaf people's participation in the development of the country's economy as well as their social participation.

This document proposes Amharic sentence to Ethiopian sign language translator. The translator accepts Amharic sentence as an input and translate it to Ethiopian sign language. The translation needs detail knowledge in input data analysis, and morphological & syntactical analysis for both languages. The translation system has three basic modules; interface, natural language processing (NLP) and eSIGN 3D avatar animation module.

The interface module handles the input text and checks whether the inputted text exists in Amharic-English (Latin) knowledge database. If it exists, it passes the equivalent Latin word to the NLP module. For example, if the user input "በ ቀዩ መኪና ሄደ", it splits each word and checks whether each word exists in the dictionary or not, and translates to Latin word and pass the word sequence "be qeyu mekina hEde" to the NLP module. In addition, it receives the words sequence in EthSL from NLP module and passes their corresponding SiGML script to 3D avatar module.

The NLP module is responsible on the processing of both languages (Amharic and EthSL). It translates the source language (Amharic text) to the target language (EthSL) which is displayed by 3D avatar animation module. In NLP module different tasks will be done, such as morphological and syntactical analysis (linguistic analysis) between the two languages. This module uses different tools or components to accomplish its task; these are Morphological analyzer, Romanizer, Part of Speech tagger, grammar translation, and preposition removal. Each component has its contribution on the translation process. And the result (EthSL text) is attached with signing gesture markup language (SiGML) that is an XML-compliant representation of gestures and passes to the animation technology. The eSIGN 3D avatar animation module is a virtual human, which can be displayed on a computer screen and is capable of performing sign language sequences.

Part of speech taggers are very useful in modern natural language processing, and have potential applications in machine translation, speech recognition, and information retrieval. They are usually used as a first step on a block of text, producing a tagged corpus that can then be worked with. Of course, the better the tagger is, the more accurate overall results will be. Therefore, to enhance the performance of the part of speech tagger we combine the previous work with the new brill tagging approach.

1.2. Statement of The Problem

In the age of globalization, information is a key for facilitating everyone's activities and it fulfills individual or organizational interests. Information is exchanged between different individuals, organizations, and countries. Individuals may have different personality, behavior and culture; in addition, some of them may have disability such as hearing impairment. Those differences can be a barrier for communication and may lead to misunderstanding. Beside these barriers, language differences have a negative impact in communication such as communication gap between the deaf and non-deaf people.

For a long period of time the deaf people communicate with only deaf people and there is no problem at all when the communication is limited between them. However, they face problems when they struggle to get information from hearing people due to the misunderstanding between the two parties; the deaf people with hearing people. Since years ago, the two parties communicate using a person as an intermediate for the translation of these two languages.

In Ethiopia using a translator person is a solution for filling the communication gap between the deaf people with non-deaf people. Even though the translators play a great role in the translation process, sometimes it's difficult to exchange confidential information between these two parties, for example in closed court, medical area, and different social issues. In addition, using translator person is not economical cost, time, and effort wise.

In most developed countries the above stated problems can be eliminated using translation tools which perform translation and fill the communication gap instead of using translator person. But this tool is not available in Sub-Saharan Africa especially in Ethiopia and also no research has been conducted in the development of Amharic sentence to EthSL translation. Due to the lack of Amharic sentence to Ethiopia sign language translator, the only way of translation is done using an intermediate person. Therefore, developing such model as well as tools for EthSL eliminates the existing problem and can be a solution for filling the communication gap between the deaf and non-deaf people.

1.3. Motivation

Developing automatic translation of Amharic sentence to Ethiopian sign language has a very good role on facilitating the communication between the deaf and hearing people. Communicating without an intermediate translator (person) is difficult if the two parties need privacy, but the proposed model solves this problem by letting them communicate directly without the intervention of other parties. Another benefit of the proposed model is it supports the translator by reducing their burden in the translation process and it also has a positive contribution on development of standardized sign language within a country.

The previous researches conducted in Ethiopia were limited at word, letter, and number level. For example if we feed the word በላ to the prototype system, it analyzes the stem word using morphological analyzer and translates it to the corresponding English text (Latin) and displays its sign. But the researches did not handle or translate the word sequences. In contrast to the previous researches, our proposed model translates simple Amharic sentence to Ethiopian sign language. For example, if we feed sentence ዳንኤል ቀይ መኪና ነዳ it translates to the corresponding sign that is represented by the sentence ዳንኤል መንዳት መኪና ቀይ.

1.4. Objective

1.4.1. General Objective

The general objective of this research is to design a model for Amharic sentence to Ethiopian sign language translator.

1.4.2. Specific Objective

To achieve the general objective, the following specific objectives are performed

- Study the linguistic structure of both languages as well learn about EthSL
- Design model to Amharic sentence to Ethiopian sign language translator
- Develop a prototype system
- Test a prototype system

1.5. Scope and Limitation

The scope of this research is on translation of simple Amharic sentence to Ethiopian sign language. Sentences are a small group of words that align together. A simple Amharic sentence is a sentence which constructed from noun and verb phrases. Both nouns and verb phrase can be constructed from መስተዓምር፣ አጎላምሽ፣ መሙያ እና መሪ [4].

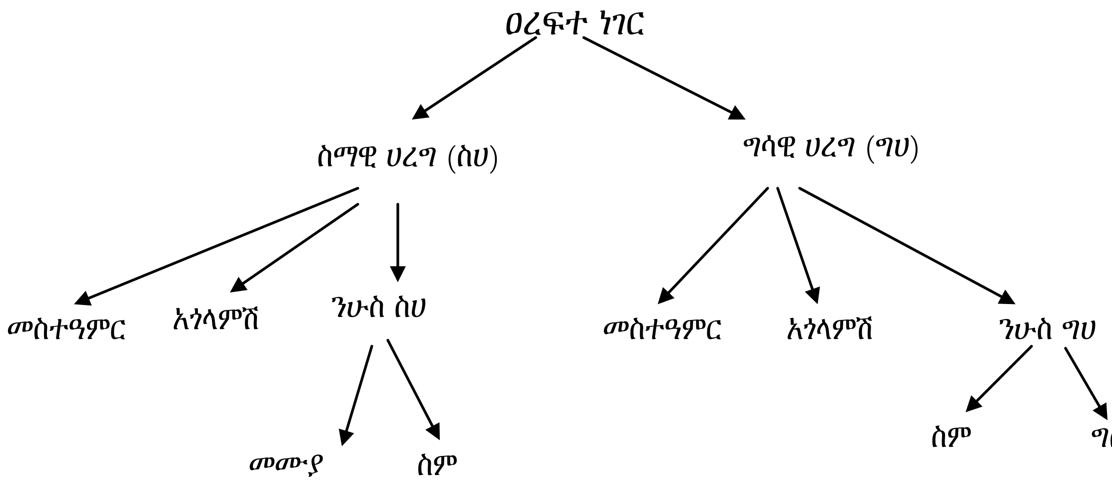


Figure 1.1: *Structure of the expected input sentences*

The verbs can be plain or inflecting/indicating verbs. Plains verbs are verbs which does not incorporate subject and object information, e.g መንዳት፣ መብላት፣ ወዘተ. And inflecting/indicating verbs are verbs which incorporate subject and object information in a verb, e.g. ነዳ፣ በላች፣ ወዘተ and the nouns are common nouns (eg. መኪና፣ ዉሃ) and/or proper nouns (eg. ኢትዮጵያ፣ ማርታ). And the paper only address the preposition which is separated from words, for example instead of typing በመኪና by combining the preposition with noun it accepts the preposition በ and noun መኪና separately; በ መኪና.

Although the study achieves its objective, there were some limitations because of the time constraint. The research conducts simple Amharic sentence without considering the tense, even if the system translate complex Amharic sentences with some extent, it has poor quality, due to the time constraint we unable to study complex nature of the language (EthSL). In addition object/subject word contrastive and time placement in a sentence for unknown object was out of the scope.

1.6. Methodology

1.6.1. Literature Review

For providing a background for the topic, to evaluate the depth of the research, and to clearly identify the tools, approach, and method of the research, literature review has been conducted. Previous works, dictionaries, newspapers, learning materials, and online documents in different languages were reviewed.

In the recent years several works have been done in machine translation. Even if various researches on text to sign language translation for different languages within a text or sentence level have been conducted, no research has been done on Amharic sentence to Ethiopian sign language translation.

1.6.2. Data Collection Methods

Data Collection is an important aspect of any type of research study. Our data collection starts with determining what kind of data are required followed by techniques of data collection. There are different techniques of data collection which have their advantages and limitation; we select three techniques to collect data according to the research requirement.

- observing directly
- interviewing in depth and
- analyzing documents

1.6.2.1. Interview

We use this method to collect data about the user requirements and identify the communication gap between the deaf and non-deaf people in addition to understanding the grammatical structure of late and born deaf way of communication. We will interview different individuals that have direct relation to the language from different area; one person from AAU linguistic department (sign language department), 4 translators from different academic and non-academic organization, 2 deaf people (born and late deaf), and the president of ENAD.

1.6.2.2. Observation

Observation is a fundamental and highly important method. It is used to discover complex interactions in real settings. Using the observation technique we will observe the translation process to get knowledge about the two languages relationship especially in the actual manual translation process and in sign language training center. In parallel with observing the language we are trained to understand what sign language means through attaining sign language training class at Addis Ababa University for one month. By conducting the real environment we can be able to collect information about both languages.

1.6.2.3. Document Analysis

The actual signs and corresponding texts are gathered from different documented materials; dictionaries, newspapers, online documents, and training materials which prepared by ENAD and some non-governmental organizations. These documents give information that will help us to achieve the general objective of the research.

1.6.3. Approach

Machine translation is an automated process that computer software uses to translate one language to another natural language, such as Amharic text to Ethiopian sign language. To develop machine translation system there are different approaches or technologies.

We prefer rule-based machine translation approach. This approach works on written rules by experts and use bilingual dictionaries. It guarantees and improves quality; users can also improve the translation quality by adding their terminology into the translation system. Rule-based translation model is better especially when getting bilingual corpora is difficult or hard. And also its quality is predictable besides using less resource utilization. Therefore, for the natural language translator module of the proposed system we use rule-based translation model.

1.6.4. Tools

To achieve the objective of the research we will use different tools.

- **Web Development**

Macromedia Dreamweaver 8 was used to develop the graphical user interface. Dreamweaver is an integrated web development environment. It is used primarily to design and code HTML and CSS; in addition, it can also be used for other languages such as PHP.

- **Browser**

To interact with system and run eSIGN 3D avatar animation plug-in Microsoft Internet Explorer 6 (IE6) or latest version were used.

- **NLP Tools**
  - HORNMORPHO

    It is a python program which analyzes, generates, and segments the given Amharic, Oromo, and Tigrinya words. We use morphological analyzer for analyzing the given source language (Amharic) and to get the root word in addition to extracting morphological information about word.
  - POS tagger

    We develop a POS Tagger using python 2.7 program and NLTK tool kit which reads text in source language (Amharic) and assigns parts of speech (such as noun, verb, adjective, etc) to each word.
  - Romanizer

    Romanize is a python program which is the conversion of writing from a different writing system to the Roman (Latin) script. This program is used to Latinize the given Amharic words and also to Geezify the Latin words.
  - Grammar translation

    We develop a python program to translate the source language (Amharic) to a target language (EthSL) grammar.

- **eSIGN**
  - eSIGN editor

    The eSIGN editor software allows us to write signed text in terms of HamNoSys and convert to SiGML file for signing using eSIGN 3D Avatar animation.
  - eSIGN avatar plug-in

    It is virtual signing technology which uses virtual humans or Avatars. This technology is still in development by German Hamburg University, virtual signing can be better than using videos of real human signers. Virtual signing will be quicker to download than videos and does not take up lots of space on Internet servers. It is plug-in for Internet Explorer 6 and above.

- **MSQL Database**

It is characterized as a free, fast, reliable open source relational database. It is comprised of enhanced MySQL, Apache, PHP, and Perl, to create an integrated web development environment. This technology suite delivers a complete, stable environment for building and deploying database driven applications for the internet.

MYSQL is widely used for web applications. This acts as a database component for the prototype system as well as PHP used for managing database and web applications.

CHAPTER TWO

LITERATURE SURVEY

2.1. Literature Review

2.1.1. Machine Translation

Machine translation (MT) is an important sub-discipline of the wider field of artificial intelligence. AI, among other things, deals with getting machines to exhibit intelligent behavior. A machine translation system essentially takes a text in one language (called the source language), and translates it into another language (called the target language) [10, 31]. The source and target languages are natural languages such as English and Amharic.

- **Components of MT System**

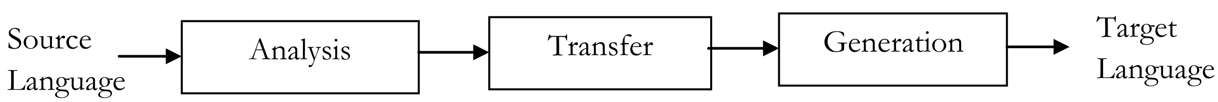


Figure 2.1: *Components of MT system [10]*

Machine translation task can be divided into two or three main phases. The system has to first analyze the source language input to create some internal representation. It then typically manipulates this internal representation to transfer it to a form suitable for the target language. Finally, it generates the output in the target language [10].

A typical MT system contains components for analysis, transfer and generation. These components incorporate a lot of knowledge about words (lexical knowledge), and the languages (linguistic knowledge) [10]. Such knowledge is stored in one or more lexicons, and possibly other sources of linguistic knowledge, such as grammar. The interface allows users to verify, disambiguate and if necessary correct the output of the system [10]. Another common feature of NLP work is the use of large corpora. A corpus is a large collection of text which has been appropriately tagged, and is used for acquiring the required lexical and linguistic knowledge.

The lexicon is an important component of any MT system [10]. A lexicon contains all the relevant information about words and phrases that is required for the various levels of analysis and generation. A typical lexicon entry for a word would contain the following information about the word: the part of speech, the morphological variants, the expectations of the word (or the typical words, phrases or constructs that this word typically goes with), some kind of semantic or sense information about the word, and information about the equivalent of the word in the target language. Some systems prefer to split the lexicon into a source lexicon, a target lexicon, and a transfer lexicon that maps between the two. The exact format of the lexicons is a matter of engineering design, and would take into account the system designer's policy about issues like how to handle morphological variations, multiple word senses, synonyms and so on [10].

- **Approaches to Machine Translation**

The approaches to MT can be divided into three major classes: - direct, transfer-based and inter-lingual. A direct MT system tries to directly map the source language to the target language, and is therefore highly dependent on both the source and target languages. A transfer-based approach first converts the source language into an internal representation (IRs) which is dependent on the source language [10]. The system then transforms IRs into a TL-oriented (target language) representation which depends only on the target language, and finally generates the target language output from TL-oriented. The inter-lingual approach converts the input into a single internal representation (IR) that is independent of both source and target languages, and then converts from this into the output [10].

- **Computational Architecture of MT**
    - **Rule-based approach**

Rule-based MT Systems rely on different levels of linguistic rules for translation [26]. This MT research paradigm has been named rule-based MT due to the use of linguistic rules of diverse natures. For instance, rules are used for lexical transfer, morphology, syntactic analysis, syntactic generation, etc. In RBMT the translation process consists of [26]:

- Analyzing input text morphologically, syntactically and semantically
- Generating text via structural conversions based on internal structures

These steps make use of a dictionary and a grammar, which must be developed by linguists. This requirement is the main problem of RBMT as it is a time-consuming process to collect and spell out this knowledge, frequently referred as knowledge acquisition problem [26].

- **Corpus-based approach**

Corpus-based machine translation, also referred as data driven machine translation, is an alternative approach for machine. Corpus based MT automatically acquires the translation knowledge or models from bilingual corpora [26]. Since this approach has been designed to work on large sizes of data, it has been named corpus-based MT. There are two types of CBMT; statistical machine translation (SMT) and example-based machine translation (EBMT) [26].

Statistical machine translation (SMT) employs a statistical model, based on the analysis of a corpus, to generate text in the target language. SMT starts with a very large data set of good translations, which have already been translated into multiple languages, and then uses those texts to automatically infer a statistical model of translation. That statistical model is then applied to new texts to make a guess as to a reasonable translation [26].

Example-based machine translations (EBMT) translate a source sentence by imitating the translation of a similar sentence already in the database [26]. EBMT requires a bilingual corpus of translation pairs, normally aligned at sentence level. Exact matches are rare and the probability of retrieving one decrease as the sentence length/complexity increases, so systems facilitate partial matching [26].

- **Hybrid approach**

Some recent work has focused on hybrid approaches that combine the transfer approach with one of the corpus–based approaches. This was designed to work with fewer amounts of resources and depend on the learning and training of transfer rules. The main idea in this approach is to automatically learn syntactic transfer rules from limited amounts of word aligned data [26]. This data contains all the needed information for parsing, transfer, and generation of the sentences.

2.1.2. Morphological Analyzer

Morphological analyzer is a software component capable of detecting morphemes in a piece of text. Amharic is morphological complex language [39], it needs to be pre-processed before it is indexed and one of the common pre-processing steps is stemming. Stemming is a process through which suffixes are removed and words are converted to their stems. For example, the word "አልመጣሁም" might be stemmed to "መጣ". Amharic has more complex morpho-syntactic characteristics (e.g. different suffixes or prefixes can be used with a single word depending on the tense, gender, number, case, etc.) and thus more complex rules for their stemming are needed.

Many works have been conducted in morphological analysis and have developed some application in different languages. HornMorpho (a project by Michael Gasser at Indiana University, school of informatics and computing) is a Python program that analyzes Amharic, Oromo, and Tigrinya words into their constituent morphemes (meaningful parts) and generates words, given a root or stem and a representation of the word's grammatical structure.

We adopt HornMorpho in the development of the prototype system to get the stem or root words and to get morphological information of a source language (Amharic text).

2.1.3. Romanization

Romanization is the process of representing a language in Latin script [38]. Latin script, or Roman script, is a writing system based on the letters of the classical Latin alphabet and extended forms thereof. Languages that use scripts that are not Romanized are sometimes put into Latin script to make language learning easier or in some cases to reduce the amount of knowledge needed to write in a language [38]. There are usually regular systems for Romanization in languages that make reading Romanized texts much easier, but sometimes there is more than one system in use.

2.1.4. Part of Speech Tagging

POS tagging is the process of assigning a part-of-speech tag such as noun, verb, pronoun, preposition, adverb, adjective or other lexical class markers to each word in a text. POS tagging is not useful by itself but it is generally accepted to be the first step to understanding a natural language. Most other tasks and applications heavily depend on it. In addition to that, POS tagging is seen as a prototype problem because any NLP problem can be reduced to a tagging problem. For example, machine translation can be seen as the tagging of words in a given language bywords of another language; speech recognition can be seen as the tagging of signals by letters and so on.

There are three approaches to solving this tagging problem based on two fundamental concepts: rules and statistics. Rule-based part-of-speech tagging is the oldest approach that uses hand-written rules for tagging. Rule based taggers depend on dictionary or lexicon to get possible tags for each word to be tagged. Hand-written rules are used to identify the correct tag when a word has more than one possible tag. Disambiguation is done by analyzing the linguistic features of the word, its preceding word, its following word and other aspects [16].

A stochastic approach includes frequency, probability or statistics. The simplest stochastic approach finds out the most frequently used tag for a specific word in the annotated training data and uses this information to tag that word in the un annotated text. An alternative to the word frequency approach is known as the n-gram approach that calculates the probability of a given sequence of tags. It determines the best tag for a word by calculating the probability that it occurs with the n previous tags, where the value of n is set to 1, 2 or 3 for practical purposes. These are known as the Unigram, Bigram and Trigram models [16].

The third approach combines the best of both concepts. None of them is perfect for all languages and for all purposes. The relevance and effectiveness of each approach depends on the purpose and the given language.

2.2. Related Works

2.2.1. Automatic translation of Amharic text to Ethiopian sign language

The input for the system is Amharic word, letter and number. The output of the system is avatar animation of Ethiopian sign language. The signs are represented by means of VGuido (the eSIGN 3D avatar) animations. For creating gesture animations, eSIGN editor is used. The author developed morphological analyzer for handling the morphological variation and its scope is limited to perfective form verbs (e.g. ሰበረ፣ ሮጡች) (bound with personal pronoun suffixes) and nouns that are simplex (e.g. ቤት) (bound with possessive pronoun suffixes).

This study focuses on Amharic personal and possessive pronouns as they are bound morphemes on verbs and nouns. The system architecture of the proposed Amharic text into ESL translation system is composed of the following three essential modules [21].

1. Input Text Analysis
2. Text to Sign Mapping
3. Sign Synthesis/Generation

The first module accepts Amharic text (word, letter, or number) and analyzes the input depending on the nature and type of the input and transfers to the next module [21]. The second module transfers the written Amharic text to equivalent sign structures and map input words with corresponding signs, and sends signed sequence to final module. Finally, in the last module converts script of the animation into ESL animation and the ESL is displayed. The general architecture of the system is illustrated diagrammatically as follows:

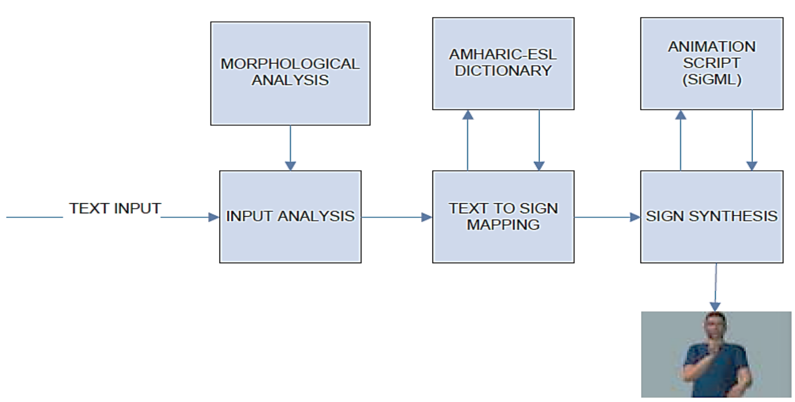


Figure 2.2: *Architecture of the developed prototype system [21]*

The author took two deaf and two ESL interpreters in the evaluation of the AmESL-T. The participants were presented with selected signed words, letters and numbers and asked to write down what they understood. 30 words, 34 letters and 20 numbers were selected randomly for this purpose to check the accuracy of identification of words, letters and numbers across four users. The average accuracy of identification was 51% for words, 76% for letters and 80%for numbers [21].

