

**ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
SCHOOL OF INFORMATION STUDIES FOR AFRICA**

**DESIGN AND DEVELOPMENT OF WORD PARSER FOR AMHARIC
LANGUAGE**

**BY
ABIYOT BAYOU**

**A thesis submitted to the School of Graduate Studies of Addis Ababa
University in partial fulfilment of the requirement for
the degree of Master of Science in Information Science.**

MAY, 2000

**ADDIS ABABA UNIVERSITY
LIBRARIES
P.O. BOX 1176
ADDIS ABABA, ETHIOPIA**

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
SCHOOL OF INFORMATION STUDIES FOR AFRICA

DESIGN AND DEVELOPMENT OF AMHARIC WORD PARSER

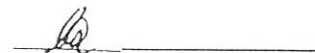
BY
ABIYOT BAYOU

Name and Signature of Members of the Examining Board

Ato Getachew Birru, Chairman, Examining Board



Dr. Nega Alemayehu, Advisor



Dr. John Cowell, External Examiner



DECLARATION

This thesis is my original work and has not been submitted for degree in any other university and that all sources of material used for the thesis have been dually acknowledged



Abiyot Bayou

The thesis has been submitted for examination with my approval as university advisor

 *19 May 2000*

Dr. Nega Alemayehu

ACKNOWLEDGMENT

I am deeply grateful to my advisor Dr. Nega Alemayehu, for his invaluable advice, constructive suggestions and sharing his resources.

My special thanks goes to the German Academic Exchange Services (DAAD) for its financial support throughout my study. I want to mention my greatest gratitude to my friend Ato Solomon Dessalegn for his co-operation in locating and selecting relevant materials on Amharic word morphology.

I am also indebted to Prof. Baye Yimam , Ato Wondesen and Ato Layne Brhane who gave me their valuable comments on this study. Finally I would like to thank all my friends for their encouragement.

Table of Content

DECLARATION	II
ACKNOWLEDGMENT	III
List of tables	VII
List of figures	VIII
List of Algorithms	VIII
Abstract	IX
 CHAPTER ONE	 1
INTRODUCTION	1
1.1 Background	1
1.2 Natural Language Processing System	4
1.3 Applications of Natural Language Processing	5
1.4 Developments in Amharic Language Processing	7
1.5 Statement of the Problem	9
1.6 Objective of the Study	10
1.7 Methods	10
1.7.1 Data Collection and Literature Review	10
1.7.2 Development of the Parser	11
1.7.3 Testing technique	11
1.8 Organization of the Thesis	11
 CHAPTER TWO	 13
WORD PARSING	13

2.1 Introduction.....	13
2.2 Words and Meaning.....	13
2.2.1 Word Feature	16
2.2.2 Categories of Words	18
2.3 Morphological Parsing.....	20
2.3.1 Knowledge Required by a Word Parser	21
2.3.2 The Lexicon of Natural Language Processing.....	22
2.3.3 Output of Morphological Parsing	23
2.4 Approaches to Morphological Parsing.....	24
2.4.1 Bottom up Approach	24
2.4.2 Top-to-Bottom Approach	25
2.4.3 Two Level Model of Morphology	26
2.4.4 Unification Based Word Grammar	29
CHAPTER THREE.....	32

THE STRUCTURES OF AMHARIC WORD 32

3.1 Introduction.....	32
3.2 The Amharic Writing System	32
3.3 Vocabulary and word Shapes of Amharic	35
3.3 Amharic Verb Classes.....	36
3.3. Conjugation of Amharic Verbs	39
3.3.1 Derived Verbs.....	40
3.3.2 Inflectional Affixes of Amharic verb stems	43
3.3.2 Combination of Suffixes and Suffix Orders of Amharic Verb	50
3.4 Prefixes To Amharic Verbs.....	54
3.4.1 Clause Markers	54
3.5 Orders of Prefixes	58
3.6 Derived Nouns	60
3.7 Suffixes to Nouns.....	69
3.8 Prefixes to Nouns.....	72

CHAPTER 4 73

DESIGN OF LEXICONS FOR AMHARIC WORD PARSER..... 73

4.1 Introduction.....	73
-----------------------	----

4.2 Approaches to the Design of Word Parser for Amharic Language	73
4.3 Designing the Lexicon	76
4.4 Designing the Affix Database	78
4.4.1 Designing the Suffix Databases	79
4.4.2 Designing the Prefix Database	88
4.5 Phonological Properties Used	92
4.5.1 Vowel Changes	92
CHAPTER FIVE	100
ALGORITHM DESIGN AND THE EXPERIMENT	100
5.1 Introduction	100
5.2 The Word Parsing Algorithm	100
5.2.1 Extracting Morphological Information	101
5.2.2 Deciding the categories of the word	115
5.3 Results of the Experimental	120
CHAPTER SIX	123
CONCLUSION AND RECOMMENDATION	123
6.1 Introduction	123
6.2 Conclusion	123
6.3 recommendation	124
Reference	126

LIST OF TABLES

Table 3.1 Amharic Consonants.....	34
Table 3.2 Major Division of Amharic Verbs	38
Table 3.3 Sub-Division of Amharic Verbs	38
Table 3.4 Subject Marker Affixes Of The Imperfective/Jerundive Stem	45
Table 3.5 Subject marker suffixes of the Perfective stem.....	46
Table 3.6 Subject marker affixes of the Jerundive stem.....	46
Table 3.7 Object Marker Suffixes.....	48
Table 3.8 Affermative Suffixes.....	49
Table 3.9 Orders sufices to the Amharic verb	50
Table 3.10 Possible Combinations of Suffixes to Amharic Verbs	53
Table 3.11 Amharic Verb Suffixes with their orders.....	59
Table 3.12 Combinatios of Amharic prefixes.....	60
Table 3.13 Suffixes To Amharic Nouns that indicates Possesion	69
Table 3.14 Suffixes to Amharic Nouns With Their order.....	70
Table 3.15 Combination of Amharic Suffixes	71
Table 4.1 Structure of the suffix File	85
Table 4.2 Structure of the concatnated File	86
Table 4.3 Structure of The concatnated Suffix File	90
Table 4.4 Vowel Changes	92
Table 4.5 Dental consonants with the corresponding Palatal Consonants.....	95
Table 4.6 Palatalization doue to 'ia'	96
Table 4.7 Palatalization doue to 'e'	97

LIST OF FIGURES

Figure. 2.1 Different forms of ‘sbr’	15
Figure 2.2 Components of the word ‘sädäbat’	20
Figure 2.3 Surface and underlying forms of ‘Kfäčiylat’	25
Figure 2.4 Components of the Kimmo Parser.....	28
Figure 2.5 Tokens of the word ‘enlargement’	29
Figure 2.6 Unification based Parsing for the word ‘enlargement’	30
Figure 3.3 Vowel and consonant layer of ‘säbär’	40
Figure 4.1 The suffix ‘ - ačhun’ in different conditions.....	87
Figure 5.1 Overview Diagram of the Parser	101
Figure 5.5 Possible prefixes of the word ‘slämayanqäsaqsulačhu’	107
Figure 5.6 Alternative affix combination results in legal root but wrong stem.....	108
Figure 5.7 Wrong stem for sub type ‘a’ roots	108
Figure 5.10 Flow Chart for Determining Inflectional Catagories.	116

List of Algorithms

Figure 5.1 The Algorithm that extracts all possible prefixes from a word	102
Figure 5.2 The algorithm that extracts possible suffixes	103
Figure 5.3 The algorithm for checking alternative affixes.....	105
Figure 5.4 The algorithm for checking stem when the word has only prefix	106
Figure 5.8 The algorithm to check the verb stem	111
Figure 5.9 The Algorithm to Checks Derived Nouns	112
Figure 5.9 The algorithm that determines the inflectional category of a noun.....	120

Abstract

Most of the human knowledge is recorded using natural language. This language is too complex to be understood by computers. Only computers that have the capability to understand Natural language can access all these information efficiently. In addition, Natural Language interfaces to computers would allow complex systems to be accessible to every one.

Activities in the area of Natural Language Processing tries to design and implement computer programs that can understand Natural Language and act appropriately on the information contained in the text or utterance. Enabling computers to understand natural language involves extraction of meaning from words in texts. Word parsing is a process of extracting the meaning expressed by the words of a text by tokenising words to their basic components.

This study is an attempt of designing and implementing word parser for Amharic words. The study focuses on Amharic verbs and their derivations, which covers more than 75% of the words in the language. The paper describes the morphology of Amharic verbs and attempts to extract features that enables to implement automatic Amharic word parser. The morphological data described and analysed in this study is taken from study outputs in the area of Amharic word morphology.

An algorithm that can use the morphological property of the language has been designed and presented. Even though this study focuses on Amharic verbs and their derivations the algorithm and the discussion can be easily extended for other word groups. The study also describes the information contained in affixes and designs a database that can hold this information for word parsing and further analysis works.

CHAPTER ONE

INTRODUCTION

1.1. Background

find similar list { As the cost of computers decreases together with increasing performance, computer technology continues to penetrate and reshape the society. It is hardly possible to name an area on which computer has no impact. A number of studies have been conducted to increase the capabilities of computers and to make computers more accessible to the society. Among these studies computer processing of natural language is the one. The academic discipline that studies computer processing of natural language (NL) is known as Natural Language Processing (NLP) or Computational Linguistic (Gazdar,1996). NLP involves the development of computational models of a language and based on this develop computer programs which can analyse Natural Language (NL) and act appropriately on the information contained in the text or utterance (Grishman,1984).

One of the reasons that has created a great interest on NLP is the fact that most of the human knowledge is recorded using natural language. This language is too complex to be understood by computers. Only computers that have the capability to understand NL can access all these information efficiently. In addition, NL interfaces to computers would allow complex systems to be accessible to every one. Where computers are already involved in creating, transmitting, storing, searching, or reproducing text; we have the chance at little marginal cost to add new features.

NLP involves language understanding. But the complexity and richness of human language make the language understanding process very difficult, but not impossible (Allen, 1996:3). The first step in understanding natural language involves recognition of individual words (Cottrell, 1987:1). Word parsing is one of the method by which one can make individual words to be understood by the computer (Anteworth, 1994). Therefore any study in accordance with this is crucial for implementing systems that will understand natural languages.

Natural language is any language that human being learn from their environment and used to communicate with each other. Natural languages are used to express our knowledge and emotions and to convey our response to others. Humans process natural languages whenever they interact with other peoples and to the surrounding. In any language, whether it is spoken or written, every message has a structure and the elements of a language relate to each other in recognisable ways.

A sentence expresses a complete thought and is made up of words, which individually or in combination with other words represent ideas (Harris,1985:15). Words are formed from morphemes, which constitute the smallest meaningful unit of a language. Morphemes can be divided in to phonemes. Phoneme is smallest distinctive unit of speech in a language. A phoneme may be a group of sounds. Phonemes do not carry meaning in the usual sense, but must be combined into established patterns to produce meaningful units

One of the assumption of language analysis is that the elements of a language, both phonemes, and morphemes, do not merely form sequential chains, but are integrated in to units of a higher order of complexity which function as a whole. This whole is characterised by certain overall

qualities that transcend the characteristics of the mere sum of their components (Garvin, 1963:17). Language therefore can be seen as a system of linguistic units, which can be represented by signs. In each instances of use of a language a selection is made among the signs available within the system. The selection of these linguistic units is based on the function of the unit. The function of each linguistic unit may be closely related to, or even governed by, the presence of other units with in the sequence. The selection and the sequential arrangement of the unit give the meaning of the whole. For example, the Amharic verb root ‘gdl’ ገደለ ‘kill’ can combine with other units in the language as in the example below;

gädäl	ገደለ	‘ Kill’	stem of the word
gädäl-ku	ገደለ-ከ	‘ I kill’	1 st person singular
gädäl-ku-at	ገደለ-ከ-አት	‘ I killed her’	3 rd person singular female
gädäl-ku-l-at	ገደለ-ከ-ለ-አት	‘I killed (some thing) for her’	

In the above example, the basic units can form a number of different forms by combining with other units in different ways. Each instance of combination gives a unique meaning or has a specific interpretation. Understanding of the meaning of the whole therefore involves identifying the components out of which the whole is formed and the sequence of arrangements of the components.

- Linguistic entities that can be formed from other lower units can have features that can transcend the mere sum of the units. For example, the Amharic word ‘hayläña’ ሃይለኛ ‘powerful’ is an adjective given to express a powerful-thing. However its components ‘hayl’ ሃይለ ‘power’ and ‘-ña’ -ኛ ‘suffix’ do not express such quality of an object before they are combined. Such type

of interpretation can be possible by looking at the features that show the behaviour of the unit when combined with other units. For instance in the previous example 'hayl' is a noun, and '-añ' is a suffix that can be attached to a noun and when attached to a noun it changes the category of the new word to adjective. Such features of a language helps to identify the category and hence the meaning of the word. Therefore, a system that needs to understand a language should identify such behaviours of the language.

1.2. Natural Language Processing System

All applications of NLP have the common objective of understanding and extracting meaning from a text input, which involves transforming the text into a form where the meaning is explicit and easily usable by the application program. Therefore, the implementation of such type of system requires enabling computers to understand the meaning of the text, they are processing. Meanings in a sentence can be represented by using different linguistic units and syntactic structures. Therefore we should have a mechanism to capture and represent this rules and units, so that the computer can recognise the meaning of the text.

Parsing is one of the major activities towards this. Parsing performs an analysis of the natural language utterance or sentence and resolving ambiguities to the extent possible using various sources of linguistic and other statistical information. The major purpose of parsing is extracting structural and semantic information from the input text (Gungordu & Kemal, 1995). To carry out parsing, the parser should be supplied with information about each word in the sentence. Therefore the first step in any natural language processing involves recognising and identifying individual word forms from the input text (Salton, 1989:329).

In some languages, like English, such information may be supplied by a lexicon that simply lists all word forms with their part of speech and inflectional information such as number and tense. Because such languages have a relatively simple inflectional system, the number of forms that must be listed in such a lexicon is manageable. But an exhaustive lexical listing is simply not feasible for many other languages that have a number of inflected forms for each noun or verb (Antworth, 1994). This is because each lexical word may have literally thousands of distinct surface forms differing in inflectional properties but other wise the same vocabulary elements. Therefore NLP for these languages could be possible if only there is a word parser that will use the morphological system of the language to compute the part of speech and inflectional categories of the word (Antworth, 1994; Kuritutumun, 1993).

1.3 Applications of Natural Language Processing

There are a number of areas that NLP can be applied ranging from a small hand held dictionary to a sophisticated system that can translate a text from one language to another.

Grishman (1984) indicates that NLP has two primary roles to play in the storage and retrieval of large bodies of information. In retrieval, NLP offers the potential for a friendly, flexible interface through which the user can state his information requirements.

The objective of Natural Language interfaces is to take inputs in human language and extract from them something that is meaningful to a computer. A natural language query to a database system provides end users with a way to formulate queries in a native natural language.

For information storage, NLP offers the potential for structuring large bodies of textual information in order to facilitate retrieval of individual facts. This application of NLP can play a great role in the areas of automatic indexing and automatic abstract generation. Automatic indexing is based on the assumption that there are some parts of a text that are expected to contain useful information than others (Artundi, 1976, Sparc Jones, 1974). NLP researches in machine indexing has largely involved in the study of meaning implicated by the relation of signs, one to another. They identify the relation of signs to the objects to which the signs are applied and then enable the computer to recognise the relationship of the signs and their designation (Lewis, 1996).

✓ Automatic abstract generation involves sentence selection based on the semantic content of the sentence (Johnson, 1997). The complex syntactical rule of a language represents a set of constraints that give structure to the system. These constraints will create redundancy (something said or written than is strictly necessary to convey the message). We can expand our phrases and sentences and express our thoughts in several different ways that result in semantic redundancy. Automatic abstracting involves selection of those sentences that do not repeat a concept and introduces the general content of a text.

✓ Machine Translation (MT) is another application of NLP. MT is a processes of converting information from one language to another with a computer (Vasconcellos, 1993). MT analyzes a text in the source language (original language) and automatically generates sentences in the target language. MT works best if the subject matter is specific or restricted. MT systems has a great potential to break the language barriers for information transfer.

Other applications of NLP include spelling correction and Grammar checking in word processing, language learning, and analysis of language, text to speech synthesis and text compression. All the applications discussed involve tokenizing a text to an understandable segments, in order to processes input texts. Therefore, to develop application of such types one needs a mechanism to enable the system to recognize the language by which the text is created.

1.4 Developments in Amharic Language Processing

Ethiopia is a multilingual country with a population of around 62 million. Amharic is the official language of the federal government. Although many languages are spoken in Ethiopia, Amaharic is clearly dominant in that it is spoken by a substantial segment of the population. It is more highly developed than other languages in Ethiopia, in that it has a well-established writing system and well standardised norms of spoken and written usage (Bender and Ferguson, 1976). Amharic is one of the highly inflected Semitic languages. Nouns as well as verbs are inflected for a number of grammatical cases (Girmay, 1992).

To the best of my knowledge, there is no research conducted on NLP of Amharic Language, which enables to implement systems that can understand the language. Nevertheless, there are few related studies in relation to computerisation of Amharic language.

Worku (1997) has designed an algorithm for Optical Character Recognition of Amharic characters, and subsequent study is conducted by Dereje (1999). They develop an algorithm for

optical recognition of type written Amharic texts. In the area of Text to Speech Synthesis Laine (1998) has developed a text to speech system for Amharic language.

The study by Wedekind (1992) to determine the minimal form that can predict other forms of a verb, can be considered as the first attempt to characterise Amharic words using computer. In his study, Wedekind concludes that the root form can predict other forms un- ambiguously.

There are also few researches done on Amharic text retrieval. Biru (1992) has customised the CDS/ISIS software for Amharic Language. A recent study on development of stemming algorithm for Amharic is studied by Nega (1999). He develops a stemming algorithm for Amharic text retrieval that conflates different forms of a word to a common single form.

There are a number of keyboard drivers and character sets to represent Amharic Fidel scripts. Since there is no standard followed to develop these applications, most of them are not compatible. Thus, it is not possible to convert a text written using one character set to another. To date there is no word-processing tool for Amharic language that can support language specific utilities such as spell checks, grammar, thesaurus etc.

The limited numbers of researches in relation to Amharic language, indicates that there is an urgent need to conduct further research in the area of computer processing of Amharic language in general. Moreover, the absence of application that can recognise Amharic language indicates that there is a need to conduct a research that can identify such possibilities.

1.5 Statement of the Problem

The absence of word-processing tool for Amharic language that can support language specific utilities such as spell checks, grammar, thesaurus etc. made word processing activities using the Amharic language incomplete. Implementation of such utilities requires handling the morphological variants of words of the language.

Today, a large amount of information is available on the Internet. A country like Ethiopia can enhance its development by making this huge amount of information accessible to its local peoples. One of the barriers to make this information accessible to local people is a language problem. Online Machine Translation systems that can translate text from a foreign language to local language can solve the problem greatly. Therefore the need for having NLP systems is unquestionable.

Similar to other highly inflected languages, Natural Language Processing for Amharic language is difficult without having a word parser, which uses the morphological system of the language. Studies on the entries of Amharic dictionaries shows that 75% to 80% of the entries conform that most entries can either be traced back to verbs or they themselves are verbs (Leslau, 1973:300). The Amharic verb roots predict the meaning of many nominal entries of the lexicon. Therefore it is possible to say that the Amharic verb can more or less represent the language (Wedikend, 1992). Thus, the purpose of this study is to find a possibility of developing automatic word parser for Amharic verbs and their derivations, based on the morphological system of Amharic language.

