ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

AUTOMATIC MORPHOLOGICAL SYNTHESIZER FOR AFAAN OROMOO

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

By: **Abebe Abeshu Diro**

June 2010

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

AUTOMATIC MORPHOLOGICAL SYNTHESIZER FOR AFAAN OROMOO

By: **Abebe Abeshu Diro**

Advisor: **Dida Midekso (PhD)**

Signature of the Board of Examiners for Approval

	Name	Signature
1.	Dida Midekso (PhD),Advisor	_____
2.	_____	_____
3.	_____	_____
4.	_____	_____
5.	_____	_____

Dedicated to

My father, Abeshu Diro and My mother, Kumeshi Furgassa, who have raised me to be the person I am today

ACKNOWLEDGMENTS

Though my name is the only author on this work, many people have contributed to its completion: those who provided insight and comments, those who provided ideas and suggestions, and those who provided love and support. My advisor, **Dida Midekso (PhD)**, is naturally at the top of this list. He has always been an enthusiastic supporter of my work, providing an unending supply of ideas and comments. He gave me the independence to pursue my own interests, and in the end, gave me the guidance needed to clear the final hurdles. Thank you very much!

Abebe Keno deserves special awards for his willingness to give me comments at all times, and without complaint on linguistic aspects, and in evaluating the system. His selflessness made the completion of this thesis much easier. It is also my pleasure to express my gratitude to **Tolemariam Fufa (PhD)** and **Aseffa W/Mariam** for providing linguistic materials and providing valuable comments.

Sebsibe H/Mariam (PhD) has supplied a tremendous amount of ideas in constructing this thesis. His clear thinking provided practical and insightful comments at key moments. He paved a way to go on, and showed an approach to follow.

In addition to being a friend, coffee buddy, and research partner, **Mandefro Legese, Tesfaye Guta, Getasew Tsedalu, Teklay G/ Egziabher, Abel T/Mariam, Moges Ahmed, Selama G/Meskel, Michael Shiferaw, Esmael Kedir and Mequanint Muniye** are due a huge debt of gratitude for thier assistance in everything that I needed. I am lucky to have such classmates.

I will be forever thankful to my closest friends: **Gonfa**, **Regassa, Efrem, Gemechu, Tadele** and **Admasu**. While they were always there to help when the going was rough, I'm most thankful for encouragements and valuable support. **Moybon Wolde,** you are my always '**hiriyaa dhugaa'**, and our intimacy persists forever. I can never thank my family enough, especially Mom and Dad, for always providing encouragement. Both provided nothing short of unconditional love and support. **Ebisa Dhaba, Abebe Lemessa, Hunduma, Daniel**, **Temesgen**,**Obsa Tesema** you are thanked for responding the questionnaire to evaluate my system and providing constructive crticisms.

I offer my regards and blessings to all of those who supported me in any respect during the completion of the thesis as well as expressing my apology that I could not mention all.

And last, but absolutely not least, I thank God for giving me the wisdom and the strength I need to discharge my duty.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

NL Natural Language

NLP Natural Language Processing

NLG Natural Language Generation

POS Part-of-speech

CDs Compact Discs

CV Consonant-Vowel

NLU Natural Language Understanding

CSA Central Statistics Agency

NLS Natural Language System

ABSTRACT

Computational morphology is an important component of most natural language processing tasks. Morphological generation, the process of returning one or more surface forms from a sequence of underlying (lexical) forms, can provide fine-grained parts of speech information and help resolve necessary syntactic agreements. In addition, morphological synthesis systems are used as components in many applications, including machine translation, spell-checker, speech recognition, dictionary (lexicon) compilation, POS tagging, morphological analysis, conversational systems, automatic sentence construction and many others. Generally, the thesis describes processes of automated morphological synthesis ranging from manually synthesizing words to developing a prototype and conducting an experiment. The automated generation of word forms avoids the storage of exhaustive lexicons and thereby saves memory requirement. The devepment of such systems demand an in-depth study of the morphology of the language used.

Morphological synthesizers have been developed for languages like English. But there is no such a system for Afaan Oromoo, the working language of Oromia national regional sate, and one of the major languages in Ethiopia. This study is, thus, an attempt to develop automatic morphological synthesizer for Afaan Oromoo.

Algorithms that take the morphological properties of Afaan Oromoo into consideration are developed from scratch and applied, as there are no previous such attempts. We employed rule based computational model to design and develop the prototype referred to as ***HORSIISAA****. The performance of the system on average is 96.28% for verbs and 97.46% for nouns. The result obtained encourages the undertaking of further research in the area, especially with the aim of developing a full-fledged Afaan Oromoo morphological synthesizer.*

Keywords: *Morphology, Morphological synthesis, Natural language processing, Morphological generation, Morphological processing, Afaan Oromoo, word forms*

CHAPTER ONE: INTRODUCTION

1.1. BACKGROUND

Language is one of the fundamental aspects of human behavior and it constitutes a crucial component of our lives. In its written form it serves as a means of recording information and knowledge on a long term-basis and transmitting what it records from one generation to the next. In its spoken form, it serves as a means of coordinating our day-to-day life with others [1]. Linguistics can be defined most simply as the scientific study of languages, particularly, natural languages. An approach to linguistics that employs methods and techniques of computer science is called Computational Linguistics or Natural Language Processing (NLP). NLP is concerned with computational processing of natural languages in order to provide such novel products as computers that can understand everyday human speech, translate between different human languages, and otherwise interact linguistically with people in ways that suit people rather than computers [2]. The goal of natural language processing is designing and building systems that will understand and generate natural languages. Having such types of systems will enable to communicate with machines as though one is communicating to human agent. The machine should understand first the natural language before processing it. Thus, NLP demands deep Natural Language Understanding (NLU) and modeling the natural language so that computer programs that act appropriately on the information contained in the text can be developed.

In every language, whether it is spoken or written, every meaningful pattern has its own structure and the elements of language should relate to each other in understandable manner. These relationship requirements together with the existence of large number of classes in natural language make the processing task complicated [6, 8]. The understanding of natural language by machine has ambiguity at structural, semantic and lexical levels. Besides ambiguity of words, inflection and derivation of the language are other reasons that make natural language understanding very complex [41]. Such complication becomes even worse for Afaan Oromoo

which is a highly inflected, resource scarce language and whose structure has not been studied extensively. In such languages, many word forms can be generated from a single basic unit such as the stem/root. The details are presented in Chapter Four.

A natural language system must use considerable knowledge about the structure of the language itself, including what the words are, how words combine to form sentences, what the words mean, how word meanings contribute to the sentence meanings and handling word variants [1]. This further complicates the task for computers to understand NL. Though, understanding natural language by machines is a complicated problem, there are various approaches under investigation and some of them have succeeded to some extent in making machines to understand natural language at different levels of processing.
There are different levels of natural language processing [10]:

- **Phonological** deals with sounds or combinations of sounds
- **Morphological** deals with processing of individual word forms
- **Lexical** deals with the procedures operating on full words
- **Syntactic** deals with grouping words of a sentence into structural units
- **Semantic** adds contextual knowledge to the purely syntactic process in order to restructure the text into units that represent the actual meaning of a text, and
- **Pragmatic** uses additional information about the social environment in which a given document exists.

There are, for instance, systems developed for processing NL at phoneme, word, sentence, and pragmatic levels. These systems are developed in such a way that the output of a lower system can serve as an input to the next higher level. For instance, the output of a morphological synthesizer that works at word level could serve as an input for syntactic parsers that work at sentence level [12]. In addition, the words generated by morphological synthesizer can be used as a suggestion list for spell checker [11]. This study is based on the morphological level of

natural language processing. Morphology in language refers to the study of rules for forming admissible words i.e. grammatically right and acceptable words in the respective language. The morphological processing problem can be summarized in two separate tasks: given a word form, determine its stem and its grammatical description (this is known as morphological analysis) and given a stem and a set of intended grammatical features, determine the corresponding inflected form (this is known as morphological synthesis). Generating correct inflectional word forms requires more information than analyzing word forms into stems and affixes [51]. An analysis grammar needs only an inventory of affixes and morphotactic processes that can be applied to stems, with no requirement that certain affixes or morphotactic processes be associated with particular stems. Assume, for example, that an analysis grammar of English contains the plural suffixes *s* and *es.* Assume further that the word *hisses* is encountered in a text. A straightforward morphological analyzer will strip off the affix *s* and look up *hisse* in the lexicon, finding no match. Then it will strip off the affix *es* and look up *hiss* in the lexicon, finding a valid entry. There is no real need for the analyzer to know in advance that *hiss* must take the suffix *es* rather than *s*: the latter case simply will not ever be encountered. A generation grammar, by contrast, would have to know that *es* and not *s* is the pluralizing suffix for *hiss* since there is no room for ambiguity in generation. Morphological synthesis or generation, which we deal with in this thesis, is a process of returning one or more surface forms from a sequence of morpheme glosses. The morphological synthesizer in this case will enable one to generate the surface form (e.g. books) from its constituent distinct parts called morphemes (e.g. /book/ and /s/). That means /book/ + /s/= /books/. Thus, the morphological synthesizer will accept /book/ and /s/ as an input to generate the surface form /books/ [23].

Morphological synthesizers have vital role in NLP systems. They are used to generate surface word forms, which are the ones that are found in everyday communication, from lexical components that could be stored separately in different databases (lexicons). Such systems are used as a subcomponent of NLP in applications like machine translation, dictionary (lexicon) development, and spelling and grammar checking, and etc. Thus, it is the purpose of this study to

explore the possibility of developing an automatic morphological synthesizer useful for generating Afaan Oromoo words.

1.2. STATEMENT OF THE PROBLEM

Afaan Oromoo (also known as Oromo) is one of the major languages that are widely used in Ethiopia. Currently, it is a working language of Oromia national regional state (which is the largest region/state in Ethiopia). Unlike Amharic, another major language and working language of Ethiopia which belongs to Semitic family languages, Afaan Oromoo is part of the lowland east Cushitic group within the Cushitic family of the AfroAsiatic phylum. In this Cushitic branch of the Afroasiatic language family, Afaan Oromoo is considered as one of the most extensively spoken languages among the forty or so Cushitic languages [19]. Based on the 2007 Census conducted by the Central Statistical Agency of Ethiopia (CSA), it is spoken by more than 27.6 million Oromos within Ethiopia [49]. In addition, the language is also spoken in Somalia, Kenya, Uganda, Tanzania and Djibouti [20].

Although Afaan Oromoo is today spoken by such a large number of people, few advances have been made in computational linguistics or natural language processing in the language. Computational approaches to linguistic analysis of Afaan Oromoo so far have been hindered due to non availability of well studied linguistic resources. This scenario has recently started to change, primarily through the theses that are being carried out at Addis Ababa University.

The amount of accessible electronic information has exploded in recent years thanks to the Internet and other related distributed international networks. Due to the rapidly expanding use of the Internet for communication and dissemination of information throughout the world, electronic information sources are now available in an ever-increasing number of languages. Users of such globally distributed networks (including digital libraries and World Wide Web) need to be able to access and retrieve any relevant information in whatever language and form it may have been recorded and stored. However, most developing countries have no systematic programs for the collection, analysis and dissemination of available information to the potential

users [22]. One of the barriers to this is the absence of online machine translation systems that can translate texts from a foreign language to a local one say, from English to Afaan Oromoo. Thus, the existence of machine translation systems that require morphological synthesizers as a component are of paramount importance for the delivery of electronic resources (such as Internet and CDs) to the population at large in their mother tongues. Therefore, the need for NLP systems such as morphological synthesizer is unquestionable for Afaan Oromoo.

The module of morphological analysis and/or synthesis is unavoidable in any language engineering tool for Afaan Oromoo because of its rich morphology. In information retrieval systems, text based information retrieval (IR) focuses on matches between text based representations of human information needs and textual representations of documents. The match between the query and documents is seldom perfect, because both representations are expressed in natural language and have different origins and characteristics [13, 14]. As stated in [5] the effectiveness of information retrieval system entirely depends on the indexing and searching techniques employed. In a very crude term, the search result from a given query can be an indication of how effective a given information retrieval is. It is common that in a textual database, words occur in various forms (either inflectional or derivational). It is awkward to give the different forms of the word in order to retrieve related documents. In this case, the system either fails to retrieve documents or the query should be given in all variants. In order to improve the effectiveness of an information retrieval, one of the popular approaches is to give base form of words as query term so that all variants are automatically searched that is, it is usually desirable to make queries using semantic entities, not using special morphological forms of a word. This technique is usually employed by search engines that use stemmers in morphologically rich languages. Obviously, these extensive inflectional and derivational features of the language are presenting various challenges for text processing and information retrieval tasks in Afaan Oromoo. This again ensures the importance of morphological synthesizer.

Machine translation and information retrieval are some of the applications of natural language processing in which morphological synthesizer is crucial. There are many more tasks that need synthesizer in the area of NLP of Afaan Oromoo. For instance, the absence of morphological synthesis systems will have an effect on researches in spell-checker, speech recognition, dictionary compilation, POS tagging, conversation systems, automatic sentence construction, etc. [23]. But, no research has been conducted so far in the area of automatic morphological synthesis for Afaan Oromoo.

Therefore, it is worth conducting research and also developing an automatic morphological synthesizer for Afaan Oromoo based on the characteristics of the language.

1.3. OBJECTIVES

1.3.1. General Objective

The general objective of this study is to develop an automatic morphological synthesizer for Afaan Oromoo.

1.3.2. Specific Objectives

The specific objectives are to

- Review the various techniques (or approaches) suggested for the development of an automatic morphological synthesizer, and adopt the one appropriate for Afaan Oromoo;
- Study the morphological property of Afaan Oromoo in general and that of verbs and nouns in particular to identify properties useful for automatic morphological synthesis;
- Study the type of lexicons required for morphological synthesis, and design the lexicons accordingly;
- Develop algorithms for morphological synthesizer for Afaan Oromoo
- Develop and test a prototype of an Afaan Oromoo morphological synthesizer;

1.4. SCOPE OF THE STUDY

The scope of the thesis is limited to demonstrating the potential Rule- Based approach to develop an automatic morphological synthesizer for Afaan Oromoo nouns and verbs forms. The study excludes other grammatical categories as nouns and verbs are the most productive categories, and the work can be easily extended to the othert parts of speech. Compound word formation is also out of the scope of this research due to the absence of clearly stated rules of formation in literatures.

1.5. LIMITATIONS OF THE STUDY

The main limitation while conducting the study is the absence of readily available annotated data in the form of lexicon suitable for morphological synthesizer and well written linguistic materials in Afaan Oromoo. In addition, as every affix cannot be applied to every stem it was difficult to predict which affix is for which stem. Preparing details of lexicons and specification of stem class were very difficult and manually annotated by the researcher and linguists, so it needed much time and effort. The absence of well studied linguistic materials presented various challenges to classify stems into classes according to their similarity by the affixes they take.

1.6. METHODS AND TOOLS

For the successful completion of this study, the following methods have been used.

1.6.1. Literature Review and Discussion with Experts

Developing morphological synthesizer for Afaan Oromoo needs thorough understanding of the language. The principles and rules of the language in the area of phonology, lexical, morphology and parts of speech have been carefully studied. This was done for the purpose of studying the morphological structure of Afaan Oromoo words in general and that of nouns and verb forms in particular, and to know how to develop lexicon for morphological synthesis research work. A

number of resources including books, research reports, journal articles, manuals and other published and unpublished documents (including those from the Internet) and discussion with linguists have been used. In addition, related works in the same area for different languages have been thoroughly studied to study techniques or approaches in morphological synthesis and to adopt one that is found appropriate for our study.

1.6.2. Lexicon and Algorithm Development

A database **OroMorpho** consisting of 4 tables has been developed as a knowledge base. The tables include Stem, NounSuffix, VerbSuffix and BoundaryChange tables. The majority of the algorithms are designed from scratch as there are no previously designed algorithms for this purpose based on the morphological properties of the language to generate Afaan Oromoo words from an input stem and suffixes. Some algorithms have also been adapted from different languages particularly from Amharic.