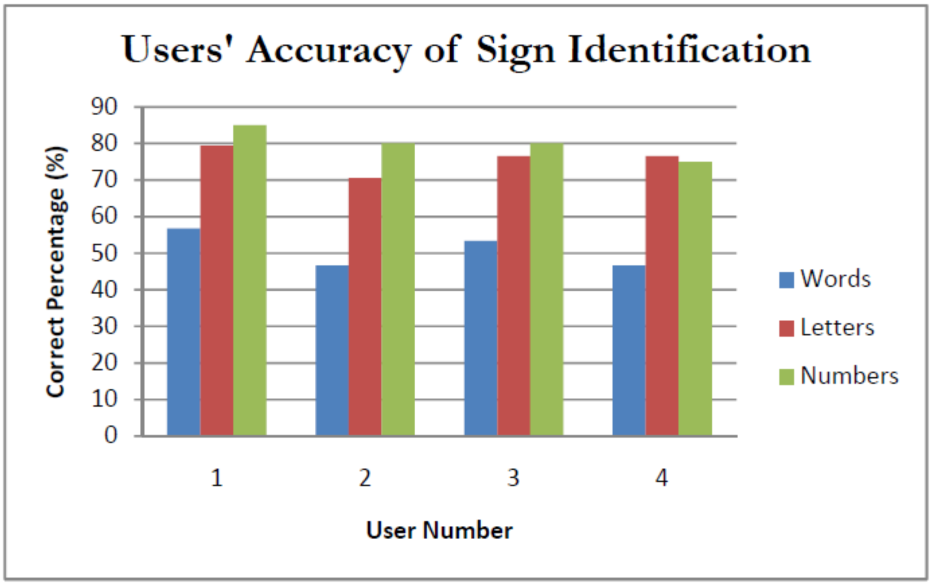


Figure 2.3: *The performance of the developed system [21]*

Participants were asked to rate how acceptable the words, letters and numbers were as an example of ESL on a 3-point scale (1="Low" 2="Medium", 3="High"). The percentage of words, letters and numbers rated in each category of acceptability [21].

Table 2.1: *Percentage of words, letters and numbers acceptability rate*

ACCEPTABILITY RATING		% WORDS	% LETTERS	% NUMBERS
HIGH	3	33.3	35.0	35.3
MEDIUM	2	46.7	50.0	50.0
LOW	1	20.0	15.0	14.7

The author points out that there is scope for improving the quality of the avatar's signing and it can be improved by correcting HamNoSys transcription of a sign to correctly convey the meaning in sign language [21].

2.2.2. Machine translation system for Amharic text to Ethiopian sign language

The MT system function is to translate a given Amharic text words to their equivalent representation in ESL. The system is designed for limited vocabulary, and rule-based machine translation (RBMT) approach is deployed in the research. The thesis' scope is limited to morphological level translation; i.e. direct RBMT architecture was followed. The HamNoSys and SiGML tools were applied for the displaying ESL, and randomly selected fifty words and all alphabets were used to test the performance [8]. The system was designed to map a given Amharic word to ESL equivalent words using morphological analysis and synthesis. If the word is not in the bilingual dictionary, it'll finger spell rather than ignoring it.

The system accepts Amharic words in text format and outputs the equivalent in ESL by using signing avatar. To achieve this function three main processing and one supporting systems are required. The figure below shows the block diagram of the overall system

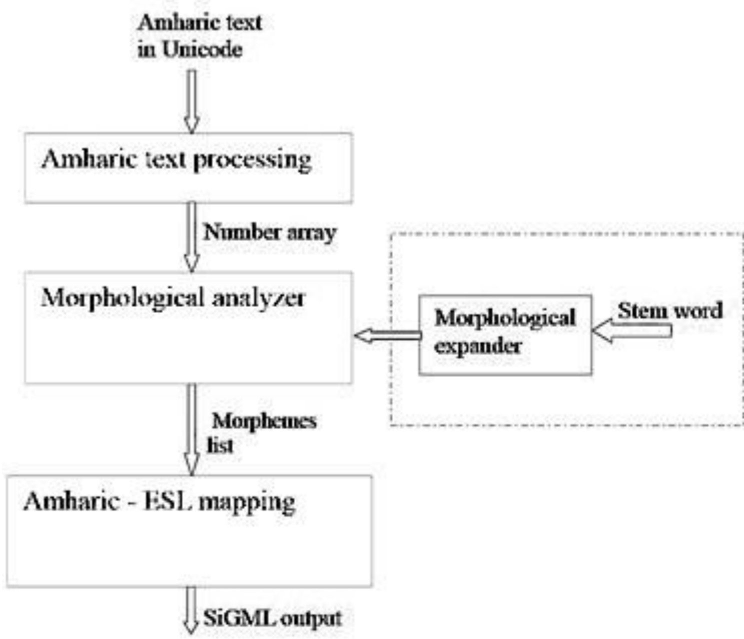


Figure 2.4: *Architecture of the developed system [8]*

Text processing part is responsible to change a given Unicode written Amharic input to ASCII representation, by using the SERA representation [8]. This part also implements the conversion from Ge'ez or SERA to number array and vice versa.

Morphological analyzer contains the morphological expander, maintains the Amharic dictionary and when invoked, it'll load the dictionary on the memory with all the needed information.

The goal of Amharic-ESL mapping sub system is to change a given text word(s) in Amharic to ESL [8]. The morphological analyzer sub-system prepares the word(s) and this sub-system holds the conversion key. The researcher used three deaf, which are native ESL for evaluating the Amharic-ESL mapping sub system, and they were asked to write down the ESL word signed by the avatar from a decoded input [8]. They were also asked to write any comment they had on that specific word, such as, needs little refinement, satisfactory and so on. For the finger spelling part, each alphabet is presented and they were asked to give values 1(very poor), 2(poor), 3(satisfactory), 4(good), and 5 (perfect) [8].

Table 2.2: *Assessment of finger spelling*

Signer	No of alphabets	Assessment value				
		1	2	3	4	5
1	34	5	1	3	0	25
2	34*	2	8	6	5	12
3	34	0	4	8	5	17

Table 2.3: *Words evaluation in ESL dictionary*

Signer	No of words	Correct	Incorrect	
			Replaced	Jumped
1	28	21	1	6
2	28	23	1	4
3	28	25	1	2

From evaluation table for finger spelling more than 50% of the alphabets are perfectly transcribed [8]. If a word which is not in ESL dictionary is given to the system, the output sign to be perceived is greater than or equal to 3, which is about 80% i.e. 81 out of 101 [8]. If the word is in the ESL dictionary, they can perceive more than 82% (67 out of 84). The overall system performance is, therefore, more than 80% [8].

2.2.3. Amharic speech training for the deaf

The thesis addresses automated speech training. The author proposed modeling of a lip for the articulation of Amharic characters which is part of an automated speech training system. He used an analysis-synthesis approach to first analyze a real lip in speech making and applied the output of the analysis on his lip model to articulate different Amharic characters. The solution proposed is implemented in a prototype developed for selected Amharic characters and its efficiency is tested with some students of Mekanissa deaf school.

The system accepts user input character for which a lip articulation is to be shown [9]. In Amharic language one letter represents CV (Consonant Vowel) combination. So, whenever a person chooses a letter the system identifies the consonant as well as the vowel to be articulated. Once the consonant and vowel are identified, the next step is finding which lip gesture can make up the required articulation. Then the database class identifies the coordinates of control points on the lip poses that correspond to the chosen letter and fetches it to the drawing class in which the curves will be painted [9]. The system then shows different modeled lip gestures of the chosen letter in sequence with some time interval. The figure below shows the architecture overall system [9].

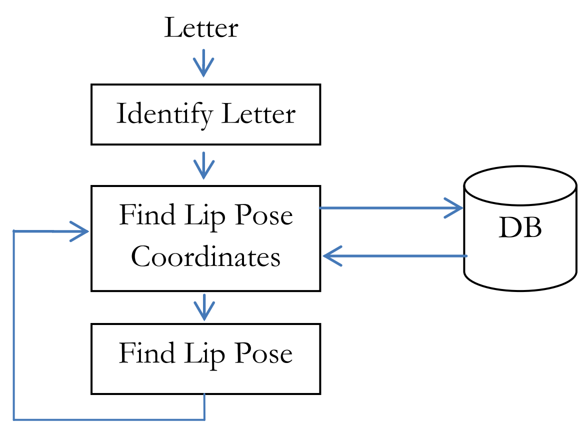


Figure 2.5: *The architecture of Amharic speech training for the deaf*

The relational DBMS, Microsoft Access is selected for persistent data storage. The database comprises tables to store the inner and outer contours' control point coordinates. For each lip pose of any given letter stored the letter's x, y and z coordinates of each control point, and lip pose number as a given letter has more than one lip pose. The Lip model is tested with 15 male and 15 female hearing impaired students of "Mekanissa deaf school" [9]. The group comprises students from KG level up to grade 8. They showed the lip model to the students articulating each of the seven orders of the Amharic letter "መ" and the result is shown in next table [9].

Table 2.4: *Result of student articulation*

	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
No. of Students with correct Articulation	17	23	15	20	24	21	27
Percentage of Students with correct articulation	56.7%	76.7%	50%	66.7%	80%	70%	90%

Table 2.5: *Result of students by gender*

		መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
No. of Students with correct articulation	Male	10	10	6	11	12	8	13
	Female	7	13	9	9	12	13	14

Generally the model was tested with 30 students (15 male and 15 female) from "Mekanissa deaf school" and it shows 70% accuracy. The result shows the letters with less similarity than the others and can only be seen that are easily readable by lips and the letter that includes visible tongue movement along with the lips has some difficulty to be identified. This shows that our lip model can be more efficient if it is integrated with other articulators [9].

2.2.4. A hypermedia learning tool for Ethiopian sign language

The study examined the development of a prototype of web-based hypermedia learning tool for Ethiopian sign language taking health environments as a case to contribute in bridging the communication gap [11]. The tool was designed using a UML based hypermedia designing methodology. The study has made its focus on health centers for the sake of narrowing the scope of the research, and bearing in mind that health is the primary issue for anybody [11].

Evaluation of the tool was done in two phases [11]. Primarily, the researcher was working closely with the intended users. During this time the users gave feedback and the researcher has considered all their needs. This was taken as the first phase. With the second phase the evaluation was done by forming groups, having 7 health professionals at Shiromeda clinic, 6 health professionals at Minilik-ii hospital. The evaluation result showed that the users can easily understand and use the tool for the intended purpose [11].

2.2.5. A machine translation system to translate Amharic text to Ethiopian sign language

The research document presents a machine translation system to translate Amharic text to an equivalent Ethiopian sign language (ESL) by finger spelling representation with the help of 2-D animating avatar. He adopts the three paths of translation [25].

- Direct
- Transfer
- Interlingua

The author use Macromedia Flash 8.0 tool and test the result by 10 deaf people and 66.6% of them responded that the overall performance is good and supportive [25]. The author limits his scope:

- Input specific to an Amharic text typed with the phonetic model of Geez alphabet
- Internal process use multi-path machine translation approach
- Output limited to only the application of Ethiopian finger spelling to convey meaning by ESL because he believed fingers are quite manageable.

The author was not concerned about signing (conceptual sign), but only about finger. In addition, he was only concerned about a single word, not applicable for phrase, sentence, or document level translation [25].

2.2.6. Offline candidate hand gesture selection and trajectory determination for continuous Ethiopian sign language

In this thesis, a system that extracts candidate gestures for EMA (Ethiopian manual alphabet) and determines hand movement trajectories is proposed [1]. The system has two separate parts:

- Candidate gesture selection (CGS)
- Hand movement trajectory determination (HMTD)

The CGS combines two metrics namely speed profile of continuous gestures and modified hausdorff distance (MHD) measure and has an accuracy of 80.72% [1]. The HMTD is done by considering each hand gesture centroid from frame to frame and using angle, x- and y-directions [1]. A qualitative evaluation of the CGS in a correctly divided video clip is found to be 94.81% [1]. The HMTD has an accuracy of 88.31%. The overall system performance is 71.88% [1].

The author considers EMAs except the base EMA (the first letter and the last or 7th form in this work) [1]. The base alphabet is left out because the proposed design works based on speed profile between successive gestures to divide the sequence of video frames into blocks while the base EMA is created using the gesture without any movement. EMA is created by starting with the valid EMA and followed by a hand rotation. So it is not convenient to include this EMA because it just rotates around a fixed axis without moving to any of the directions. And this is not suitable to deal with speed profile [1]. The overall architecture for the proposed system is show bellow [1]:

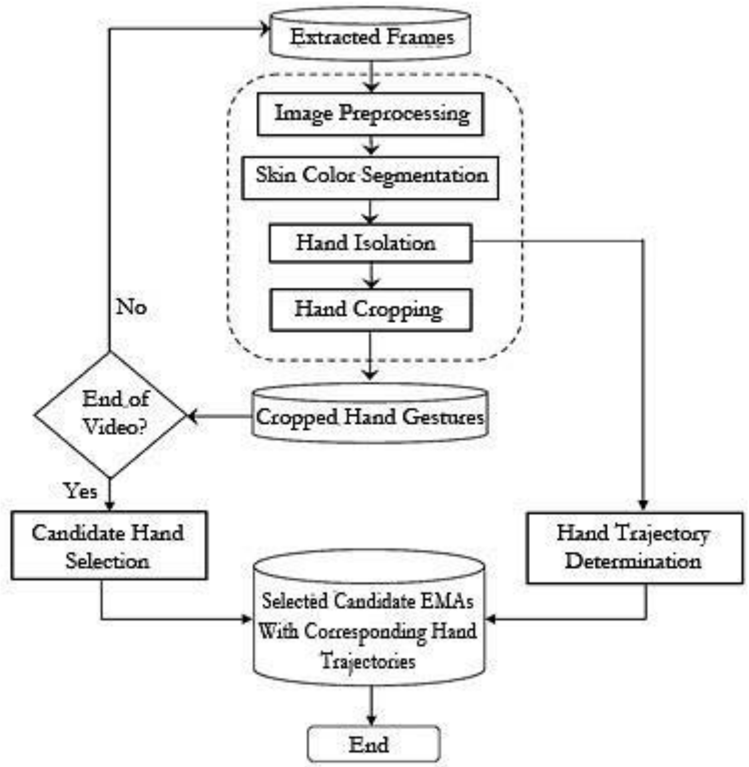


Figure 2.6: *The architecture of offline candidate hand gesture selection and trajectory determination*

2.2.7. Speech to sign language translation system for Spanish language

The research was the first document in spoken Spanish (for applying or people renewing for identity card) to Spanish sign language translation system. The translation system is made up of speech recognizer, natural language translator, and 3D avatar animation module. In addition, two approaches were evaluated for natural language named; rule-based translation and Statistical translation [30]. A speech recognizer is used for decoding the spoken language into a word sequence. After that, a natural language translation module converts the word sequence into a sequence of signs belonging to the Spanish sign language (LSE). In the last module, a 3D avatar plays the sign sequence.

The two proposals for the natural language translation module have been implemented and evaluated. The first one consists of a rule-based translation module reaching 31.60% SER (sign error rate) and 0.5780 BLEU (bilingual evaluation understudy). The confidence measures from the speech recognizer have been used to compute a confidence measure for every sign. This confidence measure is used during the sign animation process to inform the user about the reliability of the translated sign [30].

The second proposal for natural language translation is based on statistical translation approach where a parallel corpus was used for training. The best configuration has reported a 38.72% SER and a 0.4941 BLEU [30].

The rule-based strategy has provided better results on this task because it is a restricted domain and it has been possible to develop a complete set of rules with a reasonable effort. Another important aspect is that the amount of data for training is very little and the statistical models cannot be estimated reliably [30]. The overall architecture of the system is shown below [30]:

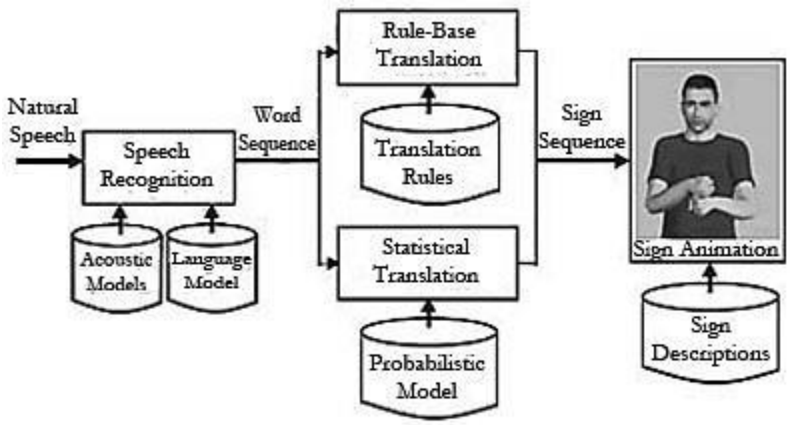


Figure 2.7: *The architecture of speech to sign language translation system*

2.2.8. Statistical sign language translation

The authors suggest a statistical machine translation system for sign language and written language, especially for the language pair German sign language (DGS) and German. They propose a system for translating a sign language into a spoken language and vice versa. Such a complete system translating from sign language to spoken language needs a gesture recognizer as input, the translation system and a speech translator as output.

The complete system translating from spoken language to sign language needs a speech recognizer as input, the translation system and a graphical avatar as output. In this paper the focus is held on the translation part. The general architecture of automatic sign language translation system looks as follow [19]:

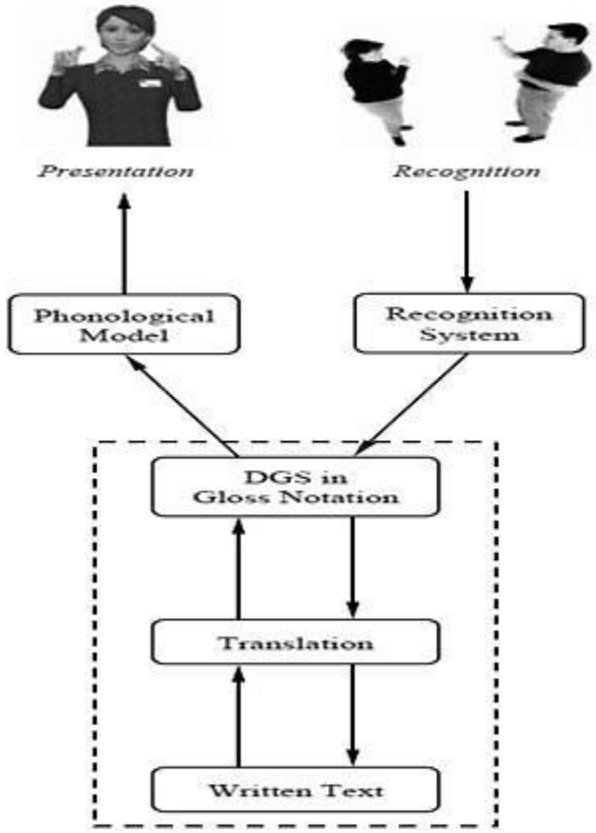


Figure 2.8: *The architecture of statistical sign language translation*

Statistical machine translation trained using bilingual corpora containing full sentences. For starting basis the corpus collected by the DESIRE team Aachen consisting of 1399 sentences in DGS (German sign language) and German. The corpus is divided into training data samples (83% of the sentence) and testing sample (17% of the sentence). The training is performed by using various statistical models like IBM model 1-4 and another like Hidden Marcov model (HHD). The mWER (word error rate) and mPER (position-independent word error rate) error rate, on the performed translation of German to DGS is as follow [19]:

Table 2.6: *Word error rate*

	mWER (%)	mPER (%)
Single word	85.4	43.9
Alignment templates	59.9	23.6

2.2.9. English to American sign language machine translation of weather report

The author performed on the English text to American sign language translation specifically in the area of national weather service forecasts. The author developed a prototype machine translation application using Don Newkirk's literal orthography, a system that uses the Roman alphabet for writing signs. To produce a fluent translation the author used a lexicon and grammatical principles [3].

2.2.10. The architecture of an English text to sign languages translation system

The authors present an overview of the overall architecture of the language processing component of an English text to sign languages translation system. The authors present the overall components of language processing to English to sign language translation. They also present the syntactic, semantic and discourse oriented natural language processing techniques, which are implemented to generate the DRS-based semantic representation from English text [15].

2.2.11. Automatic part of speech tagging for Amharic

He is the pioneer for Amharic POS tagging experiments. He developed a tagging prototype using Hidden Markov models, which he trained and tested on a text of one page. His contribution also included the definition of a tagset of 25 that has served as a basis for the tagsets used by subsequent researchers [18].

2.2.12. Part of speech tagging for Amharic using conditional random fields

The author collected five news articles and manually annotated them, which he then used for both training and testing of a stochastic model based on conditional random fields. He obtained an average accuracy of 74%. The main reason for the poor performance is the small size of the dataset. 80% of the words in the test files consist of unseen words. From this result and successful experiences in other experiments for large datasets, it became clear that Amharic POS-annotated data is necessary to achieve performances comparable to the state-of-the-art results [32].

2.2.13. Methods for Amharic part-of-speech tagging

In 2006, a medium-sized corpus of reportedly 210, 001 tokens annotated with parts of speech was released (Demeke and Getachew, 2006). The corpus consists of 1065 news articles collected from Walta Information Center (WIC), a private news agency located in Addis Ababa. It is tagged with 312 parts of speech and is publicly available on the internet. This corpus has been a useful resource for the recent experiments on Amharic POS tagging.