1.6 Objective of the Study

The general objective of this study was to investigate a development option for automatic word parser for Amharic verbs and their derivations. With the aim of achieving the general objective, the study has attempted to address the following specific objectives:

1. Studying Amaharic verbs and investigate patterns which can allow computer representation;
2. Studying morphological property of the language, and identifying those properties appropriate for word parsing;
3. Studying what type of lexicons are needed to be stored for Amaharic word parser, and design the lexicon accordingly;
4. Select word parsing algorithm, which best feet for Amaharic word parsing, if any, or develop a new one ,and test the algorithm for effectiveness;
5. Develop a prototype that will implement the findings of the study.

1.7 Methods

To come up with a successful research output, the following methods were employed.

1.7.1 Data Collection and Literature Review

In order to identify the property of Amharic words that can enable to develop automatic word parser one has to do research on the language in general. Nevertheless, this is costly and can take a long time. Alternatively it is possible to use the output of other researches. Therefore in this study, literatures in the area of Amharic language morphology have been reviewed for identifying

morphological property of the language. Together with these literature in the area of computational linguistics in general, and on word parsing in particular has been reviewed.

1.7.2 Development of the Parser

Based on the analysis results, a word parser algorithm is designed. In order to parse words more meaningfully, lexicons are designed. In designing the lexicons, the linguistic elements are taken into consideration. The prototype is coded using Visual C++ Programming language. This programming language is selected because;

- The program can be easily integrated with other application,
- The source code can be modified easily for further modifications,
- Easy to add menus and other features

1.7.3 Testing technique

The prototype developed is tested for effectiveness by using selected Amharic words. Based on the results the algorithm is modified, and further tests are conducted again to see the effectiveness of the final algorithm

1.8 Organization of the Thesis

This thesis is organised in six chapters. Chapter one introduces what a Natural language processing is and the role of word parser for natural language processing. This chapter also presents the objective and statement of the problem. Different approaches for developing a word parsing and issues related to word parsing is discussed in chapter two. The third chapter describes the Amharic language and the word formation processes of the language. In chapter four the

features of Amharic words that are used in developing the word parser are discussed. This chapter also describes the design of the lexicon. The algorithm designed and the experiment conducted is discussed in chapter five. The conclusions and recommendations made in the study are presented in chapter six.

CHAPTER TWO

WORD PARSING

2.1. Introduction

As indicated in the previous chapter the main objective of the study is to design and implement Amharic word parser. Word parsing is a process of tokenizing a word in to its components or individual morphemes. Different from stemming words for purposes such as statistical analysis, in word parsing the word should be tokenized in to morphologically valid components. The component tokenized in to parts will be further analyzed for their contribution to the meaning and categorization of the whole word.

In this chapter, different approach to morphological parsing and issues related to word parsing for NLP is discussed. The first section discusses about linguistic features of words and their meaning. The second part discusses different approaches of morphological parsing.

2.2 Words and Meaning

Every language enables its speakers to communicate each other. Human being can exchange ideas by using string of words arranged in a specified manner. A word is an arbitrary symbol representing a physical object or a concept. The grammar of a language dictates the way words are arranged and combined during communication. String of words arranged based on the

grammar of the language will form a sentence. The meaning of a sentence can be understood from the meaning of the individual words and the way they are arranged.

Words have their own structural properties. In many cases they have a complex internal constituent structure and a set of idiosyncratic properties that are crucial for determining aspects of the syntactic structure of sentences (Gazder and Pullum, 1985).

Using the words and grammar of a language we can express unlimited number of ideas. According to Pinver (1999), this is possible because our knowledge of languages resides in an algorithm that combines abstract symbols such as 'noun' and 'verb' as opposed to concrete concepts such as 'man' and 'dog'. These abstract symbols embraces a vast set of concepts and can be combined freely into an even vaster set of propositions. Therefore the grammar can express a remarkable range of thoughts. For this reason a grammar can be seen as a finite algorithm with the ability to generate an infinite number of potential sentences, each corresponding to a distinct thought (Pinver, 1999).

Words can be created by combining morphemes or Words. In Most languages words are created by a sequence of word formation rules which can be applied iteratively to each other's output. This process generally creates new words from existing words. Words that are created from the same root represent different forms of the same word. As indicated by Lyons (1995:52), these different forms of the same word will generally differ in meaning. They will share the same lexical meaning but they differ in respect of their grammatical meaning. For example the Amharic word;

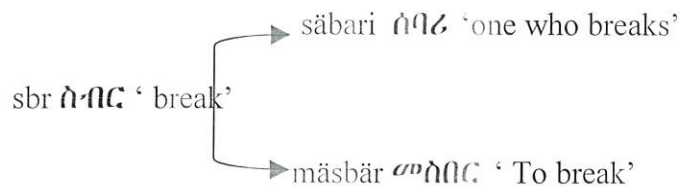


Figure. 2.1 Different forms of 'sbr'

Both words, ሰባሪ and መስባር share the same lexical meaning ስብር but they differ grammatically. ሰባሪ indicates the agent who do the underlying action while መስባር is a name of the underlying action. As the meaning of the sentence depends on words these phenomena is also relevant to sentences meaning.

The processes of word formation are generally based on the grammar. Therefore, it is possible to say that the meaning of the words, of which a sentence is composed of and its grammatical meaning, determine the meaning of a sentence.

All words do not have equal contribution to the meaning of a sentence. The contribution of a word to a meaning depends on the category and feature of the word. The category or class of the word determines the possible syntactic rules to be applied on the word. Syntax is the set of hidden dependency relationships among the words of a language. The syntactic rule of the language makes all sequence of words not equally acceptable. Thus to determine the contribution of a word to the meaning of a sentence and the rules that can be applied to a particular word, identifying the class or category of the word should be the first step.

2.2.1 Word Feature

In order to define classes for linguistic entities that need to be referred to in rule sets and other constraint systems, an appropriate classification system for linguistic entities (phonemes, syllables, morphemes, words, phrases and sentences) must be identified. A linguistic theory provides a system of constraints on linguistic entities. In formal theory, constraints are expressed as relations among or operations on representation of linguistic entities. The linguistic feature is a formal mechanism to define a system of such classes (Prasad, 1993:114).

Feature of linguistic entities may be represented by feature structure. A feature structure is a set of feature specifications where each feature specification is an "attribute value" pairs feature, i.e. for each attribute there is a corresponding value. Feature structures may be combined by the operation of unification. Unification enables the building up of a description of a linguistic object by unifying information from various sources that represent the mutual constraints that words and their sentential environments put on each other (Whitelock, 1988.). This unification can help us to determine the categories of words based on the values of the attributes. These attributes may include:

- Polarity

Polarity indicates the affirmativeness of the word. Its value can be either negative or positive.

- Tense

Tense refers to the time of an action or state in relation to a temporal focus. Tense can have a value of past, present or future.

- Evidentiality

Evidentiality expresses whether the narrator actually saw what is being described, or inferred it, or was told about it, and so forth.

- Aspect

Aspect refers to a temporal composition or the completion of an event, which usually indicate perfective/imperfective. It generally refers a sense of grammatical specification in the present discussion

- Person

Person make a distinction between 1st person (referring to the narrator), 2nd person referring to the addressee) and 3rd person (referring to someone other than speaker and addressee)

- Reference classification

This includes gender feature whose value can be masculine /feminine, or animate and inanimate.

- Number

Number describes how many individuals are referred to by a predicate argument. Its value can be singular or plural

- Case

Case marks the function of the predicate argument. Case indicates the syntactic function of the NP in the clause. Case marking can be clitics or particles and inflection. Case generally marks, subject, object, dative, instrumental, locative, ablative)

2.2.2 *Categories of Words*

Lexical items can be classified based on phonological, syntactic and semantic features. Syntactic features include category features, inherent features and contextual features (Jacobson, 1978:113).

2.2.2.1 *Categorical Features*

The category of words can be determined semantically, morphologically or syntactically. Words may be semantically categorised based on the function they have in the sentence (Radford, 1988: 57) as in: -

1. Words that denote actions - Verbs
2. Words that denote entities - Nouns
3. Words that denote states - Adjectives
4. Words that denote the manner in which something is done -Adverbs
5. Words that serve to specify - Determiners

In a semantic categorisation, a word can be categorised in more than one category. Therefore the semantic categorisation of words is unreliable. Hence, other mechanisms should be devised to determine the category of a word.

Morphological categorisation is one of the mechanisms to the above-mentioned problem. Morphological categorisation depends on the fact that certain types of inflections are attached only to specific categories of words noun (Radford, 1988:57). For example, only nouns take the plural marker ‘-oč’ in Amharic, and any word with suffix –oč can be categorised as a

The syntactic criteria are based on the distribution property of words. The distribution property of a word is the set of sentence positions it can occur in. For example in Amharic language only verbs can come at the end of a sentence. Therefore any word which come at the end of Amharic sentence can be considered as a verb.

2.2.2.2 Inherent Features

Inherent feature of a lexical item indicates the sub category of a lexical item (Jacobson, 1978:113). Members of a given category may belong to a variety of different categorical subclasses. For example, a word can be countable or unaccountable noun, or transitive or intransitive verb. The sub category of words determines the type of morphological and syntactical processes that are possible on the given word (Radford, 1988: 337).

2.2.2.3 Contextual Features

The contextual features determine how the word shall be used in the sentence given a certain context. These features are strict sub-categorisation features. For example, some verbs can be used without object while others do not. The contextual feature also includes the selection feature of the lexical class. The selection feature of a word indicates the features required by certain categories of words from other categories to function together in a given context (Jacobsen, 1978:114). For example, it is not possible to say, 'The man is pregnant '. Even though the sentence in the example is grammatically legal sentence it is not acceptable because that the word 'pregnant' requires a 'female animate' subject.

2.3 Morphological Parsing

In NLP before a syntactic parser can parse a sentence, it must be supplied with information about each word. Morphological parsing helps us to identify what information a word form carries (Palmer & Zhibiao, 1995). For example, in translating a word from one language to another there may not be equivalent words. A single word in one language may require a sentence in another language to represent the full meaning of that word. For this reason the translating system should get information about the structure of a word. For example the Amharic word 'sdb' ሰደብ has the meaning of 'insult' in English, but the meaning of 'sädäb-ä-at' ሰደብ-ኧ-አት can only be represented using a full sentence, 'he has insulted her'. Therefore, to get such information for further analysis the word shall be broken in to its morpheme as in the example.

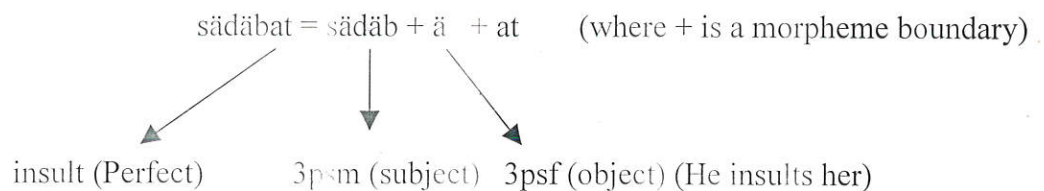


Figure 2.2 Components of the word 'sädäbat'

In analysing the structure of words it is impractical to list all the words in the language together with an indication of their relevant properties. For most languages, listing all the words in the lexicon is both costly and linguistically inadequate. This is because-for example in agglutinative languages –each lexical verb may have literally thousands of distinct surface forms differing in inflectional properties but otherwise the same vocabulary elements. These words have a considerable degree of predictability, i.e. using basic forms and word formation rules of the language, it is possible to predict the other forms. It is therefore redundant to store exhaustive list of the words in a language. (Anderson, 1988; Golding & Thomson, 1985).

2.3.1 Knowledge required by a Word Parser

In order to write efficient parsing algorithms one needs to have a secure definition of the language to which he designs the parser. Much of the knowledge for NLP purpose must come from Linguistics studies. This helps us to understand better the organisation and operation of the language. Knowledge of the grammar of the language also helps to have an idea how the system should behave under a wide range of conditions. For this reason NLP system will have to be based on what linguists can determine about the structure of the language (Gazder and Pullum, 1985; Spark Jones, 1974).

The grammatical knowledge of the language should be represented in appropriate form for the parser in order to carry out the parsing process effectively. Even though some basic knowledge is common for most parsers, the detail knowledge required can vary from language to language. Pullman et. al. (1988) notes the following as the three main types of knowledge that needs to be represented;

- i. Knowledge about spelling or phonological changes upon affixation;
- ii. Knowledge about the syntactic or semantic properties of affixation (That is , inflectional and derivational morphology); and
- iii. Knowledge about the properties of the stored base forms of words.

In addition, the parser should have knowledge about the syntactic or semantic properties of the vowel patterns of the stem, particularly for Semitic words. This is because vowel pattern of the stem in Semitic languages can determine the inflectional and derivational category of a word. For Example two Amharic words ‘sbäri’ ሰበረ (break! (2psf)) and ‘sbäri’ ሰባረ (a fragment) derived

from the same root 'sbr' ሰብር (break), but due to different vowel pattern they have different meanings.

2.3.2 The Lexicon of Natural Language Processing

The lexicon for NLS contains the vocabulary that the system knows. The lexicon of NLP should provide the syntactic, semantic and pragmatic information necessary for each lexical item to be processed by the system (Harris, 1985:322). The lexical entries are different for different parsers, based on the goal of the parser. This is because the domain of the application limits the type of language that the parser is likely to encounter (Harris: 322; Metzler, 1989).

There are two sets of categories of lexical items: open categories and closed categories. Open categories represent the primary function of a word: nominal, action, nominal modifier, and action modifier. The vast majority of words in the vocabulary fall into these categories which essentially correspond to the standard word classes: noun, verb, adjective and adverb. The closed categories represent those words, which include a finite, restricted number of instances; for example the class containing conjunction or determiners in a language can not be extended. Words in open category are also called content words, while those in closed categories are called function words.

Function words make up the skeleton of the sentence. They carry a great deal of information about the syntactic structure of the sentence (Jonsen et al., 1997). The part of speech category of a function word proved useful to distinguish more categories than the standard dictionary

categories. The majority of content words not listed in the lexicon can be tagged using morphological information about suffixes (Jonsen et al., 1997; Park et al., 1996).

2.3.3 Output of Morphological Parsing

It is obvious that any analysis task is governed by the final goal it seeks to achieve. The success or failure of the analysis should also be determined from the goal it seeks to achieve. The output of morphological analysis of a language for NLP and Linguistic can be different as they have different goals to achieve.

Analysis of NL for NLP involves identifying computational characteristics of the language relevant to the application at hand, and constrained by limitations of computational effectiveness and available technologies. The analysis of NL for linguistic purposes is usually motivated by the problem of language acquisition, and identifying universal property of a language. Analysis of NL for linguistic purposes involves characterising observable property of the language and description of an entire language. Shieber (1985) Word parsers that are designed to serve the purpose of NLP should provide information that enables to determine;

- What categories are actually indicated by a given word (such as case, number agreement etc)
- An indication of lexical identity with a pointer to the syntactic and semantic properties of this lexical item and
- A morpho-syntactic representation that shows the inflectional categories conveyed by this word form

2.4. Approaches to Morphological Parsing

There are a number of approaches for morphological parsing. This section reviews few approaches that have close relevancy to the study.

2.4.1 Bottom up Approach

Bottom up starts from the given form and attempts to find a way of grouping its parts into larger structure. This approach is proposed by Key (1977). Key proposed that the procedure for analysing words essentially follows the following steps;

1. Starting from the surface form, undo the phonology by inverting the rules. Construct a set containing all of the phonologically possible underlying representations that could have given rise to the form being analysed.
2. Set if (at least one of) these representation could correspond to a well-formed sequence of lexically listed elements.

For example if such system is supplied with the Amharic word ‘kfäčylat’ ክፈረሃት (you (3psf) open (something) for her). The parser first attempts to find the way this word is formed by undoing the word formation process i.e. from the surface form it will attempt to find the underlying form, as indicated in the following example.

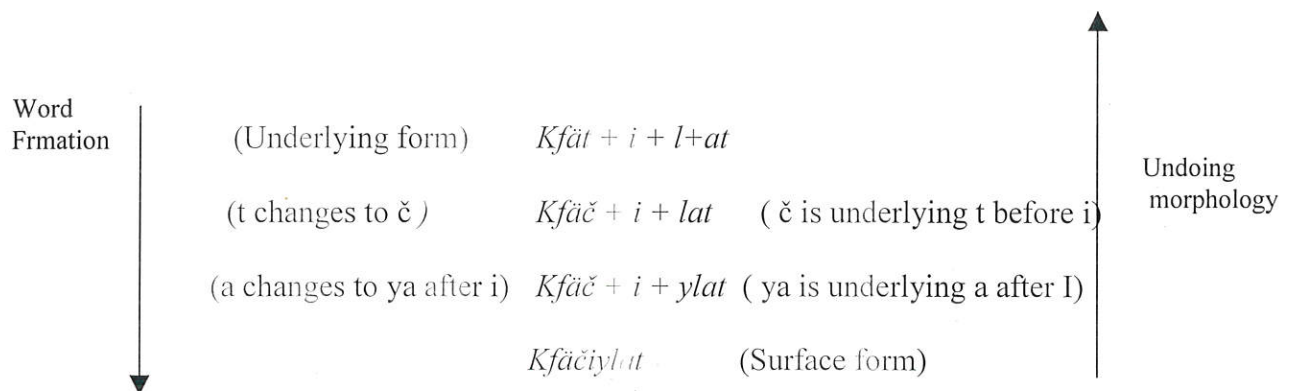


Figure 2.3 Surface and underlying forms of 'Kfäčiylat'

A major problem with this approach arises if the phonology of the language involves a non-trivial, amount of neutralisation (through assimilation, detentions, insertions, etc)(Anderson, 1988). Anderson notes that, when that happens, it is necessary to calculate all of the possible combinations of alternatives allowed by various rules, which may be individually large when neutralisations are involved and whose product grows exponentially as the amount of significant rule interaction (order) increases that may make the approach impractical.

2.4.2 Top-to-Bottom Approach

As referred to by Andeson (1988), the Top-to-Bottom approach is developed by Hunkamer (1983) based on a representation of the phonological system of Turkish (its morphotactics) in the form of finite state-machine. The parser starts at the left edge of the word, and attempts to identify as much as possible the word with some lexical stem. Once a stem is found, and assuming that it does not exhaust the word, the finite state machine representing the morphology is consulted to see what materials could come next. Each of this is tried in succession, by

attaching it to the stem and then applying the phonological rules of the language to the result. At any point, when none of the morphologically acceptable possibilities is confirmed, the system backtracks to the first previous choice point to attempt an alternative path. If there is no such point a failure is reported.

This approach works in a deterministic fashion. It relies on the fact that at any point in the analysis, the possibilities for the identity of the next piece of material in a word are severely constrained by the material that has already been identified. For example, if it finds a noun stem it looks only those suffixes, which can be attached to the noun. Since this system operates in a completely determinate fashion, i.e. in a top-to-bottom direction, there is no combination problem deriving from a multiplicative interaction of alternatives as in bottom-to-top system.

This system has been found efficiently working. However, as noted by Anderson (1988), the efficiency is due in essential ways to some accidental properties of Turkish that do not allow its direct extension to a full range of other languages. It is based on the assumption that concatenating of prefixes, root and suffixes forms words (Golding and Thompson, 1985). However, this assumption failed to include changes inside words like infixation and reduplication. Therefore some extension or revision will be necessary to extend the approach for language that involves extensive infixation and reduplication.

2.4.3 Two Level Model of Morphology

The two level model was suggested by Koskeniemi (1983). This model is developed for Finnish, which is an agglutinative language in which words can have thousands of inflected forms. As

indicated by Antworth (1994) NLP systems for Finnish could get no where without first parsing its morphology.