1.6.3. Development Tools

Rule-based approach was adopted in the design of the morphological synthesizer referred as **HORSIISAA**, in this study. In the development, this approach was employed for morphophonemic processes like assimilation, epenthesis, deletion, classification and reduplication. It is based on this approach that lexicons and most algorithms are designed from scratch as there are no previously developed ones. The prototype synthesizer has been developed using Java Programming language.

With regard to the stem not stored in the lexicon, rules of classification have been formulated to predict the type of stem, after which the corresponding suffixes are retrieved from the lexicon and word formation continues by applying the necessary concatenation and boundary change rules. For the identification of syllables, and boundary change processes, we have developed full-fledged rules (patterns) using regular expressions.

1.6.4. Evaluation

In the first phase, the test has been done personally by the researcher by generating words from selected stems and comparing the generated words with the structure of words in the grammar books. The test has been conducted iteratively to increase prototype's performance. The errors encountered during experimentation have been corrected and the experiment has been done iteratively until the result is found to be satisfactory. Most of the errors that arose were from the missing rules of boundary changes, and corrected accordingly before the second phase evaluation.

In the second phase, eight linguists were used for evaluation. The performance of the system was found to be 96.28% for verbs and 97.46% for nouns. The details of experimentation and the performance of the prototype are discussed in Chapter Seven.

1.7. APPLICATION OF RESULTS

Morphological synthesis systems are useful in many areas of NLP for Afaan Oromoo. The beneficiaries of this study include researchers who are, or want to be, involved in increasing the capability of computer processing in Afaan Oromoo. The study could also be used:

- As a component for the development of higher forms of NLP systems such as machine translation, speech recognition, parts of speech tagging, automatic sentence construction, etc. for Afaan Oromoo;
- For Afaan Oromoo teaching and learning;
- To build a morphological dictionary for Afaan Oromoo;
- To generate suggestion lists for spell checker
- In Web search to automatically search for the inflected forms of the word even if the user only typed in the base form

1.8. ORGANIZATION OF THE THESIS

The rest of the thesis is organized as follows. Chapter 2 discusses literature review on different issues in morphology. In this chapter, computational morphology and its constituents, types and approaches of morphology, issues about morphotactics and two-level morphology are discussed. Chapter 3 is devoted to discuss related works done on morphological synthesizers developed in different languages. Chapter 4 specifies the morphological and phonological properties specific to Afaan Oromoo. Many language specific issues such as the writing system, syllable structures, inflections and derivations have been extensively presented. Chapter 5 discusses design requirements, techniques and approaches, the architectural and design issues of our system. The lexicon and architectural design are dealt with in this chapter in detail. The implementation issues are presented in chapter 6. This chapter details stem classification issues, signature building, algorithms and implementation of rules. Chapter 7 is devoted to the evaluation of the system. Chapter 8 concludes the thesis by outlining the benefits obtained from the research work and limitations of the system. It also shows some research directions and recommendations that can be accomplished in developing a full fledged morphological synthesizer for Afaan Oromoo.

CHAPTER TWO: LITERATURE REVIEW

In this chapter, we concentrate on addressing morphological synthesizer development strategies and approaches. The formulations of morphological synthesis and typical architecture of morphological synthesizer are discussed in section 2.1. Then, discussion of basic concepts of computational morphology is presented in section 2.2. The next section deals with constituents of morphology. The morphotactics is dealt with in section 2.4. The types and approaches of morphological synthesis that have close relevance to this study are discussed in sections 2.5 and 2.6 respectively. The last section discusses the two-level morphology.

2.1. MORPHOLOGICAL SYNTHESIS

Morphological synthesis or generation is a process of returning one or more surface forms from a sequence of morpheme glosses. The process can be formally defined as follows: Given a stem r (verb, noun) and a list of relevant attributes $a_1a_2a_3$ …, a_n, the morphological synthesizer generates the word w, which has r as stem and $a_1a_2a_3$…, a_n as the morphological attributes. For example, consider the English root verb "*go*" that takes the form "*goes*" in an *indicative sentence, when the subject is third person, singular and the tense is simple present.* Here, r is "go" and the attributes are <*indicative, present simple, singular, 3rd person* > [24]. Figure 2.1 depicts a typical architecture of morphological synthesizer.

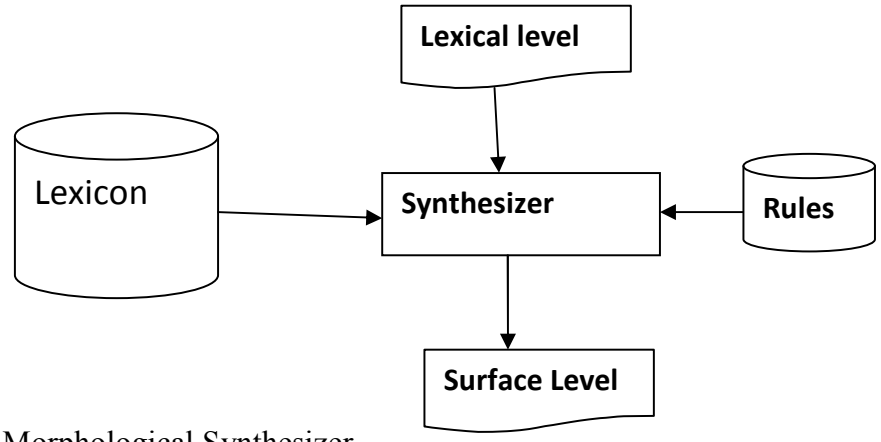

Fig 2.1: Typical Morphological Synthesizer

The general architecture of typical morphological synthesizer depicted in Figure 2.1 has 3 main components: Lexicon, Rules and Synthesizer. Lexical level represents the input stems and associating affixes that are given to the synthesizer. The lexicon module stores sample stems, and serves as a knowledge base for the word synthesis process. The rules component is stored as a mini lexicon to govern the boundary change when morphemes combine. The synthesizer returns all surface forms of the stem stored in the lexicon by applying necessary rules of formation. The surface level is the output of the synthesizer.

Even though some basic knowledge is common for most synthesizers, the detailed knowledge required can vary from language to language. As stated in [25], the following are the three main types of knowledge that need to be represented for synthesizers.

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

In addition, knowledge of syllable structure plays vital role as some suffixes change forms according to syllable structure particularly for Afaan Oromoo stems. For instance, the plural suffix '**oota**' changes its form to '**ota**' when attached to the nominal stem whose sound before the last consonant is long, as when **gaala** 'camel' becomes **gaalota** 'camels' rather than **gaaloota** during plural formation.

The morphological synthesizers developed so far function generally in two different ways. The first one is generating well-formed words from a sequence of morphemes (/try/+/s/=/tries/), for instance Flex, and the other one is generating as many different word forms as possible from a given stem, for instance Englex. This study will consider the possibility of generating all possible words from an input stem and affixes.

2.2. COMPUTATIONAL MORPHOLOGY

Computational morphology is the study of computational analysis, synthesis, and treatment of word forms for eventual use in natural language processing (NLP) applications. It is intended to handle the task of morphology automatically with the use of computers and computational methods. The purpose of such work is to aid in the effective means of the storage of words in lexicons, and provide time-efficient lookup capabilities. Computational morphological methods also give linguists the ability to create grammars and specify how word forms should be stored in lexicons. This analysis also helps to answer fundamental questions of traditional morphology like which words are stored forms and which are based on derivational procedures in word formation. Generally, the tasks involved in computational morphology can be grouped into two parts: word-form synthesis and analysis and parts-of-speech (POS) or inflectional-category determination. Much work for natural languages has been done in the field of computational morphology. A number of systems have been developed with a wide variety of approaches to processing, for use in NLP systems including natural language generation, machine translation, information extraction and retrieval using natural language, text to speech synthesis, automatic written text recognition, grammar checking, and part-of-speech tagging. Most of these approaches have been developed for languages like English [42].

2.3. COMPONENTS OF MORPHOLOGY

Morphology is the study of the meaning of individual units or morphemes of language and is concerned with the structure of words. Words are the fundamental building blocks of a language. Every human language, spoken, signed, or written, is composed of words. Every area of speech and language processing, from speech recognition to machine translation to information retrieval on the web, requires extensive knowledge about words [2, 16]. This large number of words is produced from a limited collection of smaller units called morphemes. The task of morphology is thus to identify and describe the mechanisms behind this process.

The basic building blocks in morphology are morphemes. Morphemes are defined as the smallest units in a language to which a meaning may be assigned or, alternatively, as the minimal unit of

grammatical analysis. The form of a morpheme may be free or bound. A free morpheme occurs relatively freely within other words or morphemes. In other words a free morpheme may form a word on its own, e.g., */door/*. We call such words monomorphemic because they consist of a single morph. Bound morphemes, on the other hand, occur only in combination with other forms. The majority of affixes are bound morphemes. For example, the word */dogs/* consists of the free morph */dog/* and the bound morph */-s/* which is an affix [17].

Morphemes in a language are composed of affixes and stems. An affix is a bound morpheme that is attached to a base (root or stem). Affixes can be prefixes, suffixes, circumfixes and reduplication. A prefix is an affix that is attached in front of a base. In English, */re-/, /en-/, /in-/,* as in *reemploy, endanger, inaccessible,* are examples of prefixes. The hyphen (-) indicates the position of attachment. In Afaan Oromoo, /hin/, /ni/ as in /**hindeemnu**/ (we don't go) and /**nideemna**/ (let us go) are examples of prefixes. A suffix, on the other hand, is an affix that is attached after a base. The plural markers */-s/* and ***/-oota/*** of English and Afaan Oromoo, respectively, are examples of suffixes, as in *plants and* **namoota** (men). We have encountered far more suffixes than prefixes in Afaan Oromoo. A circumfix is the combination of prefix and suffix that together express some feature. In English the combination of the prefix *'em'* and suffix *'en'* as in the case of *embolden* is an example of circumfix. As there are no circumfix in Afaan Oromoo, we don't deal with it any more. Reduplication is used in Afaan Oromoo to show an action done repeatedly [37]. Example: - **cabse '**he broke' becomes **caccabse '**He broke something into pieces'

2.4. MORPHOTACTICS

Morphotactic investigates the constraints imposed on the order in which morphemes are combined. A word grammar determines the way this has to be done. Usually there are language-specific words grammars that help determine how morphemes are put together. These word grammars put constraints on morph patterning. For example, the English word *pseudohospitalization* is formed from /pseudo-/, /hospital/, /-ize/ and /- ation/. But these

morphemes can be concatenated randomly as *'hospitalationizepseudo, seudoizehospitalation, pseudohospitalationize'* if such word grammars don't restrict their formation [28, 40].

In Afaan Oromoo, from morphemes /**deem**-/(to go), and /**-eera**/ different word forms can be formed, as in /**eeradeem**/, and /**deemeera**/, but the grammatically correct one is /**deemeera**/ (he has gone) because the suffix has to follow the stem in Afaan Oromoo. Therefore, a system for morphological synthesis needs to have a component that determines the well formedness of different word forms [34].

Morphotactics is responsible for governing the rules for the combination of morphemes into larger entities. But phonological rules may apply and change the shape of morphemes. Morphophonology, a discipline that merges morphology and phonology deals with these changes and their underlying reasons. It is common to see phonological influences when morphs concatenate to form words. In many cases this concatenation process will induce some phonological change in the vicinity of the morpheme boundary. Assimilation is an example. It is a process where the two segments at a morpheme boundary influence each other, resulting in some feature change that makes them more similar. Take, for example, the English prefix /in–/ where the /n/ changes to /m/ before labials as in *in + mature= immature (m is the labial considered in mature)* [24]**.** The same effect is also observed in Afaan Oromoo; for instance, **barat-** 'to learn' + **-na** 'person marker' becomes **baranna** 'we learn'. We observe here that 't' has changed to 'n' [37].

The general rules of morphotactics in Afaan Oromoo are as follows:

- The rule for inflected nouns: Noun form = noun stem + noun suffix(es)
- The rule for inflected verbs: Verb from = (prefix) + verb stem + verb suffix(es)

The details of morphotactics rules are presented in Appendix D.

To sum up, a system for morphological synthesis should consider the morphotactic, morphophonological and phonological features in order to generate linguistically acceptable word forms.

2.5. TYPES OF MORPHOLOGICAL PROCESSES

There are two productive ways to form words from morphemes: inflection and derivation [24].

Inflectional Morphology: deals with the combination of a word with a grammatical morpheme, usually resulting in a word of the same class as the original stem, and serving some syntactic function, for example plurals of nouns. They do not change the part-of-speech category but the grammatical function (also called morphosyntactic information) is changed. The different forms of a word are produced by inflection. In English, the word '*work*' is a verb, and inflectional forms like *'works', 'working',* and *'worked'* are produced by adding the 3rd person singular maker /-s/*,* the present continuous marker /-ing/ and the perfective /-ed/ respectively. These four word forms of *'work', i.e.'work', 'works', 'working',* and *'worked'* are all verbs and there is no change in the part-of-speech category due to the affixation.

Derivational Morphology: creates new words (i.e., words with a different part-of-speech category) by adding a bound morpheme to a stem. Derivation can be applied recursively, i.e., words that are already the product of one derivation process can undergo the process again. The following is an example from English: large (adj.) = enlarge (en- + large) (v) = enlargement (enlarge + -ment) (noun), and from Afaan Oromoo: **bar- '**to know' (v) _ **barumsa** 'education' **(bar-+-umsa)**(noun).

Compounding: is defined as a process of forming new words by combining different lexical categories [44]. However, it is not the case that every two words combine to form a compound form. Rather, every language follows certain rules by which it forms its compound. In Afaan Oromoo, the combination of **abbaa** 'father' + **lafa** 'land' forms new word **abbaa lafaa** 'landlord'. The rules of compound word formation in Afaan Oromoo is unpredictable, and thus needs futher linguistic study in the language. Hence, it is out of the scope of this research.

On the basis of structural changes of the stem and other morphemes during affixation, morphological processes can also be classified as *linear* or *nonlinear* [25]. In linear morphology affixes are added to the stem without changing the internal structural of the stem, though some

changes might take place at the boundary of stems and affixes. On the other hand, morphological systems where the internal structure of the morphemes changes during the addition of suffixes are classified as nonlinear morphology. English pluralization pertains to linear category where as Semitic languages features nonlinearity. Morphological processes in Afaan Oromoo are mostly linear in nature.

2.6. APPROACHES OF MORPHOLOGY

According to [23], the different approaches to morphology are categorized as corpus based and rule-based.

- **Corpus-based approaches**: Corpus-based approaches, also called machine learning approaches, do not strictly follow explicit theory of linguistics. The approaches are completely based on training and testing corpora, which constitute the input data. Approaches in this category use some algorithms to learn, say about the stem classification process of a language from a given lexicon and perform the synthesis based on this knowledge. Moreover, the employed algorithms are subject to modification and further fine-tuning during the operation.

  Corpus-based approaches are further divided into *supervised* and *unsupervised* based on the type of training corpora they use. Unsupervised approaches use heuristics or probability information generated from the test corpora to generate the morphological synthesis system. In this approach, no sample outputs are given. According to [23, 29, and 30] this approach reduces the cost of browsing annotated corpora. Supervised machine learning, on the other hand, is the search for algorithms that reason from externally supplied instances to produce general hypotheses, which then make predictions about future instances. In other words, the goal of supervised learning is to build a concise model of the distribution of class labels in terms of predictor features. The resulting classifier is then used to assign class labels to the testing instances where the values of the predictor features are known, but the value of the class label is unknown. As we don't use this approach, we don't deal with it further.

- **Rule based approaches**: Rule-based approaches are based on a theory of morphology laid down by experts. This group of methods enables one to incorporate sophisticated linguistic theory, such as generative phonology, into computational morphology processes. The rules are either created by linguists or automatically by computer programs, and may contain a large number of morphological, lexical and/or syntactical information. With human rule creation, there is a large set of manually constructed rules based on a specific grammar, written in a formal notation so that they can be used by the computer for further parsing. Adding a rule to the system may involve over-generation, i.e., one extra rule can result in more harm to the accuracy of rule-based approach. The rule-based tagging system has also some limitations: requires hand-written rules, costly and time consuming. Because of their reliance on linguistic theories, systems developed using rule based approaches are often efficient and produce better quality outputs. Moreover, rule based approaches have been tested quite for a long period time now, and there are a number of systems developed using this approach both for commercial and research purposes.