Using the Walta information center (was released by Demeke and Getachew in 2006) corpus, the authors applied different tagging methods and obtained worse performances than the state-of-the-art results for Arabic or English. Gamback conducted detailed experiments using three different tagsets. The overall accuracies using the ELRC3 tagset are 85.56% for TnT, 88.30% for SVM and 87.87% for MaxEnt Similarly [17].

2.2.14. Part of speech tagging for Amharic

The author discusses theoretical and practical POS tagging issues with the view to improving POS tagging performance for Amharic. For tagging experiments, the author applied conditional random fields; Brill and TnT (HMM-based) have been used. With the experiments carried out on Walta information center corpus, POS tagging accuracies for Amharic have crossed above the 90% limit for the first time [5].

CHAPTER THREE

ETHIOPIAN SIGN LANGUAGE (ETHSL)

3.1. Ethiopian Sign Language (EthSL)

The deaf of Ethiopia live similarly to any other person within their given cultures, but are cut off from meaningful interaction with others. The vast majority of deaf Ethiopians, who live in rural areas, spend their lives in extreme isolation. They are looked down upon as mentally deficient and evil because of their lack of spoken communication. In many places they are misunderstood as being a result of sinful behavior, or some form of supernatural curse. They are not seen as suitable marriage partners and may even result in the entire family's loss of status [12]. For this reason they are frequently sheltered even further from the outside world and communicate only with their families or those close to them through small amounts of writing or signing, if they are able.

In towns more awareness has been generated regarding the deaf. Many parents are eager to send their children to schools, although the resources available are not sufficient for the number of potential students. Missionaries, and more so lately, the government, have established several schools for the deaf. In more recent years clubs for the deaf have been established in some towns, helping the deaf to be less isolated and allowing sign language to be brought into use [12].

A sign language is a language which, instead of acoustically conveyed sound patterns, uses manual communication and body language to convey meaning. This can involve simultaneously combining hand shapes, orientation and movement of the hands, arms or body, and facial expressions to fluidly express a speaker's thoughts. They share many similarities with spoken languages (sometimes called "oral languages", which depend primarily on sound), which is why linguists consider both to be natural languages, but there are also some significant differences between sign and spoken languages.

Sign language was first taught in Ethiopia by American missionaries and is based on American sign language (ASL) and signed English [13, 14]. It has been modified to suit Ethiopian culture but may still be intelligible with ASL.

In Ethiopia, sign language was first used formally after 1960's in connection with the appearance of American and Nordic missionaries who opened schools for the deaf [6]. The missionaries brought the sign language used in their own countries. For more than 50 years, the foreign sign languages were assimilated with Ethiopian deaf culture and sign language.

The deaf community along with their association (The Ethiopian national association of the deaf) and along with other concerned bodies have been working for the institutionalization of the Ethiopian sign language. As a result of their concerned efforts, they have now a well-developed sign language which is used as a medium of instruction in schools and as means of communication by the media.

The deaf in Ethiopia constitute a linguistic minority whose human and constitutional rights should be addressed along with other linguistic entities. To address the needs of this less privileged linguistic group by developing and promoting the national sign language that is being used now is developing and promoting equality and helping them to contribute in the development of the country [36].

3.1.1. Parameters of EthSL

It's difficult to see in detail when viewing something for the first time, especially something as complicated as sign. There are fingers, hands, arms, cheeks, lips, and eyebrows moving every way continuously and simultaneously. EthSL constitutes five parameters, each having a limited range of possibilities. These parameters will give you something to focus on:

- **Hand shape:** The proper hand shape is the first parameter needed. How the hands are shaped when making signs can change the meaning of the word or expression you are trying to communicate. The basic hand shapes used for finger spelling are also used to make words when combined with other movements or signs. There are 41 hand shapes in the EthSL which are indicated by distinctive symbols [12]. For example, figure 3.1 shows hand shape for number **ሁለት** (two) and **ሦስት** (three) respectively

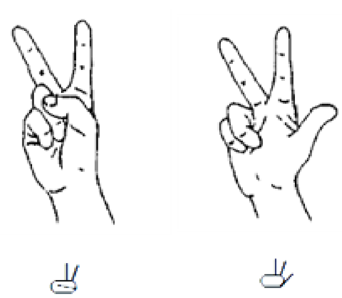


Figure 3.1: *Sign representation of number 2 and 3 [12]*

- **Hand orientation:** Orientation refers to which way the hands are facing. Does the palm face toward the signer or away? This could change the meaning of the sign. Changing the orientation of the hands can reverse the meaning of the sign. For example, changing the orientation of phrase "**እኔ እጠይቅሃለሁ** (I will ask you)" will be changed to the meaning "**አንተ ጠይቀኝ** (You ask me)" or "**እኔ እሰጥሃለሁ** (I will give you)" to "**አንተ ትሰጠኛለህ** (You will give me)". Figure 3.2 illustrate the meaning difference that comes by changing the hand orientation from **እኔ** (I) to **አንተ** (you) respectively.

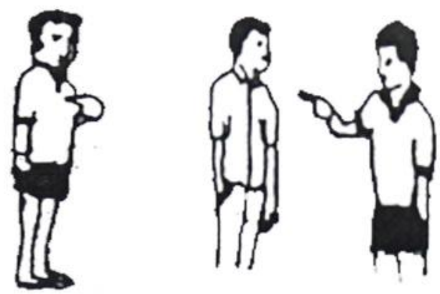


Figure 3.2: *Sign representation of "I"and "you" [12]*

- **Location:** The location or sign area relates to where the hands are held during signing. They can be against the head or other parts of the body, depending on what you are saying. The signing space is an imaginary rectangle, shoulder width, from head to just below waist. Some signs may go outside of this area, depending on what is being communicated, but the majority will be within this location. There are 12 possible locations for signing using the EthSL [12].
  Depending on what we need to sign the location will be changed. For example whenever you sign intellectual thing signing will be done around head. For example **አሰበ** (think) can be expressed as

Figure 3.3: *Sign representation of "think" [12]*

- **Movement:** Some signs also include a movement. Knowing the proper movement for what you are trying to sign is important in getting your thoughts across. Adding the wrong movement to a sign can change the meaning of the word or phrase you are using. The movement can be used to show the plural form (ዛፎች/Trees፣ ዓመታት/Years), vigor (በጣም ጥሩ/very good፣ በጣም መጥፎ/very bad), and extents (ትልቅ/big), in addition to indicating the type and direction of signing.
- **Expression (Non-Manual features):** Facial expression is included in the five parameters of sign to help communicate the feeling behind the words. A scowl, eye roll or happy expressions will let the person you are signing with know how you feel about what you are saying.

3.1.2. Characteristics of Sign Language

There are four major characteristics of sign language:

- **Simultaneity**: refers to the transfer of large idea at a particular time that is by using different signs idea or information, which considers being larger, can be transmitted.
- **Localization**: is explaining of ideas that can be compared and contrasted using sign language.
- **Movement**: is also one character, which is the nature or behavior of the hand movement.
- **Iconic:** Icons are symbols which share a physical resemblance to objects they represent. Because of the similarities between object and icon, the symbols usually are interpreted with little difficulty. Sign language is iconic or picture-like. Because icons are in most instances visible symbols, it is understandable that observers may relate shape and symbol when they study sign language.

3.2. Late-Deaf and Born-Deaf

People who are born hearing and become deaf late in life, are "physically deaf", but "culturally hearing". They grew up speaking a spoken language, using different communication tools. They think, speak, read, write and base their opinions on the world they knew before they became deaf [6].

People who are born into the deaf community, and whose first native language is a signed language, not a spoken one, are "culturally deaf". Some of them are born-deaf or became deaf at a very young age. Some of them are hearing people born into all-deaf families, and even though they can hear, even though they speak a spoken language, their first language was a signed language, not a spoken language. They base their view of the world from the deaf perspective. They are "physically hearing" but "culturally deaf" [6].

Usually hearing people are afraid of becoming blind. But to them, deafness doesn't seem nearly as bad. They forget that hearing is connected with the development of speech.

Hearing aids do not work for all deaf people and even when hearing aids do work, they work with minimal success for the profoundly deaf, sounds are muffled. It is very hard to distinguish between voices and other sounds. Oftentimes a deaf person can only hear bells or telephones with a hearing aid but cannot hear voices distinctly.

Deafness in deaf families is often genetically based. There are several genes that produce deafness. In deaf families, most often, everyone uses a signed language. They are "native signers", since it is their first language. Later they learn spoken language as their second language.

When a deaf person is born into a deaf family, they communicate in the same language of their parents. Their language development, using sign language from the moment of birth, is as normal as any hearing child's language development in spoken language. Everyone in the family uses the same language and the child starts signing at the expected age. They began absorbing language as babies.

Deaf children born to hearing parents are not always so lucky. Often the hearing parents do not realize the child is deaf until age three when they realize the child does not speak. The first three years are the crucial years for language development, so under those circumstances the child is deprived of "normal language development". Every member of the family is frustrated and communication is often poor at best.

3.3. Signed Amharic

Signed Amharic is a sign language dialect which matches each spoken Amharic word. All aspects of Amharic are signed by following Amharic grammar. People generally speak at the same time when using Signed Amharic to give deaf children the benefit of both signed and spoken Amharic. Example, Amharic phrase "አበበ ወደ ቤት ሄደ (Abebe went home)" can be signed as direct word by word "አበበ + ወደ +ቤት + ሄደ (Abebe + went + home)".

3.4. Signing Hand

Sign language is a body language which uses our body parts to convey meanings. It combines hand shapes, orientation, movement of the hand (optional), and facial expression (needs to express feeling). The main component in signing is shape which is constructed by changing the shape of our hands either using both or one hands.

The two hands can be classified as dominant and non-dominant hand. If you are right-handed, your right hand is your dominant hand. If you are left-handed, your left hand is your dominant hand. If you are ambidextrous, choose one hand to use as your dominant hand, and stick with it.

There are three types of signs when it comes to what hand you will use:

- One-handed signs: Uses only your dominant hand and you signing process will be done using a single hand. For example አንገት (neck)፣ ውሸት(lie)፣ መሳቅ(Smile) are a one-handed signs, diagram 3.4 shows their sign representation respectively.

Figure 3.4: *Sign representation of "neck, lie, and smile" [12]*

- Two-handed symmetrical signs: Uses both your dominant and non-dominant hands where they both move the same way. For example, አስተማሪ (teacher)፣ ኮት (coat)፣ ብዙ (many) are two-handed symmetrical signs and are illustrated as follows respectively:

Figure 3.5: *Sign representation of "teacher, coat, and many" [12]*

- Two-handed non-symmetrical signs: Uses both your dominant and non-dominant hand where the dominant hand moves while the non-dominant hand remains stationary. For example ሩቅ (far)፣ ባዶ (empty) are two-handed non-symmetrical signs and are signed as follows respectively:

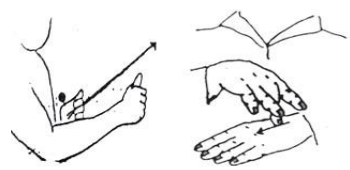


Figure 3.6: *Sign representation of "far and empty" [12]*

3.5. EthSL Grammar

3.5.1. Nouns and Plural forms

Nouns are common concepts to all languages. Like American Sign Language, EthSL does not alter the form of nouns to express plurality (for example: a 'noun' denotes a single thing and 'nouns', denotes more than one thing). In EthSL the plurality can be expressed by repeating the noun or using additional sign ብዙ (Bizu) (many).

Table 3.1: *Example for noun and plural forms*

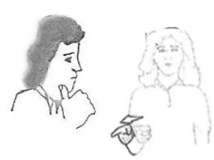
Noun	Singular form (Amharic)	Plural form (EthSL)
ውሻ (dog)	ውሾች (dogs)	ውሻ + ብዙ (dog + many)
ሰው (person)	ሰዎች (persons)	ሰው + ብዙ (person + many)
መፅሃፍ (book)	መፅሃፍት (books)	መፅሀፍ + ብዙ (book + many)
ዛፍ (tree)	ዛፎች (trees)	ዛፍ ዛፍ ዛፍ(መደጋገም) (tree tree tree …)

When the noun is unknown noun or does not have any sign that illustrates it, signing process will be done using finger spelling.

Pronouns and Indexing

Indexing is when you set up a point to refer to a person or object that is or is not present in the signing area. The following tables show how to sign pronouns in EthSL.

Table 3.2: *Pronouns and indexing*

Pronouns	**EthSL**
እኔ (I)	Pointing to myself 
አንተ (You)	Pointing to you 
እሱ (He)	Pointing to the person 
እሷ (She)	Pointing to the person 
እነሱ (They)	Sign half circle by pointing to them 
እኛ (Our)	Sign arc by pointing my chest starting from the right to the left or vice versa 

Suppose you want to talk to someone about a person who is not physically nearby, you should use contrastive structure. The rules of contrastive structure are easy. First, identify the person by finger spelling his or her name; describing a few key features such as hair color or height also help. Then, you can index the person or object to a point in space. Once you have set up this referent, you can refer back to that same point every time you want to talk about that person or object.

3.5.1.1. Possessive Noun

Signing possessive nouns (የእኔ/ mine፣ የአንተ/ yours፣ ያንቺ/ yours፣ የኛ/ ours፣ የነሱ/ them) is similar to signing pronouns except it uses our palm instead of using index finger.

Table 3.3: *Possessive noun*

Possessive noun	EthSL
የኔ (mine)	
የአንተ/ያንቺ (Yours)	
የኛ (Ours)	
የነሱ (them)	

3.5.2. Verbs

Verbs are the other common concepts in all languages. They have traditionally been defined as words that show action or state of being. In fact, without verbs, language would cease to exist. Verbs in EthSL come in two types: plain, and inflecting.

- **Plain Verbs**

A plain verb is a normal verb in EthSL. When using plain verbs the signer must designate the subject and the object and each plain verb must be signed using a single signing. Examples መሮጥ (run)፣ መዝፈን(Sing)፣ መጫወት(play)are plain verbs which do not have any indication of object and/or subject in a verb and they only show the action.

- **Inflecting/Indicating Verbs**

Inflecting/Indicating verbs incorporate the subject and object into the verb. When the signer sign inflecting verb, he/she sign the verb with the subject and/or object. Examples of inflecting verb in EthSL are:

Table 3.4: *Inflecting/Indicating verbs*

Amharic	**EthSL**
ሰጠ(he gave)	እሱ + መስጠት (He + gave)
ላከ(sent)	እሱ + መላክ (He + sent)
ከፈለች(she paid)	እሷ + መክፈል(she + paid)

3.5.3. Adjectives and Adverbs

Descriptive words are adjectives and adverbs. They modify nouns and verbs in detail. They also add imagery to our writing, speech, and signing.

Typically, EthSL puts an adjective after the noun it modifies, but one may place the adjective before the noun for stylistic purposes. Example, an Amharic phrase ቀይ ውሻ (red dog) is signed as ውሻ + ቀይ (dog red).

In Amharic the adverb is placed before the verb, whereas in EthSL it is placed after the verb. Most of the time, the placement of adverbs are similar with the placement of adjective in EthSL. Table 3.5 shows the placement of Adverbs in EthSL.

Table 3.5: *Adjectives and adverbs*

Amharic	EthSL
በፍጥነት ነዳው(he drive speedily)	(እሱ) + መንዳት + ፍጥነት / he + drive + speedily
ቶሎ ሄድኩ (I went quickly)	(እኔ) + መሄድ + ቶሎ/ I went quickly

3.5.4. Conjunction

The combining of two sentences in EthSL is different based on the conjunction needed. For example, the concept of the word "and" does not exist. Simply, sign a sentence, take a short pause and then sign the next sentence. Similar conjunctions such as "or" and "but" have signs. Example: An Amharic sentence እኔ እና አንተ አብረን ምሳ በላን (you and me ate lunch together) signed as እኔ+አንተ+አብሮ+መብላት+ምሳ (I + you + together + ate + lunch)

3.5.5. Interjections

Interjections are words used to express strong feelings or sudden emotion. They are included in a sentence usually at the start to express a sentiment such as surprise, disgust, joy, excitement or enthusiasm. Most of the time EthSL uses facial expression to express interjections but sometimes a few interjections can be expressed using signs. For example, "በየሱስስም (be 'iyesus sem)፣ በስመአብ (be seme'ab)፣ ይገርማል (yigermal)" have their sign equivalents.

3.5.6. Prepositions

All most all prepositions are not used in EthSL, because it is reserved more for signing exact Amharic. It is a good idea to avoid prepositions when signing in EthSL, because they are shown in context. For example: ከቤት መጣ (ke bEt meTa) is signed as ቤት መምጣት (bEt memeTat), and the preposition ከ (ke) is shown in the context, not signed as a word, but some preposition are expected to be signed for example in the phrase ወደ ቤት ሄደ (wed bEt hEd), the preposition wed indicating the direction of movement is expected to be signed as ወደ ቤት መሄድ (wed bEt mehEd).

3.5.7. Negation

The role of negation in EthSL is a fairly easy concept to grasp. There are two ways to sign negate in a sentence. The non-manual marker for a negated sentence is simply a shake of the head and it possible to sign "not" using hand. When signing the word not; one must remember that in EthSL syntax negation words always come at the end. Table 3.6 illustrates the negation of the sentences:

Table 3.6: *Negation*

Amharic	**EthSL**
አልመታሁትም('almetahutm)	እኔ + እሱ + መምታት + አይደለም('nE + memtat + 'aydelem)
አልበላሁም('albelahum)	እኔ + መብላት + አይደለም('nE + meblat + 'aydelem)
አልተኛሁም('alteNahum)	እኔ + መተኛት + አይደለም('nE + meteNe'at + 'aydelem)

3.5.8. Placement of Time Words

Time words are the only things that come before the sentence in EthSL, but sometimes they come in the middle when the subject is unknown. Example, ጠዋት 2 ሠዓት ላይ ላገኝህ(Tewat 2 se'at lay lageNh)can be translated as ጠዋት+ሠዓት+2 እኔ+አንተ+መገናኘት+መፈለግ(Tewat + s'at + 2 + 'nE + 'ante + megenaNet + mefeleg).

3.5.9. Word Order of Questions

In other natural languages such as Amharic and English, statements are given in a particular word order. EthSL does not invert its word order nor does it add in any helping words, it uses non-manual signals to display a question asked. These non-manuals can consist of body movements, facial expressions, or eyebrow movements. Let's examine a simple YES/NO question in EthSL and Amharic.

Amharic: ቁርስክን በላክ?

EthSL: አንተ መብላት ቁርስ?

In a YES/NO question the eyebrows are raised and the body leaned forward slightly. These non-manuals show the receptive signer that the statement is actually a question.

Another type of question is a WH-question. These types of question require more of a response than yes or no. They always include signs like WHAT, WHERE, WHEN, HOW, or WHY? These WH-words always come at the end of the question, unlike in Amharic where it is the first or middle word in the question.

Amharic: መቼ ትመጣለክ?

EthSL: መምጣት መቼ?

Similar to a YES/NO question, WH-word questions also have non-manual markers, however this time instead of raising your eyebrows, you must lower your eyebrows. In addition to lowering your eyebrows the signer must lean the body in slightly and extend the last sign for a couple seconds. This allows the receptive signer to understand they are being asked a question that requires more of a response.

The final type of question is called an RH-question. The use of an RH-question is like an Amharic speaker using the word because (ምክንያቱም). There is no sign for the word because in EthSL, therefore they sign a question and answer it themselves. The non-manual markers for an RH-question are the same as a YES/NO question.

Amharic: ቁርሴን አልበላሁም ምክኒያቱም ስላራበኝ

ASL: እኔ ቁርስ መብላት አይደለም ለምን? መራብ አይደለም

3.5.10. General Syntax

With background on how parts of speech are used in EthSL, we can now evaluate the syntax, or word order, of EthSL. The article, the word order of EthSL is different from that of Amharic. Amharic follows a SOV, subject-object-verb sentence pattern, whereas EthSL uses a topic-comment pattern and the subject is topic of the sentence.

Amharic: አበበ በሶ በላ

EthSL: አበበ መብላት በሶ

3.6. Manual Alphabet and Numbers

Finger spelling is used for proper nouns. They may include, but are not limited to movie titles, books, names, and street names. The EthSL has 33 letters and each has 7 sub-types which are identified using movement.

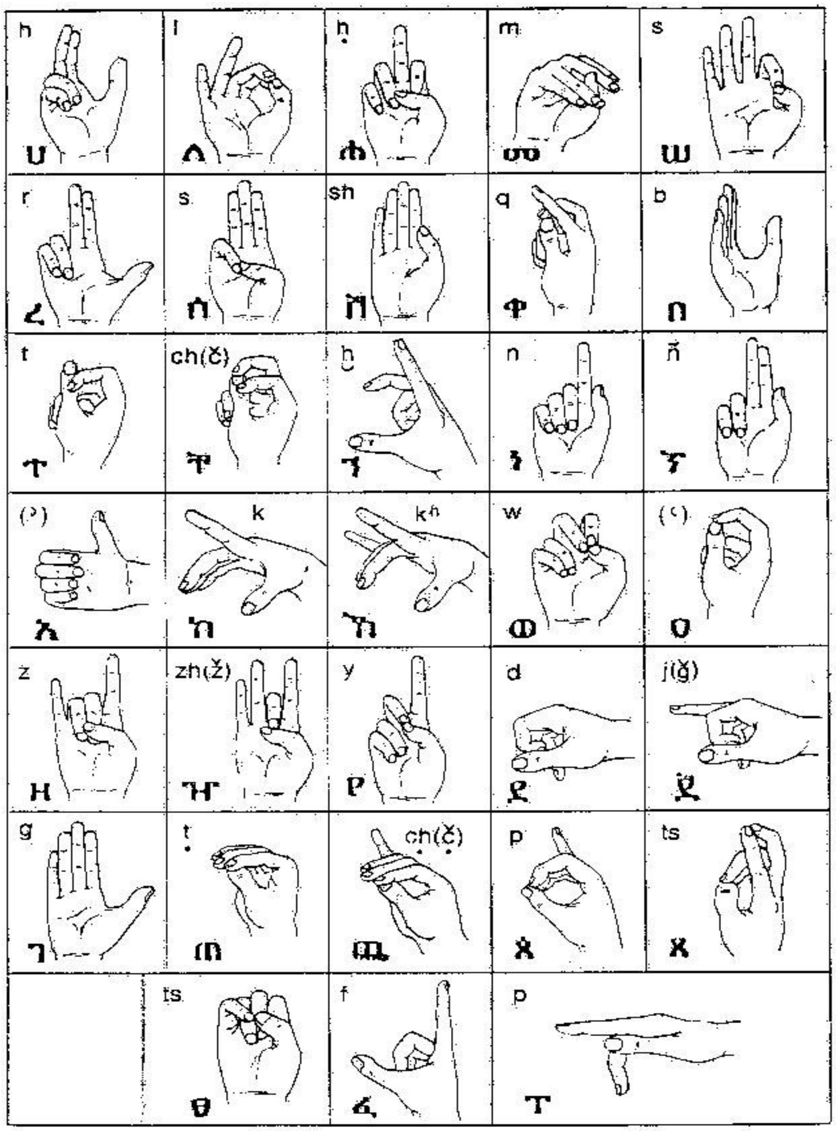


Figure 3.7: *EthSL finger spelling representation [12]*

Signing numbers is different from the normal signing that most Amharic speaker's use. All numbers below one thousand are signed using a single hand, and the second hand is only used to designate that a number is in the thousands or millions.