This model is based on morphotactics and morphophonemics properties of the language. Morphotactics enumerates the inventory of morphemes and specifies in what order they occur; while morphophonemics enumerates accounts for alternative forms or spellings of morphemes according to the phonological context in which they occur. For example, the Amharic word 'kfäči' ክፈረሻ is analysed morphotactically as the stem 'kfät' followed by the suffix '-i'. However the addition of the suffix '-i' apparently causes palatalization of the final 't' of 'kfät', thus 'kfät' and 'kfäč' are allomorphs or alternate forms of the same morphemes. This model considers letter for letter correspondence between its lexical (surface) and underlying form. For example the Amharic word 'kfäči', in the two level Model can be analysed as;

Lexical form k f ä t + i

Surface form: k f ä č + i

Koskenniemi's two-level model comprises two components:

- a rule component, which contains phonological rules represented as finite state devices, and
- a lexical component, or lexicon which lists lexical items (individual words and morphemes) in their underlying forms, and encodes morphotactic constraints.

One of the implementation which uses the two level model was the KIMMO, a general

morphological parser (Korttunen, 1983). The main components of the KIMMO parser are shown in the following figure;

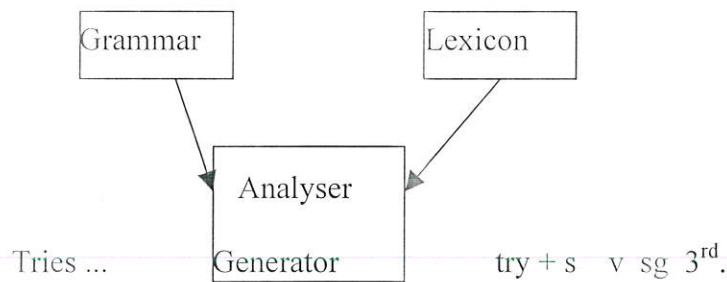


Figure 2.4 Components of the Kimmo Parser

The grammar component consists of two level rules that account for regular phonological or orthographic alternations, such as ‘try’ versus ‘tri’. Second, the lexicon listed all morphemes (stems and affixes) in their lexical form and specified morphotactic constraints. Using these two components, the processor performs the analysis and generation tasks. The generator accepts as input a lexical form and returns the surface form. The recogniser accept as input a surface form and return an underlying form divided into morphemes, plus a gloss string, as in the following example.

Example:

a. Recogniser:

Input: tries

Output: try + 3rd person, singular, present tense, verb

b. Generator:

Input: big er

Output: bigger.

The most important technical feature of the KIMMO parser is that, morphological rules are represented in the processor as automata, more specifically, as finite state transducers (FST). FST is based on the notion of finite state morphology which is based on the fact that a morpheme may have alternate realisations depending on phonological environment and the composition of a word (Kurttnen, 1983).

2.4.4 Unification Based Word Grammar

The Kimmo model tokenises a word into a sequence of tagged morphemes, but it could not directly determine the part of speech of a word. For Example; if a parser provided with word 'enlargement', it could tokenise into a sequence of morphemes as in the following example and gloss each morpheme, but it could not determine that the entire word is a noun.

Example:

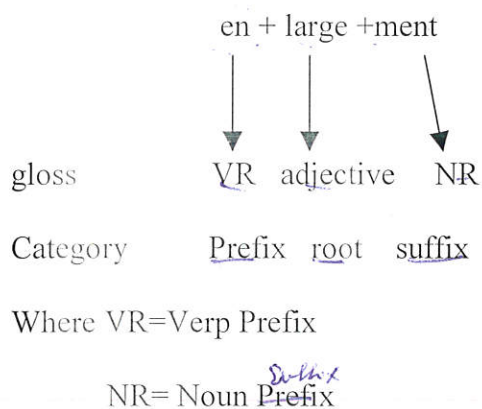
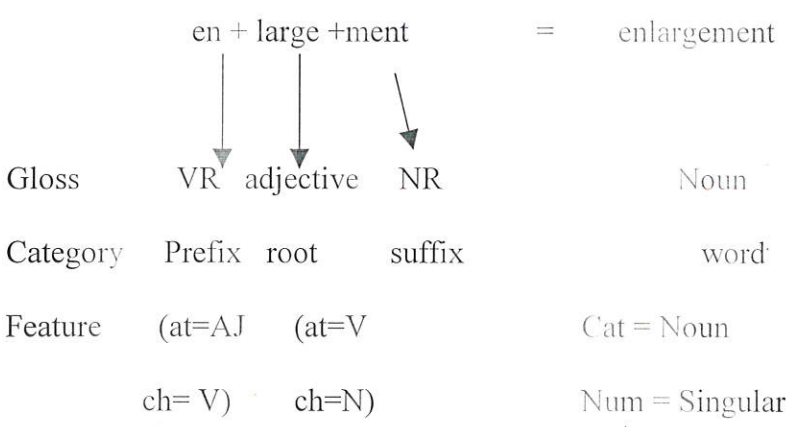


Figure 2.5 Tokens of the word 'enlargement'

Such word parser could not be adequate to act as a morphological front end to a syntactic parser (Anteworth, 1993). As a solution to this problem Anteworth (1993) has developed a word parser

that could determine the part of speech of the word based on the word grammar. This parser first tokenises a word in to a sequence of morpheme structures, which consist of a lexical form, its gloss, its category and feature the same as the above example. Then, the analysis obtained from the first step will be passed to the unification based word grammar component. This word grammar is based specifically on two specific behaviours of features of affixes. These behaviours state that there are specific categories of words (roots) to which an affix can be attached and can transform to. For example, the prefix 'en-' can be attached to an adjective and when prefixed to an adjective, the adjective becomes a verb. The example given above can be modified in the unification based word grammar as;



Where VR=Verb Prefix
 NR= Noun Prefix
 at= Attached to
 ch= Change To

Figure 2.6 Unification based Parsing for the word ‘enlargement’

In this example when ‘en-’ is prefixed to an adjective root ‘large’ it becomes a verb ‘enlarge’; the suffix ‘-ment’ has a feature of property ‘can be attached to verbs’ and ‘changes to noun’. Thus, in

the example when the suffix ‘-ment’ is attached to the verb ‘enlarge’ it will result in a noun ‘enlargement’. Therefore, the entire category of the word ‘enlargement’ becomes a noun.

CHAPTER THREE

THE STRUCTURES OF AMHARIC WORD

3.1 Introduction

This chapter discusses the structure and property of the Amharic verbs. As indicated in the first chapter this study focuses on verbs and their derivation, which covers 70 – 80% of words in the language (Wedikend, 1982). Most of the data presented in this chapter is taken from Bender and Hailu Fullas (1982), Mullen (1986), Alemayehu (1987), and Leslau (1967). Further explanation of the subject can be obtained from these sources.

3.2 The Amharic Writing System

The present writing system of Amharic is taken from Ge'ez alphabet, which was the language of literature in Ethiopia in the early time. The Amharic writing system consists of a core of 33 characters (fidel) each of which occurs in a basic form and in six other forms known as orders. The seven orders represent the different forms of a consonant. These forms are created by the combination of the consonants and vowels. The non-basic forms are derived from the basic forms by more-or-less regular modifications. For example using the consonants h, l and m the orders are;

1 st	2 nd	3 rd	4 th	5 th	6 th	7 th
ሀ	ሀ•	ሂ	ሃ	ሄ	ህ	ህ'
h ä	hu	hi	ha	he	h	ho
ለ	ለ•	ሊ	ላ	ሌ	ሎ	ሎ•
l ä	lu	li	la	le	l	lo
መ	መ•	ሚ	ማ	ሜ	ሞ	ሞ'
m ä	mu	mi	ma	me	m	mo

In addition to these there are nearly 40 others, which contain a special feature usually represent labialization. For example:

ቂ	ቂ•	ቂ	ቂ	ቂ•
q ^w ä	q ^w i	q ^w a	q ^w e	q ^w

Not including the labiovelars the alphabet has 251 letters. Each graphic symbol represents a consonant together with its vowel. The vocalic symbol can not be detached from the consonant element. Thus, the Amharic script is a syllabic rather than an alphabet (Leslau, 1967).

In Amharic writing system there is no distinction between capital and lower case letters. The individual words in a sentence are separated by two dots (:) , although this practice is now being discontinued. The end of the a sentence is marked by a square formed by four dots (::). The symbols ቶ represents a comma, while ፡ corresponded to a semicolon.

One of the deficiencies of the Amharic writing system is the lack of marking of gemination of consonants and the ambiguity between sixth-order symbols with and without vowel. For

example, 'gäna' (still, yet) and 'gänna' (Christmas) are both written in fidel as ገፍ. Similarly, እርግብ (unstretched) and እርግብ (dove) could be pronounced as 'ĩrrggɐ' or 'ĩrgb', and indeed both pronunciations are used in this case. A number of solutions have been suggested for these problems. One of the solution is to mark all geminations and sixth-order vowels by some notational device (e.g. ም for m, and ም፡ for m m)

There are 30 consonants and 7 vowels in Amharic. The table below shows the Amharic consonants with their manner of articulation.

TABLE 3.1 AMHARIC CONSONANTS

		Labials	Dentals	Palatals	Velars	Laryngeals
Stops	Voiceless	p ጥ	t ተ	č ች	k ክ	
	Voiced	b ብ	d ድ	ǵ ግ	g ግ	
	Glottalized	pʼ ጥ'	tʼ ተ'	čʼ ጭ	q ቅ	
	Rounded				kʷ ኳ, gʷ ጳ, qʷ ቈ	
Sprints	Voiceless	f ፍ	s ስ	š ሽ		
	Voiced		z ሢ	ž ሽፕ		
	Glottalized		sʼ ጽ			
	Rounded					hʷ ኳ
Nasals	Voiceless					
	Voiced	m ም	n ጥ	ṇ ጽ		
Liquid s	Voiceless					
	Voiced		l ለ, r ር			
Semivowels		w ው		y ሃ		

3.3 Vocabulary and word Shapes of Amharic

Semitic language in general make use of a system known as roots and patterns. A root is a set of consonants which has a basic "lexical" meaning. The consonants of a root are called radicals. A pattern consists of a set of vowels which are inserted among the consonants of the root. The pattern has a grammatical meaning. Since most words in such languages are derived from verbs, the consonantal skeleton of verbs are the most important (Alemayehu, 1987:67; Dawkins, 1960).

Amharic verb roots provide large numbers of verbal derivations, and in addition, they predict the noun entries of the lexicon. The Amharic radicals determine the lexical meaning of the word like other Semitic languages.

For example the radical ስብረ 'sbr' (break) in other derived noun or adjectives reappear as in the following example;

- a. säbar-i ስብረ 'one who breaks'
- b. Sbar-i ስብረ 'a fragment'
- c. Säbara ስብረ 'broken'

In the different conjugations of the verb, the core meaning remains the same due to the root, as in the following example;

* Taken from Leslau(1976,4)

- | | | |
|-----------------|------|---|
| a. 'asäbärä' | አሰበረ | 'he made somebody to break something' |
| b. 'säbabärä' | ሰበረ | 'he breaks something again and again' |
| c. 'tä-säbärä' | ተሰበረ | 'it has got broken' |
| d. 'asabärä' | አሳበረ | 'he helped in breaking something' |
| e. 'assäbabärä' | አሰበረ | 'he helped in breaking something into pieces' |

The common meaning is due to the radicals which are common to all the conjugations. The variation in meaning is due to arrangements of the radicals and vowels in the stem. Here root refer to the basic consonants which constitute the semantic core of the word while stem is a complex structure which shows additional features like aspect, signaled largely by the inclusion of one or more vowels.

3.3 Amharic Verb Classes

Amharic verbs are very regular (except few), in such a way that for all basic verbs in the language the numerous inflected forms can be predicted when the nature of the root is known. But there are different kinds of roots that create complications of different kind of forms. This complication of forms exists because forms vary depend on the nature (type) of the root. Therefore to handle these complications the root forms classified based on their nature (Bender, 1978; Dawkins, 1969). For this reason the lexicon of the Amharic verb roots shall provide the nature of the root.

Different scholars (Bender, 1978:21; Cohn, 1981:199, Habte mariam 1994) suggest different approaches to classify Amharic verbs. Their approach to classify the verbs is based on different

criteria. In this study the classification by Bender and Hailu (1978: 21-27) is taken, since it exhaustively classes all Amharic verbs based on explicit criteria. Bender (1978) has classified the Amharic verbs based on the following criteria;

1. consonantal skeleton (number of radical)

- monoliteral
- bilateral
- trilateral
- quadrilateral, and
- quinquilaterals

2. pattern of gemination of penultimate consonants as;

Type A: penultimate consonant geminates in perfect only

Type B: penultimate consonant geminates throughout the conjugation

Type C: penultimate consonant geminates in perfect and contingent only

3. Occurrence of vowels other than normal ‘ä’ ኧ

4. Identical consonants in sequence

5. Presence of w ወ, y ይ, H ሀ

6. Occurrences of initial ‘a’ አ or ‘t’ ተ in base forms

Based on these criteria Bender has found eleven major divisions, and eight subdivisions as listed below.

TABLE 3.2 MAJOR DIVISION OF AMHARIC VERBS

Major Division		Example
1.	trilateral-Type A	bsl ሰለል (cook), dñ ጽፍን (close up)
2.	trilateral – Type B	gtr ግትር (be stubborn), lmn ልምን (beg)
3.	bilateral with final ‘-a’ - አ, Type A	zgh ዝግህ (shut), wgh ውግህ (prick)
4.	bilateral with final ‘-a’ - አ, Type B	qbh ቅብህ (paint) tkh ትክህ (replace)
5.	bilateral with final ‘-ä’ - ኧ, Type A	fjh ፍጅህ (consume) qrh ቅርህ (remain)
6.	bilateral with final ‘-ä’ - ኧ, Type B	ghyh ግህይህ (burn) dhñh ድህኝህ arbitrate
7.	internally contracted verbs in ‘-a-’	bhs ብህስ (get worse) dhn ድህን (heal)
8.	internally contracted verbs in ‘-ä-’ - ኧ-, ‘-e-’ - ኡ-, and ‘-o-’ - ኡ-	b ^w z ብውዝ (be dizzy), r ^w t ርውጥ (run)
9.	quadrilaterals	brbr ብርብር (search) , brgd ብርግድ (burst open)
10.	trilateral with final ‘-a’ - አ	brth ብርትህ (be strong) fñdh ፍንድህ (burst)
11.	trilateral with final ‘-ä’ - ኧ and preceding palatal	dhyh ድህይህ (be poor) gbyh ግብይህ (buy)

TABLE 3.3 SUB-DIVISION OF AMHARIC VERBS

Subdivisions		
a.	verbs with initial ‘a-’ (አ-)	hbd ህብድ be mad
b.	verbs with ‘o’ (ኡ) in the first radical	k ^w sl ቁስል wonuded
c.	verbs with ‘o’ (ኡ) in the second radical	čk ^w l ችኩል hasten
d.	verbs with ‘o’ (ኡ) in the third radical	dnk ^w r ድንቁር be deaf
e	combination of a and c	hg ^w l ህጉል hinder
f	triradical verbs with the first two radicals are similar	brr ብርር fly
g	triradical verbs with the last two radicals are similar	ggr ግግር beak
h	verbs with ‘a’ አ in the first radical	hds ህድስ renew
i	verbs with ‘a’ አ in the second radical	dbhlq ድብህልቅ mix

Based on this classification Bender has identified 42 categories of verbs.

3.3. Conjugation of Amharic Verbs

Amharic has five verbal stems namely: perfective, imperfective, gerundive, jussive, and infinitive. Because of their regular occurrence and crucial role in the syntax of the language, these will be considered as the principal parts of the Amharic verb (Bender, 1976:54). These five verbal provide the basis for all tense, aspect and mode forms of the verb. For example the consonantal root 'sbr' ሰብር (break) will have;

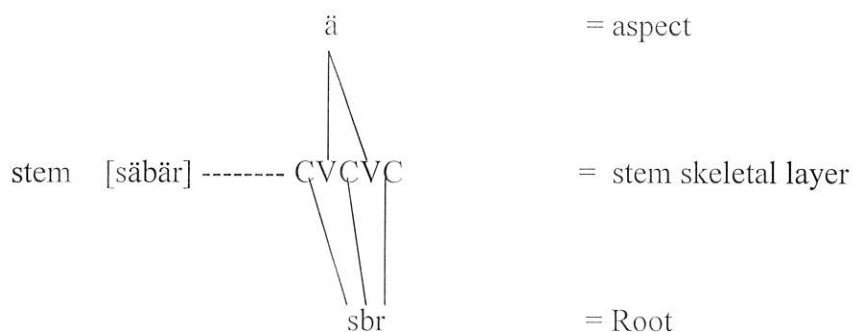
	Aspect	stem	
1.	Perfective	säbär	ሰበር
2.	Imperfective	säbr	ሰብር
3.	Gerundive	säbr	ሰብር
4.	Jussive (Imperative)	sbär	ሰበር
5.	Infinitive	säbr	ሰብር

The vowel pattern of conjugation is based on the type of the root. The vowel pattern can be predicted when the nature of the root is known. For example, the conjugations of three Amharic roots 'sbr' , 'hwq' and 'mnzr' have unique vowel pattern for each aspect in the example below.

	sbr ሰብር(break)	hwq ህወቅ (know)	mnzr ምንዝር (change)
Perfective	säbär ሰበር	awäq አወቅ	mänäzär መነዝር
Imperfective	säbr ሰብር	awq አውቅ	mänäzr መነዝር
Gerundive	säbr ሰብር	awq አውቅ	mänzr መንዝር
Jussive (Imperative)	sbär ሰበር	awq አውቅ	mänzr መንዝር
Infinitive	säbr ሰብር	awäq አወቅ	mänzär መንዝር

‘sbr’ is a root from class one (trilateral type A) verbs. This class has trilateral verbs that geminate the penultimate¹ consonant in perfect only. The second root ‘hwq’ , is in the major subdivision of type one verbs and subdivided as type ‘a’ , which has initial ‘a-’. The last one is from class nine verbs, which have quadrilateral root.

The vowel and consonant pattern of the Amharic verb stems in three layers; the layer that has the consonants; the layer that has the skeleton of the consonants and vowel pattern; and the root layer which is the consonants of the stem which can be diagrammatically represented as:



where: C = Consonant

V =Vowel

Figure 3.3 Vowel and consonant layer of ‘säbär’

3.3.1 *Derived Verbs*

The main derivations in Amharic verbs with some indication of their use in syntactico-Semantic functions involve prefixing and internal changes.

a. Prefixing

i. Prefix ‘a-’ (transitiviser)

When prefixed to an intransitive verb it make the verb transitive verb. Consider the underlined verbs in the following setence,

Example; 1. a. lj-u wha t'ät'-a ልጁ ውሃ ጠጣ

child-the water drink-(3sm)

(The child drink water)

b. lj-u saru-n wha a-t'ät'-a ልጁ ሳሩን ውሃ አጠጣ

child-the grass-the water drink-(3sm)(transitive verb)

(The child watered the grass)

2. a. Abebe msa bäla አበበ ምሳ ባላ

(Abebe eats launch)

b. Abebe lj-u-n msa a- bälä አበበ ልጁን ምሳ አ-ባላ

(Abebe feed his child a launch.)

ii. Prefix 'as-' (causative)

When a verb is prefixed with 'as-' it indicates that the subject causes the action expressed by the underlying verb.

as – säbärä አሰበረ “he made some body to break ...”

as – fälägä አስፈለገ “he made some body to search ...”

iii. Prefix 'tä -' (passive)

The prefix 'tä' changes the verb to passive.

For example; 1. a. säbärä ሰበረ “he breaks ...”

b. tä – säbärä ተ-ሰበረ “he has got broken ...”