As described earlier, most of the morphological generators and recognizers are done using the rule-based approach. Morphological systems developed using rule-based approaches have the following advantages over those developed using corpus-based approaches [23].

Data-compaction: Morphological systems developed using rule-based approaches require less storage than morphological systems developed using corpus-based approaches.

High speed: Morphological systems developed using rule-based approaches are faster than those developed using the corpus-approaches.

Better Effectiveness: Morphological systems developed using rule-based approaches are reported to have better accuracy than those developed using corpus-based approaches.

Better adaptability: Morphological systems developed using rule-based approaches are easier and more straightforward to twist or modify for the purpose of correcting errors.

On the other hand, morphological systems developed using corpus-based approaches have the following advantages over those developed using rule-baed approaches [52]:

Saving human resources: Once the system is trained, classification is done automatically with no or little human intervention

Consistent classification: The classification is done consistently on repeation

Automatic rule formation: Human resources are not needed to make rules

2.7. THE TWO-LEVEL MORPHOLOGY

A great revolution in the area of morphology started to appear in 1983 when Kimmo Koskenniemi, a Finnish computer scientist, produced his dissertation *Two-level morphology: A general computational model for word-form recognition and generation [18, 19]*. The model incorporates a general formalism for making morphological descriptions of particular languages, and a language independent program implementing the model. Thus, it is used as a framework for computational morphological recognition and synthesis. The system is based on lexicon and two-level rules. It is two-level in the sense that a word is represented as a direct, letter-for-letter correspondence between its lexical or underlying form and its surface form. An example showing the generation of the word *'chased'* is given in two-level representation as follows. In this case, + is a morpheme boundary symbol and 0 is a null character.

Lexical form	C	h	a	s	e	+	e	d
Surface form	C	h	a	s	0	0	e	d

The subsequent work after Koskenniemi's is the implementation of two-level model by Lauri Karttunen and others named as KIMMO [26]. The main components of the KIMMO parser are shown in Figure 2.2. It had two components: the rules component and the lexical component, or lexicon. First, the rules component consisted of two-level rules that accounted for regular phonological or orthographic alternations, such as /*chase*/ versus /*chas*/. Second, the lexicon

listed all morphemes (stems and affixes) in their lexical form and specified morphotactic constraints. For example, the lexicon would have included lexical entries for the verb stem /*chase*/ and the suffix /*-ed*/, and would have specified their relative order. Using these data components were two processing functions, the generator and the recognizer. The generator would accept as input a lexical form such as /*cry*/+/*s*/ and return the surface form /*cries*/. The recognizer would accept as input a surface form such as /*cries*/ and return an underlying form divided into morphemes, namely /*cry*/+/*s*/, plus a gloss string such as V+SINGULAR MAKER.

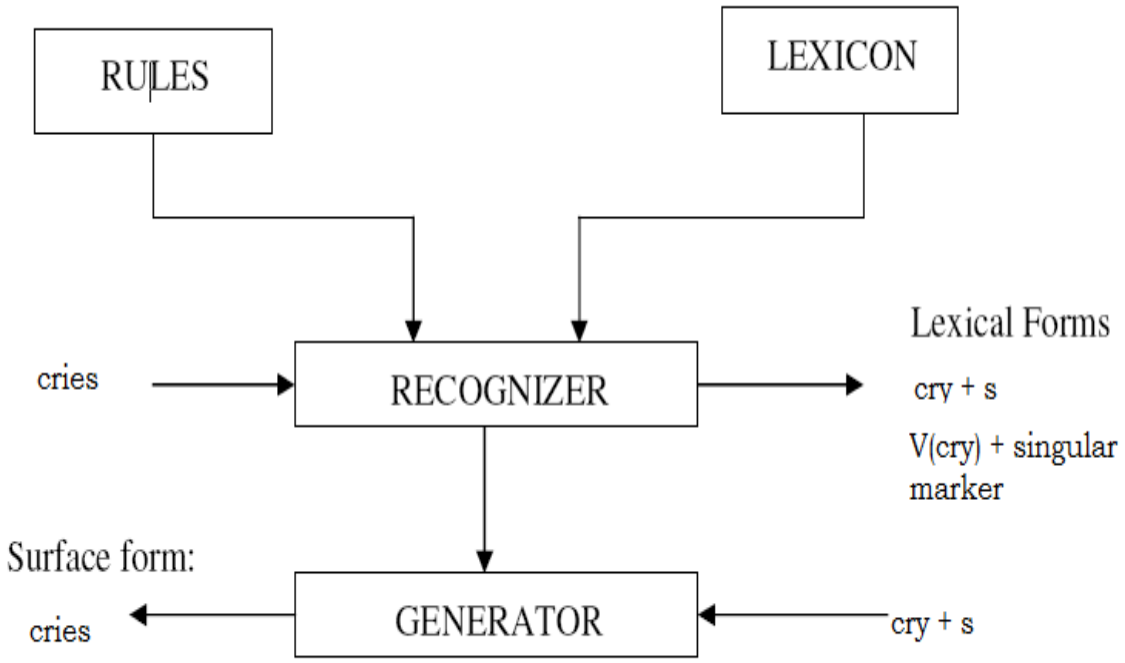

Fig 2.2: Main components of Karttunen's KIMMO Parser

As it is a general model, two-level morphology is suitable for word formation in Afaan Oromoo and hence is used in this study during suffixation in the phonological alternations. The correspondence between the underlying form and its lexical equivalent is represented as underlying rules of formation.

2.8. SUMMARY

Various morphological concepts are discussed in this Chapter. The discussions included in this Chapter on rule-based approaches will be applied in Chapter Five and Six to develop the morphological synthesizer lexicons and algorithms respectively. The next chapter discusses related works in the area of morphology that have been done for different languages.

CHAPTER THREE: RELATED WORKS

Morphological synthesis is one of the most popular research areas in the field of Natural Language Processing. Some of the research works have been done in different languages. Most of them have employed rule based approach of development based on the morphological properties of the language. The next sections briefly review the work done on morphological synthesizer.

3.1. MORPHOLOGICAL SYNTHESIZERS FOR ENGLISH

English is one of the well researched languages in the area of natural language processing. Morphological systems for the language have been done as the input for higher level package components. Here we review two of the systems: Englex and Flex

3.1.1. Englex

Englex is a description of English morphology and lexicon using the two-level model as it is implemented by PC-KIMMO [47]. It consists of the three basic components that make up any PC-KIMMO description: a set of phonological (or orthographic) rules, a lexicon, and a word grammar. First, the rules component consisted of two-level rules that accounted for regular phonological or orthographic alternations, such as *chase* versus *chas*. Second, the lexicon listed all morphemes (stems and affixes) in their lexical form and specified morphotactic constraints. For example, the lexicon would have included lexical entries for the verb stem *chase* and the suffix *-ed,* and would have specified their relative order. Third, word grammar component offers a more powerful model of morphotactics and can deduce the lexical category (part-of-speech) of a word. Using these data components were two processing functions, the Generator and the Recognizer. The Generator would accept as input a lexical form such as *spy+s* and return the surface form *spies*. The Recognizer would accept as input a surface form such as *spies* and return an underlying form divided into morphemes, namely *spy+s,* plus a gloss string such as N+PLURAL.

3.1.2. Flex

Morphological generator can be used on its own in applications that produce natural language but do not contain a standard realization component [48]. The Flex description of the morphological generator is derived automatically from the analyser through a compilation process which is computationally very cheap (taking just a few seconds). A benefit of this arrangement is that after modifications to the analyser, the generator can be updated automatically and will reflect the modifications without any further manual effort. Just like the analyser, it supports various command line options and an interactive mode. The input to the generator is expected to be a sequence of tokens of the form *lemma + inflection, label*, where *lemma* specifies the lemma of the word form to be generated, *inflection* specifies the type of inflection (i.e. s, ed, en or ing), and *label* specifies the PoS of the word form.

The compilation process inverts each analyser regular expression pattern / action pair to derive the generator. It does this by simulating the effect of the analyser action on the pattern; this produces the new generator pattern. The new action consists of a call to a function that removes the last *n* characters from the input, where *n* is the number of characters the analyser action adds, and then appends the characters that were removed by the analyser action. So, for example, the following example shows the compilation process.

{A}+{C}"ied" {return (lemma (3,"y","ed"));} (analyser)

{A}+{C}"y+ed" {return (glemma(4,"ied"));} (generator)

The accuracy of the system was found to be 99.98%. The synthesizer is freely available to the NLP research community, and is currently being applied in a number of practical NLP systems.

3.2. MORPHOLOGICAL SYNTHESIZER FOR TELUGU

Telugu is a language spoken by over 50 million people in India. Morphological synthesizer for nouns and verbs, TelMore, has been developed by Madhavi Ganapathiraju and Lori Levin at Language Technologies Institute, Carnegie Mellon University, USA [38]. It has been developed using rule based approach. Telugu nouns were classified according to the properties of their inflectional patterns (declensions) set by linguists into 8 classes. Telugu verbs also fall into three conjugations based on their morphology and end letter of the root. The system accepts a noun and a predefined lexical class as input, and generates all the applicable noun forms, namely nominative, genitive, accusative, dative, vocative and instrumental forms for masculine, feminine or neutral genders and for singular and plural numbers. A data set of nouns and verbs has been created by native Telugu speakers for testing the morphological generator. Where required, the lexical class (declension, conjugation) of the root noun is specified, and the verb is entered in the required infinitive form. A random selection of words from the word lists was used in generating morphological forms, and the results are presented to native speakers for evaluation. A second iteration after corrections to program based on errors in first iteration produced accurate results. Automatic generation of plurals is accurate for most nouns. The tool is developed in Perl® and is made available with a web interface. Current version of the toolkit is available in open source for review and enhancement by the World Wide Web community. The generator has produced promising results for both nouns and verbs.

3.3. BENGALI MORPHOLOGICAL SYNTHESIZER

The other Indian language for which morphological synthesizer has been implemented based on rule based approach is Bengali [25]. Bengali is among one of the agglutinative and highly inflectional language families having many variant forms for verbs, nouns and pronouns. This property calls for efficient morphological synthesizer. The system selects appropriate suffixes, the order among them and concatenates them to the stem according to the rules. The language is characterized by linear morphology in which affixes are added to the root without changing the internal structure of the root, though changes might take place at the boundary of the root and suffixes. In the implementation of the noun morphology, suffixes were categorized into three

classes according to their functions like case markers, emphasizers and so on. The verbs were classified based on their syllable structure into 24 classes. The system was implemented in Java and hence platform independent. The performance of the system has been evaluated on a large number of words randomly selected and was found to be mostly accurate.

3.4. MORPHOLOGICAL SYNTHESIZER FOR AMHARIC

Morphological synthesizer for Amharic perfect verb forms has also been done by Kibur Lisanu using Hybrid approach (rule based and artificial neural network) [24]. The algorithms suitable for Amharic language word synthesis have been designed from scratch by employing the linguistic properties specific to the language. The lexicon (dictionary) has been manually designed from morphological patterns of the roots. The roots have been classified into classes based on consonantal lengths. The study adopted the combination of rule-based and neural network approaches to design and develop a prototype, referred as *AmharicMorphologicalSynthesizer*. The rule-based approach generates all the roots successfully (near 100%) where as the neural network predicts the type of roots in the test data set with an accuracy of 81.48%. It was shown that morphology of Amharic is well suited to the root and template morphology like Arabic morphology. The System was implemented using Visual Basic 6 and artificial neural network tool called Brainmaker.

The result obtained using the small manually constructed root table encourages the undertaking of further research in the area, especially with the aim of developing a full-fledged Amharic morphological synthesizer

3.5. ARABIC MORPHOLOGY SYNTHESIZER

Morphological synthesizers for Arabic language have been developed by a number of researchers at different times and using different methodologies. Among such efforts, it is worth considering the works of Violetta, Abdelhadi, and Teruko [39] first. According to these researchers, Arabic has non-concatinative morphology, and generating large number of verb variants is problematic. Verb stems are formed from trilateral or quadrilateral roots by

derivational combinations of root morpheme and vowel melody. For example, the Arabic stem **katab (**he wrote**)** is composed of the morpheme **ktb (**the notion of writing**)** and the vowel melody morpheme 'a-a'. The two are coordinated according to the pattern CVCVC (C=consonant, V= vowel).

The researchers have identified 15 trilateral patterns, all of which have undergone stem changes with respect to tenses (perfect and imperfect), voices (active and passive) and moods. The implementation has been tested on verbal stems of different classes.

The other morphological synthesizer important to look for Arabic is that of M.G. KHAYAT et al [53]. This application is based on linguistic principles of Arabic morphology, and designed as three modules for particles, nouns and verbs respectively. The modules consist of rules that encode the linguistic principles of word construction in Arabic. The mode (analysis or synthesis) of operation is automatically determined by the values associated with the word and its properties. Classification based morphology has been used as development methodology, and the system was developed using prolog. It is currently being used as a component of a natural Arabic understanding system NAUS.

3.6. SUMMARY

In this chapter, different works which are done on morphological synthesis for different languages have been discussed. To the best of our knowledge, there is no research work done for Afaan Oromoo Morphological Synthesizer. From the reviewed works, we have observed that morphology is mostly language specific and morphological synthesizer systems for various languages have been developed mostly using rule based approach.

CHAPTER FOUR: AFAAN OROMOO MORPHOLOGY

Like a number of other African and Ethiopian languages, Afaan Oromoo has a very rich morphology. It has the basic features of agglutinative languages where all bound forms (morphemes) are affixes. In agglutinative languages like Afaan Oromo, Amharic and Zulu most of the grammatical information is conveyed through affixes (prefixes, infixes and suffixes) attached to the roots or stems. Both Afaan Oromoo nouns and adjectives are highly inflected for number and gender. For instance, according to [6] in comparison to the English plural marker s (-es), there are more than 12 major and very common plural markers in Afaan Oromo nouns (e.g. **-oota, -ooli, -wwan, -lee, -an, -een, -oo**, etc.). Afaan Oromo verbs are also highly inflected for gender, person, number and tenses. Moreover, possessions, cases and article markers are often indicated through affixes in Afaan Oromo. Since Afaan Oromoo is morphologically very productive, derivations and word formations in the language involve a number of different linguistic features including affixation, reduplication and compounding [7, 9]. Although Afaan Oromo words have some prefixes, suffixes are the predominant morphological features in the language. This Chapter discusses the structural property of Afaan Oromoo in general and the morphology of nouns and verbs in particular. The study focuses on these grammatical categories because they are the most productive and widely used grammatical categories, and others can easily be adapted from them.

4.1. AFAAN OROMOO WRITING SYSTEM

The Oromoo writing system is a modification to Latin writing system. Thus, the language shares a lot of features with English writing with some modification [3]. The writing system of the language is known as “qubee Afaan Oromoo” is straightforward which is designed based on the Latin script. Thus letters in English language are also in Oromoo except the way it is written [35]. A detailed description of Oromoo writing system can be found in any text related to the language but readers are referred to [5] and [43] for the detailed discussion of the language’s writing system.

4.2. SYLLABLES IN AFAAN OROMOO

Syllabification is language-dependent: each language has its own structure of syllables. For example, in English more than two consonants can come consecutively in a single word as in 'screen'. But, in Afaan Oromoo more than two consonants cannot come together except in diagraphs. Hence, there are four types of syllable structure in the language. These structures include CV, CVV, CVC and CVVC. All of these can be found at word initial, medial and final positions. A valid word can be composed from the combination of one or more type(s) of these structures. The words like **eelee, ooluu** and etc seem to start with vowels, but linguists argue that there is hidden glottal stop called **hudhaa (')** in front of any word that seems to start with vowel. In light of this, we say that every syllable in Afaan Oromoo starts with consonant. The following are just few examples.