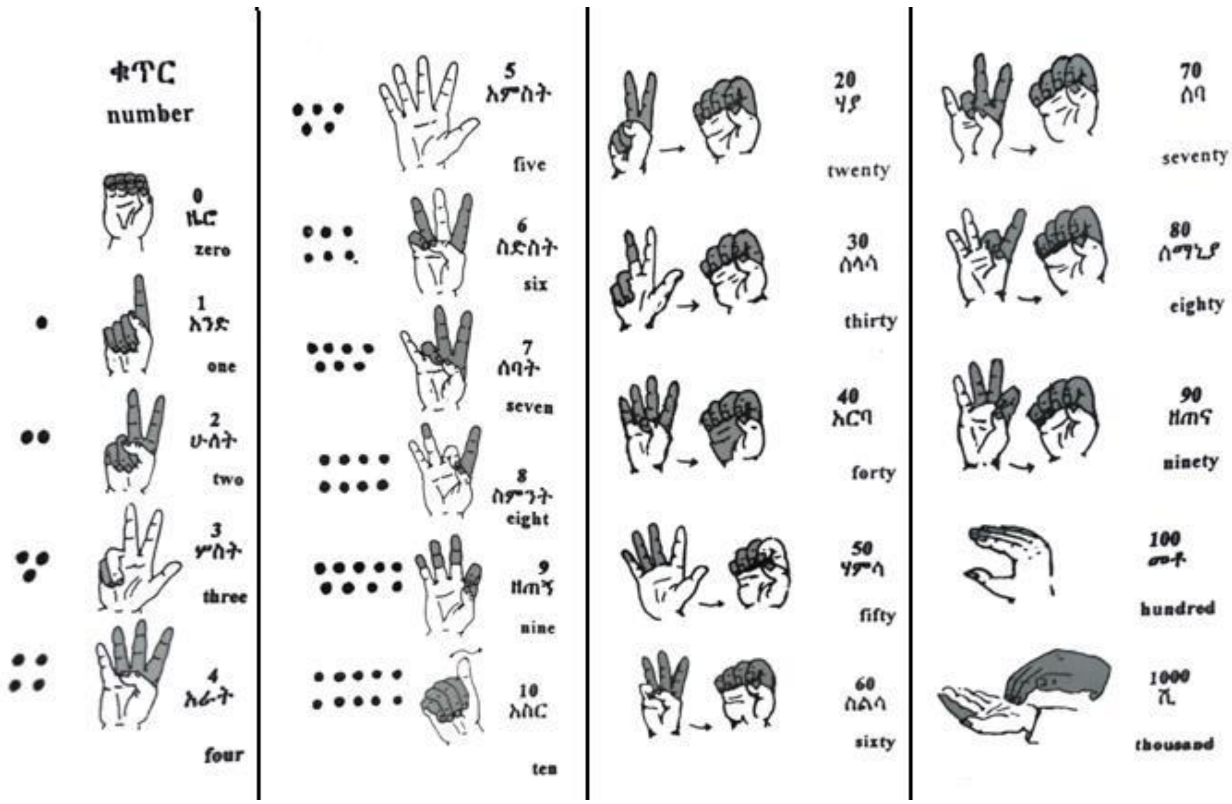


Figure 3.8: *Numbers in EthSL [12]*

CHAPTER FOUR

DESIGN OF AMHARIC SENTENCE TO ETHIOPIAN SIGN LANGUAGE TRANSLATOR

The Amharic sentence to EthSL translator is a prototype system which translate Amharic sentence to Ethiopian sign language. The main purpose of designing EthSL translator is to fill the communication gap between hearing and deaf communities as well as to develop linguistic self-confidence and security in the knowledge of deaf people's culture and forms basis for developing self-identity, active social participation and lifelong learning. The design expected to be user friendly, and the displayed animation needs to be realistic and also a correct translation.

The EthSL Translator has three main components:

- Interface component
- Back-end component
- Knowledge Database

All components play a role in the translation process, starting from accepting Amharic sentence to generating EthSL in 3D animation. Figure 4.1 illustrate the general architecture of the Amharic sentence to EthSL translator.

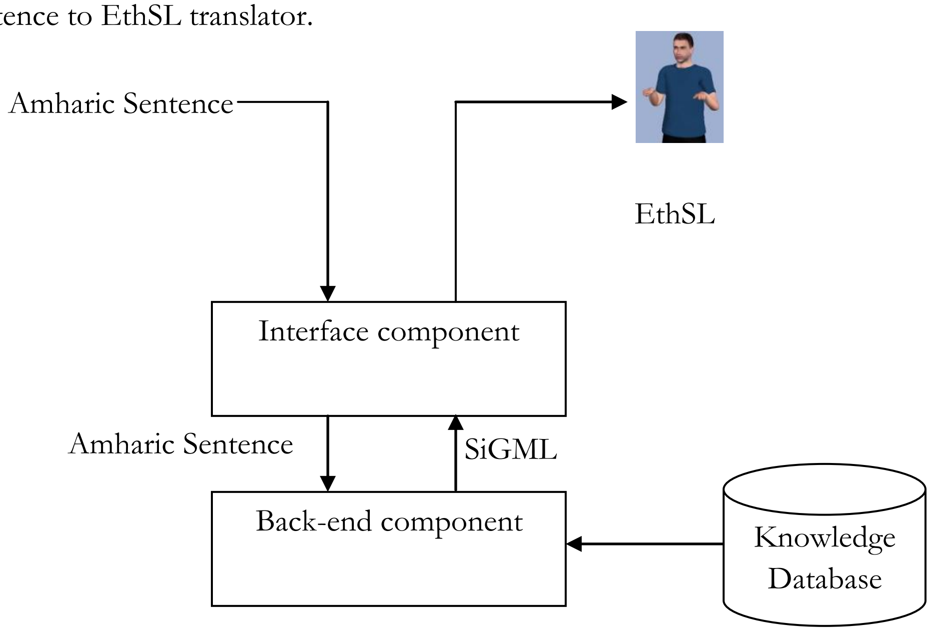


Figure 4.1: *General architecture of Amharic sentence to EthSL translator*

4.1. Interface Component

The Interface component responsible on accepting of Amharic sentence and pass the sentence to the back-end component as well as receiving SiGML file scripts from the back-end component and display the script (sign) in 3D avatar animation.

The interface components have two main parts; the web interface and the eSIGN 3D avatar. The web part acts as a bridge between the system and the users, which accepts Amharic sentence from the user and pass to the back-end component. And the eSIGN 3D avatar animation part translates the SiGML file scripts to the animated signed sequence which can easily be understood by deaf communities.

4.1.1. 3D Avatar

The gestures are represented by means of VGuido (the eSIGN 3D avatar) animations; it is plug-in for Internet Explorer 6 and above. An avatar animation consists of a temporal sequence of frames, each of which defines a static posture of the avatar at the appropriate moment. Each of these postures in turn can be defined by specifying the configuration of the avatar's skeleton, together possibly with some morphs which define additional distortions to be applied to the avatar.

In order to make an avatar sign or gesture, pre-specified animation sequences must be sent to the avatar. A signed animation is generated synthetically from an input script in the SiGML notation. The signing system constructs human-like motion from scripted descriptions of signing motions.

The eSign editor creates SiGML (signing gesture markup language) that a feed the avatar. SiGML is and XML application that enables transcription of sign language gestures [35]. SiGML builds on HamNoSys and indeed one variant of SiGML is essentially an encoding of HamNoSys manual features, accompanied by a representation of non-manual aspects.

Figure 4.2: *eSIGN 3D avatar*

4.1.2. HamNoSys

The Hamburg sign language notation system (HamNoSys) is a phonetic transcription system, which has been in widespread use since its original version; in the tradition of Stokoe-based systems, it was first developed in 1985 at Hamburg University, Germany, and it is currently in its fourth revision [35].

HamNoSys does not refer to national diversified finger spelling and therefore can be applied internationally. The writing symbols are available as a Unicode based character set for various operating systems and can be downloaded for free. The Hamburg sign language notation system is also a basis for a series of avatar controls [29].

Overall structure

The overall structure of HamNoSys is illustrated as follows:

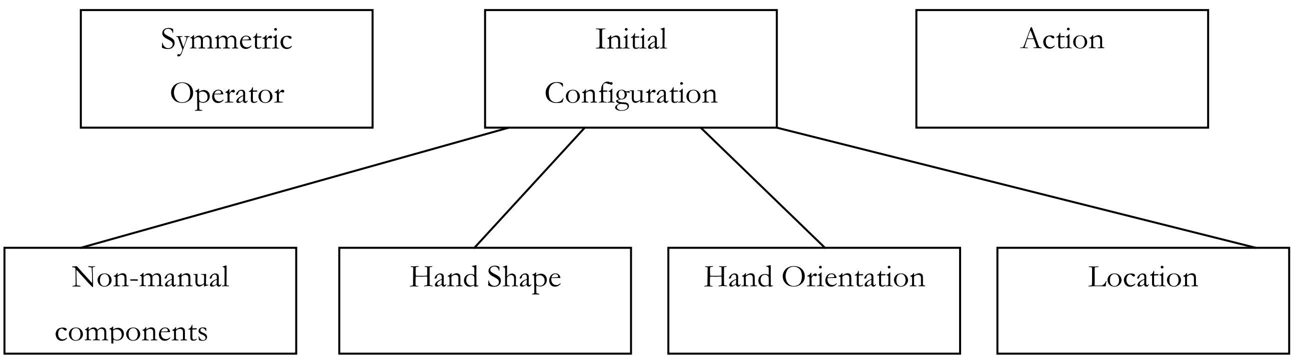


Figure 4.3: *Overall structures of HamNoSys [29]*

- **Symmetric Operator (.. or :)**

Symmetric operators are used for symmetrical two handed signs that have the same hand shape. Thus, it shows two hands involved in the signing. In the transcription of non-symmetric signs both hand shapes and their relation to each other must be describe [29].

- **Non-Manual components**

Non-manual markers consist of the various facial expressions, head tilting, shoulder raising, mouthing, and similar signals that we add to our hand signs to create meaning [29].

- **Hand shape**

Hand shape information is composed of 18 basic symbols representing hand-shape, thumb position, degree of finger extension, individual figure identification. The following diagram illustrates the basic open hand-shapes and thumb combinations [29].

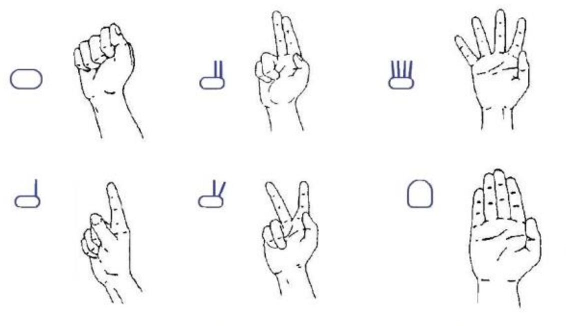


Figure 4.4: *Basic open hand-shapes [29]*

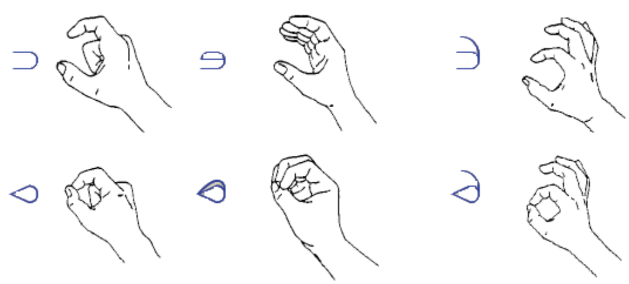


Figure 4.5: *Thumb combinations [29]*

- **Hand Orientation**

Orientation is the way the hand faces. Typically, the sign is performed facing outward away from the signer so that those reading the signs can have a better view. The most important points here are readability and comfort. Hand orientation is split in two parameters [29]:

- Extended finger direction: which define the orientation of the hand axis
- Palm orientation: which defines the palm orientation

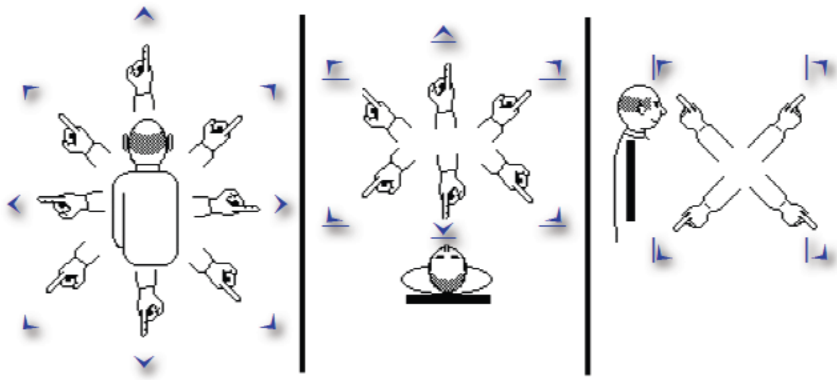


Figure 4.6: *Primary (extended finger) directions [29]*

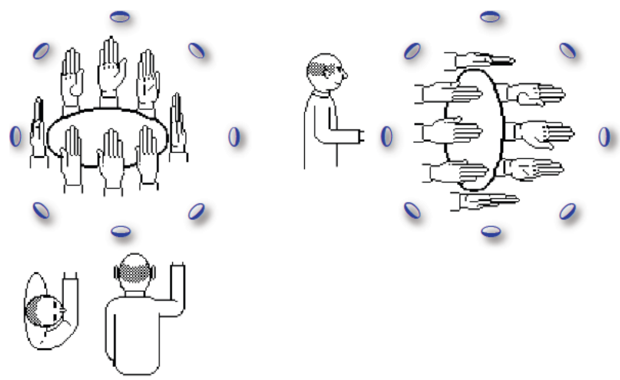


Figure 4.7: *Palm orientations [29]*

- **Location**

The location of the hand (near the forehead, near the chin, chest level, etc.) can totally change the meaning of a word. For instance, the same hand shape and motion are used for mother and father, but can alter the meaning of a sentence because of their location [29].

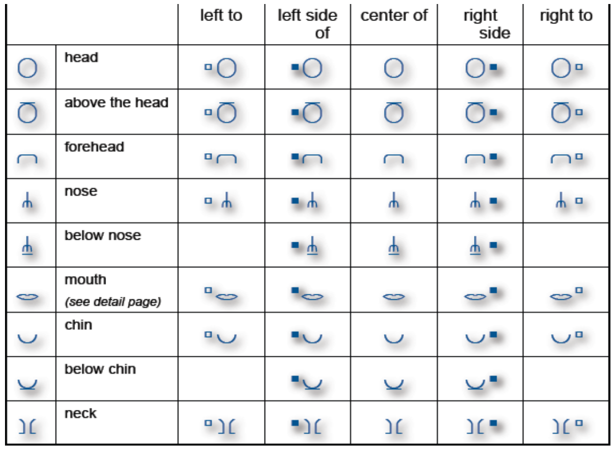


Figure 4.8: *Head location [29]*

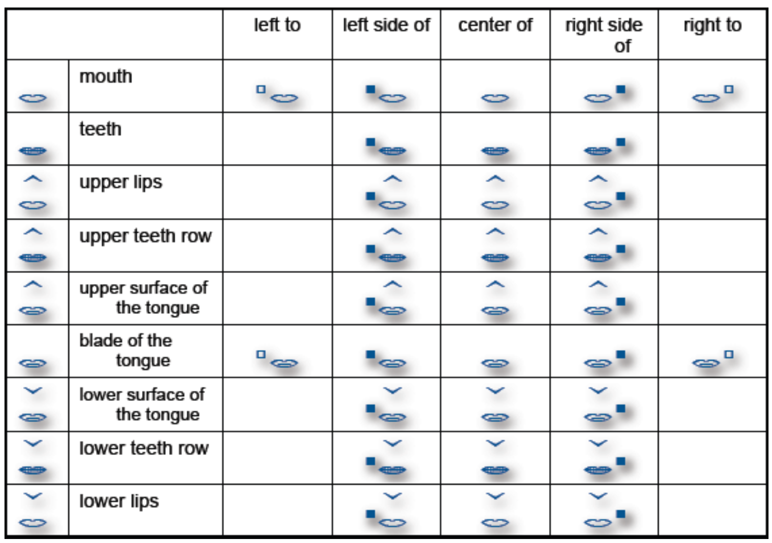


Figure 4.9: *Mouth locations [29]*

- **Action**

In sign language, movement refers to the distinctive hand actions which form words. It is a major component of sign language. There are different kinds of movements named; path movements (straight, curve, wavy/zigzag, and circular movements), in place movements (replacement of hand-shape and/or orientation, Finger play, wrist movements), and non-manual movements (head movements, etc.). The following diagrams illustrate some of those movements [29].

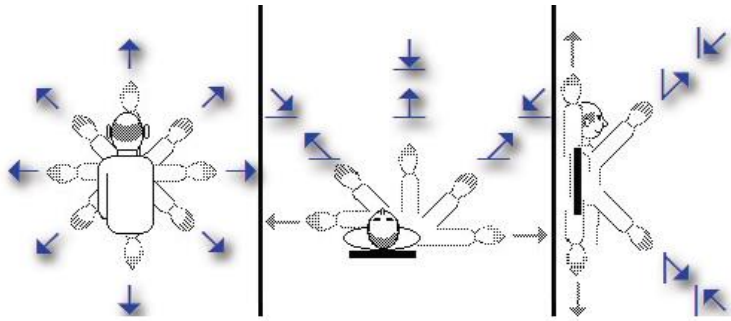


Figure 4.10: *Straight movements [29]*

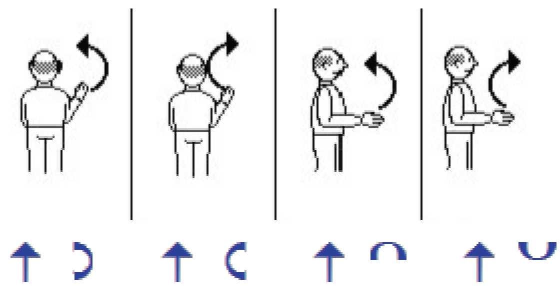


Figure 4.11: *Curved movements [29]*

4.2. Back-End Component

The back end component accepts Amharic sentence from the front end component and outputs SiGML files (sign script). Figure 4.2 illustrate the design of back-end component. The back-end component has three modules:

- Amharic text analysis
- Natural language processing (NLP)
- Text-to-sign mapping

All back-end process has been done here. The Amharic text analysis modules accept Amharic sentence from the interface component and generate Romanized Amharic sentence to the NLP module. Then the NLP module does all language processing and generate decorated sentence which is sentence in EthSL grammar. Finally the text-to-sign mapping maps each word with the SiGML and generates SiGML files.

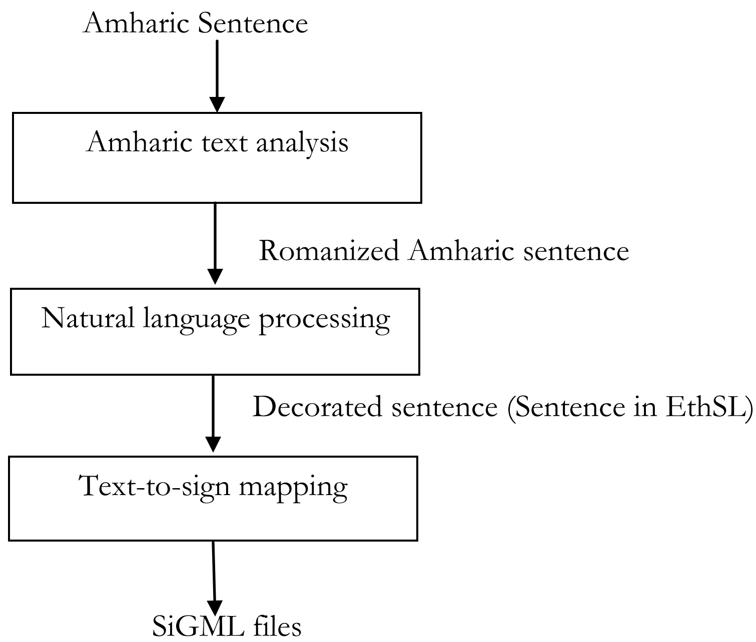


Figure 4.12: *Back-end component of Amharic sentence to EthSL translator*

4.2.1. Amharic Text Analysis Module

This module is responsible for analysis of Amharic sentence. It accept Amharic sentence and map to Latin words by splitting sentence into words. It checks each word in the in the knowledge database. If the words exist in the knowledge database the module (Amharic text analysis) reconstruct the sentence and pass to the NLP module. If the words not exist in the knowledge database it will terminate the process. Figure 4.3 illustrate the process of Amharic text analysis module.