2. a. gädälä ገደለ “ he kills.....”

b. tā – gädälä ተገደለ “he was killed.....”

iv. Prefix ‘n-’ (expressive)

It is always found with either ‘a –’ or ‘tä’ preceding, i.e. as ‘an –’ or ‘tä-’

tän-kässkäsä ተንቀሳቀሰ “he starts to move ...”

an – käbalälä አንከፃለለ “he rolled...”

b. Internal changes

The internal changes of derived verbs occur in two ways. The first one involves the insertion of vowel ‘-a-’ after the antepenult (the third from the last) radical of the root. It always occurs with ‘tä-’ prefix. Its meaning is conative extensive (reciprocal). For Example ;

tä- sabäru ተሳበሩ “they broke each other. “

tä- gadälu ተጋደሉ “they killed each other.”

If the subject is the causative for the reciprocal action the stem prefixed with ‘a-’ . For example;

a- sabäru አሳበሩ ‘ they made some things to break each other’

a- gadälu አጋደሉ ‘ they made some things to kill each other’

The second one involves duplication of penultimate (the second from the last) radical with the first of the resulting radical having vowel ‘a’. It expresses that some thing is done again and again or iteratively.

Example: tä- säbabäru	ተሰበሰቡ ‘they broke each other again and again’
tä- gädadälu	ተገደደሉ ‘ they killed each other again and again’
a- säbabäru	አሰበሰቡ ‘ they made some things break each other again and again’
a- gädadälu	አገደደሉ ‘ they made some things kill each other again and again’

The most productive of the derivations is the facative (as-), which occurs, with most verbs. The expressive prefixes are relatively rare and occur mainly with quadrilateral verbs. All verbs do not undergo these derivational processes. Verbs take or fail to take the various derivations in idiosyncratic patterns according to semantic, syntactic, and phonological considerations. Therefore, in most of the cases it is difficult to predict whether it is possible to apply a particular derivational process to a specific root. Bender and Hailu (1978) suggest the following criteria as some of the considerations to apply as a rule;

- A verb with no antepenultimate radical cannot undergo the first internal change.
- A verb with initial ‘a’ does not take the ‘a-’ prefix.

3.3.2 Inflectional Affixes of Amharic verb stems

Except for the 2nd person imperative the verb in Amharic does not stand without inflection. For example ‘sbär’ አበር (break (3psm)) can stand as a word on its own. Apart from this exception,

the minimally inflected stem must have at least a subject marker to function as a word in syntax (Alemayehu, 1987:65). To make these stems full words inflectional affixes must be added to them. For example the perfect stem ‘SäBäR’ ሰበረ, to be used in the syntax it must have an inflectional affix such as ‘-ä’ to become a word as ‘SäBäR-ä’ ሰበረ (he breaks).

One of the characteristic of Amharic word formation is inflectional affixes cannot be added until stems are formed. For example it is not correct to say ‘sbr- ä’. Together with this phonological rules (such as palatalization) must apply only when fully inflected words are derived from verbal stems. For example, kfät + -e = käfče ክፈሮ (t changed to č).

The inflectional affixes of Amharic verb stems indicate features of person, number and gender. Based on the case they indicate Amharic verbs have three types of inflectional affixes to verb stems;

1. Subject markers
2. Object markers
3. Affirmative affixes

3.3.2.1 Subject Feature Indicators

Subject feature indicators indicate the feature of the subject i.e. the features of the doer of the action. The subject feature indicator affixes are obligatory affixes. All Amharic verbs, except 2nd person singular can not function without these affixes.

Example:

1. X zaf tākäl-ä

X ገዢ ተከለ

(X has planted a tree)

2. X zaf tākäl –äč

X ብፍ ተከለች

(X has planted a tree)

The subject ‘X’ in the first example has the feature value of person = third, number = singular and gender = male. These feature values are the features of the suffix ‘– ä’ of the verb ‘tākäl –äč’. In the second example the subject ‘X’ has the feature value of ‘person = singular, number = singular and gender = female, which is the features indicated by the suffix ‘–äč’ of the verb ‘tākäl –äč’.

Based on the stems to which they can be attached, Amharic subject indicator affixes can be grouped in to three paradigms (Alemayehu, 1987). Except some affixes of the imperfective and jussive, all the remaining affixes are suffixes. The following tables present the affixes in each paradigm.

TABLE 3.4 SUBJECT MARKER AFFIXES OF THE IMPERFECTIVE/GERUNDIVE STEM

Place(s) of Affixation		Features			Example	
Prefix	Suffix	Person	number	Gender	jussive	Imperfective
ī- ‘		1	s			‘säbr ሰብር
n- ን-		1	pl		‘n-sbär ንስብር	‘n-säbr ንስብር
t- ት-		2	s	m	sbär ስብር	t-säbr ትስብር
t- ት-	-i -ከ	2	s	f	sbär-i ስበሪ	t-säbr-i ትስብሪ
t- ት-	-u -ከ	2	pl		sbär-u ስበሩ	t-säbr-u ትስብሩ
y- ይ-		3	s	m	y-sbär ይስብር	y-säbr ይስብር
t- ት-		3	s	f	t-sbär ትስብር	t-säbr ትስብር
y- ይ-	-u -ከ	3	pl		y-sbär-u ይስበሩ	y-säbr-u ይስብሩ

TABLE 3.5 SUBJECT MARKER SUFFIXES OF THE PERFECTIVE STEM

Suffix	Features			Examples
	Person	Number	Gender	
-hu/ku -ሀ/ከ	1	s		säbär-ku ሰበርኩ
-n -ን	1	pl		säbär-n ሰበርን
-h/k -ሀ/ከ	2	s	m	säbär-k ሰበርክ
-š -ሽ	2	s	f	säbär-š ሰበርሽ
-ačhu -አችሁ	2	pl		säbär-ačhu ሰበራችሁ
-ä -ኧ	3	s	m	säbär-ä ሰበረ
-äč -ኧች	3	s	f	säbär-äč ሰበረች
-u -ሁ	3	pl		säbär-u ሰበሩ

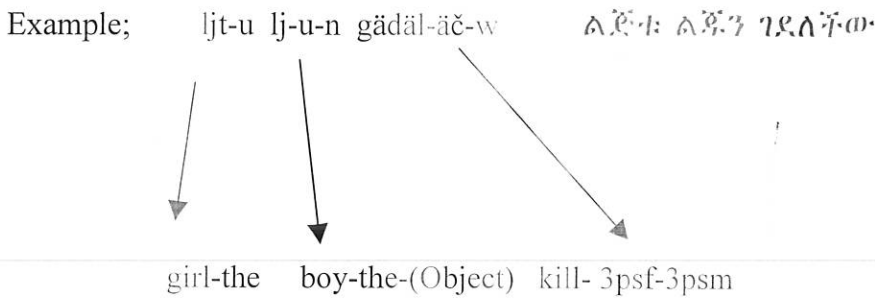
TABLE 3.6 SUBJECT MARKER AFFIXES OF THE GERUNDIVE STEM

Suffix	Features			
	Person	Number	Gender	Example
-e -ኤ	1	s		säbr-e ሰብፎ
-än -ኦን	1	pl		säbr-än ሰብረን
-äh -ኦህ	2	s	m	säbr-äh ሰብረህ
-äš... -ኦሽ	2	s	f	säbr-äš ሰብረሽ
-ačhu -አችሁ	2	pl		säbr-ačhu ሰብራችሁ
-o -ኦ	3	s	m	säbr-o ሰብሮ
-a -ኦ	3	s	f	säbr-a ሰብራ
-äw -ኦው	3	pl		säbr-äw ሰብረው

3.3.2.2 Object marker Affixes

The object marker affixes are attached to a verb after the subject marker affixes are attached to the stem. All object indicators of Amharic verbs are suffixes. They indicate the feature of the object of the verb to which they are attached to.

For example:



The subject 'ljt-u' has features, Person=3rd, Number = singular and gender = female; and the object 'lǝ-u' has features, Person=3rd, Number = singular and gender=male. These features are indicated by the suffixes '-äč' and '-w' of the verb 'gädäl-äč-w'. These suffixes are optional i.e. the stem can function in a sentence without the presence of these suffixes. One of the important features of these suffixes is that their presence indicates the presence of an inflected verb and, the person of the object markers should be different from the person of the subject.

The object marker suffixes can be preceded by the prepositional suffixes '-l' or '-b' to indicate whether the result of the action is to the merit or demerit of the object respectively. When preceded by these suffixes the object instead of receiving the action by the underlying verb directly, it becomes a dative-affected by the result of the action.

Example:

fäläg-ä-š	ፈለገሽ	“he looked for you (female)”
fäläg-ä-l-š	ፈለገልሽ	“he has looked something for (the benefit of) you (female) ”
gädäl-u-at	ገደሏት	‘they killed her’
gädäl-u-l-at	ገደሉላት	‘they killed some thing for (the benefit) of her’
gädäl-u-b-at	ገደሉባት	‘they killed (against) her’
gädadäl-u-l-at	ገደደሉባት	‘they killed something again and again against her’

The following table presents the object indicator suffixes with their features.

TABLE 3.7 OBJECT MARKER SUFFIXES

Suffix			Features		
			Person	Number	Gender
(l-) ል	-ñ	-ኝ	1	s	-
(b-) ብ	-h	-ህ/ክ	2	s	m
	-š	-ሽ	2	s	f
	-wo(-wot)	ዎ/ዎት	2	respect	-
	-w;-t	-ው/ት	3	s	m
	-at	-አት	3	s	f
	-ačäw	-አቸው	3	respect	-
	-n	-ን	1	pl	-
	-ačhu	-አችሁ	2	pl	-
	-ačäw	-አቸው	3	pl	-

3.3.2.3 The Positive Affirmative Suffixes

Affirmative suffixes can be attached to the inflected stems (verbs) These suffixes have the same

gender, number and person of the subject markers (the obligatory inflectional affixes of the stem). All the affirmative suffixes to the Amharic verb are suffixes. They can be attached after the subject indicator or after the object indicator affixes if the verb has the object indicator affixes. These suffixes are not attached to the verb if the stem is in a perfect aspect. The following table presents the affirmative suffixes together with their features.

TABLE 3.8 AFFIRMATIVE SUFFIXES

Suffix	Person	Number	Gender
- alähu -አለሁ	1	s	-
- alän -አለኝ	1	Pl	-
-aläh -አለህ	2	S	M
- alačhu -አላችሁ	2	pl	-
- aläš -አለሽ	2	S	F
- alu -አለሁ	3	Pl	-
- al -አለ	3	S	M
- aläč -አለች	3	s	F

Except a verb with the gerundive stem, all the Amharic verbs can take the negative marker affixes. The prefix ‘al-’ expresses the negative sense of the verb in Amharic. These prefixes can be strengthened by the negative marker suffix ‘-m’.

säbär-ä ሰበረ (he breaks)

al- säbär-ä-m አልሰበረም (he didn’t break)

If the suffix ‘-m’ is attached to a word without the ‘-al’ or with any prefix before ‘-al’ , it functions as a conjunction .

Positive affirmative suffixes and negative markers can not come in the same word, i.e. the presence of the negative markers excludes the positive markers and vice versa. For example it is not possible to say ‘ al- säbr-äw-al’.

3.3.2 Combination of Suffixes and Suffix Orders of Amharic Verb

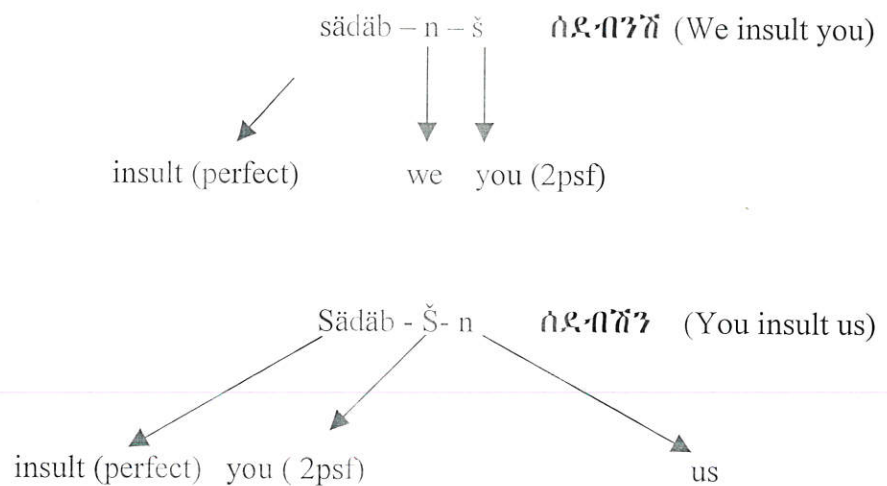
The verb should have one of the obligatory suffixes and any one of or a combination of the optional. The following table summarises the possible obligatory and optional Suffixes of Amharic verbs.

TABLE 3.9 ORDERS SUFFIXES TO THE AMHARIC VERB

No	Obligatory			Optional			
	IM	PE	JE	S1	S2	S3	S4
1	-i	-hu	-e	b	-ñ	-alähu	m
2	-u	-n	-än	l	-n	-alän	
4	ø	-h	-äh		-h	-alächu	
5		-k	-äk		-k	-aläš	
6		-š	-äš		-š	-aläh	
7		-aču	-aču		-wo	-al	
8		-ä	-o		-wot	-aläč	
9		-u	-a		-aču		
10		-ku	-ä w		-t		
11					-w		
12					-at		
13					-ačäw		

The order of suffixes in a word can affect the grammar and/ or the meaning of the word. For example two Amharic words: sädäbnš ሰደብኝሽ (we insult you) and ሰደብኸሽ (we insult you)

have different meaning because of the order of the suffixes ‘-n’ን and ‘-š’ሽ as in the example below;



From the above discussion it is possible to see that given a verb stem the order of the suffixes to Amharic verbs can be expressed as;

X-(IM|PE|JE)– S1-S2-(S3|S4)

Where:

X= Stem

IM = subject markers in the imperfective stem

PE= subject markers in the Perfective

JE = subject markers in the Gerundive

S1 = benefactive S2= Object markers S3 = Affirmative

S4 = Negative Marker suffix

*Items in the bracket indicate selection.

For example the word: sädäb-n-š can be write as

STEM = sädäb

Subject marker in the perfect=n

Beneficative=0

Object Marker=š

Affirmative Suffix=0

Negative Marker=0

Where 0 indicate no value for this category.

A word can not be positive and negative at the same time and they can take only a single subject marker at a time. Thus, the subject markers IM, PE and JE can not come at the same time in the word; and the suffixes S3 and S4 (the negative and affirmative suffixes) also do not come at the same time in a single word. Therefore it is possible to put the order of the suffixes to Amharic verbs as;

X-S1-S2-S3-S4

Where X=stem

S1=Subject Marker

S2=Beneficative

S3=Object marker

S4=Polarity (Affirmative or negative)

Therefore the above case can be put as:

These suffixes can be combined in a number of ways. The possible combinations of the optional suffixes together with the obligatory suffixes are summarised in the following table.

TABLE 3.10 POSSIBLE COMBINATIONS OF SUFFIXES TO AMHARIC VERBS

No	Suffix combination	Example		
		IM	PE	JE
1	S1S2S3S4	t-säbr-i-b-ñ-aläš ትሰብራብኛለሽ	al-säbär-ku-l-at-m አልሰበርኩላትም	säbr-e-b-at-alähu ሰብራባታለሁ
2	S1S2S3	sbär-i-b-at ስበራባት	säbär –ku-b-at ሰበርኩላት	säbr-e-b-at ሰብራባት
3	S1S3S4	t-säbr-i-w-aläš ትሰብራዋለሽ	al-säbär-ku-at-m አልሰበርኳትም	säbr-e-at-alähu ሰብራያታለሁ
4	S1S3	t-säbr-i-at ትሰብራያት	säbär –ku-at ሰበርኳት	säbr-e-at ሰብራያት
5	S1S4	t-säbr-i-aläš ትሰብራያለሽ	al-säbär –ku-m አልሰበርኩም	säbr-e-alähu ሰብራላችሁ

Generally Muller (1986:199) Notes that the strict order of the affixation and the conjugation pattern of the Amharic verbs indicates the word formation process in Amharic has three levels;

1. Creation of stems from roots
2. produce fully inflected words (from stems as well as to derive new words from existing words, mainly by affixation of subject markers)
3. Cliticization (Affixation of other inflectional affixes to the fully inflected word) and compounding.

3.4 Prefixes To Amharic Verbs

Prefixes in Amharic are attached to the beginning of a word. With the exception of 2psm imperative and subject marker prefixes of the imperfect; prefixes can be attached to a stem only after the stem form a full word by attaching a subject marker. For example ‘slä-säbär’ ስለሰበረ is incorrect while ‘slä-säbär-ä’ ስለሰበረ ”for the reason that he breaks’ is correct.

Based on their function prefixes to Amharic verb can be classified as:

1. Clause markers
2. Negative marker
3. Subject markers of the imperfect stem
4. Derived verb prefixes

3.4.1 Clause Markers

This group includes the prefixes that are used to

1. Indirect object indicator
2. Relative clause marker
3. Reason clause
4. Time clause
5. Conditional clause

Indirect object indicator

The indirect object can be expressed by the prefix lä or läm. lä- is prefixed to the perfect, while läm is prefixed to the imperfect.

Example;

- a. čgr-u-n lä-atän- u - t gänzäb Sät-ä-ačäw ቸግሩን ላጠነት ገንዘብ ሰጣቸው

Problem-the (obj) Study -3pp-it money give-he-them

 ‘He gave money for those who have studied the problem’

b. čgrun lā-m-yatān-u-t gānzāb sāt-ā-ačw ትግሩን ለሚያጠኑት ገንዘብ ሰጣቸው

 Problem-the (obj)- Study-(imp.)-3pp-it money give-he-them

 ‘He gave money for those who will study the problem’

Relative clause marker

The relative clauses in Amharic are expressed by the prefix "yā" or "yām" attached to a verb. yā- can be prefixed to the perfect for the expression of the past, while Yām is prefixed to the imperfect for the expression of the present future.

Example;

a. Fātāna-w-n yā-Wādāq-ā Yālām ፈተናውን የወደቀ የለም

 Exam + the (relative clause)- fail (per.) there is no

 (There is no one who failed the exam)

b. Fātāna-w-n yām-yfātān-u-t māt'u ፈተናውን የሚፈተኑት መጡ

 Exam + the (relative clause)- take(imp.) arrive-they

 (those of them who are going to take the exam are arrived)

Reason Clause

Reason clause indicates why the action expressed by the main verb of the sentence is done. This clause is expressed in Amharic by using a verb with a prefix 'Slä' ስለ- .

Example:

a. säw slä-gädäl-ä police yaz-ä-w ሰው ስለገደለ ፖሊስ ያዘው

↓ ↓ ↓ ↓
person (reason)-kill-he police arrest-he-him

(He is arrested by the police man for killing a person)

b. lä-mämar släm-y-fälg sraw-n läqäq-ä ለመማር ስለሚፈልግ ስራውን ለቀቀ

↓ ↓ ↓ ↓
to-learn (reason)-he-want job-his resign-he

(He resign from his job because he wants to learn)

ia
ya
ya
na
na

Time clause

Time clause indicates the time when an action indicated by the main verb of the sentence is performed or will be performed. In Amharic these clause can be expressed by the prefix 'iskä' , 'isk' , 's', indä.

Example:

a. znab-u iskä-m-y-abara lela sra sära-n ዝናቡ እስከሚያባራ ሌላ ስራ ስራን

↓ ↓ ↓ ↓ ↓
rain-the until-(imp.)-it-stop another job do-we(per.)