CVC	shan (five)
CV	na-ma (man), tu-re(stay), ku-ma(thousand)
CVVC	deem-(go)
CVV	Boo-naa (personal name)

Majority of words in Afaan Oromoo are disyllabic with considerable number of trisyllabic ones. Monosyllabic and quadrisyllabic words are rare [34]. Of course this description is concerning mostly disyllabic words and morphological processes can produce a different result.

4.3. NOUN MORPHOLOGY

Broadly, it is possible to categorize the major types of suffixes in Afaan Oromoo into three basic groups: *derivational*, *inflectional,* and *attached* suffixes. Afaan Oromoo *attached suffixes* are particles or postpositions like *-arra*, *-bira, -irra, -itti* and *-dha* while *inflectional suffixes* comprises the most frequent and dominant suffixes such as *–n, -lee, -een, -icha, -tu, -oo, -oota* and *-wwan*. Afaan Oromoo *derivational suffixes* such as *-achuu, -eenyaa, -ina* and *-ummaa* are often used for formation of new words in the language following the stems or base forms of Afaan Oromoo words. In complex word structure, certain set of suffixes conventionally come in

a particular sequence before or after other suffixes. The most common order/sequence of the above major three Afaan Oromoo suffixes (within a given word) is: *<stem><derivational suffixes><inflectional suffixes><attached suffixes> [13]*. The order is relative because one or more of these suffixes may be absent. The next subsections are primarily concerned with the description of noun inflection and derivation.

4.3.1. Noun Inflections

Afaan Oromoo nouns are words used to name any of categories of things, people, places or ideas [3]. Nouns are inflected to indicate different grammatical functions such as number, gender, definiteness and case. Inflectional suffixes are combined with stem usually resulting in a word of the same class as the original stem. The principles of noun inflection here apply to nouns and adjectives.

4.3.1.1. Pluralization

A singular is marked by zero morphemes where as a plural noun is marked morphologically by suffixing the morpheme like **-oota, -oolii, -een, -lee, -wwan, -yyii, -eetii, -ii, -oo** to the base as free alternates [13,33]. It is difficult to predict which suffix is for which noun, but there is a possibility of using all these suffixes as plural makers. In certain more complicated situations Afaan Oromoo noun may take more than one plural markers concatenating and suffixing them one after the other, just to indicate the double plural form of the noun as in: *manneenota (mana + een + ota)* or *manneenotaawwan (mana + een + ota + wwan).* Even though the usage of some suffixes is not common, they can be used and accepted by the speakers of the language. Linguists agree that some groups of suffixes are most preferably applied to almost all nouns, and the others are used with only some words. Hence, we can categorize them according to universal usage, those attached to stems ending in specific consonantal phoneme and those that end in some group of phonemes.

The first category of suffixes, **-oota, -oolii, -oolee,** is affixed to virtually any noun to form plural. But all the terminal vowels, except nouns having the phoneme /o/ as the last vowel(s), in the citation form get deleted when these plural formative suffixes are attached to nouns. These

suffixes are used with nouns that end with both long and short vowels. Suffixes **–wwan, -lee** don't delete the final vowels and are used with nouns terminating in long vowels. Though not common, these suffixes can also be used as universal suffixes. In Afaan Oromoo nouns pluralized by the suffixes set **-oota, -oolii, -oolee** in long form or **-ota, -olii, -olee** in short form depend on the vowel present in the syllable that precedes the last syllable. If the vowel in the syllable that precedes the last syllable is short, the noun takes **-oota** as in the case of **barataa** (student) becomes **barattoota** (students), while the noun having long vowel in the syllable prior to the last syllable takes the suffix **-ota** as in the case of **mammaaksa** (proverb) becomes **mammaaksota** (proverbs). Following are some examples corresponding to the four suffixes.

Noun	**Gloss**	**Plural**	**Gloss**
Barataa	student	barattoota	students
Warra	parent	warroolii	parents
Gaaffii	question	gaaffiiwwan	questions
Jabbii	calf	jabbiilee	calves

The other plural making suffixes that behave differently from the above ones are **-an**, and **-een**. All nouns that take these plural maker suffixes make plural noun by doubling the consonant in the last syllable. These nouns mostly end in the consonantal phoneme **l, m** and **r** in the case of **–an,** and **b, d, g, k, n** in the case of **-een** followed by short vowel. Following are some examples.

Noun	**Gloss**	**Plural**	**Gloss**
Beera	old woman	beerran	old women
Eessuma	uncle (maternal)	eessumman	uncles
Wasiila	uncle (paternal)	wasiillan	uncles
muka	tree	mukkeen	trees
mana	house	manneen	houses

The third category of plural makers is the suffix **–eyyii**. It forms plural by dropping complex endings that form nouns like **–eessa, eensa, eettii** and attaching the plural maker **-eyyii.**

Noun	**Stem**	**Suffix**	**Plural**
sooressa	soor-	-eyyii	sooreyyii
waraabessa	waraab-	-eyyii	waraabeyyii

Even though there may be several plural maker suffixes in different dialects in Afaan Oromoo, the suffixes discussed above are the common ones. All these suffixes cannot be applied for all nouns, but more than one suffix can be applied for one noun. It is difficult to categorize nouns according to the suffixes they take for making plural as the way plural makers are attached to the nouns may or may not be similar. Table 4.1 summarizes the categories

Table 4.1: Suffixes Category

Suffix	**Category**
-oota,- oolii, oolee	Suffixes that delete the last vowel
-wwan, -lee	Suffixes that don't delete the last vowel
-een, -(a)n	Suffixes that double last consonant
-eeyyii	Suffixes that drop –eessa/eensa

4.3.1.2. Definiteness

Afaan Oromoo has no indefinite articles (corresponding to English *a*), but it indicates definiteness (English *the*) with suffixes on the noun: **-icha** for masculine nouns and **-ittii** for feminine nouns. Vowel endings of nouns are dropped before adding these suffixes: **karaa** 'road',**karicha** ' the road', **nama** 'man', **namicha** 'the man', **haroo** 'lake', **harittii** 'the lake'. Note that for animate nouns that can take either gender, the

definite suffix may indicate the intended gender: **qaalluu** 'priest', **qaallicha** 'the priest (masculine)', **qallittii** 'the priest (feminine)'. The definite suffixes appear to be used less often than *the* in English, and they do not to co-occur with the plural suffixes [21].

4.3.1.3. Cases

Case is a grammatical category of nouns that indicates the nature of their relationship to the verb in sentences [37]. The number of cases varies from language to language. In this regard, nouns in Afaan Oromoo are inflected for nominative, ablative, instrumental and locative cases. Each case is described and a summary is given in Table 4.2.

a) Nominative case

The nominative case is used for nouns that are the subjects of clauses. Most nouns ending in short vowels with the preceding single consonant drop the final vowel and add ***-ni*** to form the nominative. Following certain consonants, assimilation changes either the ***n*** or that consonant.
E.g. **nama** 'man', **namni** 'man (nom.)'

If a final short vowel is preceded by two consonants or a geminated consonant, **-i** is suffixed.
For instance, ibsa 'statement', **ibsi** 'statement (nom.)'

If the noun ends in a long vowel, **-n** is suffixed to this.
For instance, **maqaa** 'name', **maqaan** 'name (nom.)'

If the noun ends in **n,** the nominative is identical to the base form.
For instance, **afaan** 'mouth, language (base form or nom.)'

b) Instrumental Case

The instrumental is used for nouns that represent the instrument ("with"), the means ("by"), the agent ("by"), the reason, or the time of an event.
The suffixes **–n, -tiin** following a long vowel or a lengthened short vowel, **-iin** following a consonant, and **-dhaan** following a long vowel indicates instrumentation.

For instance, **harka** 'hand', **harkaan** 'by hand, with hand'

For instance, **Afaan Oromoo** 'Oromo (language)', **Afaan Oromootiin** 'in Afaan Oromoo'

For instance, ***halkan*** 'night', **halkaniin** 'at night'

For instance, **yeroo** 'time', **yeroodhaan** 'on time'

c) Locative Case

The locative is used for nouns that represent general locations of events or states, roughly *at*. For more specific locations, Afaan Oromoo uses prepositions or postpositions. Postpositions may also take the locative suffix. The locative also seems to overlap somewhat with the instrumental, sometimes having a temporal function. The locative is formed with the suffix **-tti**.

For instance, **harka** 'hand', **harkatti** 'in hand'

For instance, **guyyaa** 'day', **guyyaatti** 'per day'

For instance, **jala, jalatti** 'under'

d) Ablative Case

The ablative is used to represent the source of an event; it corresponds closely to English *from*. The ablative, applied to postpositions and locative adverbs as well as proper nouns, is formed in the following ways:

- When a word ends in a short vowel, this vowel is lengthened.

For instance, **biyya** 'country', **biyyaa** 'from country'

For instance, **keessa** 'inside, in', **keessaa** 'from inside'

- When the word ends in a long vowel, **-dhaa** is added (as for one alternative for the dative).

For instance, **Finfinneedhaa** 'from Finfinnee (Addis Ababa)'

For instance, **gabaa** 'market', **gabaadhaa** 'from market'

- When the word ends in a consonant, **-ii** is added (as for the genitive).

For instance, **Hararii** 'from Harar'

- Following a noun in the genitive, **-tii** is added.

For instance, **mana** 'house', **buna** 'coffee', **mana bunaa** 'cafe', **mana bunaatii** 'from cafe'

- An alternative to the ablative is the postposition *irraa* 'from' whose initial vowel may be dropped in the process:

For instance, **gabaa** 'market', **gabaa irraa**, **gabaarraa** 'from market'

Table 4.2: Summary of case makers

case	Declensions						
	Class1	Class2	Class3	Class4	Class5	Class6	Class 7
Nominative	n	i	ni	i	-	n, ni	n
instrumental	tiin, dhaan	-	-	-	-iin	-	tiin, dhaan
locative	tti	tti	tti	tti	-	tti	tti
ablative	dhaa, rraa	rraa, a	rraa, a	rraa, a	-	-	dhaa, rraa

As it can be seen from Table 4.2, the first column shows the available cases, and the rest columns represent the declensions of nouns in Afaan Oromoo. Each cell of the table is an intersection of a case and a class containing the case suffix applicable for the respective class. Table 4.2 shows the usage of the above case maker suffixes for the available cases in each declension by using examples.

Table 4.3: Examples of case maker suffixes usage

Declensions	**Noun Stem**	**Examples**
Class1	Maatii	Maatii**n**, maatii**dhaan**, maatii**tiin**, maatii**rraa**, maatii**dhaa**, maatii**tti**
Class2	Beera	Beerr**i,** beera**tti**, beera**a,** beera**rraa**
Class3	Mana	Man**ni**, mana**tti**, mana**a**, mana**rraa**
Class4	Waraabessa	Waraabess**i**, waraabessa**tti**, maraabessa**a**, waraabessa**rraa**
Class5	Halkan	halkan**iin**
Class6	Gamna	Gam**ni**, gamna**tti**
Class7	Fayyisaa	Fayyisaa**n**, Fayyisaa**tti**, Fayyisaa**rraa**

4.3.2. Noun Derivation

Afaan Oromoo is very productive in word formation by different means. One method is the use of different derivational suffixes. In Afaan Oromoo, derivational suffixes enable a new word, often with a different grammatical category to be built from stem/root of other words. But the distribution of suffixes is unpredictable since some nouns are formed with different suffixes. There are three derivational processes that create nouns in Afaan Oromoo.The subsequent sections discuss the processes in detail.

4.3.2.1. Nouns Derived from another Nouns

Abstract nouns are derived from other nouns by adding the suffix **-ummaa,-eenya or -ooma** to the noun stems. Thus, when these abstract noun formative morphemes are added to nouns, the final vowels of these words are deleted as the following set of examples illustrate.

Noun	***gloss***	***derived noun***	***gloss***
Ollaa	neighbor	ollummaa	neighborhood
Bilisa	free	bilisummaa	freedom
Fira	relative	firummaa	relationship
Garba	slave	garbummaa	slavery
Gooftaa	boss/lord	gooftummaa	lordship

Nama	man	namooma	humanity
Nagaa	peace	nageenya	peaceful
Jabaa	strong	jabeenya	strength

4.3.2.2. Nouns Derived From Verbs

In Afaan Oromoo the nominal can be derived from the verb stem by suffixing the morphemes like **-aa**,**-eenya, -tuu, -ina, -noo, -ii, -ee, -iinsa, -iisa, -umsa, -maata, -aatii.** The following examples indicate the derivation of such nouns.

Verb	***gloss***	***noun***	***gloss***
qab-	to have	qabeenya	property
rak-	to suffer	rakkina	problem
hubat-	to understand	hubannomoo	understanding
falm-	to argue	falmii	argument
tiks-	to shepherd	tiksee	shepherd/guardian
dalag-	to work	dalaga	work/job
barsiis-	to teach	barsiisaa	teacher
bulch-	to govern	bulchiinsa	government
qot-	to farm	qotiisa	farming
bar-	to learn	barumsa	education
fur-	to solve	furmaata	solution
lol-	to fight	loltuu	soldier

4.4. VERB MORPHOLOGY

Verbs are words that tell us the state of doing or being. Verbs are morphologically the most complex POS in Afaan Oromoo, with many inflectional forms; numerous words with other POS are derived primarily from verbs. Generation of syntactically and semantically correct sentences requires appropriate choice among the different forms of verbs. There are two major criteria to identify verbs from other word categories: syntax and morphology. In the former case, verbs function as predicates in a simple sentence and they are found at the end of a sentence. In the later case, the agreement of verb with the number, gender and/or person of the subject, proper case markers for the different nominal forms and expression of the tense, aspect of the verb and

number, specificity of the of nouns are some of the important morphological constraints governing correct generation.

An Afaan Oromoo verb consists minimally of a stem, representing the lexical meaning of the verb, and a suffix, representing tense or aspect and subject agreement. For example, in **dhufne** 'we came', **dhuf-** is the stem ('come') and **-ne** indicates that the tense is past and that the subject of the verb is first person plural.

As in many other Afro-Asiatic languages, Afaan Oromoo makes a basic two-way distinction in its verb system between the two tensed forms, past (or "perfect") and present (or "imperfect" or "non-past") [31]. Each of these has its own set of tense/agreement suffixes. There is a third conjugation based on the present which has three functions: it is used in place of the present in subordinate clauses, for the jussive ('let me/us/him, etc. verb', together with the particle **haa**), and for the negative of the present (together with the particle **hin**). For example, **deemne** 'we went', **deemna** 'we go', **akka deemnu** 'that we go', **haa deemnu** 'let's go', **hin deemnu** 'we don't go'. There is also a separate imperative form: **deemi** 'go (singular).

4.4.1. Verb Forms (Conjugation)

The rules of morphophonemics in Afaan Oromoo operate on consonant -consonant sequences, consonant -vowel or vowel-vowel sequences across morpheme boundaries. The consonant -consonant sequences originate in the following situations.

a. The last consonant of the verb stem can be followed by the initial consonant of person makers such as **–ta, -ti, -na, -tani, -tu, -nu** and **–tana**, and the negative suffix **–ne**. Table 4.3-Table 4.5 present these suffixes in the present, past and imperative tenses respectively, categorizing them under affirmative(AFF) and negative indicators(NEG).