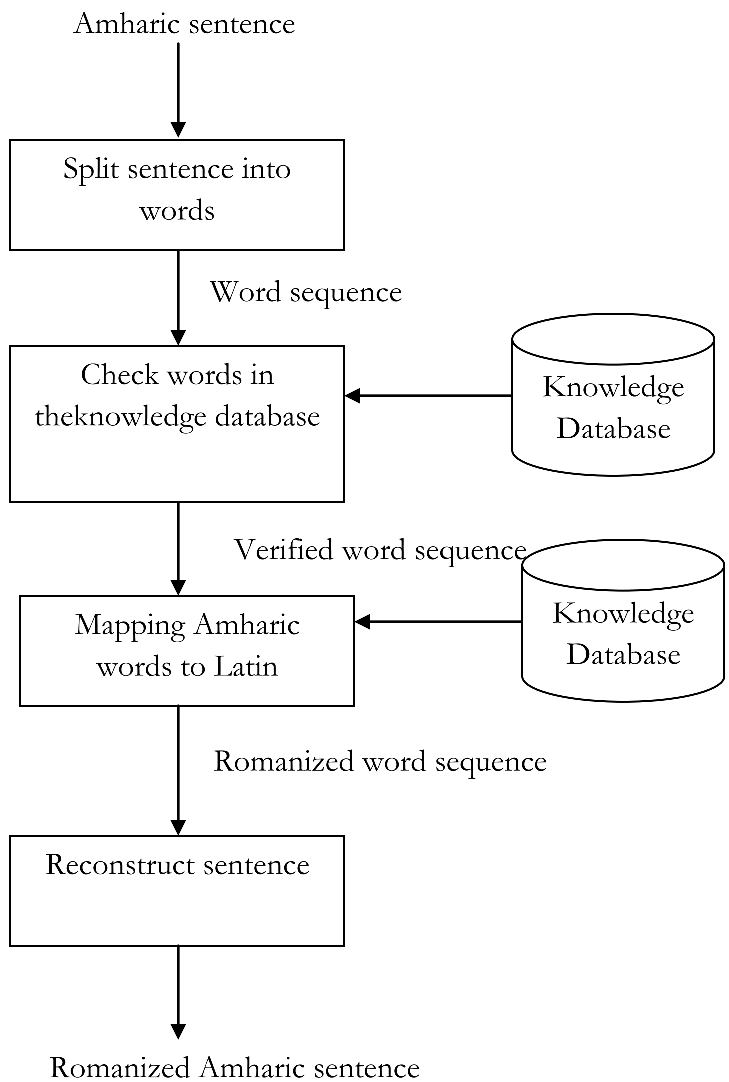


Figure 4.13: *Process of Amharic text analysis module*

4.2.2. NLP Module

The NLP module handles all natural language processing tasks. This module is responsible for processing both languages: Amharic and EthSL. Figure 4.4 illustrate the NLP module. This module has five sub-components which facilitate the task of NLP. The sub-components of the NLP module are:

1. POS tagger
2. Preposition removal
3. Romanizer
4. Morphological analyzer and
5. EthSL grammar translation

The POS tagger tags each word with their part of speech tag. For example, if the module accepts the Romanized sentence "be qonjo mekina hEde", it will generate tagged word sequences [(be, PREP), (qonjo, AD), (mekina, N), (hEde, V)] and return the result to the next sub-component, Preposition removal. Preposition removal component filters words which belong to preposition part of speech as they are not usable in EthSL and generate preposition free sentence. Therefore the previous tagged words will be converted to "qonjo mekina hEde" and return the result to Romanizer sub-components.

Romanizer is responsible to convert the given text (Romanized) to Amharic letter (Geez). For example, if it accepts the mapped (Romanized) text "qonjo mekina hEde" it converts it to "ቆንጆ መኪና ሄደ". And then by tokenize the sentence it passes each word to the morphological analyzer. Morphological analyzer generate morphological information which indicates information about gender, number, subject, negation, and root word and pass all morphological information and inputted Amharic sentence to EthSL grammar translation component.

The EthSL grammar translation component analyzes the grammatical rules of both languages and then translates the Amharic sentence to the equivalence EthSL sentence. For example the inputted words sequences "qonjo mekina hEde" (from the previous component) will be synthesized to '"su hEde mekina konjo" and return the result to the next module; text-to-sign mapping module. Then the text-to-sign mapping module check each words in the knowledge database and maps each word with their SiGML file and pass the SiGML files to the interface component.

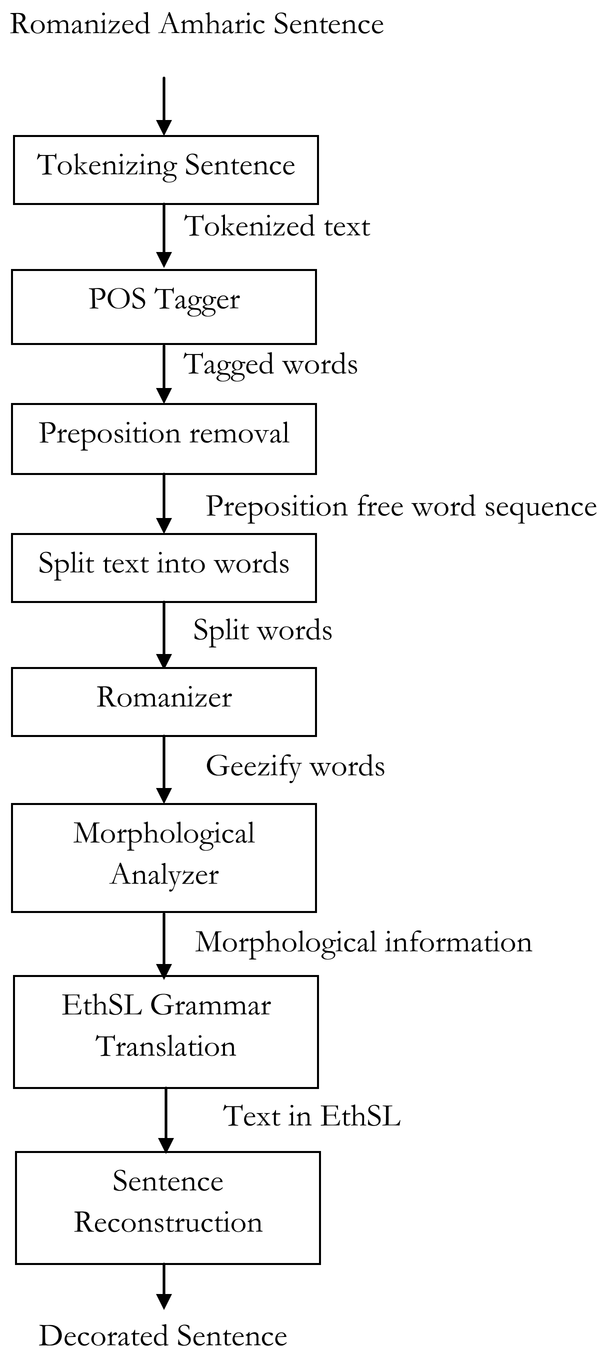


Figure 4.14: *NLP module*

- **Romanizer and Morphological Analyzer**

Romanizer (GeezSERA 1.0) and morphological analyzer (HornMorpho) are open source programs which were developed by Michael Gasser [22]. Even if the role of these two programs is different, they are interrelated with each other.

Before the morphological analyzer starts its task, the inputted Latin text must be Geezified. The role of the Romanizer is to translate the Latin text into Amharic phonetic. Then morphological analyzer can simply analyze the text and generate morphological information.

- **POS Tagger**

A Part-Of-Speech tagger (POS tagger) is a piece of software that read text in pre-specified language (such as Amharic) and assigns parts of speech to each word, such as noun, verb, adjective, etc. POS-tagger can be used in extraction of words of a specific word class, to decide which word class a word belongs to in a given position. Figure 4.2 illustrates the process of assigning a part-of-speech to each word in a sentence.

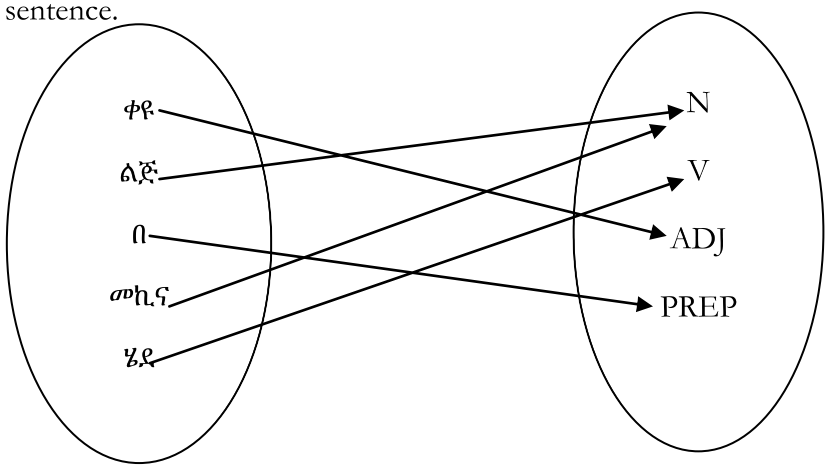


Figure 4.15: *Process of assigning a part-of-speech tagger*

We use a supervised POS tagging approach that requires a pre-tagged corpus which is used for training, to learn information about the tagset, word-tag frequencies, and rule sets. The performance of the model generally increases with the increase in size of this corpus. We use Amharic corpus that contains 8070 tagged Amharic sentences.

We combine two taggers to enhance the performance of tagger; which contains Brill tagging approach and naive Bayes classifier. Brill tagger can be summarized as an error-driven transformation-based tagger. Transformation-based in the sense that a tag is assigned to each word and changed using a set of predefined rules. If the word is known, it first assigns the most frequent tag, or if the word is unknown, it naively assigns the tag "noun" to it. Applying over and over these rules, changing the incorrect tags, a quite high accuracy is achieved. Diagram 4.6 shows how a transformation based learning.

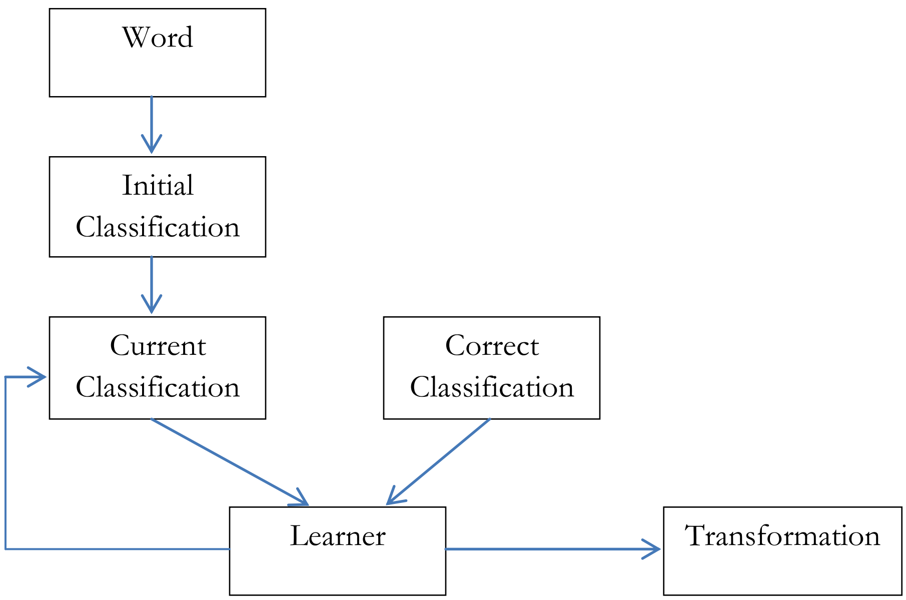


Figure 4.16: *Process of transformation based learning*

A simple way of dealing with unknown words is to assign them the most common part-of speech and then rely on the contextual rules to correct the errors. Brill suggests a special set of rules to apply only to the unknown words, generated in basically the same way as the contextual rules.

A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong independence assumptions. It assumes that the value of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. A naive Bayes classifier considers each of these features to contribute independently to the probability, regardless of the presence or absence of the other features [5].

As we mentioned above we combine the two approaches. To get a better result of tagging we select the tagger which has a better performance from previous work; naive Byes classifier which has been done by Biniyam Gebrekidan. And we develop a tagger which uses a Brill tagging approach and then by combining the two taggers we try to get a better performance. The tagging process conducts first using a Brill tagger, if the Brill tagger unable to tag the word, it pass untagged words to the next level tagger; naïve Bayes classifier. Then finally we get the best effort of the two taggers. Diagram 4.7 illustrates how the combined tagger works.

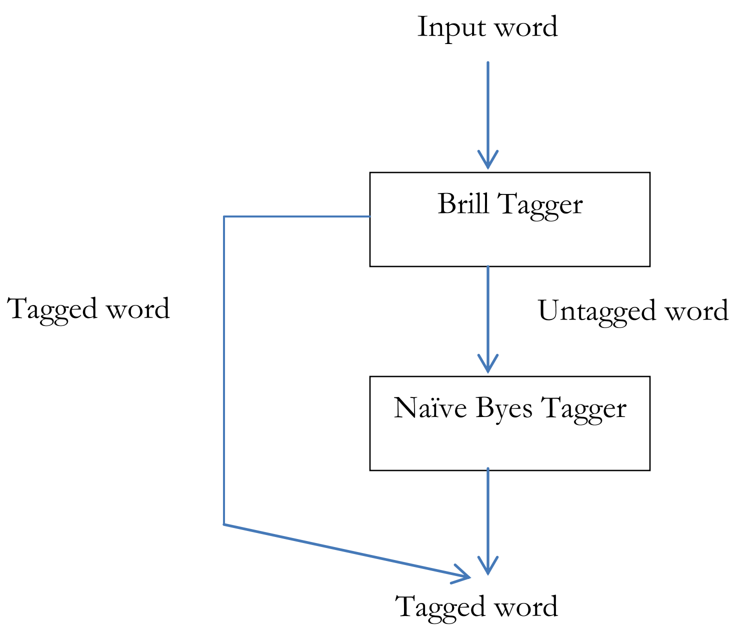


Figure 4.17: *Structure of POS tagger*

- **Preposition Removal**

One basic difference between Amharic and EthSL is part of speech tag usage. In Amharic we use different words that can be tagged in preposition tagset, such as ከ፣ በ፣ ለ. These words are not used in EthSL specifically among pre-lingual deaf except some specific words such as ወደ. The purpose of preposition removal is to avoid a preposition in a sentence.

- **EthSL Grammar Translation**

There is a grammatical difference between Amharic and EthSL. For more information about EthSL language grammar, refer to section 3.5. The role of this component is to translate the sentence which follows Amharic grammar structure to a sentence in EthSL grammar.

In Ethiopia sign language (EthSL) syntax is conveyed through word order and non-manual markers. EthSL sentences follow a topic-comment structure. However the topic of a sentence is the subject.

EthSL grammar translation component analyze the syntax of both language; Amharic and EthSL and performs translation based on pre-specified syntactical rules. Diagram 4.8 illustrates detail process of EthSL grammar translation.

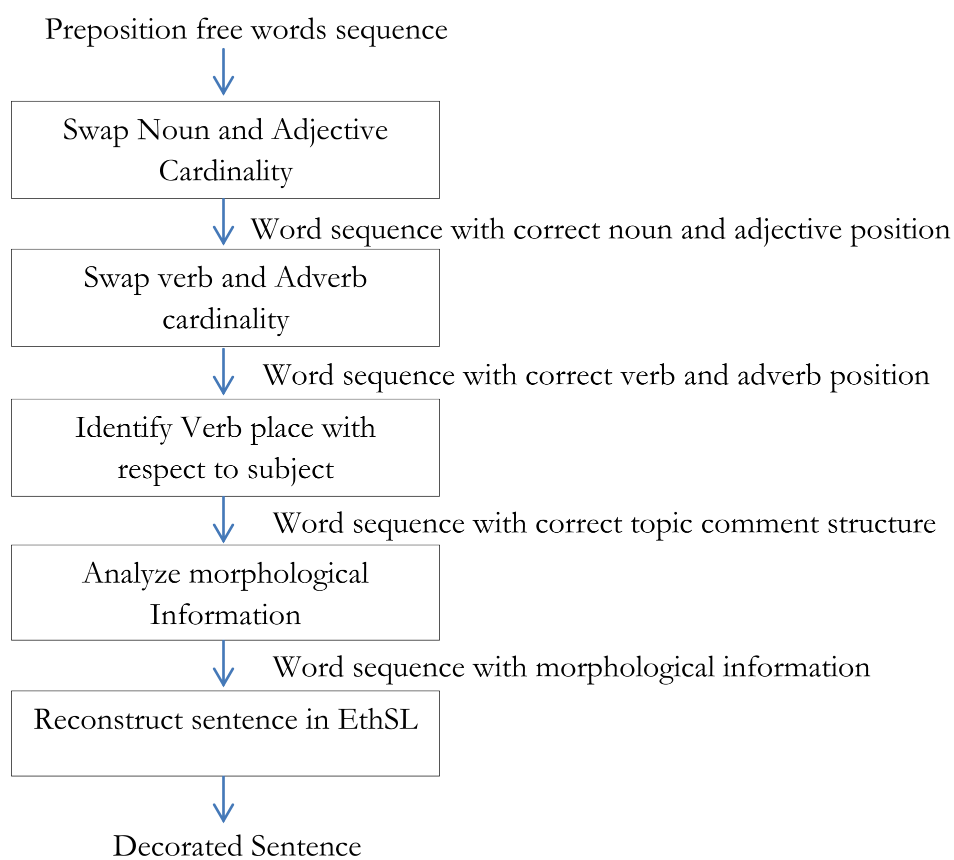


Figure 4.18: *The general design of grammar transformer*

4.2.3. Text-to-Sign Mapping Module

Text-to-sign mapping module responsible for mapping of text to SiGML file sign script. This module accept sentence in EthSL grammar and mapping each word with their SiGML file by splitting sentence into word sequence. Figure 4.9 shows how text-to-sign mapping module work.

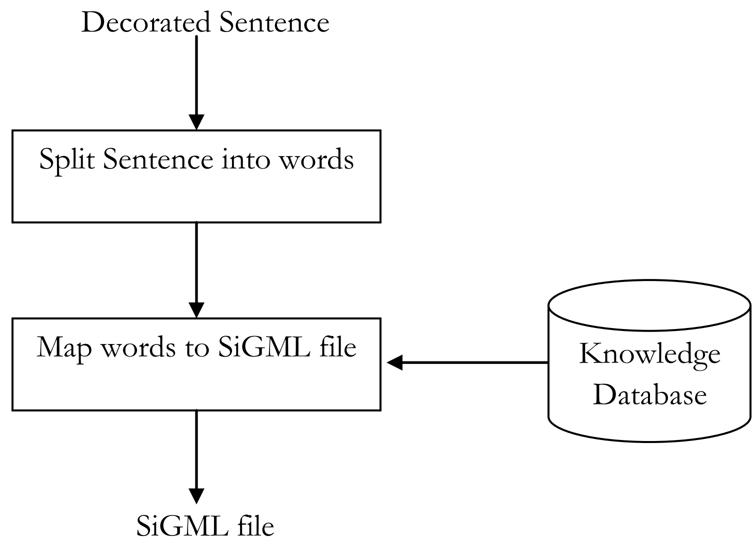


Figure 4.19: *Text-to-sign mapping module*

4.3. Knowledge Database

The final component of Amharic sentence to EthSL translator is a database component which holds information about Amharic words and their corresponding Latin words. In addition, it holds information about Amharic words and Ethiopian sign language representation in terms of SiGML file. Therefore, the back-end-component and the text-to-sign mapping module fetch all required information, to accomplish their task, is from the EthSL knowledge database. All information is stored in MySQL database except the numbers which stored in text files.

The knowledge database has three tables; “amharicenglishmap”, “dictionary”, and “letter”. Each table holds information about their entities. Amharicenglishmap has two fields named; Amharic, English (Latin) Word. This table holds information about Amharic words and their corresponding Latin equivalent, and this information is used to map of the inputted Amharic text to English (Latin) word which could be an input to the NLP module.

Dictionary table has four fields, AmharicRoot, SigmFile, EthSLWord, and Descreption. It stores information about Amharic root words with their equivalent EthSL representation and words description. It is used to map a words sequence in the Ethiopian sign language to the Sign sequences, SiGML files, which can be displayed by eSIGN 3D avatar animation. Finally the letter table holds information about letters, letter orders, and their SiGML file.

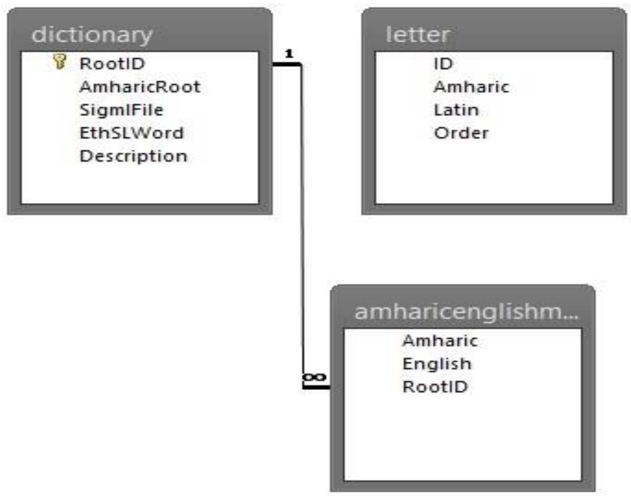


Figure 4.20: *Tables*

4.4. The General Description of Ethsl Translator

The EthSL Translator has different function. The entire function categorized into three main functions. The main function of the system is to synthesize Amharic sentence, Amharic letter (finger spelling), and numbers. In addition to translation, the system allows the user to insert new Amharic words with their correspondence Latin words and add new signs that do not currently exist in the database.