(We have done another job until the rain is stop)

b. znab-u s-y-zänb äl-näbär-ku-m ዝናቡ ሲዘንብ አልነበርኩም

↓ ↓ ↓ ↓
rain-the when-it-rain not-around-I-(not)

(I was not around when the rain is raining)

c. isk-t-mäläs t'äbq-alhu እስክትመለስ እጠብቃለሁ

↓ ↓

until-you-come back wait-I

(I will wait until you come back)

d. indä-säma-h ngär-ä-n አንደሰማሕ ንገረን

as soon as-hear-you tell-you-us

(Tell us as soon as you hear)

Conditional clauses

Conditional clauses indicates an action that can occur (could occur) if some condition is full filled. This type of clauses expressed in Amharic using prefixes b,kä,bä

Example:

a. nāgä b-t-māt'a t-agāñ-ä-w-aläh ነገ ብትመጣ ታገኝዋለሕ
 ↓ ↓ ↓
 Tomorrow if-you-come you (imp)-meet-you-him-(affr.)
 (If you come by tomorrow, you can meet him)

b. nāgä kä-māt'a-h t-agāñ-ä-w-aläh ነገ ከመጣሕ ታገኝዋለሕ
 ↓ ↓ ↓
 Tomorrow if-you-come you (imp)-meet-you-him-(affr.)
 (If you come by tomorrow, you can meet him)

Negative marker

The negative marker prefix changes the sense of the underlying verb to negative. In Amharic the prefix 'al' prefixed to a verb indicate negativness. This prefix is further strengthened by the suffix '-m'.

Example:

lj-u al-mäta-m ለጄ አልመጣም
 ↓ ↓
 child-the not-come-(not) (the child is not coming)

3.5 Orders of Prefixes

Amharic prefixes have specific position in relation to each other. A word can have all the five categories of prefixes at once, but only one from each category. If all categories of prefixes are prefixed in a single at the same time, the order of the prefixes starting from the word is ; Derived verb makers, Subject markers of the imperfect, negative marker, clause makers of the imperfect and clause makers. Thus the Amharic verb with its prefixes can be put us;

P1-P2-P3-P4-P5-X

Where : X=Verb

P1=Clause makers

P2=Clause makers in the imperfect

P3=Negative makers

P4=Imperfective Subject prefixes

P5=Driven verb makers (transitive/intransitive)

The following table summarises the Amharic verb prefixes based on their order.

TABLE 3.11 AMHARIC VERB PREFIXES WITH THEIR ORDERS

No	P1	P2	P3	P4	P5
1	slä	m	al	y	a
2.	yä			t	tä
3.	lä			n	as
4	l				tän
5	indä				
6	ind				
7	bä				
8	kä				
9	s				

These prefixes can be combined in a number of ways. The possible combination of these prefixes is summarised in the following table.

TABLE 3.12 COMBINATIONS OF AMHARIC PREFIXES

No	Prefixes
1	P1
2	P1 p2 p3
3	P1 p2
4	P1 p2 p3 p4
5	P1 p2 p3 p4 p5
6	P1 p2 p3 p5
7	P1 p2 p4
8	P1 p2 p4 p5
9	P1 p2 p5
10	P1 p3
11	P1 p3 p5
12	P1 p5
13	P2
14	P2 p3
15	P2 p3 p4
16	P2 p3 p4 p5
17	P2 p3 p5
18	P2 p5
19	P3
20	P3 p4
21	P3 p4 p5
22	P3 p5
23	P4
24	P4 p5
25	P5

3.6 Derived Nouns

There are a number of nouns that can be derived from Amharic verb roots. The nouns derived from the verbs has at least the radicals of the root that carries the basic lexical meaning of the root. Such nouns are also formed from the derived verbs. This section presents the type of nouns that can be derived from Amharic verbs, and their inflectional property. Most of the data presented in this section is taken from Leslau (1965) , Hailu (1966) and Baye (1986).

The Verbal Noun

A verbal noun is formed by prefixing ‘mä-’ to the verb stem. Such a noun names the abstract action derived from the meaning of the underlying verb. For example;

Root: flg	ፍለግ
Noun: mä – fäläg	መፈለግ (to search)

Root: zll	ዝለለ (jump)
Noun: mä-zläl	መዝለለ (to jump)

Root: g ^w dh	ጉድህ (injure)
Noun: mä-gudat	መጉዳት (to injure)

The prefix mä-(m- before /-a-) can characterise the verbal nouns. The verbal noun has different forms depending on the verb class. This noun can also be formed from derived verbs. For example;

m-as-f ä l äg ማስፈለግ (to make something to search)

m-a-f ä l äg ማፈላለግ (to help in searching)

m-ä-f ä l äg መፈላለግ (to search each other)

Instrumental

An instrumental noun expresses something used as an instrument to perform the action expressed by the underlying verb. It is derived from the verbal noun by attaching the suffix /-iya/. For example;

Root: sbr	ስበር (break)
Noun: mä-sbär- <i>iya</i>	መስበሪያ (an instrument used to break)
Root: ftr	ፍጥር (create)
Noun: mä-ft'är- <i>iya</i>	መፍጠሪያ (an instrument used to create)

One of the properties of these nouns is the probability of paltalization due to the presence of 'i' at the beginning of the suffix. For example;

Root : qrt'	ቅርጥ (sharpen)
Noun: mä-qräč'a	መቅረጫ (sharpener)

The t' ጥ of the root is changed to č' ጭ in the instrumental noun. Therefore in addition to the suffix '-iya' this category of nouns can be characterised by having a penultimate palatal consonant and a vowel 'a' at the end.

Manner of doing

The manner of doing represents the way or the manner by which the action expressed by the underlying verb is performed. These nouns are derived from reduplicative stems by

prefixing ‘as-’ or ‘a-’.

Root: sbr	ሰብር (break)
Reduplicative stem: säbabär	ሰባብር (break again and again)
Noun: as-säbabär	አሰባብር (manner of breaking some thing)

Participative Noun

Participative nouns indicate an occasion or event in which two objects can participate in doing the action expressed by the underlying verb. It is formed from the root by duplicating the middle radical of the try radical verbs.

Root: lmd	ለምድ (train)
Noun: lmmd	ለምምድ ‘training’

These nouns can also be formed from other class of verbs as follow:

i. Stems with o in the first radical

Root: f ^w kr	ፋክር ‘compete’
Noun: fukkr	ፋክክር ‘Competition’

ii. Stems with initial ‘a’

Root: awq	ህወቅ (know)
Noun: twwq	ትወወቅ (Introducing each other)

iii. Stems with final a

Root: šnä	ሸኑ (accompany)
Noun: šnäṯ	ሸኑኑት (accompanying each other)

Participle

Participle indicate the agent of the action expressed by the underlying verb. The agent can be a doer of the action or the one undergoes the action.

Root:	sbr	ሰብረ
Noun:	säbari	ሰባሪ (one who breaks)
Root:	q ^w tr	ቁጥረ
Noun	qotari	ቆጣሪ

This noun characterised by the vowel -i after the last radical of the stem. Its pattern differs according to the type of verb, but in all cases it has the vowel 'a' before the last radical of the stem and the vowel 'i' after the last radical of the stem. The presence of 'i' at the end change the last radical of the root to a palatal consonant if it is a dental consonant.

Example :	Root:	lbs	ለብሰ
	Noun:	läbaš	ለባሽ
	Root:	lmd	ለምድ
	Noun:	lämaj	ለማጅ

Name of an action

The name of an action indicates the abstract fact stated by the base verb. This pattern is only possible on triradical and queadriradical verbs.

Root: sbr	ሰብረ (break)
-----------	-------------

Noun: säbära ሰበራ (breaking)

Root: flg ፍለግ (search)

Noun: fläga ፍለጋ (searching)

Root: mnzr ምንገር (exchange)

Noun: mnzära ምንገራ (exchanging)

Root: q^wt'r ቁጥር (count)

Noun: qot'r ቁጠራ (Counting)

The vowel pattern of these nouns varies according to the type of the verb. A common pattern for all is the vowel 'ä' before the last radical and a vowel 'a' after the last radical.

A part remains after an action

The noun patterns in this category indicates a small part created or obtained, due to or by the application of the action indicated by the underlying verb.

Example:

Root: sbr

Noun: sbari ሰባሪ (a small part broken from a big object)

Root: wdq

Noun: wdaki ወዳቂ (a leftover)

The pattern of vowels vary according to the type of underlying verb. Common to all is the vowel 'a' before the last radical and the vowel 'i' after the last radical.

To disambiguate it from the participle. This pattern do not usually have vowels between the 1st and 2nd radical; if there is it must be the vowel 'u'. While the participle always have one of the vowels 'ä' or 'o' in this position.

Observable property due to an action

This noun denotes an object which shows manifestation of observable property or characteristics as a result of the action expressed by the underlying verb. This property most often refers to a defect.

Root: sbr

Noun: sābara ṣḥḥ (a broken)

Root: wlq

Noun : wālaqa ṣḥḥ (a person with pulled out teeth)

This noun patterns can be used as an adjective. The common pattern for this class of nouns is the vowels 'a' before and after the last radical.

The name of what the action is

This noun expresses what the agent does by Performing the action of the underlying verb.

Root: tyq

Noun: tyaqe ጥያቄ (question)

Root: mnzr

Noun : mnzare ምንገራ (exchange)

The common pattern for this class of nouns is the Vowel 'a' before the last radical and the vowel 'e' after the last radical of the stem.

Occurrence of an action

There are only a few verbs that produce this noun pattern. The denotation or referents of this nouns are not observable apart from something which does the action or to whom the action is done.

Root: grf

Noun: grfiya ግርፊያ (beating)

The common pattern of this noun class is the suffix '-iya' to the root.

General name of an action performed

These nouns denote both abstract and concrete objects. The distinguishing pattern of this nouns is they do not have vowels. They expresses the general name of the action expressed by the underlying verb.

Root: sdb

Noun: sdb ስድብ (insult)

Root: fqr

Noun: fqr ፍቅር (Love)

General name of an action

These nouns denote mostly abstract nouns. The distinguishing pattern for this class of nouns is the suffix ‘-ät’ or ‘-at’ to the root.

Root: kbd

Noun: kbdät ክብደት (weight)

Root: brt

Noun: brtat ብርታት (Strength)

The patterns of the type of noun differ according to the root type.

3.7. Suffixes to Nouns

Nouns can be inflected for number. The plural form of a noun in Amharic is expressed by attaching the suffix ‘-oč’ to the singular noun. Amharic nouns can also attach possessive pronoun suffixes. These suffixes indicate the feature of the possessor of the object or concept expressed by the underlying noun. For example ‘Immd-e’ (my training).The following table summarises the possessive pronoun suffixes of Amharic.

TABLE 3.13 SUFFIXES TO AMHARIC NOUNS THAT INDICATES POSSESSION

Possessive pronoun	Person	Number	Gender
e	1	S	-
-ačn	1	P1	-
-k	2	S	m
-h	2	S	m
-š	2	S	f
-ačhu	2	P1	-
-u	3	P1	m
-w	3	S	m
-ua	3	S	f
-ačw	3	P1	-

the object marker suffix ‘-n ‘ and the emphasis marker suffix ‘-m’ can also be attached to nouns. the following table summarizes the inflectional suffixes of amharic nouns with their orders.

TABLE 3.14 SUFFIXES TO AMHARIC NOUNS WITH THEIR ORDER

No	S1	S2	S3	S4
1	oč	e	n	m
2		ye		
3		ačn		
4		h		
5		š		
6		ačhu		
7		u		
8		ua		
9		ačāw		
10		wo		

Where:

- S1 Plural maker
- S2 Possessive pronouns
- S3 object marker
- S4 emphasis marker

A noun can take one or a combination of these suffixes in a number of ways. The following table summarizes the possible combination of the suffixes to nouns.

TABLE 3.15 COMBINATION OF AMHARIC SUFFIXES

	Suffix	Example
1	S1	-oč
2	S1S2	-oč-e
3	S1S2S3	-oč-e-n
4	S1S2S3S4	-oč-e-n-m
5	S1S2S4	-oč-e-m
6	S1S3	-oč-n
7	S1S3S4	-oč-n-m
8	S1S4	-oč-m
9	S2	-oč
10	S2S3	-en
11	S2S3S4	-enm
12	S2S4	-em
13	S3	-n
14	S3S4	-nm
15	S4	-m

3.8 Prefixes to Nouns

Nouns only take the following prefixes that are similar to the first category of suffixes of verbs. However the meanings of these prefixes is different for nouns and verbs. The prefixes to nouns are usually conjunctions. For example;

bä-mämäkina by car

lä-tāmariwoč for the students

kä-agär-u from his country

Therefore words that have other categories of prefixes can not be considered as noun.

This chapter has discussed about the word formation processes in Amharic , specially about verbs and nouns derived from verbs. The next section discusses fetures of Amharic words considered in designing the word parser for Amharic words. The chapter also describes the procedures and assumption taken to deign lexicons used to parse a word.

CHAPTER 4

DESIGN OF LEXICONS FOR AMHARIC WORD PARSER

4.1 Introduction

In this chapter the word parser designed for Amharic language is discussed. The designing process is based on the morphological property of the language presented in chapter three and the assumptions and approaches discussed in chapter 2.

The first section discusses the approaches taken to design the parser and the features of the language considered in designing the parser. The next section discusses the design and structures of the different databases implemented for representing the data required by the word parser. The last section presents phonological changes that may occur in the language and analyses these properties for word parsing.

4.2 Approaches to the Design of Word Parser for Amharic Language

There are a number of algorithms designed for word parsing. However, most of these algorithms are designed for western languages, some of the language such as English have relatively simple morphological processes when compared with other languages like Amharic. In this study the approaches discussed focuses mainly on those approaches designed for morphologically complex languages.

Most of the approaches followed in designing the word parser for Amharic language is based on the assumptions of Unification Based Word Grammar Approach (Antiwort,1994) and the Two

Level Model* (Karttunen,1983) approaches. The Two Level Model uses a morphological property of the language. This approach uses the type and order of the morphemes and the possible changes that take place during word formation. This approach is designed for handling languages, which involve intensive inflectional and derivational processes. The Unification Word Grammar approach is based entirely on the Two-Level model. In addition, this approach uses the categorial property of the affixes in a unified manner.

In Amharic even though there are specific suffixes for verbs and nouns, there are some suffixes with the same spelling but can be attached to different categories. These type of suffixes create ambiguity in categorising a word using suffixes. The ambiguities further increase when concatenating suffixes. Therefore in this study in order to predict the categories of words based on their suffixes the concatenated suffixes are classified as;

- Suffix combinations applied to only verbs
- Suffix combination applied to only nominals, and
- Suffix combinations that can be attached to both categories.

Before the parser decides the category using the suffix it must check for the validity of the stem. Therefore once the suffixes are found, it checks deterministically whether the remaining lexical component is a well formed one using the following rules;

1. If the suffix found has a category of verb check the stem only for being a verbal stem
2. If the suffix found has a category of noun check the stem only for being a well formed noun
3. If the suffix found has ambiguous category

* Discussed in chapter 2 section 2.4.3 and 2.4.4

- First check the stem for being a verb
- If not verb check it for being a derived noun

This approach minimises the parsing process by avoiding unnecessary attempts.

To handle phonological variations occurred during word formation a mechanism is designed which is based on the two-level rule approach. Before a word is parsed, a pre-processing is made in order to bring different phonological (spelling) variants to the same. Unlike the two level rules discussed in chapter two, in this study these rules are used only for recognising words. The vowel patterns of Amharic are one of the properties of the language that can give substantial information about the word form. Therefore the word parser designed in this study also uses this property of the language for parsing a word.

Generally in this study to design the parser the properties of the language which includes: the categories and arrangement of affixes; the root and the vowel patterns of the stem; and phonological properties of the language; are used. Therefore the activities performed to design the parser are;

- Designing the root lexicon
- Designing the affix database
- Ordering the phonological rule

These activities are discussed separately in the following sections.

4.3. Designing the Lexicon

Many researchers (Golding, 1985; Anteworth, 1995; Karttunen, 1983; Güngördü, 1995) suggested that highly inflected languages should not store numerous inflected forms of a word in the lexicons of NLS. Storing these forms drastically inflates the size of the lexicon. Therefore, the lexicons of NLS for such languages should have a list of root forms, together with rules to recognise and generate the inflected variant forms. Similarly, since Amharic is one of the highly inflected languages that a basic form have a number of inflected forms, the above recommendation is also true for Amharic.

As indicated in the previous chapter, most of the words in Amharic are either verbs or derived from verbs. The basic lexical element of Amharic verbs is the root. The root has one or more consonants. Stems are formed from the root by inserting vowel patterns. Thus, most of the words in the language can be reduced to consonant and vowel pattern. The vowel patterns are not the same for each root. The type of root determines the structure of the stem. Therefore, the type of the root should be known to determine the structure of the stem and other forms.

In this study, a lexicon that enables the word parser to decide the well-formed stem and its category is designed. The design is based on the classification of Amharic verbs suggested by Bender (1978)*. The lexicon has the root of the Amharic verbs and syntactic and semantic information about the root. Each root in the lexicon has the following attributes;

- **NAME:** String of characters, which are made up of two or more consonants
- **CLASS:** The class to which the root belongs; which can be 1-11

- **SUBCLASS-1:** The subcategory the root belongs, which may have one of the

Values

- subtype a
- subtype b
- subtype c
- subtype d
- subtype f
- sub-type g
- sub-type h
- sub-type i

- **SUBCLASS-2:** The subcategory the root belongs, which may have one of the value

- subtype b
- subtype c
- subtype d
- subtype f
- subtype g
- subtype h

TRANSITIVITY: This indicates whether the verb is transitive or not, its value can be

1. Transitive
2. Intransitive

* Discussion in section 3.3

DERIVED TYPE_a whether the root permit the derivation of stems with prefix -a

DERIVED TYPE_{as} whether the root permit the derivation of stems with prefix -as DERIVED

TYPE_t whether the root permit the derivation of stems with prefix _t

Based on this analysis a database is designed which have the following structure.

TABLE 4.1 ATTRIBUTES AND DATA TYPES OF THE LEXICON.

Attribute	Type	Remark
Name	Character	A maximum of 7 characters, and can not be empty
Class	Integer	From 1 to 11 and can not be empty
Subclass-1	Integer	From 1 to 8, for a to h orderly. It can be empty
Subclass-2	Integer	From 2 to 8, for b to h orderly. It can be empty
Transitivity	Boolean	0 or 1
Derived type-a	Boolean	0 or 1
Derived type-as	Boolean	0 or 1
Derived type-t	Boolean	0 or 1

The parser does not use all the attributes in the table. But these attributes must be there for further processing.

4.4 Designing the Affix Database

The affixes carry different types of syntactical and semantic information, which help in interpreting different forms of a word. In Amharic once stems are formed from the roots other forms can be formed by affixation and infixation. The features of affixes considered in designing the parser in this study are:

- Except for the second person singular imperative, the Amharic verb stem should have at least a subject marker suffix to function as a word
- Affixes can be grouped based on the type of information they will contribute to the word to which they are attached
- Affixes that indicate the same category of information will have a specific position in relation to other affixes
- Affixes from the same category can not come at the same time in a single word
- There are some groups of affixes whose presence do not permit the occurrence of other affixes in the same word, and
- Some Affixes can have different interpretations based on the category of the word to which they are attached.

In this study to get the necessary information the following files are created:

- Non concatenated suffix file (basic forms) ✓
- Concatenated suffix file ✓
- Concatenated prefix file ✓

The design and the assumptions considered to design these files are discussed below.

4.4.1 Designing the Suffix Databases

The position of the suffixes indicates the type of information they have when attached to other words while the value of the attributes of the suffix give the information expressed (contributed) by the presence of the suffix in the word. Therefore, the data structure to represent the information expressed by suffixes should have a mechanism to hold these values. Suffixes can

be concatenated each other in a number ways based on the rules of the language. The presence or absence of other suffixes can determine the position of the suffix in relation to the stem. Therefore it is important to see suffixes in two forms as;

- Suffixes in non-concatenated form (basic suffixes), and
- Suffixes in concatenated form

The two forms are discussed below.

4.4.1.1 Suffixes in non-concatenated form

Suffixes in non-concatenated forms have a name and a position, which identify one from the other. The values of the attributes of suffixes indicate the semantic and syntactic information that the suffixes contribute to the word to which they are attached.