Table 4.4: Inflectional suffixes that indicate present tense

Present main clause				**Present subordinate clause**	
Person		AFF	NEG	AFF	NEG
singular	1 2 3male 3female	-a -ta -a -ti	-u -tu -u -tu	-u -tu -u -tu	-ne -ne -ne -ne
plural	1 2 3	-na -tani -ani	-nu -tani - ani	-nu -tani - ani	-ne -ne -ne

Table 4.5: Inflectional suffixes that indicate past tense

Past main clause				**Past subordinate clause**	
Person		AFF	NEG	AFF	NEG
singular	1 2 3male 3female	-e -te -e -te	-ne -ne -ne -ne	-e -te -e -te	-ne -ne -ne -ne
plural	1 2 3	-ne -tani -ani	-ne -ne -ne	-ne -tani -ani	-ne -ne -ne

Table 4.6: Inflectional suffixes that indicate imperative

	Imperative	
Person	**AFF**	**NEG**
singular	-i	-in
plural	-aa	-inaa

b. The last consonant of the verb root can be followed by the initial **s** of the causative verb extensions **-s, -sis, -siis**

c. The last consonant of the verb stem can be followed by nominal agents suffix **–tuu**

E.g. tum tumtuu

As can be seen from the tables, we observe two kinds of aspects namely perfective (to indicate the action that has been completed) and imperfective (the action that has not been completed yet but can be completed at any time) aspects indicated by different set of suffixes. The former is indicated by the suffixes **–e, -ne, -te, -tan,** and the latter is indicated by the suffixes **–a, -na, -ta, -ti, -tu, -u**. In Afaan Oromoo person maker suffixes ending in **-e** and **-an** are perfective markers while those ending in **-a, -i** and **-u** are imperfective markers [4]. The suffix **–ne** is used as negation marker in subordinate clauses of imperfective aspect, and both main and subordinate clauses of perfective aspect. The suffixes **-i, -in, -inaa**, and **-aa** are imperative indicators. The suffixes starting with consonant interacts with verbal stems in the consonant-consonant association. In addition, causative suffixes such as **–siis-** can interact with consonantal start of verbs as in **barat-** 'to learn' +**-siis** becomes **barachiis** 'make to learn'. Noun agent suffix **–tuu** can also participate in the consonant-consonant interaction as in the case of **hat-** 'to steal' +**-tuu** produces **hattuu** 'thief'.

n Afaan Oromoo verb stem pattern includes those that end in consonant, and apostrophe (hudhaa). We take the stem **beek-**'to know' and **ka'**'to stand' to illustrate the two paradigms.

i. Main clause affirmative present

Table 4.7: The present affirmative in main clause

Person		**Beek-**	**Ka'**
singular	1	beeka	Ka'a
	2	beekta	Kaata
	3m	beeka	Ka'a
	3f	beekti	kaati
plural	1	beekna	Kaana
	2	beektani	Kaatani
	3	beekani	Ka'ani

ii. Main clause negative present

Table 4.8: The present negative in main clause

Person		**Beek-**	**Ka'**
singular	1	hinbeeku	hinKa'u
	2	hinbeektu	hinKaatu
	3m	hinbeeka	hinKa'u
	3f	beektu	hinkaatu
plural	1	hinbeeknu	hinKaanu
	2	hinbeektani	hinKaatani
	3	hinbeekani	hinKa'ani

iii. Main clause affirmative past

Table 4.9: The past affirmative in main clause

Person		**Beek-**	**Ka'**
singular	1	beeke	Ka'e
	2	beekte	Kaate
	3m	beeke	Ka'e
	3f	beekte	kaate
plural	1	beekne	Kaane
	2	beektani	Kaatani
	3	beekani	Ka'ani

iv. Main clause negative past, subordinate clause negative past and subordinate clause negative present have forms that are invariant for person

e.g**. beek**- becomes **hinbeekne**

v. Subordinate clause negative past have forms that are invariant for person

E.g. **dhaq**-(to go) becomes **hindhaqu**

vi. Imperatives

Singular	**plural**	
Dhaabi	dhaabaa	stop!
hojjedhu	hojjedhaa	work!
hindhaabin	hindhaabinaa	don't stop!

4.4.2. Verb Derivation

An Afaan Oromoo verb stem can be the basis for three derived voices, passive, causative, and autobenefactive, each formed with addition of a suffix to the stem, yielding the stem that the inflectional suffixes are added to [20].

a) Passive voice

The Afaan Oromoo passive corresponds closely to the English passive in function. It is formed by adding **-am** to the verb root. The resulting stem is conjugated regularly. For instance, **beek-** 'know', **beekam-** 'be known', **beekamani** 'they were known'**; jedh-** 'say', **jedham-** 'be said', **jedhama** 'it is said'

b) Causative voice

The Afaan Oromoo causative of a verb V corresponds to English expressions such as 'cause V', 'make V', 'let V'. With intransitive verbs, it has a transitivizing function. It is formed by adding **-s, -sis,** or **-siis** to the verb stem, except that stems ending in *-l* add *-ch*. Verbs whose stems end in **'** drop this consonant and may lengthen the preceding vowel before adding ***-s***. For instance, **beek-** 'know', **beeksis-** 'cause to know, inform', **beeksifne** 'we informed'; **ka'-** 'go up, get up', **kaas-** 'pick up', **kaasi** 'pick up (sing.)!'; **gal-** 'enter', **galch-** 'put in', **galchiti** 'she puts in'; **bar-** 'learn', **barsiis-** 'teach', **nan barsiisa** 'I teach'.

c) Autobenefactive voice

The Afaan Oromoo autobenefactive (or "middle" or "reflexive-middle") voice of a verb V corresponds roughly to English expressions such as 'V for oneself' or 'V on one's own', though the precise meaning may be somewhat unpredictable for many verbs. It is formed by adding **–adh** *and* **–at** to the verb stem. The conjugation of a middle verb is irregular in the third person singular masculine of the present and past (**-dh** in the stem changes to **-t**) and in the singular imperative (the suffix is **-u** rather than *-i*).

For instance, **bit-** 'buy', **bitadh-** 'buy for oneself', **bitate** 'he bought (something) for himself', **bitadhu** 'buy for yourself (sing.)!'; **qab-** 'have', **qabadh-** 'seize, hold (for oneself)', **qabanna** 'we hold'. Some autobenefactives are derived from nouns rather than verbs, for example, **hojjadh-** 'work' from the noun **hojii** 'work'.

The voice suffixes can be combined in various ways. Two causative suffixes are possible: ***ka'-*** 'go up', **kaas-** 'pick up', **kaasis-** 'cause to pick up'. The causative may be followed by the passive or the autobenefactive; in this case the **s** of the causative is replaced by **f.**

For instance**,** **deebi'-** 'return (intransitive)', **deebis-** 'return (transitive), answer', **deebifam-** 'be returned, be answered',**deebifadh-** 'get back for oneself'.

Another derived verbal aspect is the frequentative or "intensive," formed by copying the first consonant and vowel of the verb stem and geminating the second occurrence of the initial consonant. The resulting stem indicates the repetition or intensive performance of the action of the verb. For instance, **bul-** 'spend the night', **bubbul-** 'spend several nights', **cab-** 'break',**caccab-** 'break to pieces, break completely'; **dhiib-** 'push, apply pressure', **dhiddhiib-** 'massage'.

The infinitive is formed from a verb stem with the addition of the suffix ***-uu***. Verbs whose stems end in ***-dh*** (in particular all autobenefactive verbs) change this to ***ch*** before the suffix. For instance, **dhug-** 'drink', **dhuguu** 'to drink'; **ga'-** 'reach', **ga'uu** 'to reach'; **jedh-** 'say', **jechu** 'to say'. The verb **fedh-** is exceptional; its infinitive is **fedhuu** rather than the expected **fechuu**. The infinitive behaves like a noun; that is, it can take any of the case suffixes. For instance, **ga'uu** 'to reach', **ga'uuf** 'in order to reach' (dative case); **dhug-** 'drink', **dhugam-** 'be drunk', **dhugamuu** to be drunk', **dhugamuudhaan** 'by being drunk' (instrumental case).

4.5. PHONOLOGICAL PROPERTIES

4.5.1. Consonant and Vowel Phonemes

Like most other Ethiopian languages, whether Semitic, Cushitic, or Omotic, Afaan Oromoo has a set of ejective consonants, that is, voiceless stops or affricates that are accompanied by glottalization and an explosive burst of air. Afaan Oromoo has another glottalized phoneme that is more unusual, an implosive retroflex stop, "**dh**" in Oromoo orthography, a sound that is like an English "d" produced with the tongue curled back slightly and with the air drawn in so that a glottal stop is heard before the following vowel begins.

Afaan Oromoo has the typical Southern Cushitic set of five short and five long vowels, indicated in the orthography by doubling the five vowel letters.

Table 4.10: **Afaan Oromoo Vowels**

	Front	**Central**	**Back**
close	i , ii		u , uu
Mid	e , ee		o , oo
Open		a	aa

The difference in length is contrastive, for example, **hara** 'lake', **haaraa** 'new'. Gemination is also significant in Afaan Oromoo. That is, consonant length can distinguish words from one another, for example**, badaa** 'bad', **baddaa** 'highland'.

Table 4.11 :Afaan Oromoo Consonants

		Bilabial/ Labiodental	Alveolar/ Retroflex	Palato-alveolar/ Palatal	Velar	Glottal
Stops and affricates	Voiceless	(p)	T	ch /t/	k	' /ʔ/
	Voiced	B	D	j /d/	g	
	Ejective	ph	x	c /t/	q /k/	
	Implosive		dh			
Fricatives	Voiceless	F	S	sh		h
	Voiced	(v)	(z)			
Nasals		m	N	ny		
Approximants		w	L	y /j/		
Rhotic			R			

In Qubee alphabet, a single letter consists either of a single symbol or a digraph ("ch", "dh", "ny", "ph", "sh"). Gemination is not obligatorily marked for the digraphs, though some writers indicate it by doubling the first symbol: ***qopphaa'uu*** 'be prepared'. In the Table 4.11, the International Phonetic Alphabet symbol for a phoneme is shown in brackets where it differs from the Oromoo letter. The phonemes /p v z/ appear in parentheses because they are only found in recent loan words [20].

4.5.2. Morphophonemic Processes

One of the main occurrences of morphophonemic changes is the change that takes place between the boundary of stems and inflectional or derivational suffixes. In Afaan Oromoo the change may be assimilation, epenthesis, metathesis, deletion, reduplication and so on. In the following section, we briefly discuss each of them.

4.5.2.1. Assimilation

The phonemes that come next to each other at morpheme or word boundary may take the form of the previous or next. This produces the combinations of a variety of stem-final consonants followed by **t** (third person singular feminine, second person singular and second person plural), **n** (first person plural, neutral common), **s** (common, causative-common singular) and so on.

The change can take place between prefix and stem or stem and suffix. Some of the changes are optional because they differ according to the dialect spoken. Table 4.12 summarizes the change.

Table 4.12: Assimilation

Combination of phonemes	Result	Example
d +s	Ch	Duud +sa=duucha
Dh +s	ch	Nyaadh +sisa=nyaachisa
dh + n	n	Fuudh +na=fuuna
d + n	nn	Did +n=dinna
T +n	nn	Dhaloot +ni=dhaloonni
T +ch	ch	Hojjet +chisna=hojjechisna
X +s	Cc or ch	Fix-+siise=ficcisiise/fichisiise
T+dh	dh	Barat+dhu=baradhu
Dh +t	t	Fuudh +tan=fuutan
L +s	ch	Awwaal+sise=awwaalchise
B+t	bd	Waraab +te=waraabde
S+t	ft	Baas+te=baafte
D+t	dd	Yaad+te=yaadde
L+n	ll	Gal+ne=galle
G+t	gd	Dhug-+te=dhugde
X+t	xx	Fix-+te=fixxe
C+t	cc	Boc+te=bocce
J+t	jj	Ajaj-+te=ajajje
R+n	rr	Abaar-+ne=abaarre
S+n	fn	Baas-+ne=baafne

4.5.2.2. Deletion

In Afaan Oromoo for the convenience of speaking, phonemes at word or morpheme boundaries are deleted. This process usually occurs in noun derivations or inflections.

Mana 'man' + **-oota=manoota** 'men'

Nama 'man' **+-ummaa=namummaa** 'personality'

In verbs, deletion usually takes place in stems ending with 'h, dh, hudhaa(') '.

Hodh- +te=hoote

Bah-+te=baate

4.5.2.3. Epenthesis

In Afaan Oromoo, more than two consecutive consonants cannot occur together. When more than two consonants occur consecutively /i/ or others will be inserted between them. This process is sometimes called insertion, and is triggered on the basis of phonological information. For instance, **Elm- + -na= Elmina,** and **Sirb- +ta=sirbita**

4.5.2.4. Reduplication

Reduplication is formed by copying the first consonant and vowel of the verb stem and geminating the second occurrence of the initial consonant. The resulting word indicates the repetition or intensive performance of the action of the verb. Generally, if the stem starts with consonant, reduplication has the form of CV(C) + stem, where C=consonant and V= vowel. But, it has the form of V (') + stem if the stem starts with vowel. Though adjectives can undergo reduplication, we only discuss the reduplication process in verbs in this thesis.

E.g. **Deemuu=deddeemuu, ciruu =cicciruu, affeeluu=a'affeeluu**

4.6. SUMMARY

This chapter has discussed word generation processes in Afaan oromoo, especially for verbs and nouns through inflections and derivations. The morphophonemic processes that take place in the word boundaries have also been dealt with. The next chapter will discuss the procedures and assumptions taken in this chapter to design the lexicons used to synthesize a valid word forms, and hence constitute the core of this study.

CHAPTER FIVE: DESIGN

In this chapter, we discuss design requirements, approaches and techniques for our morphological synthesizer which we henceforth refer to as **HORSIISAA**. The design of lexicons is detailed accordingly.

5.1. DESIGN REQUIREMENTS

In designing morphological synthesizer for Afaan Oromoo, typical features of the language play a pivotal role. The design and realization of every synthesizer system must consider the language feature that it is intended for. Having efficient and effective lexicon look up mechanisms is one of the requirements in designing every morphological generation system. Important properties for the morphological synthesis systems for Afaan Oromoo are: portability, extensibility, and effectiveness. In addition, there are certain desiderata that should be expected from a morphological generation system for any language. These include (1) coverage of the language of interest in terms of both lexical coverage (large scale) and coverage of morphological and orthographic phenomena (robustness); (2) the mapping of surface forms from a deep level of representation that abstracts over language-specific morphological and orthographic features; and finally, (3) availability for the research community [46]. These three issues were essential in the design of **HORSIISAA** for Afaan Oromoo morphological generation. As a pre-condition for a useful integration as a package a wide range of morphological phenomena has to be dealt with.

5.2. DESIGN APPROACHES AND TECHNIQUES

Synthesis can be done either manually or automatically. The manual aspect is done by domain experts or linguists. In this research, automatic synthesizer that returns surface of words from a given stem and set of affixes is considered. There are various approaches which are used for building generators for different languages. The most popular ones are: Rule-based, Artificial Neural Network, and Hybrids. Rule-based approach requires set of rules to define the appropriate suffixes for a stem, and accordingly suitable boundary change during assimilation, etc. The rules

can be designed either manually by linguists or using machine learning techniques. Artificial Neural Network can be used to extract patterns and detect trends that are complex to be noticed by humans. Hybrid approach uses two or more of the above approaches for determining the class of a given word. This research focuses on rule based approach with hand-crafted rules.

In designing the system, it is worth looking at different design goals for evaluation of the system. Among the many goals, achieving good accuracy, speed and less storage area are few to mention.

5.3. LEXICON DESIGN

It has been estimated that average speakers of a language know between 45,000 to 60,000 words. This means that we as speakers must have stored these words somewhere in our heads, our so-called **mental lexicon [27].** The combination of sense and form in a morph, and the possibility to identify the governing rules are the incentives to attempt to build an engine which can automatically generate the same processes taking place in the brain of a native speaker [45]. Lexicon is the heart of any natural language processing systems, even though the format is different for each of them according to their specific need. A lexicon may be defined as a list of words with additional word-specific information, i.e. a dictionary. Lexemes are formed with the help of morpho-syntactic rules [15]. Large and rich lexicons are needed for all types of NLP. To obtain this kind of lexicon a lot of lexical information is needed.

This section discusses the design of lexicons essential for the prototype of morphological synthesizer developed for Afaan Oromoo using relational approach. The design process is based on the morphological properties of the language discussed in Chapter Four and the assumptions and approaches discussed in Chapter Two.

5.3.1. Design of Stem Lexicon

Storing of many inflected forms of word for highly inflected language is not recommended due to the fact that

- It increases the size of the lexicon drastically.
- It is a real bottleneck if we want to port the applications on small handheld devices with limited memory.
- Manually creating this resource is a laborious task, and prone to errors.
- Storage of all possible inflected words is practically impossible since Afaan Oromoo is productive i.e. new stem generalizes to the existing forms.
- It fails to represent linguistic generalization

Therefore, the lexicons for such languages should have a list of stem, together with rules to recognize and generate the inflected variant forms. This recommendation also applies to Afaan Oromoo NLS since it is one of the highly inflected languages [7].