4.4.1. Amharic Sentence to EthSL Translator

The translator accept simple Amharic sentence and synthesizes it to EthSL. For more information about the simple Amharic sentence, refer to section 1.5. When the user insert simple Amharic sentence, the system first split the words sequence and maps each word with their corresponding Latin word.

After mapping each word with Latin word, it sends the words sequence to POS tagger, and the POS tagger tags each word and then passes the result to word filter. Then word filter removes words (prepositions) which are not supported by EthSL and return preposition free word sequence to Romanizer. Rominizer geezify each word and return to morphological analyzer. Morphological analyzer analyzes morphological information each word and generates morphological information and returns the result to EthSL grammar translation.

EthSL grammar translation transforms the Amharic word sequence to EthSL, based on the preset grammar rule. Finally the EthSL text is mapped with the equivalent SiGML files and passed to eSIGN 3D avatar animation which is responsible for animating the sign sequence (SiGML files).

4.4.2. Amharic Letter to EthSL Translator

The Amharic letter to EthSL translator accept Amharic letter and synthesize it to EthSL. When the user inserts Amharic letter, before any processing it maps the letter to Latin word. Next it checks the letter type and order from the database. Then it maps the SiGML file of the letter and its order.

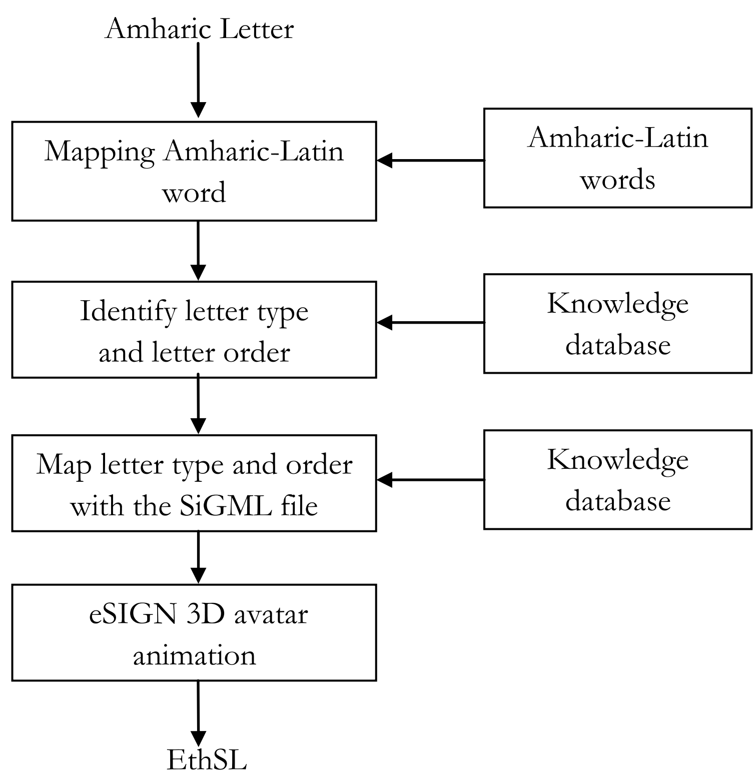


Figure 4.21: *Process sequence of Amharic letter to EthSL translator*

4.4.3. Number to EthSL Translator

The numbers can be categorized into two classes; basic and derived numbers. Basic numbers are numbers which have unique sign representation, and derived numbers are numbers which are derived from basic numbers represented by combining the basic numbers' sign. For example, number between 0 and 9, 100, and 1000 are basic numbers and they have their own sign representation; in contrast, 145 is a derived number which is derived from 100, 40, and 5.

The translator accepts a number and identify whether it is basic or derived. If the number is basic number it fetches the SiGML file from the file and passes the file to eSIGN 3D avatar animation. But if the number is a derived number, it expands to basic numbers and integrates the basic numbers SiGML files and passes the integrated file to eSIGN 3D avatar animation.

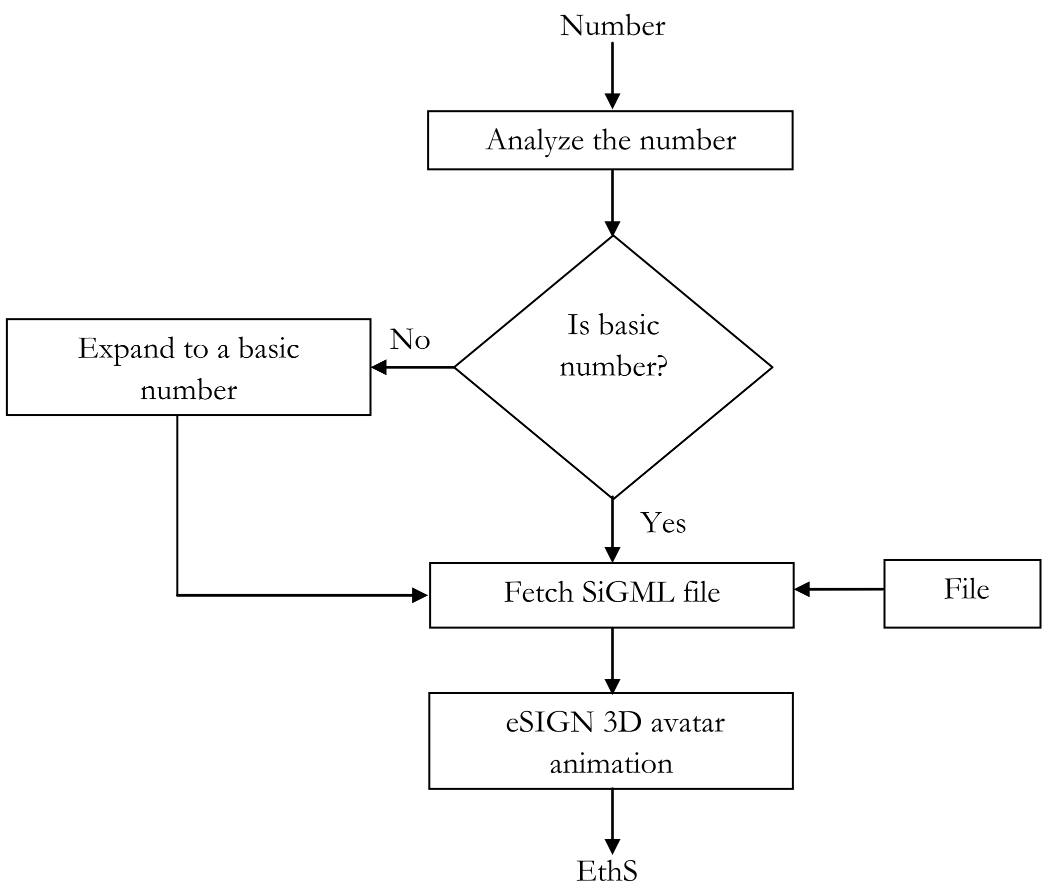


Figure 4.22: *Process sequence of number to EthSL translator*

CHAPTER FIVE

IMPLEMENTATION AND EXPERIMEN

This section discusses the implementation and testing of Amharic sentence to EthSL translator. Moreover, the development tools and prototype of the system are also detailed in this chapter.

5.1. Development Tools

The designed system is a web base application and to make the system complete different tools were applied such as scripting language, HTML editor, web server, database management tool, and eSIGN editor.

5.1.1. Scripting Language

For developing of the back-end component both python and PHP scripting language were used. For more information about back-end component, refer to section 4.2. Python program were used for the developing of NLP module, which is a sub-component of back-end component. We develop three NLP tools (POS tagger, preposition removal, and EthSL grammar translation) using a python program and adopt two NLP tools (Rominizer and morphological analyzer) which already developed and available as open source. In addition we used PHP scrip for text analysis and text-to-sign mapping (sub-component of back-end component).

5.1.2. HTML Editor

For interface designing we use Dreamweaver 8 which is a proprietary web development tool developed by Macromedia. It is a web design and development application that provides a visual WYSIWYG (what you see is what you get) editor and a code editor with standard features

5.1.3. Web Server

For delivering the developed web application we use a WampServer2.0i. It is a Windows web development environment. It allows you to create web applications with Apache2.2, PHP and a MySQL database. Alongside, PhpMyAdmin allow managing easily our databases.

5.1.4. Database Management Tool

MySQL database is a web hosting database that be used to store persistent data of the system. It is the most popular type of relational database on the web today. This is partly because it is completely free but also very powerful.

5.1.5. eSIGN Editor

eSIGN (essential sign language information on government networks) was originally funded under the information society technologies (IST) program of the European Union's fifth framework and supported by the European eContent program [35]. eSIGN technology offers tools which can provide information in sign language, using virtual humans (avatars).

The signed material can be easily updated by anyone with sign language translation skills. So this method of providing signed information can be implemented economically. In some situations creating signed content can be partially automated. In such cases, site maintenance could be undertaken by people with no signing skills at all.

Although virtual signing can never be expected to replace human interpreters, or to significantly reduce demand for them, this technology can provide a readily available alternative, increasing access to information in situations where an interpreter would not usually be an option [35].

- **eSIGN content creation**

Content creation starts from the creation of a sign lexicon, signed sequences are constructed and then added to web browsers or tailored applications. The original text is entered into the eSIGN editor. The corresponding sign language translation is entered alongside the original text, gloss by gloss.

Glosses with their HamNoSys strings are taken directly from the lexicon database for the appropriate language. Due to the diversity of sign there is often more than one entry for a given gloss. In order to choose the correct sign, it is possible to either

- have a look at one or more video clips that can accompany each sign in the database;
- pick a sign and let the avatar sign it; or
- read the HamNoSys string that describes the manual components of the sign.

Signs that are not in the database are transcribed in HamNoSys and may be entered into the database. Mouth pictures can be retrieved from a pronunciation table in the database by simply writing the mouthing in standard orthography.

It is possible to alter signs that are retrieved from the database. For example it may be necessary to change the HamNoSys string so that a sign performance suits the given context. In some instances, mouthing instructions can be edited, or added if they were not stored alongside the sign in the database.

In the eSIGN editor it is also possible to add any necessary facial expressions, body movements and pauses, or to alter the position or location of a sign.

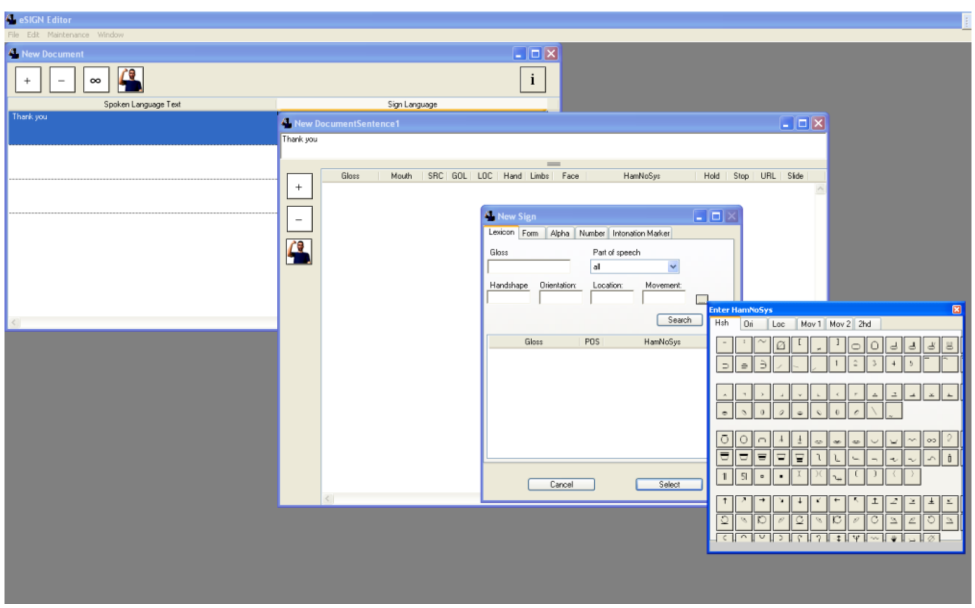


Figure 5.1: *eSIGN editor*

The eSign editor creates SiGML (signing gesture markup language) that feeds the avatar. Example: eSIGN editor creates SiGML XML language for the HamNoSys representation of car (መኪና).

(HamNoSys representation)

```
<sigml>
        <hns_sign gloss="$car">
                <hamnosys_nonmanual>
                </hamnosys_nonmanual>
                <hamnosys_manual>
                        <hamfist/>
                        <hamfingerbendmod/>
                        <hamextfingeru/>
                        <hampalmd/>
                        <hamshoulders/>
                        <hammover/>
                        <hamarcu/>
                        <hamrepeatfromstart/>
                        <hamplus/>
                        <hamfist/>
                        <hamextfingeru/>
                        <hampalmd/>
                        <hamshoulders/>
                </hamnosys_manual>
        </hns_sign>
</sigml>
```

5.2. Natural Language Processing Tools

As we mention before we develop three NLP tools using a python program and the remaining two NLP sub-components adopted from HornMorpho.

5.2.1. POS Tagger

When implementing the POS tagger we use supervised POS tagging approaches. For more about supervised POS tagging approach, refer section 4.2.2. For training and testing the brill tagger we use an Amharic corpus that contains 8070 tagged Amharic sentences and 90% of the total sentence used for training the tagger and we generate 3000 useful rules. The remaining 10% used for testing the performance of tagger.

The performance of part of tagger has a great role on the quality of the translation. Therefore the tagging process conducted using two taggers; Brill tagger (our own work) and naive Byes classifier (previous work) in combination. The system first try to tag the word using Brill tagging approach, if it unable to tag the word it pass untagged words to the next level of tagger, naïve Byes classifier. Finally we got a better result from the combination of both taggers and the result will facilitate the translation process (refer to the Brill tagging code in appendix D).

5.2.2. Preposition Removal

The purpose of preposition removal is to avoid a preposition in a sentence. It accepts tagged Amharic word sequence as input and returns preposition free tagged Amharic word sequence (refer to the preposition removal code in appendix E). Before avoiding the words with preposition POS it always check whether the word is in preposition exception, if it is it leave as it is.

5.2.3. EthSL Grammar Translation

EthSL grammar translation component analyze the syntax of both language; Amharic and EthSL, and performs translation based on pre-specified syntactical rules. We develop a python program which processes the syntax of both languages. The program accepts preposition free tagged Amharic word sequence. Then it checks the existence of adjective in a sentence, if it exist it search noun which position on adjective +1 and interchange the position of nouns and adjective. After identifying the correct position of the noun and adjective the program check the existence of adverb in a sentence. If it exists it search verbs and put the adverb location to verb + 1.

Finally the program rearranges the sentence with topic comment structure which is the syntax of EthSL by identifying the correct location of verbs with respect to subject. In addition the program appends all morphological information to the sentence in the correct order with referring of EthSL syntax (refer to the EthSL grammar translation code in appendix F).

5.3. Prototype

As we describe in the previous section (section 4.4) the translator has three main functions:

- Amharic sentence to EthSL translator
- Amharic letters to EthSL translator
- Numbers to EthSL translator

5.3.1. Amharic Sentence to EthSL Translator

The translator accepts amharic sentence and translate to EthSL. For example if the user types "ቆንጆው ልጅ በ መኪና ሄደ" the translator splits each word and maps them to Latin word and reconstruct the sequence "qonjow lj be mekina hEde". And then it passes the word sequence to POS tagger and it returns the result [(qonjow, ADJ), (lj,N), (be, PREP), (mekina, N), (hEde, V)] to preposition removal. The Preposition removal avoids unnecessary words; therefore, as the preposition "be" is not used in EthSL, it removes the preposition and return the result [(qonjow, ADJ), (lj,N), (mekina, N), (hEde, V)] to Romanizer. The Romanizer Geezify each word and return to morphological analyzer. The morphological analyzer analyze subject, negation, number, gender, and root word information and send the word sequence to EthSL grammar translation.

EthSL grammar translation transforms the given sentence to EthSL (lj qonjow hEde mekina) as well as analyze and morphological information to the word sequence, e.g "**እሱ**" is integrated with the sentence and the result ('su lj qonjow hyd mekina) will be returned. Finally the each word will be mapped with their SiGML file and the file passes to eSIGN 3D avatar animation.

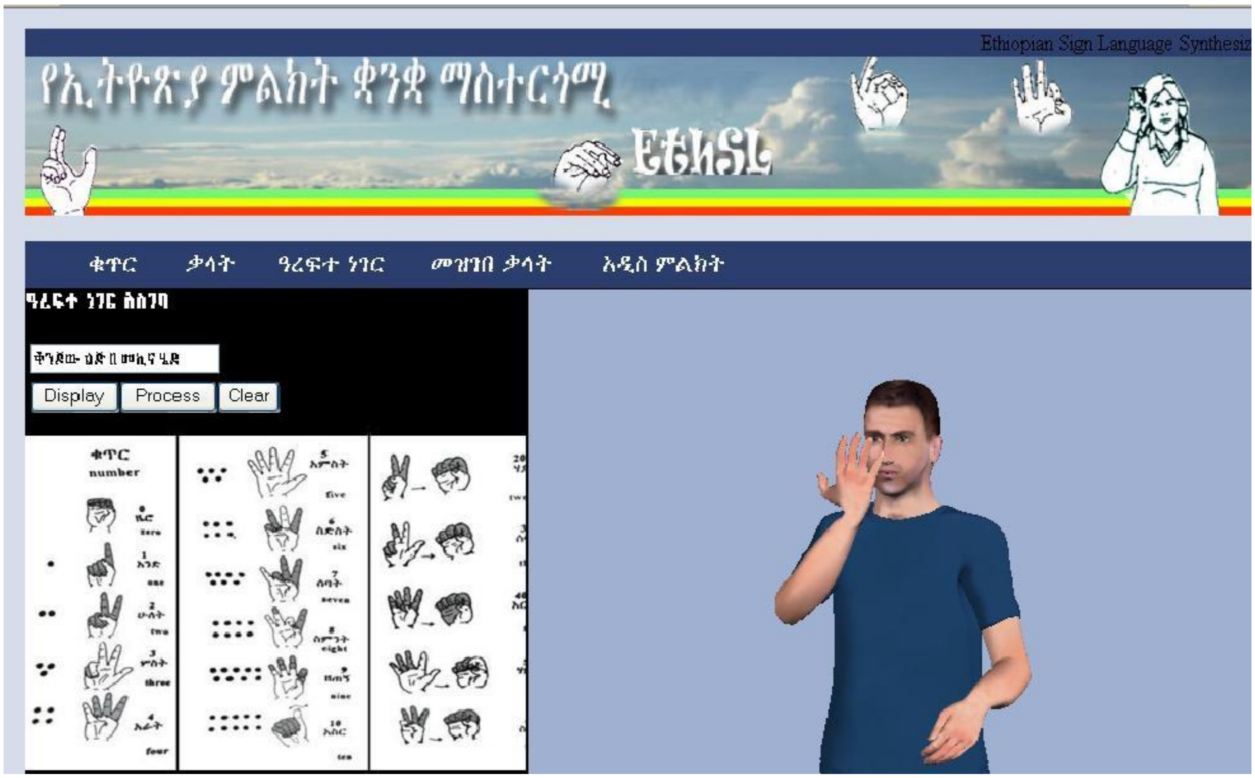


Figure 5.2: *Screen shot of Amharic sentence to EthSL translator*

Figure 5.3 illustrate the output of the above ('su lj qonjow hyd mekina) sentence.

Figure 5.3: *Output of Amharic sentence to EthSL translator*

5.3.2. Amharic Letter to EthSL Translator

The translator accept Amharic letter as input and generate EthSL. For example, if the user types the letter “ሉ”, it first maps the letter to a Latin word (le). Then it identifies the category of the letter that is “ለ” as well as the order of the letter that is “2nd”. Finally it maps the letter type and it orders with the corresponding SiGML file, and integrates both files and passes the file to the eSIGN 3D avatar animation.

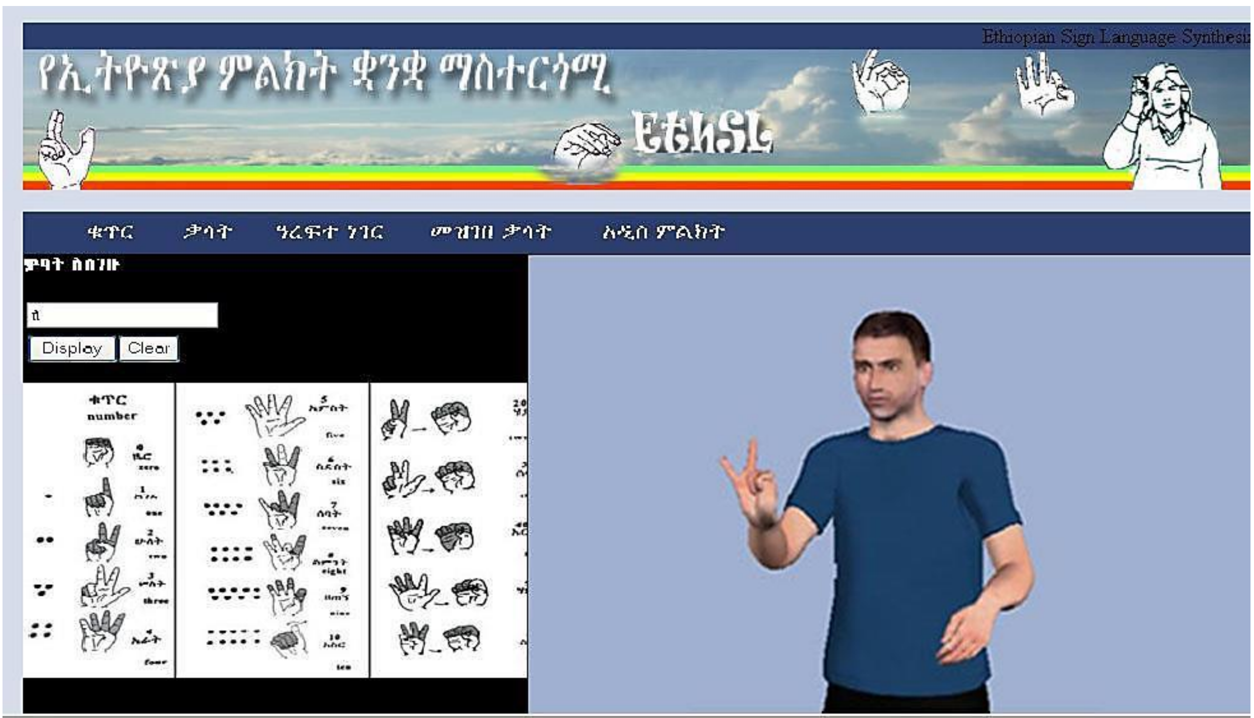


Figure 5.4: *Screen shot of Amharic letter to EthSL translator*

5.3.3. Number to EthSL Translator

The translator accept number as input and generate signed number as an output. For example if the user types “4”, the translator check the number type, the number is basic number. Therefore, the translator fetches the SiGML file from file and passes to eSIGN 3D avatar animation.