Attributes of suffixes to verbs

Name = String of one or more Character(s)

Position = The position the suffix holds in relation to the stem when all possible suffixes are present. The position indicates the type of the information the suffix contains.

The values of the position can be any one of the following;

- ① Subject marker
2. Beneficative
3. Object marker
- ④ Polarity

A suffix must have one of the four values. There is no suffix with out these values.

Tense = Indicates the aspect represented by the suffix, this can have a value of:

1. Not Applicable
2. Perfective
3. Imperfective
4. Gerundive
5. Not perfect

This attribute is applicable only for suffixes at position one and four. 'Not perfect' shows that the aspect indicated by the underlying verb is not perfect.

Person = This is an attribute that indicates the person represented by the underlying suffix. It can have any of the following values:

1. Not applicable
2. 1st person
3. 2nd person
4. 3rd person

This attribute is not applicable for suffixes at position two and for the negative marker at position four.

Number = This attribute indicates the number indicated by the suffix. Its value can be any of the following;

1. Not applicable
2. Singular

3. Plural

This attribute is not applicable for suffixes at position 2 and for the negative marker at position 4.

Gender = This attribute shows the gender indicated by the suffix. Its value can be any one of the following

1. Not Applicable
2. Male
3. Female

This attribute is not applicable for suffixes at position 2 and for the negative marker at position 4.

Polarity = This attribute indicates positiveness or negativeness of the sense of the verb.

Its value can be:

1. Not Applicable
2. Positive
3. Negative

This attribute is only applicable for suffixes at position 2 and 4.

CATAGORY: This attribute indicates the lexical category of the word to which this suffix will be attached. Its value can be:

1. Verb
2. Noun

Attributes of Suffixes to Nouns

Suffixes that can be attached to nouns in Amharic Language have four categories. These can be in one of the following categories:

1. Plural marker
2. Possessive pronouns
3. Object marker
4. Conjunction

Except four suffixes at position 2, only knowing their presence or absence in a word is enough to get the inflectional information contributed by these affixes. For Example, the plural marker ‘-oč’ -አኝ in the word ‘tāmari-oč’ ተማሪዎች (Students) indicates the word is a plural noun, while its absence indicates the word is singular. Therefore the relevant attributes for these suffixes among the attributes discussed above for verb suffixes includes; Name, Position, and Category. The remaining attributes are not applicable for these suffixes. These suffixes can be interpreted as:

Plural marker: its absence indicates that the noun is singular

Possessive indicators: have the attributes Person, Number, Gender

Object marker: their presence indicates that the noun is the Object of the action underlying the verb in the same clause.

Emphasis marker: its presence indicate the presence of a conjunction.

The Possessive Pronouns have the attributes of Name, Position, Gender, Number, Person, and category. The remaining attributes discussed for verb suffixes are not applicable for these suffixes. This basic suffixes concatenated in a number of ways. The attributes of the concatenated suffixes are discussed below.

4.4.1.2 Suffixes in concatenated form

Amharic words form a number of inflected forms by concatenating the suffixes at different positions. The suffixes out of which the concatenated suffix is formed must be identified to get the correct syntactic and semantic information from the concatenated suffixes. Therefore the database handling this should have a mechanism to represent this information. In this study the following attributes of concatenated suffixes are identified to represent the information contained by them.

Name: String of characters that may have one or more concatenated suffixes.

Category: The category to which the suffix is attached. This can be

1. Verb
2. Noun
3. Ambiguous

Position 1: String of characters at position one. Its value is either empty or a name from the subject marker suffixes

Position 2: String of characters at position two. Its value is either empty or a name from the benefactive suffixes.

Position 3: String of character at position three. Its value is either empty or a name from the object marker affixes.

Position 4: String of character at position four. Its value is either empty or a name from the polarity indicator affixes.

The information contained by both the concatenated and non-concatenated suffixes is crucial for word parsing. Based on the above analysis a database that can hold this information is designed.

4.4.2.2 *Generation of the Suffix Database*

Based on the analysis two types of suffix files are created. The following steps are followed to generate these file:

1. Files that contain all the Amharic verb suffixes and noun suffixes (Listed in table 3.9 and table 3.15) are created separately. Each suffix is represented in the database based on the following structure:

TABLE 4.2 STRUCTURE OF THE NON-CONCATNATED SUFFIX FILE

Attribute	Data Type	Length
Name	Character	5
Position	Integer	2
Category	Integer	2
Person	Integer	2
Number	Integer	2
Gender	Integer	2
Tense	Integer	2
Polarity	Integer	2

2. Other files are created for nouns and verbs separately by concatenating the suffixes based on the possible combination patterns as shown in tables 3.10 and table 15 respectively. Each Concatenated suffix has the following structure.

TABLE 4.3 ATTRIBUTES OF THE CONCATNATED SUFFIX FILE AND THEIR DATA TYPE

Attribute	Data Type	Length
Name	Character	20
Category	Integer	2
Position-1	Character	5
Position-2	Character	5
Position-3	Character	5
Position-4	Character	5

A category value of verb is given for all combined suffixes that are from verbs and Noun for those combinations from noun suffixes. For example, the suffixes ‘-Kulačhu’ ከ-ላችሁ as in the word ‘gädäl-ku-l-ačhu’ ገደል-ከ-ል-ላችሁ (I killed for (the benefit of) you) is formed by concatenating suffixes at Position-1, Position-2, Position-3; i.e. it has suffixes that indicate subject marker, benefactive marker, and object marker respectively, and therefore this concatenated suffix is represented in the database as:

Name: Kulačhu
 Position 1: Ku
 Position 2: l
 Position 3: ačhu
 Position 4: “ “

Category: Verb

Then the two files are compared for similar spellings. Suffixes found with similar spellings but in different categories are given the category value of Ambiguous . For example, the suffix ‘-ačhu-n’ as in ‘sädäb-ačhu-n’ ሰደብ-አችሁ-ን (you insult us) ‘särg-ačhu-n’ ሰርግ-አችሁ-ን (your wedding). In the first case, it is attached to a verb stem and in the second one, it is attached to a noun. The structure of this suffix in two cases is as shown.

‘- ačhun’ as in ‘ sädäb-ačhu-n’		‘-ačhun’ as in ‘särg-ačhu-n’	
Name:	ačhun	Name:	ačhun
Position 1:	ačhu	Position 1:	“ ”
Position 2:	“ ”	Position 2:	ačhu
Position 3:		Position 3:	n
Position 4:	n	Position 4:	“ ”
Category:	Verb	Category:	Noun

Figure 4.1 The suffix ‘ - ačhun’ in different conditions

Therefore, it is not possible to determine the possible category of the word using ‘-ačhun’. Thus, such types of suffixes in the database are changed to Ambiguous. Finally the two files are merged and arranged in decreasing order of string length of the suffixes so that to make them ready for longest match search

Besides the possible concatenation patterns of the suffixes to verb the following rules of the language are considered;

- The person of the subject markers should be different from the person of the object marker;
- Subject markers of the perfective stem (Suffixes with Tense=Perfective) do not combined with the affirmative suffixes; and
- The person, number, and gender of the affirmative suffixes should be the same with the person, number and gender of the subject markers.

These considerations can decrease the size of the file, and avoids incorrect and accidental matches.

4.4.2. Designing the Prefix Database

Based on the type of information they contain, Amharic has five categories of prefixes. Prefixes that indicate the same category of information always have the same position in relation to the stem and other prefixes. Prefixes in Amharic language can be concatenated in a number of ways. Therefore, similar to suffixes, prefixes can be treated in two forms separately as:

- Prefixes in non concatenated form (basic prefixes)
- Prefixes in concatenated form

4.4.2.1 Prefixes in non concatenated form

The name and position of the prefix is enough to decide the information the prefix can contribute to the meaning of the word. Prefixes in non-concatenated form have the following attributes:

NAME: a string of one or more characters

POSITION: The position the prefix has when the maximum number of prefixes are present in the same word. The value of the position can be

- 1 Clause marker
- 2 Clause marker in the imperfect
- 3 Negative marker
- 4 Subject markers in the imperfect
- 5 Derived stem marker

All prefixes can be attached to verbs, while Nouns only take prefixes at position one. Therefore, it is possible to conclude that if a word has prefixes other than the one at position one, the word is from the category of verbs. Prefixes from the same category can not be attached to the word at the same time.

4.4.2.2 Concatenated prefixes

Concatenated prefixes have one or more non-concatenated prefixes. The type of the prefixes out of which a concatenated prefix is made of, affects the syntactic and semantic meaning of the underlying word. Therefore identifying the prefixes which form the concatenated prefix helps to explicate this meaning. Thus, in this study the following attributes are identified to represent the information contained by a concatenated prefix:

NAME: String of characters made of one or more concatenated prefixes

Position 1: String of characters, its value can be either empty or name of a prefix from Clause marker prefixes

- Position 2: String of characters, its value can be either empty or name of a prefix from Clause markers in the Imperfective
- Position 3: String of characters, its value can be either empty or name of a prefix from Negative marker prefixes
- Position 4: String of characters, its value can be either empty or name of a prefix from Subject markers in the Imperfect prefixes
- Position 5: String of characters, its value can be either empty or name of a prefix from Derived verb marker prefixes

4.4.2.3 Generating the Prefix Database

Based on the above analysis a prefix database is created by concatenating the prefixes based on the permissible combination enumerated on page 57 in table 9. The file created has the following structure.

TABLE 4.4 STRUCTURE OF THE CONCATENATED SUFFIX FILE

Attribute	Type	Length
Name	Character	20
Position 1	Character	4
Position 2	Character	4
Position 3	Character	4
Position 4	Character	4
Position 5	Character	4

This structure helps to tokenize concatenated prefixes to their basic forms. For example the prefix ‘Slämal-’ as in the word ‘Slämalwädat’ ስለማልወዳት (for the reason that I don't like her) is represented in the database as ;

Name:	Slämal
Position-1	Slä
Position-2	m
Position-3	al
Position-4	“ ”
Position-5	“ ”

Therefore, ‘Slämal-’ is tokenized as a concatenated prefix made up of three prefixes, Slä+m+al which fill the position 1, 2, 3 respectively. The presence or absence of a prefix in a certain position has a semantic implication. For instance in the previous example the presence of a prefix (being not empty) at position 1, 2 and 3 indicates;

1. The word is a clause
2. The word has an imperfect verb stem
3. The word has a negative sense

The databases designed in this section are used by the word parser to decide the inflectional and derivational categories of words.

When affixes are concatenated each other or when they attached to words, they may change some of their spellings. Therefore pre-processing must be done on different forms before using this database. The following section discusses the phonological changes that

may occur during word formation.

4.5 Phonological Properties Used

When words are formed by concatenation, phonological changes that may alter the sounds of the original can be occurred. To determine the basic forms of the word, phonological changes that have been taken place during word formation should be reversed accordingly. As the objective of this study is on parsing words in the text, only processes that cause changes in grapheme are considered.

4.5.1 Vowel Changes

Change of vowel can occur when two vowels are come in sequence. The following table summarises the possible change of vowels that can possibly occur in Amharic Language;

TABLE 4.4 VOWEL CHANGES

	a	e	i	o	u	ä
a	a	aye	ay	awo	aw	a
e	eya	eye	ey	ewo	ew	ä
i	iya	iye	i	iwo	iw	iya
o	owa	oye	oy	owo	ow	o
u	uwa	uye		uwo	uw	wa
ä	a	äye		o	äw	ä

The process that changes $a + a \rightarrow a$ and $\ddot{a} + a \rightarrow a$ creates ambiguity in finding its underlying form. This ambiguity create difficulty specially in concatenated suffixes. These ambiguities includes;

* The table is taken from Bender (1978:39)

a	+	ačhu	→	ačhu
a	+	at	→	at
a	+	ač äw	→	ačäw
a	+	aläc	→	aläc
ä	+	ačhu	→	ačhu
ä	+	at	→	at
ä	+	ačäw	→	ačäw

These processes cause ambiguities during word parsing. For example, sädäbačhu ሰደባችሁ can have the meaning of ‘he insult you’ or ‘you insult...’. These ambiguities can be solved using syntactic information. For example;

lgu ‘nantän säd äbačhu ልጁ እናንተን ሰደባችሁ

‘The boy insulted you’

‘nantä lgun sädäbačhu እናንተ ልጁን ሰደባችሁ

‘ You insulted the boy’

In the first sentence as the subject of the sentence is a singular noun, the verb should have a singular subject marker therefore the suffix ‘-ačhu ’ is an underlying ‘ä + ačhu ’ In the second case as the subject of the sentence is plural , the suffix ‘ačhu’ have similar surface and underlying forms. The same ambiguity can be solved for ‘a+ačhu’ and ‘ačhu’ using syntactic information.

For example:

lgtu ‘nantän sädbačhu hedäc ልጆቹ እናንተን ሰደባችሁ ሂደኝ

‘After insulting you, she has gone away’

lgtun sädbačhu hedäčhu ለጅቱን ሰድባችሁ ሄዳችሁ

‘after insulting her, You have gone away’

The suffix ‘-ačhu’ in the first sentence is underlying ‘a+ ačhu’. The subject is indicated by the gerundive suffix ‘-a’ which have a feature of ‘3rd person singular female. The ‘-ačhu’ in the first sentence indicates the object of the sentence, which have a feature of 2nd person plural. In the second sentence the suffix ‘-ačhu’ is a subject marker, which have a feature of 2nd person plural and gerundive aspect. As the objective of the study is to parse a word using morphological information only, the parser designed does not use syntactic information to disambiguate this occurrences.

Some of the ambiguities can be minimised by considering other factors. As indicated in the previous section the Amharic suffixes concatenated orderly as <subject marker +beneficative indicators +object marker+polarity indicators> if all suffixes are found. Using this the above ambiguities can be minimised by the following considerations.

- The forms ‘a+ ačhu’ and ‘ä+ačhu’ already attach the object marker ‘-ačhu’, after this they only attach the polarity indicator suffixes.
- As the suffix ‘a’ in the ‘a+ ačhu’, has the aspect gerundive it does not take the negative marker suffix ‘-m’. Therefore it only take the affirmative suffix ‘aläč’ . And any of the two does not take it, because the affirmative suffixes should have the same person number and gender with the subject marker.
- The ‘-ä’ in the ‘ä+ačhu’ has a perfective aspect, therefore it only attach the negative marker suffix ‘- m’

Therefore it is possible to conclude the following unambiguously as

- ačhualäč = a + ačhu + aläč

- ačhu* = ačhu + * where * is any suffix or combination of suffix except suffix '-m'

Similar criteria can be used for the remaining;

'-at' can be 'a+at' or 'ä+at'

Therefore the two cases can be disambiguated by considering the aspect of the stem to which they are going to be attached. '-ä' is attached to perfect stems and '-a' to verbs with gerundive aspect; therefore.

Perfect stem + at = perfect stem + ä + at

Gerund stem + at = Gerund stem + a + at

Perfect stem + ačäw = perfect stem + ä + ačäw

Gerund stem + ačäw = Gerund stem + a + ačäw

Perfect stem + at = perfect stem + ä + at

Gerund stem + at = Gerund stem + a + at

The above discussion indicates that the ambiguities can be solved after the aspect of the stem is identified.

4.4.5.2 Palatalization,

Palatalization causes change of sound that affect the grapheme when a dental consonant is followed by the vowels 'e' and 'i' (palatal vowel). The following table shows the corresponding dental and palatal consonants in Amharic.

TABLE 4.5 DENTAL CONSONANTS WITH THE CORRESPONDING PALATAL CONSONANTS

Dentals		Palatals	
ደ	d	j	ጅ
ት	t	č	ች
ጥ	t'	č'	ጥ'
ገ	z	ž	ገገ
ስ	s	š	ሸ
ዕ	s'	č'	ጥ'
ን	n	ñ	ኸ
ል	l	y	ይ

In Amharic, the common changes due to palatalization occur in the following contexts. Where d is a dental consonant and p is a palatal consonant

1. Context one $d+ia \rightarrow d+iya \rightarrow pa$

This context has an intermediate step where, $ia \rightarrow iya$

2. Context two rule two $d+e \rightarrow pe$
3. rule three $d+i \rightarrow pi$ or $d+i \rightarrow p$

The following tables give examples of these changes with the contexts;

TABLE 4.6 PALATALIZATION DUE TO 'IA'

$d+ia \rightarrow pa$	surface form	intermediate form	Underlying form
$sia \rightarrow ša$	ገርሻ	ገርስ+ኢያ	ገርስ + ኢአ
$tia \rightarrow ča$	ዛቻ	ዛት+ኢያ	ዛት + ኢአ
$dia \rightarrow ja$	ፍርጃ	ፍርድ+ኢያ	ፍርድ + ኢአ
$zia \rightarrow ža$	ግብገ	ግብዝ+ኢያ	ግብዝ + ኢአ
$nia \rightarrow ña$	ንደኛ		ንደን+ ኢአ
$ya \rightarrow lia$	ግድያ	ግድል+ኢያ	ግድል+ ኢአ
$t'ia \rightarrow ča$	ምርጫ	ምርጥ+ኢያ	ምርጥ + ኢአ

TABLE 4.7 PALATALIZATION DUE TO 'E'

d+e→pe	Surface form	Underlying form
se→še	አርሼ	አርሰ+ አ.
te→če	ገዝቺ	ገዝቲ+ አ.
de→je	ወድጄ	ወድድ+ አ.
ze→že	ታዝዝ	ታዝዝ+ አ.
ne→ñe	ወሰኝ	ወሰን+ አ.
te→će	ገልጪ	ገልጥ+ አ.

TABLE 4.8 PALATALIZATION DUE TO 'I'

d+i→pi	Surface form	Under laying form
si→ši	ቀንሺ	ቀንሰ+ አ.
ti→či	ገምቺ	ገምቲ+ አ.
di→ji	ፍረጄ	ፍረድ+ አ.
zi→ži	ጠርዝ	ጠርዝ+ አ.
ni→ñi	ወሰኝ	ወሰን+ አ.
tī→cī	ቁረጪ	ቁረጥ+ አ.

To handle, such cases the parser pre-processes the word from the surface form to the underlying form by reversing the changes indicated. However, there are some ambiguities in cases where the root has originally palatal consonants. For example 'šāñi' ሸኝ (escort) does not have underlying 'šān+i' ሸን + አ., rather it have underlying 'šāñ+ i' (ሸኝ+አ.). Therefore, to handle such cases whenever the parser get a root with palatal consonant from the lexicon it changes the palatal consonant to the corresponding dental, and after comparison it returns the palatal one.

The change from 'ya→lia' can also create a problem, because there are words which have the characters 'y' and 'a' successively in both underlying and surface forms. For example, the surface form 'yasäbr' ያሰብር (let him made others to break) is an underlying 'y +a+säbr', if we apply the rule that changes 'ya→lia' it give as 'liasäbr' which is incorrect in this case. Therefore,

this rule must be applied with a constraint. As the two vowels, 'ia' occurs at the suffix position, and there is no probability to be occurred at the beginning of the word, this rule should be applied only when it is found at the suffix position.

Another change that may occur at morpheme boundaries is when a consonant character is followed by 'y'. For example

släm + ysäbr → slämisäbr

yäm + yasäbr → yämiyasäbr

In the above two cases, the condition of 'm' and 'y' ('m+y') results in 'mi'. Therefore, the surface 'mi' should be changed to 'my' to get the underlying form. However, there are instances where surface 'mi' will be 'underlying 'mi'. For example in the word 'täqawami' ተቃዋሚ (opposing group/object) the last characters 'mi' is also underlying 'mi'. Therefore the rule should be applied with a constraint. The only instance that this can occur in Amharic language is when a verb stem with second person singular male is prefixed with those prefixes ending with consonant this are; m, l, ind, b, s. For example;

släm+ysäbr → slämisäbr ስለሚሰበር (for the reason that he will break)

l+ysäbr → lisäbr ሊሰበር (to begin braking)

ind+ysäbr → indisäbr እንዲሰበር (inorder that he breaks)

b+ysäbr → bisäbr ቢሰበር (if he breaks)

Therefore, to avoid the ambiguity the rule specified is to use the prefix as a whole as a constraint.

slämi → slämy

li → ly

indi → indy

bi → by

These rules are applied only when the entire sequence of character is found at the beginning of the word.