The basic lexical element of Afaan Oromoo is the stem form. Stem means any part of word that is not changed in declension/conjugation [36]. There are many different categories of nouns and verbs in Afaan Oromoo. Thus, the design of the lexicon depends on the classification of these grammatical categories as to be described in the next Chapter.

Each stem in the lexicon *Stem* has the following fields:

- **StemName:** string of arbitrary length which shows name of the stem.
- **StemClass:** string of characters showing class to which the stem belongs
- **POS:** a character that indicates the stem's part of speech, in this study V or N.

5.3.2. Designing the Affix Lexicon

The affixes carry different types of syntactical and semantic information, which help in interpreting different forms of a word.

The table of noun suffixes named NounSuffix consists of fields such as SuffixName, SuffixClass, SuffixType, and alternateType where:

- **SuffixName:** string of one or more characters
- **SuffixClass**: string that indicates the subtype of nouns to which it is suffixed
- **SuffixType**: string that indicates whether the noun is plural or case maker.
- **AlternateType**: string that shows alternative types of suffixes for some nouns classes.

The table of verb suffixes is similar with that of nouns, but the last two attributes of noun suffixes are absent in the verb suffixes. The attributes of this table are as follows.

- **SuffixName**: string of one or more characters
- **SuffixClass**: string that indicates the subtype of verbs to which it is suffixed

5.3.3. Design of the Boundary Change Lexicon

Boundary changes that occur when two phonemes appear consecutively should be handled by the system so that correct and meaningful words are generated. The prototype handles this problem by accessing BoundaryChangeTable before producing the generated word form.

The table has two fields:

- **Pattern**: string of one or two characters that are at the boundary of stem and suffix.
- **ReplacedBy**: string(s) by which the pattern is replaced.

The following table illustrates some records in the BoundaryChangeTable table.

Table 5.1: sample data in BoundaryChangeTable

Pattern	**ReplacedBy**
bt	bd
gt	gd
rn	rr
ln	ll
qt	qx
st	ft
sn	fn
t/*d*/*dh*/*xn*	nn
dt	dd
dht	tt
xt	xx

Table 5.1 is not exhaustive since the rules are so many to list completely.

Generally, taking into account the knowledge of morphological regularity and paradigms of stems, we are aiming to create a very simple and flat, though relatively flexible data model for encoding lexicon to be used in morphological processing. It should be noted that while endings and declensions/conjugations are countable sets and can/have to be provided manually, list of stems is theoretically infinite. Thus, stem description (lexicon entry) has to have minimal attribute set to facilitate automatic acquisition of entries. In our model, it can be achieved easily exploiting typical machine readable lexicon, where stem, part-of-speech and inflectional paradigm are given.

5.4. ARCHITECTURE OF THE SYNTHESIZER

Developing morphological synthesizer to generate words from stems available in the lexicon is difficult. This needs detail understanding of the language at hand. Accodingly the architecture of Afaan Oromoo morphological synthesizer has been selected. The architecture has 9 components as detailed in figure 5.1.

Lexical level: is the stem to be given to the system as input

StemPresenceChecker: This is the module that checks whether stem has been stored in the lexicon or not.

Knowledge Base Module: Testing data has to be ready to acquire the necessary information in the synthesis process. The prototype is tested based on the information stored in lexicon in order to generate words. The knowledge base is constructed by extracting necessary features and information from the stored data. Lexical information of stems and their tags are the basic components to make the knowledge base. Lexical information is gained by taking into account a stem given POS category and class. This enables us to distinguish between verbs and nouns, and different subtypes in each. The rules and suffixes information are also another essential element to the knowledge base.

POS-guesser: Essentially, the problem a morphological synthesizer is faced with is the identification and classification of unknown words. With regards to this work, an unknown word can be defined as a word that is not listed in the lexicon. New words are being created everyday and the lexicon is not large enough to cover all the words. Two effective approaches to predict POS of unknown words are: i) rule-based approach based on word features, etc, ii) statistical approach based on contextual information of the word to calculate the probability of the word holding any tags. Even though both approaches have their own strengths and weaknesses, rule based approach suffices for this study as part of generator to avoid the need of annotated corpus preparation for training. The rule-based approach predicts the most-likely tag for unknown words based on the internal structure of words such as stem endings. For example, in English, words ending with "*able"* might be an adjective [32].

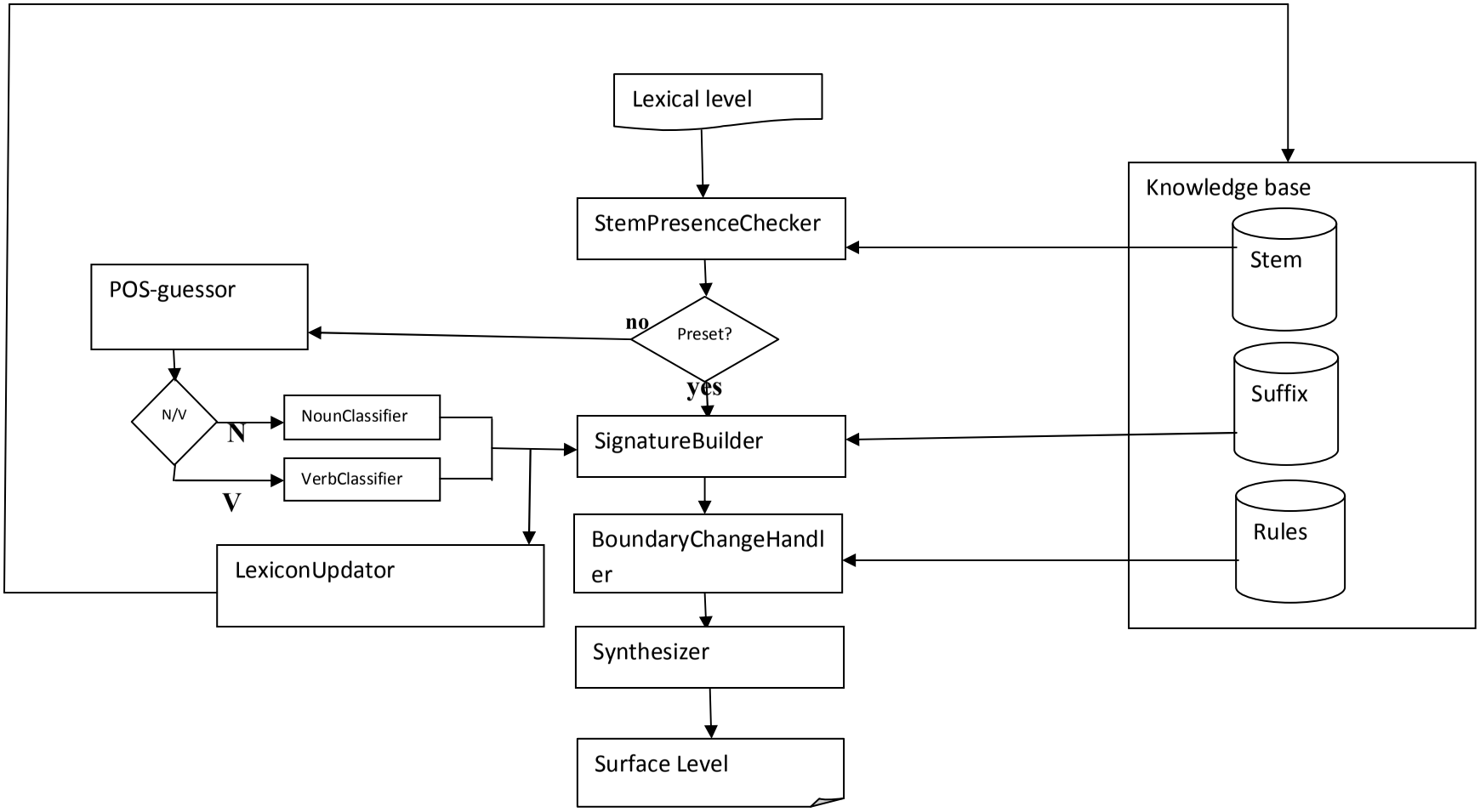

Figure 5.1: The Architecture of Afaan Oromoo Synthesizer

Most of noun stems in Afaan Oromoo end in vowels and verb stems end in single or double consonants or apostrophe. According to [37], in Afaan Oromoo only a limited number of consonantal phonemes, that is /n/, /l/ and /t/ can occur as noun final consonants. In addition to these phonemes, there is the trill sound /r/ that can occur at word final position. Except /n/, the other sounds are in words of numerals.

NounClassifier: This module identifies the inflectional type or declension of the stems for nouns. It detects the type of nouns according to the predefined classes.

VerbClassifier: After the POS-guesser identifies the stem to be verb, the subtype within the verb category should be known before signature is built. This module classifies the new stem that is not available in the lexicon into one of the predefined classes of verbal stems.

LexiconUpdator: If new word which does not exist in the lexicon is encountered, lexicon should be updated parallel with signature building. The newly arriving stem should be stored with its part of speech and subtype. The next time the word comes for generation, the system simply fetches it from the lexicon as predefined or stored stem.

SignatureBuilder: This part lists the set of available suffixes valid for a given stem from the suffix table. It relies on the type of suffixes available for the subtype of the stem at hand.

BoundaryChangeHandler: In the process of synthesis, there are various changes at the boundary of morphemes. Some stems don't allow simple concatenation with the affixes as the last consonant of the stem or first consonant of the suffix may take the form of the next or previous during affixation. These processes include assimilation, epenthesis and deletion. This module accesses rules from the boundary change table to replace some pattern by the appropriate stored form.

Synthesizer: The part that generates all possible surface forms from a given stem is crucial part of the system. It has a close contact with the knowledge base because it is the part that can get the necessary information about each stem. After acquiring this important information, the next step is to perform the actual synthesis task.

Surface level: The words generated as the output of the synthesizer. The words should be meaningful and acceptable by the speakers of the language.

5.5. SUMMARY

In this chapter the lexicon design requirements, approaches and techniques that are useful for implementing word synthesizers for Afaan Oromoo verbs and nouns have been discussed. The necessary database tables for synthesizing a word have been analyzed and designed. The system architecture has also been determined. The next chapter discusses the implementation issues, classification of words and development of algorithms based on language features and the design presented in this chapter.

CHAPTER SIX: IMPLEMENTATION

This chapter describes implementation techniques for automatic morphological synthesis. Stem classification, signature building and algorithms needed to implement the design are discussed.

6.1. MANUAL CLASSIFICATION OF STEMS

The big challenge is the grouping of nouns and verbs in such a way that the members of the same group have similar inflections and derivations. The inflections and derivations are not the same for for all nouns and verbs. A traditional way to represent the morphology of inflectional languages is through classes of similar stems. Otherwise, one has to make rules for each noun and verb, which is not feasible [46]. There are hardly any literatures that clearly indicate the groupings of stems in Afaan Oromoo. However, as the outputs of morphological analyzer described in [31] indicates there are some patterns to group the verbal stems according to their endings. The patterns show that there are groups of suffixes extracted as analyzer output from each similar ending. As described in [33], some stems of nouns and verbs are also grouped into some categories. In addition, analysis and observation of morphological/syntactic features of the language by the researcher and linguists shows some regularity of word formation. The main features used in the classification is the stem ending and/or syllable structure. Accordingly, the stems in the language are divided into classes (inflectional types). In our classification–based approach, the regularities in the way the stems take person makers for verbs, and plural makers and syllable structure for nouns are used to define classes by refining the observations in [31] and [33]. By this categorization, stems with the same morphological phenomena are grouped together into classes. Besides morphological features, syllable structure phenomena are also used in fine tuning of the classification. The stem endings in verbs of the same class have similar morphological behavior which enforces the stems toward affixing the same group of suffixes at the end slot. This categorization has been discussed with different domain linguists and got approval.

Since the classes are contained in the lexicon entries of stems, the morphological synthesis entails only simple operations: lexicon access to extract classification information, and to carry out generation as the rules permit. Every class has a unique machine number for identification and a list of rules for generation of word forms. For every stem, a signature is constructed which matches all word forms belonging to the class of this stem. A machine readable dictionary (lexicon) consists of (stem, class) entries. When an arbitrary word has to be processed the stem checker looks up a matching stem in the lexicon. If such a stem has been found, using the rules of responding inflectional type word forms is generated. Otherwise, the POS-guesser and classifier are consulted for a new stem.

A classification of inflection in view of the mentioned rules and the grammatical features of the stemss are presented in section 6.3. A part of speech is a set of classes. Every set can be divided into subsets depending on criteria pertaining to this particular part of speech. There are 17 different inflectional types which are divided into parts of speech as follows: 7 for the nouns and 10 for the verbs. Every Afaan Oromoo inflecting stem can be classified as a member of some of these types. Two stems are in the same class if their paradigms are generated in the same way. The paradigm is described as a list of word forms with concrete grammatical features for each of them. Eventhough we have set these classes, it is probable that one can increase or decrease the number of the classes according to the criteria used for classification.

Every word form can be constructed from the pattern operating with the rules of replacing and appending, described in the boundary change rules. Once known, the rules for a member of some inflectional type are the same for all other members of that type. The replacing and replaced letter rules are stored in the database.

6.2. APPROACHES OF AUTOMATIC NEW STEM CLASSIFICATION

Automatic classification is the process in which each new stem is assigned to predefined class. The goal is to assign a stem S_i to class C_i based on morphological or phonological properties.

This can be done using rule based approach or machine learning. In this thesis, we employ the former.

6.2.1. Rule Based Automatic Classification

In the section 6.1, we have discussed that classes are compiled manually based on language morphological properties and syllable structures. The classification rules are formulated during manual classification. For instance, noun stems mostly end in vowels while verb stems end in consonants. This property helps to assign a new stem to either noun or verb parts of speech. In addition, when a new stem occurs it should be assigned to predefined morphological classes. In rule based classification approach, a new stem is classified to already known classes according to linguistic theories like their endings or syllable structure as done for Afaan Oromoo, and to be discussed in section 6.3. The classification algorithm shown in figure 6.4 implements this method, and is used in this thesis.

6.2.2. Machine Learning Automatic Classification

Learning is acquiring knowledge of a subject or skill in an art as a result of study, experience or teaching. In this approach, machine learns based on human supplied examples of data or training sets. A set of predefined classes provide the experience necessary for a classifier to learn the pattern. The classification system then builds a model that is used to classify the subsequent patterns automatically. After building a model, the effectiveness of the classifier is observed on data called test set. The classifier decision should be compared with expert decisions. This approach is further divided into two: **Statistical** and **Rule learning**. The former is characterize by having an explicit probability model, which provides a probability that an instance belongs in each class, and the latter extracts rules from corpus to categorize the stems. In this thesis, we don't use this approach as rule based approach is quite better for the systems that extensively rely on language properties and for accuracy purposes.

6.3. BUILDING SIGNATURES

After listing of stems and affixes, for each stem class there should be corresponding set of affixes with which it can combine to form variants of words. This association is called signature. Signatures are used to organize the stems and suffixes in such a way that the stems in the lexicon or new stems can be organized with appropriate suffixes. For example, in Afaan Oromoo stems like **deem-,** have the suffixes **-ti, -a, -tani**, etc.

6.3.1. Verb Signatures

There is a link between stem endings of verbs and a set of suffixes. Observations confirm that stems belonging to the same category behave in the same way under morphological and phonological properties. Accordingly, based on stem finals we can categorize Afaan Oromoo verbs into the following 10 classes according to [31] and [32] with some adjustments made by linguists. The groupings are mostly done by person maker suffixes based on various aspects/tenses/.

a. Class1: Stems ending in letters f, k, m, n

These stems are nasals in the process of sound formation. The associated suffixes set for this class is **-na, -ne, -ni, -nu**

b. Class 2: Stems ending in dh, h, hudhaa(‘).

These are glottals, and have two forms. The first form is those stems having short vowel before the ending “dh” or “h”. In this form, the vowel before the ending is doubled and the ending itself is deleted prior to the affixation of subject makers in different tenses. The second form is when the vowel before the ending is long. In this case, no doubling of the vowel before the ending is necessary as three consecutive vowels are not allowed in the language. But the rest processes are similar.