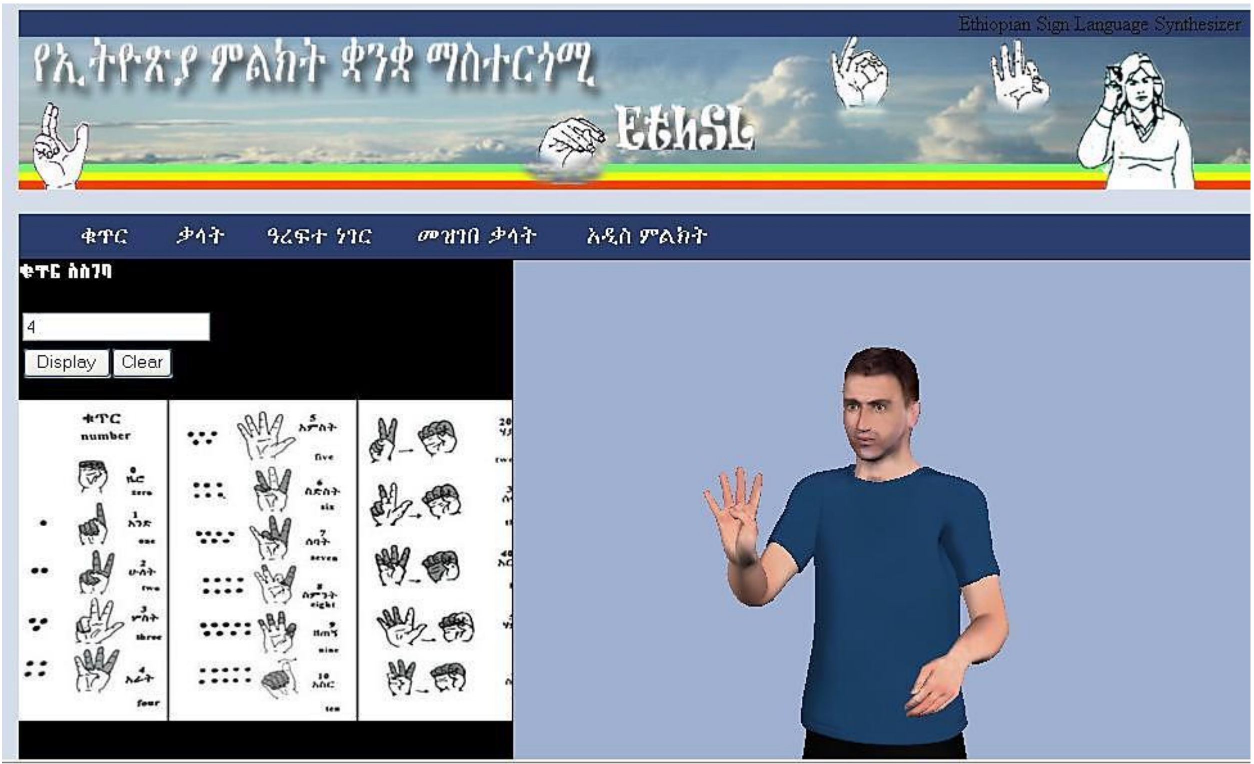


Figure 5.5: *Screen shot of number to EthSL translator*

5.3.4. Adding of New Words and Sign

In addition to translation, the system allows the user to add new words with their corresponding Latin words. This feature increases the performance of the system by increasing the amount of words in the database. In parallel with adding new word and its Latin equivalent, the system also provides a way to add new sign to the dictionary. When the user needs to add a new sign the user is expected to prepare the sign in HamNosys notation and convert to SiGML file using eSIGN editor. For more information about eSIGN editor, refer to section 5.1.5.

5.4. The Experimental Test

POS Tagger

As we describe before to enhance the quality of the translation we combines two POS tagging, which contains both Brill tagging and naïve Byes classifier approach. A tagger which uses a naïve Byes classifier is a previous work done by Biniyam Gebrekidan, and he got a performance result of 90.95%. It is a great achievement regards the previous works even if it is not enough to take the application as a component of the translation system. Therefore enhancing the performance with some degree is advantageous to get better result in the translation process. In spite of this, we combine the two methods that integrate the previous work (naïve Byes classifier) with our own work (Brill tagger). We trained the Brill tagger using 90% of 8070 Amharic tagged sentence and we use 3000 useful rules, the remaining 10% of the total corpus used to test the performance of the system and finally we got the performance of 91.84% accuracy level.

Due to the difference approach they use one word can be evaluated and tagged using the two method of tagging if the once unable to tag the word.

EthSL Translator

As we mentioned on the scope of the thesis the sentence is constructed with two phrases; noun and verb phrase. The noun phrase may have "**መስተዓምር፣ አጎላምሽ፣ መሙያ እና ስም**", and the verb phrase may contain "**መስተዓምር፣ አጎላምሽ፣ መሙያ እና ግስ**". Even if we get better performance on the translation of simple Amharic sentence, the system translate complex Amharic sentences with some extent, due to the time constraint and the complexity of the language we unable to study complex nature of the language (EthSL)

We measure the performance of the system according to the accuracy of the translation on each level; sentence, letter, and number. We calculate the system accuracy for all test data based on the following formula:

$$\text{System accuracy} = \frac{\text{number of correctly translated test data} \times 100\%}{\text{total number of test data}}$$

For testing the performance of the system at sentence level we took 5 prepositions and 12 words from each part of speech (noun, adjective, and verbs) and 5 words from adverbs. And we construct 114 simple Amharic sentences which were constructed from 46 randomly selected words and test the system accuracy at sentence level.

The system performances were not only measure in a sentence level but also at letter and number level. We took 33 Amharic letters and 50 randomly selected numbers to test the performance. The words, letters, and numbers with their corresponding signs are randomly selected from "Ethiopia sign language dictionary" which is the new EthSL dictionary. While testing, the sentences are ranked into three categories:

- Number of correctly translated sentences
- Number of understandable sentences
- Number of wrong translations

The system does not handle sentences which are complex, which has constructive words, tense, and sentences which are constructed with words that do not exist in the dictionary.

Correct output sentences without any errors are considered as correctly translated sentences. And output sentences which convey meaning but not clear sense due to sign representation and word context meaning are considered as understandable sentences. Output sentences which do not covey meaning as well as sense are considered as wrong translations.

We took the users judgments, people who engaged for testing, to identify the results were incorrect but it convey meaning or understandable.

For measuring the performance of the system two sign language translators, three voluntary sign language instructors, and 6 deaf students form Entoto poly Technique College were selected. We evaluate the system performance by feeding the randomly selected Amharic sentences to the system and let the translators or the deaf see the signed animation and let them write or explain what they understand from the displayed signed animation. When we evaluate the performance with the deaf student, we evaluate their understanding level by let them to tell what they understood to the translators and we collect data from the translators.

As we mention before we measure the system accuracy in three classes:

- System accuracy at sentence level
- System accuracy at letter level and
- System accuracy at number level

The way we use to measure the system accuracy for all categories was the same. First we were evaluate the performance of signed numbers by randomly selecting 50 numbers and evaluate the users understanding based on their respond from the displayed signed animation. Secondly we tested the Amharic letters and we select randomly 33 letters from each letter category and we evaluated the user's level of understanding. Finally we tested the performance of the system at sentence level depending on the pre-specified grammar rule. To test the Amharic sentence to EthSL translator we constructed 114 simple Amharic sentences from randomly selected words set and we check the performance by ranking the result in three categories.

Table 5.1: *Test result for number to EthSL*

Total numbers tested	50
Number of correctly translated numbers	42
Number of understandable numbers	8
Number of wrong translation	0
System accuracy (in %)	84%

Table 5.2: *Test result for Amharic letter to EthSL*

Number of letters tested	33
Number of correctly translated letters	25
Number of understandable letters	3
Number of wrong translation	5
System accuracy (in %)	75.76%

Table 5.3: *Test result for Amharic sentence to EthSL*

Number of sentences tested	114
Number of correctly translated sentences	67
Number of understandable sentences	29
Number of wrong translation	18
System accuracy (in %)	58.77%

As we described on the previous tables, we got 58.77% of system accuracy at sentence level, 75.76% of system accuracy at letter level, and 84% of system accuracy at number level translation and the overall performance of the system is illustrated graphically on Figure 4.19.

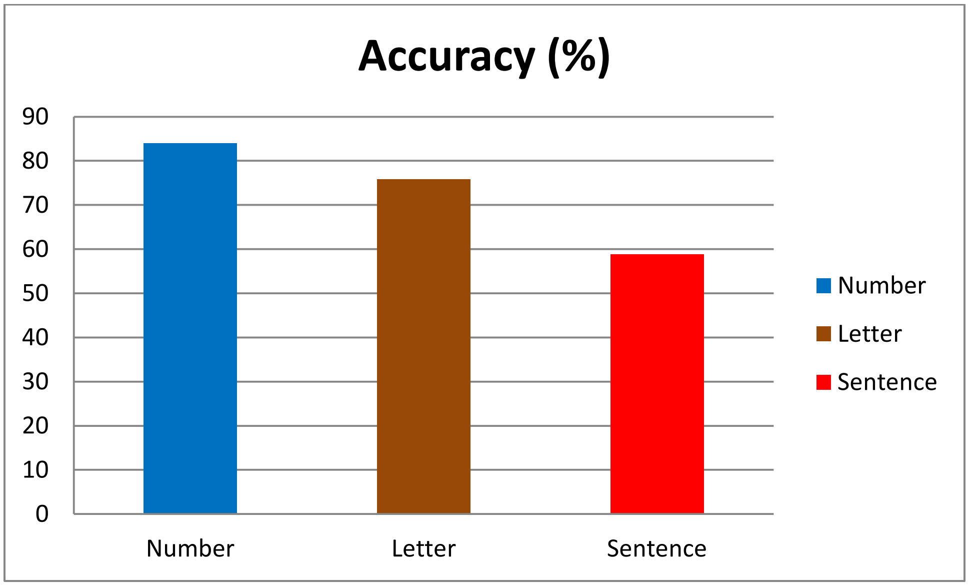


Figure 5.6: *Overall performance of the system*

The test result show as there are wrong translations on sentence and letter level. From the users feedback the errors happened because of incorrect word or letter signing which come from incorrect HamNoSys writing. In addition some sentences were understandable but not correct, it happened because of changing the signing behavior of words. For example, when we signed "**አይሮፕላኑ ሄደ**" we were not expected to sign both "**አይሮፕላን**" and "**መሄድ**" independently; instead we signed "**አይሮፕላን**" and added motion that indicated "**መሄድ**". Even if signing both words is understandable, it is not syntactically a correct way of signing.

CHAPTER SIX

CONCLUSION AND FUTURE WORK

6.1. Conclusion

The main focus of the thesis was to model for Amharic sentence to Ethiopian sign language (Ethsl) translator and develop a prototype system for translating Amharic sentence to EthSL. In parallel POS tagger, grammar translation, and preposition removal for natural language processing were developed. In addition, morphological analyzer and Romanizer were adopted from HornMorpho which already developed and available as the open source to aid the development of machine translation system. The output of sign animation is done using eSIGN 3D avatar animation, which is cost effective and flexible for signing sign languages.

Many researchers have been conducting researches on the translation of Amharic text to EthSL. But at the sentence level, this is the first step on the development of Amharic sentence to EthSL translator. Although every task has its own challenges, the big challenge on developing the model understood the grammatical rule of EthSL. Even if there is a great enhancement on the language (EthSL) development, still the language has not any written grammar rules. Due to the lack of written grammar, we were collect different information from the language professionals who are teachers at elementary, high school and university levels and also different sign language professionals that work at NGO as translators and teachers, and take time to understand the grammar of EthSL in lived experience.

The system accepts simple Amharic sentence as well as letters and numbers, and translates to EthSL. The performance of the system was measured in three categories at: sentence, letter, and number level and we got system accuracy 58.77% at sentence level, 75.76% at letter level, and 84% at number level.

Finally we conclude that in the development Ethiopian sign language, developing such models (Amharic sentence to EthSL translator) and translation software has great impact. In addition, has great contribution in the development of common signing language in a country as well as to filling of communication gap between hearing and deaf people

6.2. Future Work

This research has its own contribution on the Ethiopian sign language but to make the system and model complete and plays great roles in the development of sign language and filling the communication gap and took as a complete language tools futures works need to be conducted.

- Currently the translator is a one way communication, Amharic sentence to EthSL, and to make the system complete it needs to be a two way communication.
- The translator translates simple Amharic sentences and there is limitation on the translation of complex Amharic sentence, and also tense and subject/object constructive words are out of scope. Therefore, to make the system to be complete this gap needs to be filled.
- Word sense disambiguation system needs to be developed on the target side for Amharic to EthSL translator for avoiding semantic ambiguities.
- The system can be further enhanced by using a massive database of bilingual dictionary for better choice of words.
- To increase the performance of the system we need to enhance the signs on eSIGN 3D avatar animation.

REFERENCES

1. Abedi Tsegay Weldegebriel, *Offline candidate hand gesture selection and trajectory determination for continuous Ethiopian sign language*, master's thesis, Addis Ababa University, 2011.
2. Adafre Sisay Fissaha, *Computational Approaches to Semitic Languages*, workshop preceding, USA: University of Michigan, 2005.
3. Angus B. Grieve-Smith, *English to American Sign Language machine translation of weather reports*, University of New Mexico, 1999
4. Baye Yimam, "Amharic sewasew", published book, Ethiopia, 1986, pp. 216-303.
5. Binyam Gebrekidan Gebre, *Part Of Speech Tagging For Amharic*, ISMTCL Proceedings, UK: Wolverhampthon University, 2010
6. Birtat Magazin, monthly newspaper no. 7, Addis Ababa: ENAD, 2003
7. Bonnie J. Dorr, Eduard H. Hovy, Lori S. Levin, machine translation interlingua methods
8. Dagnachew Feleke, *Machine translation system for Amharic text to Ethiopian sign language*, master's thesis, Addis Ababa University, 2011.
9. Daniel Assefa, *Amharic speech training for the deaf*, master's thesis, Addis Ababa University, 2006.
10. Durgesh D Rao,*Machine translation, a gentle introduction,* Mumbai: national center for software technology, 1998.
11. Endale Assefa, *A hypermedia learning tool for Ethiopian sign language*, master's thesis, Addis Ababa University, 2005.
12. Ethiopian National Association of the Deaf, *Deaf culture and sign language training manual*, Addis Ababa: ENAD, 2005.
13. Ethiopian Sign Language Community, *the deaf people of Ethiopia*, people and language detail report, 2005.
14. Eyasu Hailu, *Sign language news*, Addis Ababa University, 2009.
15. Eva Safar and Ian Marshall, *the architecture of an English-text-to-sign-languages translation system*, Great Britain: university of East Anglia, 2005.
16. Fahim Muhammad Hasan, *Comparison of Different POS Tagging Techniques for Some South Asian Languages,* master's thesis, BRAC University, 2006.
17. Gamback b., Olsson f., Argaw a. and Asker, l., *Methods for Amharic part-of-speech tagging*, proceedings of the first workshop on language technologies for African languages, AfLaT, 2009.

18. Getachew, M., *Automatic part of speech tagging for Amharic*, master's thesis, Addis Ababa University, 2001.
19. Jan Bungeroth, Hermann Ney, *statistical sign language translation*, Germany: Rwth Aachen University, 2004.
20. Kathryn P. Meadow, "Early manual communication in relation to the deaf child's intellectual, social, and communicative functioning", American annals of the deaf, journal volume 148, number 1, University of California: Berkeley, 2003, pp. 18-24
21. Masresha Tadesse Erenso, *Automatic translation of Amharic text to Ethiopian sign language*, master's thesis, Addis Ababa University, 2010.
22. Michael Gasser, *HORNMORPHO*, Indiana University, school of informatics and computing, 2012.
23. Michael Nielsen, "Statically Machine Translation", July 15, 2013 [online]. Available:http://michaelnielsen.org/blog/introduction-to-statistical-machine-translation/ [Accessed: July 2013].
24. Michael W Morgan, *Complexities of Ethiopian sign language contact phenomena & implications*, Addis Abeba University. 2009.
25. Minilik Tesfaye, *a machine translation system to translate Amharic text to Ethiopian sign language*, master's thesis, Addis Ababa University, 2010.
26. Mohamed Amine, *theoretical Overview of Machine translation Chéragui*, Algeria: African University, Adrar, 2013.
27. NHS, "Introduction to hearing impairment", august 07, 2013 [online]. Available:http://www.nhs.uk/Conditions/Hearing-impairment/Pages/Introduction.aspx [Accessed: July 2013].
28. Oracle, "Introduction to mysql server", August 21, 2013 [online]. Available: http://www.oracle.com/us/products/mysql/overview/index.html [Accessed: August 2013].
29. Robert Smith, *HamNoSys4.0 for Irish Sign language Workshop Hand book*, Ireland: Dublin City University, 2010.
30. R. San-Segundo, R. Barra, R. Córdoba, L.F. D'Haro, F. Fernández, J. Ferreiros, J.M. Lucas, J. Macías-Guarasa, J.M. Montero, J.M. Pardo, *Speech to sign language translation system for Spanish,* Speech communication doi: 10.1016/j.specom.2008.02.001,2008.
31. Sign Writing Site, "two aspects of deafness", August 07, 2013 [online]. Available: http://www.signwriting.org/about/questions/quest024.html[Accessed: August 2013].

32. Sisay Fissaha, *Part of Speech tagging for Amharic using Conditional Random Fields*, Netherland: University of Amsterdam, Informatics Institute, 2005.
33. Steven Bird, Ewan Klein, and Edward Loper, "natural language processing with python", O'Reilly mediaInc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2009, pp. 179-214.
34. Thomas Hanke, *HamNoSys, Hamburg Notation System for Sign Languages*, workshop hand book, Institute of German Sign Language, 2009.
35. Uwe Ehrhardt, Bryn Davies, Neil Thomas, Mary Sheard, John Glauert, Ralph Elliott, Judy Tryggvason, Thomas Hanke, Constanze Schmaling, Mark Wells, and Inge Zwitserlood, *animating sign language*, the eSIGN approach based on eSIGN deliverable D7-2, 2004.
36. Wikipedia, "Deaf education", August 07, 2013 [online]. Available: http://en.wikipedia.org/wiki/deaf_education [Accessed: August 2013].
37. Wikipedia, "Romanize", July 17, 2013 [online]. Available: http://n.wikipedia.org/wiki/Romanization [Accessed: July 2013]
38. wiseGEEK, "Romanize", July 17, 2013 [online]. Available: http://www.wisegeek.com/what-is-romanization.htm [Accessed: July 2013].
39. Wondwossen Mulugeta and Michael Gasser, *learning morphological rules for Amharic verbs using inductive logic programming*, LREC conference, SALTMIL-AfLaT workshop: Istanbul, 2012
40. World Federation of the Deaf, "Deaf culture", August 06, 2013 [online]. Available: http://wfdeaf.org/our-work/focus-areas/deaf-culture [Accessed: August 2013].
41. W. John Hutchins and Harold l. Somers,*An introduction to machine translation*, London: academic press, 1992.
42. Yaser Al-Onaizan, Jan Curin, Micael Jahr, Kevin Knight, John Lafferty, Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, David Yarowsky, *Statistical machine translation*, report JHU Workshop, 1999.