In this chapter the property of the language and features that are useful to implementing word parser for Amharic language is discussed. More over, the necessary databases for parsing a word is analysed and designed. The next chapter discusses the algorithm designed and the result of the experiment based on the design and analysis presented in this chapter. The next chapter also discusses the results of the experiment on the effectiveness of the algorithm.

CHAPTER FIVE

ALGORITHM DESIGN AND THE EXPERIMENT

5.1 Introduction

This chapter presents and discusses about the algorithm designed and implemented in this study. The previous chapter presents the design of the data required by the word parser; and identifies the phonological and morphological properties of the language that can be useful for the parser. The implementation of the algorithm designed and the test results are discussed in this chapter. In the first section a discussion on how the algorithm works is presented. The experiment together with the results are discussed.

5.2 The Word Parsing Algorithm

Different from stemming algorithms for IR, which removes the affixes in order to conflate different forms of a word, stemers intended for NLP shall get the correct boundaries of stems and affixes. Unless they get the correct affix, it is hardly possible to determine the grammatical information represented by the affixes. The parser has three major processes as shown in the diagram.

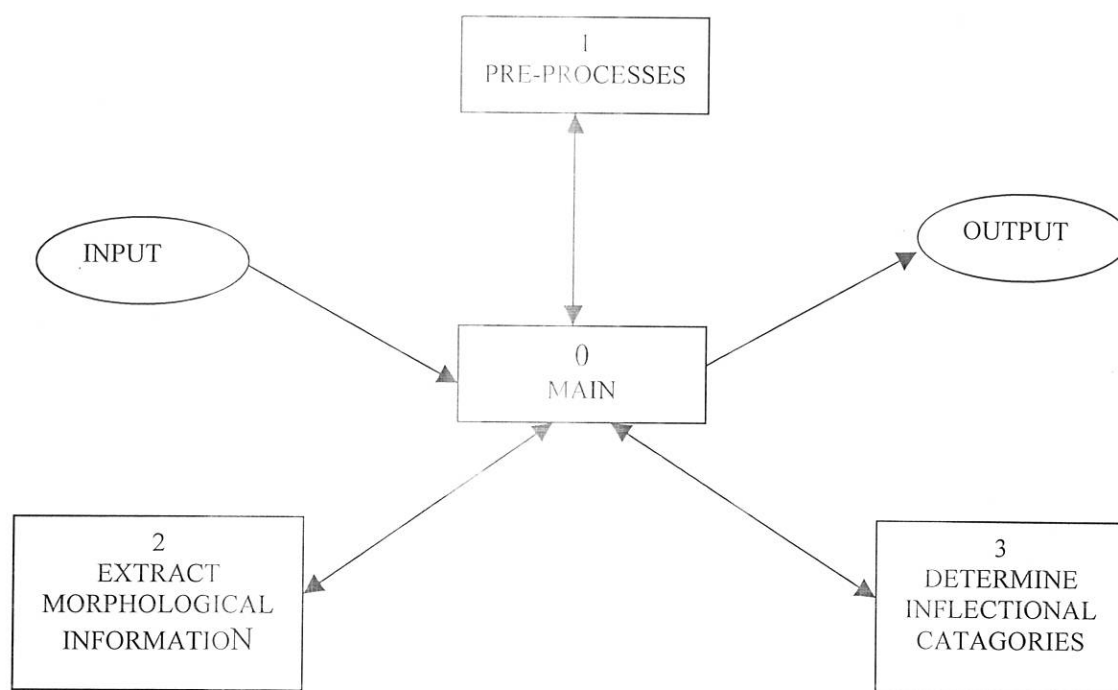


Figure 5.1 Overview Diagram of the Parser

After receiving the input word, the first process is preprocessing the input string. This process brings alternative forms (spellings) of a morpheme in to a similar form. This process works based on the phonological processes identified in the last section of the previous chapter. The second process extracts the morphological information from the word and determines lexical category of the word. The third one determines the inflectional category of the word based on the morphological information extracted by the second process. In the following sections the last two processes are discussed.

5.2.1 Extracting Morphological Information

As indicated above, the first step of the parser is pre-processing. In this stage, it changes the underlying forms to surface forms based on the phonological rules identified in the previous

chapter. The next step is extracting the morphological information, and based on that determine the lexical category of the word. This process begins by extracting all the possible suffixes and prefixes found in the word. The algorithms to extract all possible prefixes and suffixes are presented below:

Figure 5.1 The Algorithm that extracts all possible prefixes from a word

```

File      = Concatnated prefixes file;
PrfxAry  = Possible Prefix array with index i
Prefix    = Concatnated prefixes found from File

0. Start
1. Get word
   2. Set wlen= length of word-2, i=0
   3. If wlen<=0 go to 4
      3.1 Get substring (wlen,word)
      3.2 Search substring in File
      3.3 If find
            i=i+1
            PrfxAry[i]=Prefix
      3.4 wlen=wlen-1
      3.5 Goto 3
4. End

```

First it accepts a word, and assumes that the minimum length of a word is 2; it starts to check from the longest possible prefix which is equal to the total length of word minus two. The GET Substring at 3.1 extracts the first 'wlen' character from the 'word' to the 'substring'. The processes continue up to the smallest possible prefix, which is equal to the first character of the word. Whenever it gets match from the lexicon of prefixes, it put the prefix to an array of possible prefixes. Similarly, the next algorithm works for suffixes. In this case, the Get Substring gets the last 'len' characters.

Figure 5.2 The algorithm that extracts possible suffixes

```
File    = Concatnated Suffix file;
SfxAry = Possible Suffixes array with index i
Suffix  = Concatnated Suffix found from File

0. Start
1. Get word
2. Set wlen= length of word-2
3. If wlen<=0 goto 4
   3.1 Get substring (wlen,word)
   3.2 Search substring in File
   3.3 If find
       i=i+1;
       SfxAry[i]=Suffix
   3.4 wlen=wlen-1
   3.5 Goto 3
4. End
```

The result of these processes can be one of the following;

All possible suffixes and prefixes if the word has both prefix and suffix

All possible suffixes if the word has only suffixes

All possible prefixes if the word has only prefixes

0, if the word has no affix

Before the parser decides on the word class using the affix category, it must check for the validity of the part to which the affix is attached. Therefore the next process is determining the category of the word and its prefix and suffix among the possible affixes. If the first type of result is

obtained, the parser checks the remaining lexical component deterministically based on the category of the suffix using the following rules:

If the suffix found belong to the category of Verb check whether the stem is a verbal stem
If the suffix found belong to the category of noun, check whether the stem is a well formed

noun

If the suffix found has ambiguous category

First check the stem for being a verbal stem

If not verb check it for being a derived noun

This approach minimizes the parsing process by avoiding unnecessary attempts. For example, if a parser get a Suffix ‘-kulat’ as in the word ‘säbär-kulat’ ሰበረኩላት (I broak for the benefit of her), it only checks the stem ‘säbär’ for being a verb. The parser need not check the stem for being a noun because ‘-kulat’ can not be attached to a noun.

Figure 5.3 The algorithm for checking alternative affixes

```

1. Get Word ,
2. Get Possible_prefix and Possible_suffix array
3. Set W= length of word, V=False, N=False
4. For (i=0 to Total Number of Prefixes)
    i=i+1
    For (J=0 to Total Number of Suffixes)
        j=j+1
        Set P=length of prefix [i], S=length of suffix[j]
        Set A= S+P
        If (A<W)
            Set Stem = Substring (P+1, W-S, Word)
            If Suffix_Category is Not Noun V=Check_Verb(Stem);
                If v=true Go To 6
            Else If Suffix_Category is Not Verb N=Check_Noun(Stem);
                If n=true Go To 6
        Next J
    Next I
5. Return False
6. Set Suffix = Suffix[J], Prefix =Prefix[i]
    If V=True Verb_Affix_Info (prefix,Suffix)
    Else Noun_Affix_Info (Prefix,Suffix)
7. Return True

```

If the 2nd alternative is obtained the parser will decide similar to the 1st, but it only checks by removing the suffixes. If the third alternative is obtained the parser looks for the presence of prefixes in position 2,3,4 and 5, if a prefix is found in these positions the parser only checks

whether the stem is a well-formed verb stem. This is because Amharic Nouns can have a prefix at position 1 only. The algorithm for this process is presented below.

Figure 5.4 The algorithm for checking stem when the word has only prefix

```

1. Get Word ,
2. Get Possible_prefix
3. Set W= length of word, V= False, N=False

4. For (I=0 to Total Number of Prefixes)

    Set P=length of prefix [i]
    Set Stem = Substring (P+1,Word)
    If the Concatenated prefix has prefix at position one only
        N=Check_Noun(Stem);
        If N=true GoTo 6
    Else V=Check_Verb(Stem);
        If V =true Go To 6
    Next I
5. Return False
6. Set Prefix =Prefix[i]
    If V=True Verb_Prefix (prefix,Suffix)
    Else Noun_Prefix (Prefix,Suffix)
7. Return True

```

The parser tries all the possible combinations of prefixes and suffixes alternatively until it get the correct result. For Example in parsing the word ‘slāmayanqāsaqsulačhu’

ስለማንቀሳቀሱላችሁ (Because they do not move it for you), the parser generates all possible suffixes and prefixes as follows table:

Possible Affixes extracted from a single word.

Figure 5.5 Possible prefixes of the word ‘slāmayanqāsaqsulačhu’

	Prefix		Suffix
1	Slāmayan		ulačhu
2	Slāmaya		lačhu
3	Slāmay		ačhu
4	Slāma		
5	Slām		
6	Slā		
7	S		

In this particular example, the parser first gets the concatenated ‘Slāmayan-’ and the suffix ‘– ulačhu’, Then it checks the category of the suffix. Since the category of ‘– ulačhu’ is of a verb it checks the stem ‘qāsaqs’ for being a well-formed verb stem. The parser extracts the root ‘qsqs’ from ‘qāsaqs’ and searches the root lexicon to check the presence of the extracted root. When it gets the root from the lexicon, it checks whether the stem ‘qāsaqs’ is appropriate for the property of the root found from the lexicon. If it finds that the stem is correct it decides the category of the word as verb, and consequently, the parser decides the affixes removed from the stem are the correct one. This decision is made based on the assumption that only removing the correct affixes result in a correct stem. Therefore in this example the suffix ‘– ulačhu’ and the prefix ‘Slāmayan-’ are considered as correct suffix and prefix respectively.

One of the problems that needs consideration in checking the stem is the nature of affix combination. There are some alternatives of affix combination, which result in a legal root that can be found in the lexicon, although they are not the correct one. To reduce this probability of

error the parser looks whether the stem is appropriate for the type of root identified. In the figure below, for example, before the parser get the correct root of the word ‘Slämäzägäbulačhu’ ስለመዘገቡላችሁ (for the reason that they enlisted for you), which is ‘MZGB’ ምዝገባ, it founds a legal root ‘ZGB’ ዝግብ ‘Recording an event’ at the first and second attempt. In the example the stem of the root ‘ZGB’ starts with ‘ä’ i.e ‘äzägäb’, but ‘ZGB’ is a root from the category of the roots which do not take a vowel at the beginning of their stems. Therefore the parser escapes this alternative and looks for another combinations. The same is true for the next two alternatives at row 1 and column 3 (1,3); and at row 1 and column 4. The parser get the correct root ‘MZGB’ with the possibly correct stem ‘mäzägäb’ at row 2 and column 1.

Figure 5.6 Alternative affix combination results in legal root but wrong stem

		1		2		3	
	Suffix→	ulačhu		lačhu		ačhu	
	Prefix	Stem	root	Stem	root	stem	root
1	Släm	äzägäb	zgb	äzägäbu	zgb	äzägäbula	zgb1
2	Slä	mäzägäb	mzgb				

In another instance, the parser can get the reverse. For example in parsing the word ‘Slämazälačhu’ ስለማዘላችሁ (for you have carry it), the parser get a legal root at (1,1) but the category of the root requires a stem starting with the ‘a’. If the parser decides with the first attempt, the suffix that found will change the correct interpretation of the word.

Figure 5.7 Wrong stem for sub type ‘a’ roots

		1	
	Suffix→	ačhu	
	Prefix	stem	root
1	Släma	zäl	zl
2	Släm	azäl	zl

To handle such cases the parser uses the following property of roots and stems;

- The first character of the stem is either a consonant or a vowel a;

All verbal stems except those derived from a root with sub type 'a' start with consonants.

Stems formed from a root with sub type 'a' starts with 'a'

This constraint can be applied in the algorithm using the following rule.

If the stem starts with vowel other than 'a' return false

If the stem starts with 'a', it can not be a correct stems for roots which do not have sub type 'a'.

If the stem starts with consonant, compare it with only roots that don't have sub type 'a'.

Alternatively to the above rules, all verb stems that starts with 'a' represented in the lexicon by roots starting with 'h', for the reason that in Amharic when 'a' is used as a consonant it can be used alternatively with 'h'. For example, 'hagär' ሃገር or 'agär' አገር (country); 'hazäneta' ሃሕሬታ or 'azäneta' ለሕረታ (sadness).

Therefore whenever 'a' is found at the beginning of a stem it should be changed to 'h', before removing the vowels to get the root. The same is true for verb roots which ends with 'a' or 'ä'. Such types of verb roots are found in major divisions 3,4,5,6,10 and 11, according to Amharic verb classification adopted in this study. In the root lexicon the last radical of these roots is represented by 'h'. Therefore whenever 'a' or 'ä' is found at the end of a verb stem it should be changed to 'h'. However verb stems which ends with 'ä' can lose 'ä' when attach a suffix with

initial vowel. Therefore in order to get a correct match, whenever a search is failed we should try again by attaching 'h' at the end of a root and do the root search again.

Another possible problem is a root that does not carry all the radicals in different forms of a word. Roots extracted from stems of such type may not have all the radicals of the corresponding root. For example, the correct root of the word 'Slämzälačäw' ስለፖሰላቸው (for the reason that I escape them) is 'ZLL'. Nevertheless, from the stem we can only get the root 'ZL'. In another instance, they can have all the radicals in a word. For example from 'azälälut' አሳለሉት (They made him to jump), we get root 'ZLL'. Therefore, the parser should have a mechanism to handle all cases. Such type of root categorised in the lexicon as subtype 'g'. Therefore, whenever the parser encounters a root with sub type 'g' it applies the following rule.

*If previous comparison failed and if the root from the lexicon have subcategory g,
double the last radical of the root from stem and do comparison again*

The algorithm below implements all this constraints to check the verb stem. This algorithm receives 'stem' from modules that check the correctness of the verb stem. If the stem starts with 'a', it changes the 'a' to 'h'. If the stem starts with any other vowel it returns false. This is for the reason that Amharic verb stems only start with 'a' or a consonant. Then, it checks either the stem ends with 'a' or 'ä'. If any of the two is found it replace them with 'h'. If the stem ends with any other vowels it returns false, for the reason that Amharic stems only ends with a consonant or vowels 'a' or 'ä'.

Figure 5.8 The algorithm to check the verb stem

Sroot= root extracted from the input stem by removing vowels from the stem

1. *GET Stem*

2. *If Stem starts with vowel*

If stem starts with 'a' replace it with 'h'

Else return false

3. *If Stem ends vowel*

If Stem ends with 'a' or 'ä' replace with 'h'

Else return false

4. *EXTRACT Sroot from Stem*

5. *Search Sroot in the root lexicon*

If found return true

6. *Else Add 'h' to Sroot as a last letter ;*

Search Sroot in the root lexicon

If found return true

7. *Else Duplicate the last character of the Sroot*

Search Sroot from root lexicon

If found return True

8.. *Return False*

When the suffix obtained from the category of noun, after the affixes are removed the word is checked whether it is a correct word based on the affix pattern. The Algorithm decides that the word is a noun if one of the following conditions is true for the word.

Figure 5.9 The Algorithm to Checks Derived Nouns

- 1. If the word has prefix “mä” or “ma” and ends with a consonant**
- 2. If the word has prefix “mä” or “ma” and if the word have suffix ‘ia’**
- 3. The word has suffix pattern “Mia” or “ia”**
 - 3.1 And if the character at position two is a consonant or “u”, and if the word has suffix “ia”
- 4. The word has prefix ‘as’ or ‘a’**
 - If length of word > 7*
 - If the word has vowel pattern “ääää”, or “aoää”, or “aoüää”, and*
 - If character at lengthofword-2 ==lengthofword-4, and*
 - If character at word length-3==‘a’*
 - If length of word <= 7*
 - If word has vowel pattern aäa, or aoa, and*
 - If character at wordlength== character at word length-2 , and*
 - If character at word length-1=‘a’*
- 5. If vowel pattern of word is empty string or ‘u’**
 - and If character at position wordlength-1 = wordlength-2*
- 6. If vowel pattern of word is empty or ‘u’**
- 7. If word has vowel pattern ‘uai’ or ‘ai’**

If last character is 'i', and

If characters at position wordlength-1==consonant; and

If character at position wordlength-2=='a'.

8. If word have vowel pattern 'äai' or 'oai' '

If last character == i and

If character at word length-1==consonant, and

If character at word length-2=='a'.

9. If word have vowel pattern 'äaa' or 'oäa' or 'äa'

If character at word length-1 consonant

10. If vowel pattern 'äaa' or 'oaa'

If character at wordlength-1==consonant

11. If word has suffix 'ot', or 'na', or 'it' or 'at' or 'oš', and 'üt'

If word has vowel pattern o, or

if word has vowel pattern a, or

if word has vowel pattern uo, or

if word has vowel pattern ua, or

if word has vowel pattern ao, or

if word has vowel pattern aa, or

if word has vowel pattern auo, or

if word has vowel pattern aua

12. If word have suffix 'eta'

if word has vowel pattern 'ääea', or

if word has vowel pattern 'uea', or

if word has vowel pattern 'aäea', or

if word has vowel pattern 'uea',

Example of nouns that are recognised by the above condition are ;

Condition 1.

መስበር መነገድ መስንበት መቁጠር መተንፈስ መጫወት
መወፈር መውሰድ ማብረድ መውደድ

Condition 2:

መስበሪያ መነገጃ መስንበቻ መቁጠሪያ መተንፈሻ
መጫወቻ ማወፈሪያ መውሰጃ ማብረጃ መውደጃ

Condition 3:

ግብዣ ዝርፊያ ከርፊያ ግድያ ቅሚያ
ድሪያ ሩጫ ሽሚያ ልፊያ ግልበያ

Condition 4:

አሰባብበር አነጋገር አመነዛዘር አቆጣጥር አስተራረስ አስተያየት
አበላል አረዳድ አሞላል አሰራር አነዳድ

Condition 5

ንግግር ፋክክር ጭርርስ ንፍፍት ትሥሥብ

Condition 6

ውስብስብ ክርክር ስድብ ፍቅር ቁጥር ጅምር ክምር

Condition 7:

ስባሪ ቀራጭ ምራጭ ዝቃጭ ውዳቀ

Condition 8

ቆራጭ መንዛሪ ራንጣቂ ፈላገ

Condition 9:

ቆረጣ ረገጣ ቅየጣ ምንዘራ ፍልጋ

Condition 10:

ቆረጣ ደንባራ ቆማጣ ገባጣ

ሞላላ ጠማማ ቆልማማ

Condition 11:

ቁርጠት ድንቁርና ህልቂት

ስብራት ውጦሽ ትክብደት

Condition 12:

ቅሬታ ሃሜታ ሃዘኔታ

ሁኔታ አመኔታ

This section has discussed about how the morphological information is extracted and used to categorise verbs and their derivatives. The next section presents how the information extracted from a word analysed further to determine the inflectional categories of a word.