The suffixes corresponding to this class are: **-atini, -achise, -achisa, -achisan, -achiste, -achisna, -achistan,-aniiru, -anna, -anne, -annu, -ata, -atani, -ate,-atine, -atte, -attu, -attan, -atu**

c. Class3*:* stems ending in b, d, g

These phonemes have similar phonological characteristics in that they all are voiced groups, and hence they behave similarly in suffixation process [31]. Due to the fact that this endings combine with the subject maker suffix "-te" in different aspects, the inflectional suffixes of this class start with the phoneme "d" rather than "t" as a result of inherent assimilation process. In this case, the associated suffixes set for this class is **-da, -di, -dani, -de, -du, -eenya**

d. Class4: stems ending in t

This ending is called palatal, which is active participant in the process of assimilation. The corresponding suffixes of this class are**: -nna, -nnu, -dha, -dhe, -dhu, -chisiise, -chisiisa, -chisisa, -chisiisan, -chisiste, -chisisna, -chisistan, -chiise, -chiisan, -chiisna, -chiistan, -chiisne, -iinsa, -iisa, -noo**

e. Class5: stems ending in s, ch, c

This group is fricatives in the category of sound formation. The corresponding suffixes of this class are: **-ita, -iti, -itani, -ite,- itu, -ina, -inu, -ine, -ise, -isa, -isan, -iste, -isna, -ifte,-ifna, -isise, -isisa, -isisan, -isiste, -isisna, -isista, -isistan**

f. Class 6:Stems ending in l

This ending is called approximants. In this case, the suffix "n" hides itself to take the sound of the stem ending letter 'l' in the process called regressive assimilation. Hence, the corresponding inflections are **-la, -lu, -le, -lani**

g. Class 7: Stems ending in r

This class has similar characteristics with the previous class, but they differ only in the form of person maker suffixes that fill the slot after a stem.

The associated suffixes for this class are**: -ra, -ru, -re**

h. Class 8: Stems ending in x, q,ph

This class is similar with the third class, and called ejectives .The only difference is that assimilation process gives the suffix "-te" the form of **-xa, -xi, -xani, -xe, -xu** which are considered as inflections of this class as indicators of persons in various aspects.

i. Class 9: Stems ending in "aaw"

The associated inflectional suffixes are: **-oofte, -oofti, -oofna, -ooftan, -oofta.** The ending is replaced by these suffixes

j. Class 10: Stems ending j

The suffixes associated to this class are: **-ja, -ju, -je**, **-jani**

The following suffixes are common suffixes for all classes of verbal stems: **-a,-achuu,-achuuf, -adha, -adhe, -adhu, -ama, -amaa, -aman, -amne, -amoo, -amta, -amtan, -amte, -amti, -amtuu, -amuuf, -ani, -e, -eera, -I,-uu, -uuf, -neerra, -aaf, -aas, -aat, -aatu, -uuttan, -uutti, -aa, -naan, -u, -eet, -eeti,- ees, -is, -uufan, -uufi, -uufii, -adhuu, -adhee, -amani, -amanii, -ame, -amni, -amu, -amtu, -amtani, -amuu, -amuudhaa, -amuudhaaf, -amuun, -ullee, -aatii, -umsa, -iisa, -iinsa**

In addition to the suffixes compiled privately for every class, there is a significant overlap among some classes on the set of suffixes they take i.e. there are suffixes that are shared among two or more classes, but not among all classes. For instance, the suffixes **-ta, -tani, -taniittu, -te, -teetta, -teetti, -ti, -tu, -tuu** are shared among class 1, 2, 4, 6 and 7, and the suffixes **-sisa, -sisan, -sise, -siste, -sisna, -sistu, -sisne** are shared among the classes 1, 3, 6, 7 and 8 . There are also suffixes that apply for all classes as mentioned before. The suffixes private to one class cannot be used by the other, while the common suffixes or the combination of non private suffixes can be used by some or all the other classes. The surface forms of the verbal stem can be synthesized only after the generation of the signature corresponding to it.

6.3.2. Noun Signatures

The noun morphology is implemented using rule based approach by setting categories according to their morphological properties. The suffixes corresponding to specific stem group should be identified and a signature should be generated. After obtaining a complete list of affixes for a given noun, the concatenation process is done taking care of boundary changes as specified by the rules.

According to [32] the groupings of nouns are guided by the number of vowels after the last consonant, the last consonant itself and some specific endings. Hence, we have the following noun classes:

Class 1: Nouns ending in long vowels. These nouns form plural and affix case makers without boundary change of the noun stem. But, the last vowels are deleted when definiteness suffixes are concatenated. If the last consonant of the noun is 't' and the sound preceding the this letter is short , like in the case of **barataa** 'student', the last consonant doubles before common suffixes such as '**-oota**', '**-oolee**', '**-oolii**' are added. Besides, these suffixes become '**-ota**', '**-olee**', '**-olii**' when the syllable preceding the last has long sound. For instance, nouns such as **maatii** (parent), **gaaffii** (question), **jabbii** (calf) are grouped under this category.

The suffixes of this class include: **-wwan,- lee , -n,-tiin,-dhaa, -dhaan, -dhaaf,-tii**

Class 2***:*** Nouns ending in short vowel "**a**" in the last syllable and consonants like l, r, and m are of this category. The stem in this class may have two or more syllable, but the syllable preceding the last syllable must have long vowels. The group is characterized by the doubling of last consonant of the stem and deletion of last vowel when plural maker suffix is added. During case formation, they double last consonant and add case maker "i" in the nominative case. In addition, the short form of plural makers **'-ota', '-olee', '-olii'** is suffixed as common suffix instead of '**oota**', '**oolee**', '**oolii**'. For example**, beera** (old woman), **wasiila** (uncle) and **daa'ima** (child) are nouns of this class.

The suffixes of this class include: **-an, -i**

Class 3: Two syllable nouns each ending in short vowels and having end consonants like k,g,d,r, b, n. are classified here. They take plural making suffix after doubling the last consonant and deleting the last vowel. This group shares similarity with class 2 in that both double the last consonant. **Mana** (house), **laga** (river), **muka** (tree) can be taken as examples.
The suffixes of this class include: **-een, -ni, a**

Class 4: Nouns ending in **"eessa, eeysa, eecha, eettii, eeytii, essa, eensa"** take the plural suffix by deleting these endings. This class deletes the last vowel before suffixing "i" in nominative case formation. Examples are **daljeessa** (monkey), **hiyyeessa** (poor), **dureettii** (rich woman), **and bineensa** (wild animal).
The suffixes of this class include:**:** -eeyyii, -i

Class 5: This class comprises nouns that should be handled as special cases. Though most nouns end in vowels, there are a few nouns that end in consonants, particularly the letter '**n**'. They are characterized by having no plural suffixes. They only have case makers. The associated case suffixes are **–ii, -iin, -iifi, -iis, iif, -iinis, -iifuu, -iifis, -uma, -umaa, -umaanuu, -umaaf, -umaanis, -umatti, -umattuu, -umattis, -umaratti, -umarattis, itti, ittis, ittuu, -irraa, -irraahuu, -irraahis, -irraan, -irraanuu, -irraanis, -irratti, -irrattis, -irrattuu, illee, -iinillee, -umallee, -umaafillee, -umaanillee, -ittillee, -umattillee, -irraahillee, -irrattillee, -irraanillee.** The nouns such as **foon** (meat), **aannan** (milk), **ilkaan** (teeth) are instances of this class.

Class 6: Nouns that end in the same or different two consonants and nouns having three or more short syllables are categorized under this group. The main suffix of this class is **–oota**.

Class 7: This class includes proper nouns that have only case suffixes. These nouns don't have plural makers. Either they should be stored in the lexicon or identified by capitalization of the first consonant provided that typing error is avoided. The corresponding nominative suffixes are **–n, -ni or -i.**

The suffixes that are not listed as the member of plural or case makers for the above classes of nouns are common for all noun stems. They include **-oota,- oolii, -oolee, -tti,-rraa,-ummaa,-ooma,-ittii,-icha.** We have to also note that though common to all classes, the principal usage of the suffix **'-oota'** is with stems having double consonant endings or nouns which have short sound in each syllable.

6.4. SUFFIXES CONCATENATION

The process of appending suffixes to the stem is not trivial in Afaan Oromoo; that is where the information about the category of the stem is required. Depending on the category, there are changes that take place at the boundary while concatenation. The rules for handling these changes are stored in rules-table. A typical rule of concatenation for verb stem ending in **t** is of the form

X*t(stem**) + -na(**suffix**) = X*nna (**morphed form**)**

Where X=substring, and t=the last letter to be **'t**'. For example**, qot-** 'to plough' + **-na** 'person maker suffix'=**qonna** 'we plough'. After the identification of category, the appropriate rules of concatenation from rules table are referred to handle the boundary changes, and the corresponding suffixes are fetched for correct form synthesis. The implementation of rules is discussed in section 6.6.

6.5. ALGORITHMS

Afaan Oromoo is different from other languages in morphological properties, patterns of word synthesis and grammatical rules. Thus, the existing algorithms and techniques that are being used to generate word forms of English and other languages are not actually suitable for the language; rather it needs different algorithms and techniques for expected efficiency. However, some techniques and approaches can be adapted as found to be important, especially that of [23].

The lexicon for Afaan Oromoo morphological synthesizer was designed in the previous Chapter. In this section, algorithms to be used in implementing the study are developed. First, general word synthesis algorithm is dealt with, and then followed by suffixation, pre fixation, epenthesis, deletion, assimilation and reduplication algorithms.

6.5.1. Word Form Generation Algorithm

The word form generation algorithm takes stem as an input. After then it identifies the sub types of each stem and retrieves corresponding suffixes to build signature. Lastly, all possible word forms are produced by making use of transformation rules. Figure 6.1 depicts the detail of the algorithm.

Input :**Stem**
Output : **list of word forms**

Start

Get input stem

Identify class

Retrieve the suffixes corresponding to the stem from lexicon.

Build signature

Make up the word forms according to rules.

Return

End

Figure 6.1: General Algorithm for Word Form

6.5.2. Suffixation Algorithm

The algorithm performs the tasks in the following manner. First, it checks whether the stem is in lexicon or not, and if available it identifies parts of speech. If it is found in the list, the suffixes corresponding to the stem's subtype are retrieved, i.e. signature is built. Then changes at the stem and suffix boundaries are handled and surface word forms are generated. On the other hand, if the stem is missing from the lexicon POS-guesser is called to guess unknown word grammatical

category, or the parts of speech can be provided manually to guide the synthesizer. The classification process is handled by rule based classifier algorithm to be discussed in section 6.4. The stem is then given to the classifier to assign to the already known class of stems. Lastly, the word synthesis process continues as before. The algorithm is given in figure 6.2.

Input: **Stem**

Output : **Surface word forms**

Start

If stem found in lexicon

Retrieve grammatical category

Identify subtype

Build signature

Apply boundary change rule

Apply suffixation

Else

Guess POS

Classify

Build signature

Apply boundary change rule

Apply suffixation

Return

End

Figure 6.2: Suffixation algorithm

6.5.3. Pre- Fixation Algorithm

The pre fixation algorithm is applied to words generated by suffixation. It checks the endings of generated words so that it can attach correct prefix. There are inflectional suffixes that can be associated with each of the prefixes as to be discussed in the algorithm. Derivational suffixes

cannot be paired with any of the prefixes. Normally, pre fixation in Afaan Oromoo is used to indicate negation using '**hin**' or affirmative by using '**ni**'. Figure 6.3 shows the detail of the algorithm.

Input: generated word vectors

Output: all possible words

Start

For each suffixed and reduplicated word forms

If generated word ends with ***'u, ne,nu, tan, an, tu'*** *attach '****hin****' in front generated word*

Else if words end in ***'a, ta, ti, na, tan, an, e, te '*** *attach '****ni****' in front of generated word*

End

Figure 6.3: pre-fixation algorithm

6.5.4. Classification Algorithm

The algorithm starts by guessing the parts of speech for the newly fed stem. After the POS is identified, the subtype of stem should be known for further generation of words. The classification process mostly depends on the ending for both nouns and verbs. Of course, the classification of both depends on the final long or short sounds in addition of consonantal endings for nouns and consonantal endings as well as some group ending for verbs. After class identification, the normal synthesis continues as usual. Figure 6.4 shows the detail.

Input : **Stem**

Output : **classificatiuon**

Start

Guess the parts of speech of the stem

If stem ends with consonant it is verb

Else it is noun

If(noun)

If stem end with long vowels it is class1

Else if stem ends with short vowels, the sound before last consonant is among 'aa, ee, oo, uu,ii' and last consonant l, r, m it is class2

Else if stem ends with short vowels and last consonant b, d, g n it is class3

Else if stem ends with "eessa, eettii, eecha,eeysa, eeytii" it is class4

Else if stem ends with consonant letter it is class 5

Else if stem starts with capital letter it is class 7

Else it is class 6

If (verb)

If stem ends with f, k, m, n it is class1

Else If stem ends with hudhaa('), dh, h class2

Else If stem ends with b, d, g it is class3

Else If stem ends with t it is class4

Else If stem ends with s, ch, c it is class5

Else If stem ends with l it is class6

Else If stem ends with r it is class7

Else If stem ends with x, q, ph it is class8

Else If stem ends with 'aaw' it is class9

Else If stem ends with j it is class10

End

Figure 6.4: Classification Algorithm

6.5.5. Epenthesis (Insertion) Algorithm

Figure 6.5 presents the epenthesis algorithm. The algorithm works in parallel with the word synthesis algorithm to account for the changes due to three consecutive consonants. As described in Chapter Four, more than two consecutive consonants are not allowed in Afaan Oromoo. Therefore, the vowel 'i' or others is inserted when stems end and suffixes start with consonants. This process is common in Afaan Oromoo verbs but doesn't exist in nouns. For instance, the verbal stem **arg-**'to see' can simply concatenate the suffix **'-na'** to make '**argna**' which is not correct. To overcome this problem 'i' should be inserted between the stem and suffix to produce correct word **argina** 'we see'. In addition, before adding suffix families such as –**ach, -adh**, verbs ending in single consonant geminates (doubles) the last consonant in the form of insertion if the vowels before the last consonant in the stem is long. For instance, **beek-**+**achuu**=bee**k**achuu, instead of bee**kk**achuu, without taking care of boundary for epenthesis, the result which is not valid.

Input: **stem**

Output**: words**

Start

If there are two consecutive consonants in the stem end and suffix starts with consonant

*Insert the '**i**' between stem and suffix*

Return Word

End

Figure 6.5: Epenthesis Algorithm

6.5.6. Deletion Algorithm

Figure 6.6 shows deletion algorithm. This algorithm is applied mostly in the process of plural making and case formation of nouns. Among the seven classes of nouns, five of them apply this algorithm before inflections are formed. Some case maker suffixes also delete the last vowel(s) from the noun stems especially when the last sound is short. For example, the stem **mana**'house' deletes the last vowel 'a' to form the plural noun **manoota '**houses' when the suffix 'oota' is attached. Similarly, the same stem becomes **manni** 'house in nominative case' when case maker suffix **'-ni**' is applied.

Input: **Noun Stem**

Output: **word with last vowel(s) removed**

Start

Lvowels = the last vowel(s) of the Stem

Word = stem without Lvowels

Return Word

End

Figure 6.6: deletion algorithm

6.5.7. Assimilation Algorithm

We have shown that, by and large, words are composed by concatenating morphs. In many cases this concatenation process will induce some phonological change in the vicinity of the morph boundary. Assimilation is a process where the two segments at a morph boundary influence each other, resulting in some feature change that makes them more similar. The assimilation algorithm replaces the last consonant of the stem in the form that the following suffix takes. For example, the combination **fagaat-** 'to go far' + **'-ne'** produce **fagaanne** 'we went far'. In the process, the

ending**‘t’** has been changed to **‘nn’** to take the form of suffix. Figure 6.7 shows the assimilation algorithm.