APPENDIX – A

Randomly selected words from each part of speech

ተራ ቁጥር	ስም	ቅፅል	መስተዋድድ	ግስ	ተውሳከ ግስ
1	መኪና	ሌላው	በ	መሄድ	እንደገና
2	ልጅ	ጥሩ	ወደ	ነው	እዚህ
3	ቤት	ረጅም	ላይ	መቻል	ተጠቃሚ
4	አይሮፕላን	ከባድ	ለ	መቀመጥ	ዛሬ
5	ኢትዮጵያ	አጭር	ከ	መብላት	ትናንት
6	ተማሪ	ብዙ		መሸከም	
7	ውሃ	ልዩ		መለመን	
8	መርከብ	ዘመናዊ		መፍቀድ	
9	ታክሲ	ቀጭን		ማልቀስ	
10	ዳቦ	ሰፊ		መምጣት	
11	ሴጣን	ቀዩ		መቀበል	
12	አበባ	ቀዝቃዛ		ማመስገን	

APPENDIX – B

Sample sentences used to test Amharic sentence to EthSL translator

ተራ ቁጥር	ግብዓት	ውጤት
1	ቀዩ ልጅ በ መኪና ሄደ	እሱ ልጅ ቀይ መሄድ በመኪና
2	አጭሩ ልጅ በ ዛሬ ወደ ቤት ሄደ	እሱ ልጅ አጭር መሄድ ዛሬ ወደ ቤት
3	ዘመናዊው አይሮፕላን ወደ ኢትዮጵያ አልሄደም	አይሮፕላን ዘመናዊ መሄድ ወደ ኢትዮጵያ አይደለም
4	ተማሪዎች ታክሲ ተጠቃሚ ናቸው	ተማሪ ብዙ ታክሲ ተጠቃሚ ናቸው
5	አበባ ዘመናዊ መኪና ከ ልጁ ትናንት ተቀበለች	አበባ መቀበል ትናንት መኪና ዘመናዊ ልጁ
6	ረጅሙ ልጅ ትላንት ወደ ቤት አልሄደም	እሱ ልጅ ረጅም መሄድ ትላንት ወደ ቤት አይደለም
7	ሰፊ አይሮፕላን ነው	አይሮፕላን ሰፊ ነው
8	ቀጭኑ ተማሪ ውሃ ለመነ	እሱ ተማሪ ቀጭን መለመን ውሃ
9	ቀዝቃዛ ውሃ ጠጣ	እሱ መጠጣት ውሃ ቀዝቃዛ
10	ረጅሙ አይሮፕላን ወደ ኢትዮጵያ ሄደ	አይሮፕላን ረጅም መሄድ ወደ ኢትዮጵያ
11	ከባዱ መኪና ዘመናዊውን አይሮፕላን ተሸከመ	መኪና ከባድ መሸከም አይሮፕላን ዘመናዊ
12	ወደ ቤት ሄዱ	እነሱ መሄድ ወደ ቤት
13	ሴጣኖቹን ለመነች	እሷ መለመን ሴጣን ብዙ
14	ቀዝቃዛ ውሃ ተቀበለች	እሷ መቀበል ውሃ ቀዝቃዛ አይደለም
15	ቀዩ ልጅ እንደገና በ መርከብ ሄደ	እሱ ልጅ ቀይ መሄድ እንደገና መርከብስ
16	ረጅሙ ልጅ ውሃ ተሸከመ	ልጅ ረጅም መሸከም ውሃ
17	በዘመናዊ አይሮፕላን ሄዱ	እነሱ መሄድ አይሮፕላን ዘመናዊ
18	ሴጣን መኪና ላይ አይቀመጥም	ሴጣን መቀመጥ መኪና ላይ አይደለም
19	ጥሩ መርከብ ተቀበሉ	እነሱ መቀበል መርከብ ጥሩ
20	እዚህ ተቀመጠ	እሱ መቀመጥ እዚህ
21	ትላንት ዳቦ ተቀበልን	እኛ መቀበል ትላንት ዳቦ
22	ልጆቹ በ መርከብ አልሄዱም	ልጅ ብዙ መሄድ መርከብ አይደለም
23	በ ዘመናዊ አይሮፕላን ወደ ኢትዮጵያ ሄደ	እሱ ሄደ አይሮፕላን ዘመናዊ ወደ ኢትዮጵያ
24	መኪናው ላይ ተቀመመች	እሷ መቀመጥ መኪና ላይ
25	አመሰገነ	እሱ ማመስገን
26	ዳቦ ተሸከመ	እሱ መሸከም ዳቦ
27	ሌላ ዳቦ አልበላሁም	እኔ መብላት ዳቦ ሌላ አይደለም
28	በ ሰፊ መርከብ ትላንት ሄድኩኝ	እኔ መሄድ ትላንት መርከብ ሰፊ
29	ቀዩ ልጅ እንደገና አለቀሰ	እሱ ልጅ ቀይ ማልቀስ እንደገና
30	ቀዩ ልጅ ወደ ቤት በ መኪና ሄደ	እሱ ልጅ ቀይ መሄድ ወደ ቤት መኪና

APPENDIX – C

Amharic sera conversion guide

ሀ=ha	ሁ=hu	ሂ=hi	ሃ=ha	ሄ=hE	ህ=h	ሆ=ho
ለ=le	ሉ=lu	ሊ=li	ላ=la	ሌ=lE	ል=l	ሎ=lo
ሐ=Ha=ha	ሑ=Hu=hu	ሒ=Hi=hi	ሓ=Ha=ha	ሔ=HE=hE	ሕ=H=h	ሖ=Ho=ho
መ=me	ሙ=mu	ሚ=mi	ማ=ma	ሜ=mE	ም=m	ሞ=mo
ሠ=^se=se	ሡ=^su=su	ሢ=^si=si	ሣ=^sa=sa	ሤ=^sE=sE	ሥ=^s=s	ሦ=^so=so
ረ=re	ሩ=ru	ሪ=ri	ራ=ra	ሬ=rE	ር=r	ሮ=ro
ሰ=se	ሱ=su	ሲ=si	ሳ=sa	ሴ=sE	ስ=s	ሶ=so
ሸ=xe	ሹ=xu	ሺ=xi	ሻ=xa	ሼ=xE	ሽ=x	ሾ=xo
ቀ=qe	ቁ=qu	ቂ=qi	ቃ=qa	ቄ=qE	ቅ=q	ቆ=qo
በ=be	ቡ=bu	ቢ=bi	ባ=ba	ቤ=bE	ብ=b	ቦ=bo
ተ=te	ቱ=tu	ቲ=ti	ታ=ta	ቴ=tE	ት=t	ቶ=to
ቸ=ce	ቹ=cu	ቺ=ci	ቻ=ca	ቼ=cE	ች=c	ቾ=co
ኀ=^ha=ha	ኁ=^hu=hu	ኂ=^hi=hi	ኃ=^ha=ha	ኄ=^hE=hE	ኅ=^h=h	ኆ=^ho=ho
ነ=ne	ኑ=nu	ኒ=ni	ና=na	ኔ=nE	ን=n	ኖ=no
ኘ=Ne	ኙ=Nu	ኚ=Ni	ኛ=Na	ኜ=NE	ኝ=N	ኞ=No
ዐ=`a='a	ዑ=`u='u	ዒ=`i='i	ዓ=`a='a	ዔ=`E='E	ዕ=`='	ዖ=`o='o
ከ=ke	ኩ=ku	ኪ=ki	ካ=ka	ኬ=kE	ክ=k	ኮ=ko
ወ=we	ዉ=wu	ዊ=wi	ዋ=wa	ዌ=wE	ው=w	ዎ=wo
አ='a	ኡ='u	ኢ='i	ኣ='a	ኤ='E	እ='	ኦ='o
ቨ=ve	ቩ=vu	ቪ=vi	ቫ=va	ቬ=vE	ቭ=v	ቮ=vo
ኸ=he	ኹ=hu	ኺ=hi	ኻ=ha	ኼ=hE	ኽ=h	ኾ=ho
ዘ=ze	ዙ=zu	ዚ=zi	ዛ=za	ዜ=zE	ዝ=z	ዞ=zo
ዠ=Ze	ዡ=Zu	ዢ=Zi	ዣ=Za	ዤ=ZE	ዥ=Z	ዦ=Zo
የ=ye	ዩ=yu	ዪ=yi	ያ=ya	ዬ=yE	ይ=y	ዮ=yo
ደ=de	ዱ=du	ዲ=di	ዳ=da	ዴ=dE	ድ=d	ዶ=do
ጀ=je	ጁ=ju	ጂ=ji	ጃ=ja	ጄ=jE	ጅ=j	ጆ=jo
ገ=ge	ጉ=gu	ጊ=gi	ጋ=ga	ጌ=gE	ግ=g	ጎ=go
ጠ=Te	ጡ=Tu	ጢ=Ti	ጣ=Ta	ጤ=TE	ጥ=T	ጦ=To
ጨ=Ce	ጩ=Cu	ጪ=Ci	ጫ=Ca	ጬ=CE	ጭ=C	ጮ=Co
ጰ=Pe	ጱ=Pu	ጲ=Pi	ጳ=Pa	ጴ=PE	ጵ=P	ጶ=Po
ፀ=^Se=Se	ፁ=^Su=Su	ፂ=^Si=Si	ፃ=^Sa=Sa	ፄ=^SE=SE	ፅ=^S=S	ፆ=^So=So
ጸ=Se	ጹ=Su	ጺ=Si	ጻ=Sa	ጼ=SE	ጽ=S	ጾ=So
ፈ=fe	ፉ=fu	ፊ=fi	ፋ=fa	ፌ=fE	ፍ=f	ፎ=fo
ፐ=pe	ፑ=pu	ፒ=pi	ፓ=pa	ፔ=pE	ፕ=p	ፖ=po

```
ኧ='e
ሏ=lWa
ሗ=HWa=hWa
ሟ=mWa
ሧ=^sWa=sWa
ሯ=rWa
ሷ=sWa
ሿ=xWa
ቈ=qWe ቊ=qWi ቋ=qWa ቌ=qWE ቍ=qW
ቧ=bWa
ቯ=vWa
ቷ=tWa
ቿ=cWa
ኈ=hWe ኊ=hWi ኋ=hWa ኌ=hWE ኍ=hW
ኗ=nWa
ኟ=NWa
ኧ='e
ኰ=kWe ኲ=kWi ኳ=kWa ኴ=kWE ኵ=kW
ዀ=KWe=hWe ዂ=KWi ዃ=KWa=hWa ዄ=KWE=hWE ዅ=KW=hW
ዟ=zWa
ዧ=ZWa
ዷ=dWa
ጇ=jWa
ጐ=gWe ጒ=gWi ጓ=gWa ጔ=gWE ጕ=gW
ጧ=TWa
ጯ=CWa
ጷ=PWa
ጿ=SWa
ፏ=fWa
ፗ=pWa
።=. ፣=, ፤=; ፥=: ፦=: ፧=*?
```

APPENDIX – D

Brill tagging approach

```
import nltk
from secondpro import retag
from nltk.corpus import brown
def train_brill_tagger(train_data):
   from nltk.tag import UnigramTagger
   from nltk.tag.brill import SymmetricProximateTokensTemplate, ProximateTokensTemplate
   from nltk.tag.brill import ProximateTagsRule, ProximateWordsRule
   # The brill tagger module in NLTK.
   from nltk.tag.brill import FastBrillTaggerTrainer
   unigram_tagger = UnigramTagger(train_data)
   templates = [SymmetricProximateTokensTemplate(ProximateTagsRule, (1,1)),
                     SymmetricProximateTokensTemplate(ProximateTagsRule, (2,2)),
                     SymmetricProximateTokensTemplate(ProximateTagsRule, (1,2)),
                     SymmetricProximateTokensTemplate(ProximateTagsRule, (1,3)),
                     SymmetricProximateTokensTemplate(ProximateWordsRule, (1,1)),
                     SymmetricProximateTokensTemplate(ProximateWordsRule, (2,2)),
                     SymmetricProximateTokensTemplate(ProximateWordsRule, (1,2)),
                     SymmetricProximateTokensTemplate(ProximateWordsRule, (1,3)),
                     ProximateTokensTemplate(ProximateTagsRule, (-1, -1), (1,1)),
                     ProximateTokensTemplate(ProximateWordsRule, (-1, -1), (1,1))]
   trainer = FastBrillTaggerTrainer(initial_tagger=unigram_tagger,
                          templates=templates, trace=3,
                          deterministic=True)
   brill_tagger = trainer.train(train_data, max_rules=3000)
   print
   return brill_tagger
def tagger(sent):
#########
brown_train = list(brown.tagged_sents('Amharic_tagged_tr_corpus.txt'))
```

```
tagged= bt.tag(nltk.word_tokenize (sent))
    found=0
    for i in tagged:
        if(i[1]==None):
            found=1
            break
    if found==1:
        tagged=retag(tagged)
    return tagged
    print tagged
from pos_tag import main
import ast
def retag(pretagg):
    f1=open('notfound.txt', 'w')
    count=0
    pos=[]
    a=0
    for i in pretagg:
        count=count+1
        if(i[1]==None):
            if(count==0):
                pos.append(count)
           else :
                pos.append(count)
            f1.write (i[0])
            f1.write('\n')
            f1.close()
    if pos:
        main()
        words=[]
        retag=[]
        file_path = 'results.txt'
```

```
fin = open(file_path)
lines = fin.readlines()
words= ([line.strip() for line in lines])
count=0
for i in pretagg:
    if(i[1]==None):
        retag.append (ast.literal_eval (words[count]))
        count=count+1
    else:
        retag.append (i)
return retag
```

APPENDIX – E

A python program for preposition removal

```
def filter_insignificant(chunk, tag_suffixes=['PREP']):
 good = []
 index=0
 print "Starting"
 prepexc=[]
 for i in chunk:
    if i[0]!='wede' and i[0]!='lay' and i[0]!='gar' and index==0:
        prepexc=[(i[0],i[1])]
        index=index + 1
    elif i[0]!='wede' and i[0]!='lay' and i[0]!='gar' and index>0:
        prepexc.append(i)
        index=index + 1
    elif i[0]=='wede' and i[1]=='PREP' and index==0:
        prepexc=[(i[0],'EXCP')]
        index=index + 1
    elif i[0]=='lay' and i[1]=='PREP' and index==0:
        prepexc=[(i[0],'EXCP')]
        index=index + 1
    elif i[0]=='gar' and i[1]=='PREP' and index==0:
        prepexc=[(i[0],'EXCP')]
        index=index + 1
    elif i[0]=='wede' and i[1]=='PREP':
        prepexc.append((i[0],'EXCP'))
        index=index + 1
    elif i[0]=='lay' and i[1]=='PREP':
        prepexc.append((i[0],'EXCP'))
        index=index + 1
    elif i[0]=='gar' and i[1]=='PREP':
        prepexc.append((i[0],'EXCP'))
```

```
        index=index + 1
    else:
        prepexc.append(i)
        index=index + 1
for word, tag in prepexc:
  ok = True
  for suffix in tag_suffixes:
    if tag.endswith(suffix):
      ok = False
      break
  if ok:
    good.append((word, tag))
print good
return good
```

APPENDIX – F

A python program for EthSL grammar translation

Swap noun cardinality with adjective

```
def swap_noun_cardinal(sent):
    sen=[]
    temp=[]
    adj=[]
    count=0
    adjloc=0
    found=0
    print sent
    for i in sent:
        if found==0:
            if count==0 and (i[1]!='ADJ' and i[1]!='ADJP'):
                sen=[(i[0],i[1])]
                count=count+1
            elif count==0 and (i[1]=='ADJ' or i[1]=='ADJP'):
                adj=(i[0],i[1])
                found=1
                count=0
            elif (i[1]=='ADJ' or i[1]=='ADJP') and count>0:
                print i
                adj=(i[0],i[1])
                found=1
                count=0
            elif i[1]!='ADJ' and i[1]!='ADJP':
                sen.append(i)
                count=count+1
        elif found==1:
            if count==0 and i[1]=='N':
                sen.append(i)
```

```
            count=count+1
        elif count==1:
            temp=[(i[0],i[1])]
            count=count+1
        else:
            temp.append(i)
    count=0
    sen.append(adj)
    for i in temp:
        sen.append(i)
    return sen
```

Swap verb cardinality with adverb

```
def swap_verb_cardinal(sent):
    sen=[]
    adv=[]
    count=0
    ver=[]
    advloc=0
    verbloc=0
    #print sent
    for i in sent:
        if i[1]=='ADV':
            adv=i[0],i[1]
            advloc=advloc+1
            break
        else:
            advloc=advloc+1
    for i in sent:
        if i[1]=='V' or i[1]=='VREL' or i[1]=='VN':
            verbloc=verbloc+1
            ver=i[0],i[1]
            break
```

```
    else:
        verbloc=verbloc+1
print verbloc
print advloc
if verbloc>advloc:
    for i in sent:
        if i[1]=='ADV':
            sen.append(ver)
            sen.append(adv)
        elif i[1] != 'V' and i[1]!='VREL' and i[1]!='ADV' and i[1]!='VN':
            sen.append (i)
    return sen
else:
    return sent
```

Processing topic-comment structure

```
import nltk
from Filiterwords import filter_insignificant
from verbadverb import swap_verb_cardinal
from nountoadjectiveswap import swap_noun_cardinal
def Amharic_grammarprocess(sent1):
    index=0
    temp=[]
    noun=[]
    temporary=[]
    word=[]
    first=""
    second=""
    third=""
    verbloc=0
    advloc=0
    preploc=0
    adjloc=0
```

```
nounloc=0
index=0
verb=""
adverb=""
FinalN=[]
FinalN=sent1
sentstart=""
for i in FinalN:
    if i[1]=='V':
        verbloc=verbloc+1
        verb=i[0]
        index=index+1
    elif i[1]=='ADV':
        advloc=advloc+1
        adverb=i[0]
        index=index+1
    elif i[1]=='N':
        nounloc=nounloc+1
        index=index+1
    elif i[1]=='ADJ':
        adjloc=adjloc+1
        index=index+1
    elif i[1]=='PREP':
        if index==0 and i[0]=='ye':
            sentstart='ye'
        preploc=preploc+1
        index=index+1
if preploc>0:
    FinalN= filter_insignificant(sent1, tag_suffixes=['PREP', 'PREP$'])
if nounloc>=1 and adjloc>=1:
    FinalN = swap_noun_cardinal(FinalN)
if verbloc==1 and advloc==1:
```

```
    FinalN = swap_verb_cardinal(FinalN)
 index=0
for i in FinalN:
    if index==0:
        first=i[1]
        index=index+1
    elif index==1:
        second=i[1]
        index=index+1
    elif index==2:
        third=i[1]
        index=index+1
    else:
        break
index=0
if verb!='new':
    if first=='N' and second=='ADJ':#rule number 1
        for i in FinalN:
            if index==0:
                temp=[(i[0],i[1])]
                index=index+1
            elif index==1:
                temp.append(i)
                index=index+1
            elif index==2:
                temporary=[(i[0],i[1])]
                index=index+1
            elif index>2:
                temporary.append(i)
                index=index+1
        temp.append((verb,'V'))
        if advloc!=0:
```

```
        temp.append((adverb,'ADV'))
    for i in temporary:
        if i[1]!='V' and i[1]!='ADV':
            temp.append(i)
elif first=='N' and second=='N' and sentstart!='ye':#rule number 3
    for i in FinalN:
        if index==0:
            temp=[(i[0],i[1])]
            index=index+1
        elif index==1:
            temporary=[(i[0],i[1])]
            index=index+1
        elif index>=2:
            temporary.append(i)
            index=index+1
    temp.append((verb,'V'))
    if advloc!=0:
        temp.append((adverb,'ADV'))
    for i in temporary:
        if i[1]!='V' and i[1]!='ADV':
            temp.append(i)
elif first=='EXCP' and second=='N':
    for i in FinalN:
        if index==0:
            temp=[(i[0],i[1])]
            index=index+1
        elif index==1:
            temp.append(i)
            index=index+1
        elif index==2:
            temporary=[(i[0],i[1])]
            index=index+1
```

```
            else:
                temporary.append(i)
                index=index+1
        for i in temp:
            temporary.append(i)
        temp=temporary
    elif first=='N' and second=='N' and sentstart=='ye':#rule number 1
        for i in FinalN:
            if index==0:
                temp=[(i[0],i[1])]
                index=index+1
            elif index==1:
                temp.append(i)
                index=index+1
            elif index==2:
                temporary=[(i[0],i[1])]
                index=index+1
            elif index>2:
                temporary.append(i)
                index=index+1
        temp.append((verb,'V'))
        if advloc!=0:
            temp.append((adverb,'ADV'))
        for i in temporary:
            if i[1]!='V' and i[1]!='ADV':
                temp.append(i)
    elif (first=='V' and second=="") or (first=='N' and second==""):
        temp=[(i[0],i[1])]
        index=index+1
return temp
  else:
    return FinalN
```

Analyze morphological Information

```
for a in treesN:
    if a[1]=='V':
        verb=a[0]
if verb!='new' and verb!="":
    f = open ("MorphologicalInfo\Verb\Gender " + verb + ".txt","r")
    gender= f.read()
    f.close()
if verb!='new' and verb!="":
    f = open ("SegmentationInfo\Subject" + verb + ".txt","r")
    subject= f.read()
    f.close()
for i in treesN:
    if counter==0:
        if i[1]!='N':
            if gender=='Male' and subject=='3 Person Singular' and verb!='new':
                sentence=""su "
            elif gender=='Female' and subject=='3 Person Singular' and verb!='new':
                sentence=""sWa "
            elif gender=='Male' and subject=='2 Person Singular' and verb!='new':
                sentence=""ante"
            elif gender=='Female' and subject=='2 Person Singular' and verb!='new':
                sentence=""anci"
            elif subject== '1 Person Singular' and verb!='new':
                sentence=""nE"
            elif subject== '3 Person Plural' and verb!='new':
                sentence=""nesu"
        counter=counter+1
    if i[1]=='V' and i[0]!='new':
        f = open ("SegmentationInfo\Negation" + i[0] + ".txt","r")
        neg= f.read()
        f.close()
```

```
        f = open ("MorphologicalInfo\Verb\Root " + i[0] + ".txt","r")
        verbroot= f.read()
        f.close()
        if neg=='True':
            Negation='True'
            sentence=sentence + " " + verbroot
        else:
            sentence=sentence + " " + verbroot
    elif i[1]=='N':
        f = open ("MorphologicalInfo\Noun\Number " + i[0] + ".txt","r")
        numbers= f.read()
        f.close()
        f = open ("MorphologicalInfo\Noun\Root " + i[0] + ".txt","r")
        nounroot= f.read()
        f.close()
        if numbers=='Plural':
            sentence=sentence + " " + nounroot
            sentence= sentence + " " + "Bizu"
        else:
            sentence=sentence + " " + nounroot
    elif i[1] != 'N' and i[1] != 'V':
        f = open ("MorphologicalInfo\Others\Root " + i[0] + ".txt","r")
        othersroot= f.read()
        f.close()
        sentence=sentence + " " + othersroot
    elif i[1]=='V' and i[0]=='new':
        sentence=sentence + " " + i[0]
if Negation=='True':
    sentence=sentence + " " + "Aydelem”
```

APPENDIX – G

Software specification

Web Browser	Internet Explorer 6 or above
Java runtime environment	Sun java 1.4.2
Plug-in	eSIGN-SiGML-Player
Web Server	Apache server 2.2.11
Plate form	Window XP
Technology	HTML, CSS, PHP, and Python (with nltk and PyYAML tool kit)

DECLARATION

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in any other university, and that all source of materials used for the thesis have been duly acknowledged.

____

Daniel Zegeye

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

____

Sebsibe Hailemariam (PhD)

Addis Ababa, Ethiopia
June, 2014