5.2.2. Deciding the categories of the word

After the parser gets the correct suffixes and prefixes and decides the category of the word, it decides the inflectional category based on the category, suffixes and prefixes, as shown in the diagram.

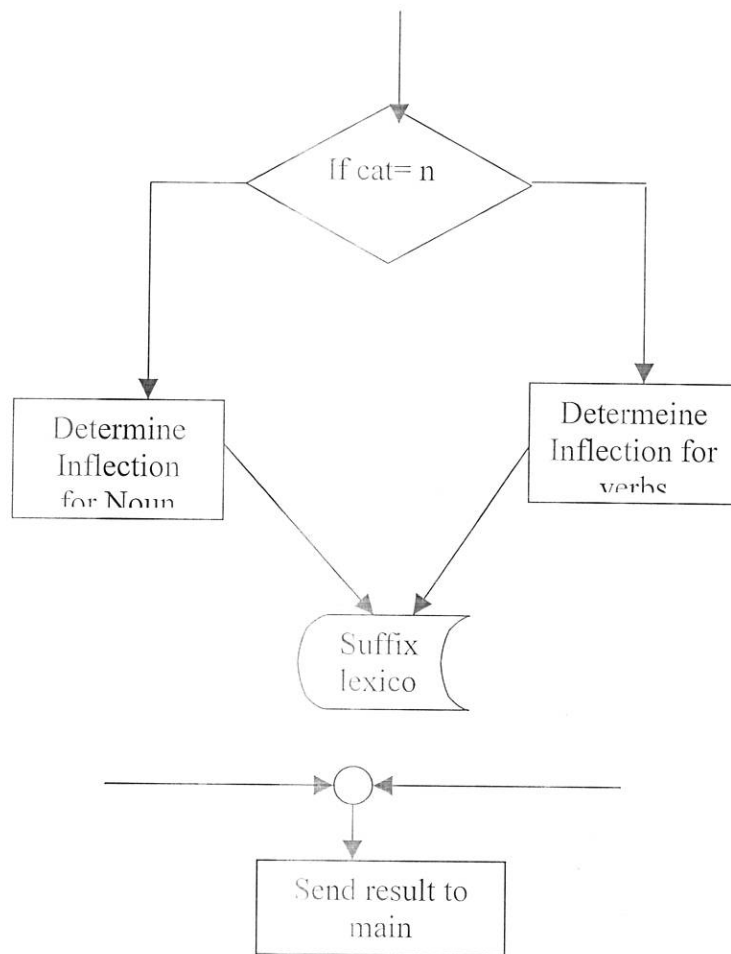


Figure 5.10 Flow Chart for Determining Inflectional Categories.

The Parser Uses The Same Suffix Lexicon For Nouns and Verbs. In the previous example ስለማያንቀሳቅሱላችሁ, the Prefix ‘Slāmayan’ and suffixes ‘ulačhu ’, represented in the database as;

Prefix		Suffix	
NAME:	Slāmayan	NAME:	ulačhu
Position-1:	Slā	Category	Verb
Position -2:	m	Position-1	u
Position-3 :	a	Position-2	l
Position-4:	y	Position-3	“ ”
Position-5	an	Position-4	ačhu

Based on each attribute the parser determines the syntactic meanings. In this particular example, with the category information it determines the category of the stem in general as verb. For the attributes from position 1 to 3 it checks the Non-concatenated suffix databases. These suffixes are represented in the database as;

NAME	u	NAME	l	NAME	ačhu
POSITION	1	POSITION	2	POSITION	4
Tense	Imperfect	Tense	0	Tense	0
PERSON	3	PERSON	0	PERSON	2
NUMBER	PLURAL	NUMBER	0	NUMBER	PLURAL
GENDER	0	GENDER	0	GENDER	0
POLARITY	0	POLARITY	0	POLARITY	0

Therefore, the parser decides based on the suffix as:

Subject features: 3rd person plural

Aspect: Imperfect

Object features: 2nd person plural

And the benefactive 'l' indicates that

the object is not a direct receiver of the action , and

the object is benefited from the action.

The presence or absence of a prefix at certain position indicates a specific meaning. The presence or absence of a particular prefix can be interpreted by the parser as

If there is prefix at position 1	Then the word is a clause
If there is prefix at position 2	Then the word is a clause with imperfect aspect
If there is prefix at position 3	Then the word have the negative sense of the stem
If there is prefix at position 4	Then the word has an imperfective aspect
If there is prefix at position 5	Then the word have a derived stem

Therefore the above example can be interpreted as- A clause that have a negative sense and imperfective aspect.

The parser interprets the affixes for verbs and nouns separately. The above example is for verbs. If the category of the word is noun the parser uses the fact that , except the possessive marker suffixes of a noun , only the presence and absence of suffixes is enough for the other. For example in parsing the word 'mäsaṛiya-oč-e-n ' መሳሪያዎቼን 'my instruments', the module that

determine the category and the correct affixes send the following to the module that determines the inflectional categories;

Concatenated Suffix = -očen

Concatenated Prefix = “ ”

Category Noun (decided based on -očen)

The structure of ‘-očen’ in the database looks:

NAME: -očen

POSITION 1: -oč

POSITION 2: e

POSITION 3: n

POSITION 4: “ ”

CATEGORY: Noun

When the affix interpreter for noun get this information it decides about the noun as: Plural noun, Object, and the Attribute of the posser of the noun is

Person= 1st

Number=Singular

Gender= ()

The algorithm that enables the parser to interpret like this is listed below:

Figure 5.9 The algorithm that determines the inflectional category of a noun

```
1. Get Concatnated_Suffix, Concatnated_Prefix
2. If Concatenated _suffix is not empty
    2.1 If the suffix have value at position one  decide the word is a plural noun
    2.2 If the suffix have value at position three  decide the word is an object
    2.3 If the suffix have value at position four  decide that the word has conjanction
    2.4 If the suffix has value at position 2
        2.4.1 Search Suffix_File (Name at position 3,Noun,3)
        2.4.2 If found the possesser of the object expressed by the noun has attributes
            person= Suffix_Person
            Number=Suffix_Number
            Gender=Suffix_Gender
3. end
```

The information provided by the prefixes interprets based on their value. For example; 'if 'bä' then the object indicated by the noun is used as instrument'

The algorithm presented here is coded using Microsoft Visual C++ and tested. The next section presents the test result and procedure.

5.3 Results of the Experimental

The test is conducted to see the effectiveness of the algorithm designed. The aim of the test was;

1. to see either the algorithm categorise a word as nominal and verb effectively

2. to check either the algorithm correctly tokenize a word to components correctly
3. to check whether the algorithm correctly interpret the inflectional categories of the word based on the tokens extracted

In the first case 100 nouns selected manually were considered

- The parser recognise 73 of the nouns correctly and decides the other incorrectly as verbs, and unrecognised word.
- out of the 73 correctly recognised noun, the parser correctly tokenize 71, only three are tokenized incorrectly
- Out of the 71 tokenized correctly the parser interprets 69 of them correctly

One of the problems to categorise a word as verb while it is a noun is identified as incorrect interpretation of the suffixes of nouns. For example when a word *sbrat* is given the parser recognise the '-at' similar to a suffix like 'sbär-at'. The parser does not perform the derived noun check processes, as it is working deterministically. Therefore it returns unrecognised word because 'sbr' has no vowel pattern (hence there is no stem with no vowel). We make changes in order that the parser when ever it gets a word it considers it as a noun with out affix so that it first used the derived noun check module. Therefore words like 'sbrat' can be interpreted correctly before they are considered as a verb. After this adjustment a result of 97 correctly recognised words are obtained.

For any word with the correct vowel and consonant pattern found in the conditions the parser decides it as a noun. Mistaken Words with a correct pattern are also considered as correct. For example the string 'wtgg' is considered as a noun because it fulfils condition 5 of the rule for

nouns. If a root lexicon is used to determine the nouns it can consider such words as mistake, but it also considers correctly spelled words but that are not found in the lexicon.

- Test using verbs

The test for verbs is conducted on 100 Words. These Words are generated by two individuals manually. They write 10 forms for 5 roots each. The test shows the following results.

- 93 out of 100 parsed correctly
- Out of the 93 all but one tokenized incorrectly
- All the 92 interpreted correctly

One of the reason for incorrect recognition was the pre-processing of the suffix 'awo' as 'ao'. Correction is taken to apply this particular pre-processing only to nouns because this process do not occur in Amharic verbs, and 'awo' is a correct suffix in Amharic as in the word 'sädb-awo'. Using this correction only one word can be re-recognised correctly and increases the accuracy to 94%.

Using a similar step with the above the test is conducted for different words. Together with the above the parser correctly recognizes 84 % of the verbs and 86 % of the nouns.

CHAPTER SIX

CONCLUSIONS AND RECOMMENDATIONS

6.1. Introduction

The purpose of this study was to design a word parser for Amharic verbs and their derivatives. A discussion on Amharic verb morphology was given. A word parser is designed to recognise. The algorithm designed uses the morphological property of the language to determine the categories of the words derived from verbs. Mainly the algorithm identifies a word as a verb or nominal, and the inflectional categories in the respective categories. An algorithm that can change surface forms to underlying forms are designed to handle phonological variations. A database of affixes and roots, which helps the parser to tokenize and interpret components, are also designed and implemented. This chapter provides a summary of the conclusion and recommendation for future research.

6.2 Conclusions

Amharic is a morphologically complex language. Most of the words in the language are verbs or derived from verbs. Verbs are the most productive classes of words in the language. The root is a basic lexical unit of the language. The root form of a base verb can be bilateral, trilateral, quadrilateral and quantilateral. Verbs are derived from roots by inserting vowel patterns to the root. Therefore, most of the words in the language can be reduced to root and vowel patterns. Roots form their stems based on their consonants nature. If the root type is known, it is possible to predict the other forms. It is shown that word formation follows regular patterns: prefixing, suffixing and internal doubling of consonants. Thus it is possible to conclude that, even though

Amharic is morphologically complex its regular patterns make it manageable for algorithmic processing.

In this study root pattern and the affixes are used to determine the lexical and inflectional category of the word. Even though Amharic has specific suffixes to nouns and verbs there are suffixes that have similar spelling but in different categories. These make the categorization using only suffixes impossible. The ambiguity further increases when suffixes are concatenated. Therefore in addition to affixes, vowel patterns and the property of the root is used to parse Amharic language.

In order to test the accuracy of the algorithms, 200 verbs and 200 nouns were considered. The test result shows that 86 % (172) of verbs and 84 % (168) of the nouns are recognised correctly. This is a promising result for further research in the same direction.

These algorithms can be extended for regular nouns by designing a lexicon. As Amharic has a handful of basic adverbs and preposition the algorithm can be extended for these categories with some modification.

6.3 Recommendations

The study does not include the property of words at syntactic level. Words in the syntax have more constraints than words out of syntactic contexts. Including the syntactic property of the Amharic words can increase the efficiency of the parser. Together with this, as derived nouns go through similar inflections as of regular nouns, the algorithm can be extended easily for regular nouns by

designing a lexicon that can handle regular nouns. More over this as Amharic has a handful of prepositions and regular adverbs the efficiency of the parser can be extended easily to include these categories. To this end further research should be undertaken to develop a full-fledged Amharic word parser.

Even though an attempt is taken to include most of the fetures of the word, one can identifay other fetures that have not used in this study. Therefore, further researches should also consider other fetures. The algorithm is tested on limited number of words, so it is not possible to say that the resylt indicate the correct accuracy. Therefore other testes using large number of data must be conducted, so that it is possible to estimate the correct effectiveness of the parser.

Word parsers are not an end by themselves they are ways to an end. Therefore further studies should also focus on implementing application that can use the outputs of word parsers.

Reference

- Alemayehu H. (1987) *An Auto Segmental Approach to Amharic Intonation*. PHD. Thesis at the School of Oriental and African Studies University of London.
- Allen, J. (1996). *Natural Language Understanding*. Benjamin/Cummings: Redwood.
- Anderson, S.(1988) Morphology as a Parsing Problem. *Linguistics* 26, 521-544.
- Anshen, F. and Mark A. (1988) Producing Morphologically Complex words. *Linguistics* 26(1988), 641-655.
- Antworth, E. (1994) *Morphological Parsing with a Unification Based Word Grammar*. North Texas Natural Language Processing Workshop, University of Texas, 1994
- Antworth, E.(1990). PC-KIMMO: *A two level processor for morphological analysis*.
http://www.sil.org/pc-Kimmo/about_Pc-Kimmo.htm
- Aronof, M. (1976) *Word Formation In Generative Grammar*. MIT:MA.
- Artrandi, S. (1976) Machine Indexing: Linguistics and Semiotic Implication. *JASIS*, 27(4) 235-239
- Baye Yimam (1988) Towards a definition of the Nominal Specified in Amharic. In *Proceeding of the 8th International Conference of Ethiopian Studies* vol. 1, edited by Dr. Tadesse Beyene, Addis Ababa Ethiopia, 599-612.
- Bender , L. and Hailu F.(1978) *Amharic Verb Morphology*. Michigan State university
- Bender, M. ; Ferguson, C. (1976). *Introduction. Language in Ethiopia*; edited by Bender, M. and Ferguson, C. [et al]; Oxford :London, 1976
- Brru Dori (1992) Customizing CDSIS/IS for Amharic Texts. Master Thesis at The School of Information Studies For Africa, Addis Ababa

- Bybee, J. (1985) *Morphology : a Study of the Relation between meaning and form*. Benjamins:Amsterdam.
- Culy, C. (1985) The Complexity of the Vocabulary of Bambara. *IN the Formal Complexity of Natural Language* , edited by Walter J. [et al]; Redial: Boston 349-357
- Dereje Teferi (1999) *Optical Character Recognition of Type written Amharic Text*. Master Thesis at the School of Information Science, Addis Ababa.
- Gazdar, G. (1996). Paradigm Merger in Natural Language Processing. *Computing Tomorrow: Future research Directions in Computer Science*. Edited by Ian Ward and Robin Milner. Cambridge: NY. PP 88-109.
- Gazder ,G. and G. Pullum (1985) *Computationally Relevant Properties of Natural Language and Their Grammars*. *IN the Formal Complexity of Natural Language* , edited by Walter J. [et al]; Redial: Boston 387-438
- Girma H. (1994) *The Syntax of Functional Categories: a Study of Amharic*. PHD. Thesis in Linguistic at the University' Du Que'bec A' Montrial.
- Girmaye B. (1992) Word Formation in Amharic. *Journal of Ethiopian Language and Literature*.No.2 1992. PP. 50-74
- Golding, A. and H. Thompson (1985) A morphological Component for Language Programs. *Linguistics* 23 , 263-289.
- Gravin, P..(1963) *Natural Language and the Computer*. McGraw-Hill: New York
- Grishman, R. (1984) Natural Language Processing. *JASIS* 35(5):291-296
- Grishman, R. (1986) *The Theory of Linguistic an Introduction*
- Güngördü, Z. ald Oflazer, K. (1995) Parsing Turkish Using Lexical Functional Grammar formalism. *Machine Transulation* 10 ,293-319.

- Habte Mariam Markos (1994) Towards the Identification of the Morphomic Component of the conjugation forms of Amharic. In Proceedings of *The eleventh International Conference of Ethiopian Studies*. Vol. I pp. 465-486.
- Hailu Fullas (1966) *Derived Nominal Patterns in Amharic*. PHD. Thesis at the University of California, Los Angeles.
- Harris, M. (1984) *Introduction to Natural Language Processing*; Reston: Virginia.
- Jirku, P. (1984) *Logical and Linguistic aspects of Computer-Based Inference Processing*
- Johnson, F., C. Paice, W. Black (1999) The Application of Linguistic Processing to Automatic abstract generation. *Readings in Information Retrieval* edited by K. Spark Jones and P. Willett.
- Leslau, W. (1973) *English-Amharic Context Dictionary*. Wiesbaden: Harrassowitz
- Leslau, Wolf (1965) *An Amharic Text Book of Everyday Usage*. University of California, Los Angeles.
- McCarthy, J (1981) A Prosodic Theory of Non Concatinative Morphology. *Linguistic Inquiry*. Vol. 12 (3), 373-417.
- Metzler, D., S. Hoas, C. Cosic, and L. Wheeler (1981) Constituent Object Parsing for Information Retrieval and Similar text Processing Problem. *JASIS* 40(6): 398-423.
- Mullen , Dana (1986) *Issues in the Morphology and Phonology of Amharic: The Lexical generation of pronominal clitics*. A Dessertation for the Degree of Doctor of Philosophy at the School of Graduate Studies and Research, Departement of Linguistic University of Ottawa, 1986.
- Nega Alemayehu (1999) Developpment of a Steming Algorithem for Amharic Texts Retrival. PHD. Thesis At University of Shifeld.
- Pacak, M. (1967). Computational Morphology. In *Papers presented in honour of Levin Dosant*; edited by Wiliam M. Austin; Mouton:Hague.

- Palmer, Martha and Zhibio, Wu (1995) Verb Semantics for English-Chinese Transulation. *Machine Transulation* v.10 59-92, 1995
- Park, B., B. Yun, H. Rim (1996) Recognizing Unregistered Nouns for an Automatic Indexing System. In *Proceedings of the Workshop on Information Retrival with Orential Languages*. June 28-29, Jacjon, Koria. pp. 3-8.
- Pinker, Steven (1999) Words and rules. *Lingua* , v.106 , 219-242
- Polacek, Zedenek (1989) Functional Sentence Perspective in Amharic: Priliminary notes. 8th *International Ethiopian Studies* 561-566
- Polak, Milos (1967) Computational Morphology. In *Papers presented in honour of Levin Dosant*; edited by Wiliam M. Austin; Mouton:Hague.
- Prasad, A.(1993)*Application of Computer Based Natyral Language Processing Tools and Technique in Developing Subject Indexing Language*. PHD thesis at Karanatak University, Dhaward. Departement of Library and Information Science.
- Pullman, s., G. Russel, G. Ritchie, and A., Black (1988) Computational Morphology of English. *Lingustic* 26, 545 – 560.
- Radford(1988). Introduction to Linguistics. MIT, MA.
- Salton, G. (1989) *Automatic Text processing: The Transformation, analysis, and retrieval of Information by computer*; Wsely: Masachusete.
- Shiber ,Staurt (1985) Separating Lingistic Analysis from Linguistic Theories. In *Natural Language Parsing and Linguistics theories*, edited by U. Reyle and C. Rohrer, Redial:Dordecht V. 35 , 33-68.
- Sparc Jones, K. (1974) Progress in Documentation Automatic Indexing. *Jouranal of Documentation*, 30(4) 393-432

- Thompson, F and B. Thompson (1975) Practical Natural Language : the REL Systeem as prototype. *Advance in Computers*, 13 110-169
- Warner, A. (1987) Natural Language Processing. *ARIST* Vol. 22, 1987 79-108
- Wedekind, K. (1992) Which Form Predict all other Best? Variation on the Amharic Verb "Theme". *Journal of Ethiopia Studies* 25, (73-92)
- White Lock, Pete (1988) A Feature based Categorical Morpho-Syntax for Japanese. In *Natural Language Parsing and Linguistics theories*, edited by U. Reyle and C. Rohrer, Redial:Dordrecht V. 35 , 230-261.
- Worku Alemu (1997) *The application of OCR techniques to the Amharic Script*; Master Thesis at The School of Information Studies For Africa, Addis Ababa University , Addis Ababa.

DECLARATION

This thesis is my original work and has not been submitted for degree in any other university and that all sources of material used for the thesis have been dually acknowledged



Abiyot Bayou

The thesis has been submitted for examination with my approval as university advisor

 *19 May 2000*

Dr. Nega Alemayehu