Input: Stem and Suffix

Output: Generated word

Start

Lchar= the last character of the Stem

Fchar=first character of the suffix

Pattern = Lchar + Fchar

Get the SurfaceForm for Pattern from the BoundaryChngeTable

If the SurfaceForm found

Word = stem without Lchar + SurfaceForm + Suffix without Fchar

End

Figure 6.7: Assimilation Algorithm

6.5.8. Reduplication Algorithm

As discussed in Chapter Four, reduplication is used to indicate the repeated action. In the process, the first syllable repeats itself by doubling the consonant. The algorithm at Figure 6.8 extracts the first consonant with its first vowel, and then the first consonant again. The reduplicated word is formed by concatenating the first two letters, the first consonant and the stem itself in sequential manner. After that the normal inflection process follows.

Input:**Inflected words**

Output**: reduplicated word**

Start

For each suffixed word forms

If(stem starts with consonant)

twoStartchars=the first two characters of the verb stem

Fchar= the first character

Word= twoStartchars + Fchar+stem

Else

oneChar= the first character of the stem

word= oneChar +hudhaa +stem

End

Figure 6.8: Reduplication Algorithm

6.6. IMPLEMENTATION OF THE RULES

There have been various rules considered while developing morphological synthesizer for Afaan Oromoo. The rules contain a large number of morphological, lexical and/or syntactical information. They are based on linguistic knowledge of specific languages. With human rule creation, there is a large set of manually constructed rules based on a specific grammar, written in a formal notation so that they can be used by the computer for further parsing. The first one is morphotactics i.e. the ordering of morphemes. The general order of morphemes in Afaan Oromoo is prefix + Stem +suffix, where prefix is optional. The second rule considered was the change that occurs at morphemes boundary during affixation process such as assimilation, insertion and deletion. The deletions of last vowel(s) in nouns have been done by taking the

substring of some portion of the stem leaving those deleted characters. The assimilation has been employed by replacing some patterns in word forms of verbs using simple regular expressions during concatenation process. The replacing and to be replaced pattern is fetched from databases. The third implemented rule was the insertion of 'i' if three consecutive consonants appear. It has also been done by using regular expressions. The details of rules are described in Appendix D.

6.7. SUMMARY

In this chapter, stems are classified according to their ending sounds and/ or syllable structures based on morphological-phonological properties. The signatures of stems have been discussed along with stem categorization. The algorithms have been formulated based on the language properties set in Chapter Four. The next chapter discusses the experimentation and the results on the effectiveness of the algorithms.

CHAPTER SEVEN: EXPERIMENTATION AND RESULTS

In Chapter Five and Six, the design of lexicons and algorithms for Afaan Oromoo Morphological Synthesizer were designed respectively. In this chapter, we test the prototype experimentally and the results are communicated.

7.1. DATA COLLECTION

Most of the time, natural language processing is a data-intensive field. The success or failures of most NLP applications depend on the quality and availability of appropriate data. The data used in computational linguistic tasks generally takes the form of corpora. Corpora can be divided into two categories: annotated corpora and unannotated corpora. Unannotated corpora are simply large collection of raw text, where as annotated corpora add additional information to the text, such as phonetic transcription, part-of-speech tags, parse trees, etc [34].

Annotated data in the form of lexicon is useful for morphological synthesizers. In this experiment, data have been collected from different sources like thesis, reference books, and the Internet. To evaluate the algorithm a test data of 1000 stems and 252 affixes from different sources have been collected. Out of total stems collected, 600 are verbs and 400 are nouns. In the collected affixes, 160 are verbal suffixes, 90 are nominal suffixes and 2 are verbal prefixes. The stems in the lexicon are not exhaustive as there are infinitely many stems in a given language. But the number of suffixes is finite and listed as much as possible, though not exhaustively compiled. After collecting the data, it has been coded in the database in the form suitable for computational scheme. Normally, the collected stems have been stored with their tags and subtypes. The suffixes were also compiled according to the type of stem to which they apply. At the same time, the data collected have been checked by linguists for validity.

According to [50] corpus is expected to have two important characteristics: sampling and representativeness, and machine readable. Stems are sampled and used being the representative of other stems as languages have large number of stems. The sample has to represent all stems considering the morphological and structural variation of the stems. In this regard, sample stems have been taken from each class of stems equally in number, and are believed to represent the rest of stems in the respective class for our morphological synthesizer. We have prepared lexicon for the system to use it as a source of stems and suffixes.

7.2. THE PROTOTYPE

The prototype has been developed using Java having Microsoft Access database as back end. The main screen of the system is depicted in Figure 7.1.

Figure 7.1: The screenshot of prototype's user interface

The input for the prototype is stems of either verbs or nouns, and the output is all the possible generated word forms. The stems available in the database are directly chosen from the combo box for valid words generation. The new stems are entered through text box, after which the system predicts POS and morphological classes for normal word forms synthesis. After the stem is selected or entered, generate button is pressed to have list of valid words below GENERATED WORDS in the list.

Table 7.1 shows sample words generated from the stem **beek-**'to know', and more samples are included in the Appendix C.

Table 7.1: The sample output of the prototype

1. beekna
2. beekne
3. beeknu
4. beeka
5. beekkachuu
6. beekkachuuf
7. beekkadha
8. beekkadhe
9. beekkadhu
10. beekama
11. beekamaa
12. beekaman
13. beekamne
14. beekamoo
15. beekamta
16. beekamtan
17. beekamte
18. beekamti
19. beekamtuu
20. beekamuuf
21. beekani
22. beeke
23. beekeera
24. beekI
25. beekuun
26. beekuu
27. beekuuf
28. beekneerra
29. beekaaf
30. beekaas
31. beekaati
32. beekaatu
33. beekuuttan
34. beekuutti
35. beekaa
36. beeknaan
37. beeku
38. beekeeti
39. beekees
40. beekis
41. beekuufan
42. beekuufi
43. beekuufii
44. beekkadhuu
45. beekkadhee
46. beekamani
47. beekamanii
48. beekame
49. beekamni
50. beekamu
51. beekamtu
52. beekamtani
53. beekamuu
54. beekamuudhaa
55. beekamuudhaaf
56. beekamuun
57. beekullee
58. beekaatii
59. beekumsa
60. beekiisa
61. beekiinsa
62. beekan
63. beekkadhaa
64. beekameera
65. beekta
66. beektani
67. beektaniittu
68. beekte
69. beekti
70. beektu
71. beektuu
72. beekteetta
73. beekteetti
74. beeksisa
75. beeksisan
76. beeksise
77. beeksisna
78. beeksiste
79. beeksisne
80. bebbeekna
81. bebbeekne
82. bebbeeknu
83. bebbeeka
84. bebbeekkachuu
85. bebbeekkachuuf
86. bebbeekkadha
87. bebbeekkadhe
88. bebbeekkadhu
89. bebbeekama
90. bebbeekamaa
91. bebbeekaman
92. bebbeekamne
93. bebbeekamoo
94. bebbeekamta
95. bebbeekamtan
96. bebbeekamte
97. bebbeekamti
98. bebbeekamtuu
99. bebbeekamuuf
100. bebbeekani
101. bebbeeke
102. bebbeekeera
103. bebbeekI
104. bebbeekuun
105. bebbeekuu
106. bebbeekuuf
107. bebbeekneerra
108. bebbeekaaf
109. bebbeekaas
110. bebbeekaati
111. bebbeekaatu
112. bebbeekuuttan
113. bebbeekuutti
114. bebbeekaa
115. bebbeeknaan
116. bebbeeku
117. bebbeekeeti
118. bebbeekees
119. bebbeekis
120. bebbeekuufan
121. bebbeekuufi
122. bebbeekuufii
123. bebbeekkadhuu
124. bebbeekkadhee
125. bebbeekamani
126. bebbeekamanii
127. bebbeekame
128. bebbeekamni
129. bebbeekamu
130. bebbeekamtu
131. bebbeekamtani
132. bebbeekamuu
133. bebbeekamuudhaa
134. bebbeekamuudhaaf
135. bebbeekamuun
136. bebbeekullee
137. bebbeekaatii
138. bebbeekumsa
139. bebbeekiisa
140. bebbeekiinsa
141. bebbeekan
142. bebbeekkadhaa
143. bebbeekameera
144. bebbeekta
145. bebbeektani
146. bebbeektaniittu
147. bebbeekte
148. bebbeekti
149. bebbeektu
150. bebbeektuu
151. bebbeekteetta
152. bebbeekteetti
153. bebbeeksisa
154. bebbeeksisan
155. bebbeeksise
156. bebbeeksisna
157. bebbeeksiste
158. bebbeeksisne
159. nibeekna
160. hinbeekne
161. hinbeeknu
162. nibeeka
163. nibeekkadha
164. nibeekkadhe
165. hinbeekkadhu
166. nibeekama
167. hinbeekaman
168. hinbeekamne
169. nibeekamta
170. hinbeekamtan
171. nibeekamte
172. nibeekamti
173. nibeeke
174. hinbeeku
175. nibeekame
176. hinbeekamu
177. hinbeekamtu
178. hinbeekamtani
179. hinbeekan
180. nibeekta
181. hinbeektani
182. nibeekte
183. nibeekti
184. hinbeektu
185. nibeeksisa
186. hinbeeksisan
187. nibeeksise
188. nibeeksisna
189. nibeeksiste
190. hinbeeksisne
191. nibebbeekna
192. hinbebbeekne
193. hinbebbeeknu
194. nibebbeeka
195. nibebbeekkadha
196. nibebbeekkadhe
197. hinbebbeekkadhu
198. nibebbeekama
199. hinbebbeekaman
200. hinbebbeekamne
201. nibebbeekamta
202. hinbebbeekamtan
203. nibebbeekamte
204. nibebbeekamti
205. nibebbeeke
206. hinbebbeeku
207. nibebbeekame
208. hinbebbeekamu
209. hinbebbeekamtu
210. hinbebbeekamtani
211. hinbebbeekan
212. nibebbeekta
213. hinbebbeektani
214. nibebbeekte
215. nibebbeekti
216. hinbebbeektu
217. nibebbeeksisa
218. hinbebbeeksisan
219. nibebbeeksise
220. nibebbeeksisna
221. nibebbeeksiste
222. hinbebbeeksisne

7.3. EVALUATION OF THE ALGORITHMS

To evaluate the performance of the synthesizer, the following procedures are used.

In the first phase, the test has been done personally by the researcher by generating words from selected stems and comparing the generated words with the structure of words in the grammar books. The test has been conducted iteratively to increase prototype's performance. The errors encountered during experimentation have been corrected and the experiment has been done iteratively until the result is found to be satisfactory, and before presented to the linguists. The errors encountered during this phase were mostly missing rules, and were fixed accordingly.

In the second phase, different stem types from both verbs and nouns have been selected. The selection was done by linguists according to their representativeness. From each class of verbs, one stem corresponding to every class were chosen. For nouns, stems have been selected to represent the classification criteria used in the previous chapter, such as endings and syllable structure. From every type, we have taken one stem intentionally from each class of verbs and two stems from each class of nouns randomly. Hence, for verbs the stem **beek** from class 1, **fedh** from class2, **eeg** from class3, **barat** from class4, **barreess** from class5,**awwaal** from class6, **abaar** from class7, **cuuph** from class8, **bushaaw** from class9, and **ajaj** from class10 have been taken. For nouns, the stem **maatii, jabbii** of type 1, **beera**, **wasiila** of type 2, **laga, mana** of type 3**, hiyyeessa, dureettii** of type 4, **halkan, foon** of type 5, **gamna, dhukkuba** of type 6 **and Fayyisaa, Roobeeraa** of type 7 were selected. The generated words are presented to the linguists with the aim of identifying errors, and then to correct iteratively until the results are found to be satisfactory.

In this experiment, the error counting approach was adopted to evaluate the word generation algorithm. The number of correctly generated words and incorrectly generated ones are counted for analysis. Through questionnaires, the output from the synthesizer for verbs was then checked against the respective valid words by domain experts. Only single stem from each class has been

used for evaluation by questionnaire respondents for verbs. The enumerous number of surface word forms that are produced from single verbal stem makes difficult to take more stems for evaluation. For all nouns, evaluation of the system has been done by linguists directly running the system since the number of surface forms are very small compared to verbs. The errors were then described in terms of correctness and incorrectness of the produced words. The produced word is said to be correct if it is accepted by speakers of the language in terms of actual and possible words, otherwise incorrect. Eventhough many stems were collected from different sopurces, a few stems have been used in the experiment as evaluating large number of stems with their forms is very difficult. Throughout the evaluation, the statistics used to measure the performance of the system is accuracy. Accuracy refers to the closeness of agreement between a test result and the accepted reference value. The accuracy of the system is calculated as the number of correctly generated words divided by the total number of words generated by the system multiplied by 100%. That is,

$$Accuracy = \frac{Total\ number\ of\ correctly\ generated\ words}{Total\ Number\ of\ generated\ words} * 100\%$$

7.4. RESULTS OF THE EXPERIMENT AND PERFORMANCE ANALYSIS

The experiments are carried out in order to test the performance of the synthesizer. The experiment is done to evaluate the performance of the synthesizer on representative stems chosen from the respective classes. Tables 7.2 and 7.3 present the results of the test based on the selected verb and noun stems. In both tables, R1, ..., R_n indicate questionnaire Respondent1, ..., $Respondent_n$ respectively. The number of correctly and wrongly generated from each respondent was counted for analysis, and the accuracy for each has been calculated. Finally, the average accuracy of all respondents has been taken to get the total accuracy.

Table 7.2: Test results on some selected verb stems

Stem Name	Class	Number of Words Generated	Number of Correctly Generated words				Number of Wrongly Generated words				Accuracy by each respondent				Average Accuracy
			R1	R2	R3	R4	R1	R2	R3	R4	R1	R2	R3	R4	
beek	V1	222	213	207	206	218	12	9	10	4	95.95	95.95	95.50	98.20	96.40
fedh	V2	286	276	196	223	282	10	8	16	2	96.50	97.20	94.41	98.60	96.68
eeg	V3	200	189	188	191	195	11	4	1	5	94.50	98	99.50	97.50	97.38
barat	V4	254	238	240	245	254	16	12	7	0	93.70	95.28	97.24	100	96.56
barreess	V5	250	236	239	242	243	14	7	4	7	94.40	97.20	98.40	97.20	96.80
awwaal	V6	224	212	202	208	218	12	12	6	1	94.64	94.64	97.32	97.32	95.98
abaar	V7	226	214	199	203	223	13	15	15	3	96.43	93.36	93.36	98.67	95.46
cuuph	V8	200	189	183	181	199	11	9	11	1	94.50	95.50	94.50	99.5	96
bushaaw	V9	230	213	165	152	230	17	14	20	0	92.61	93.91	91.30	100	94.46
ajaj	V10	210	196	180	195	210	10	6	4	0	93.33	97.14	98.10	100	97.14
														Total Average Accuracy	96.28

Table 7.3: Test results on some selected noun stems

Stem Name	Class	Number of Words Generated	Number of Correctly Generated words				Number of Wrongly Generated words				Accuracy by each respondent				Average Accuracy
			R1	R2	R3	R4	R1	R2	R3	R4	R1	R2	R3	R4	
maatii	N1	29	29	29	27	28	0	0	2	1	100	100	93.10	96.55	97.41
jabbii	N1	29	29	28	28	27	0	1	1	2	100	96.55	96.55	93.10	96.55
beera	N2	22	22	22	21	20	0	0	1	2	100	100	95.45	90.90	96.59
wasiila	N2	22	22	22	20	22	0	0	2	0	100	100	90.90	100	97.73
laga	N3	22	22	22	22	22	0	0	0	0	100	100	100	100	100
mana	N3	22	22	20	22	22	0	2	0	0	100	90.90	100	100	97.73
hiyyeessa	N4	19	19	19	16	17	0	0	2	2	100	100	82.21	89.47	92.92
dureettii	N4	19	18	17	16	17	1	2	3	2	94.44	89.47	82.21	89.47	92.92
Fayyisaa	N7	10	10	10	10		0	0	0	0	100	100	100	100	100
Roobeeraa	N7	10	10	10	10		0	0	0	0	100	100	100	100	100
Halkan	N5	39	39	39	39		0	0	0	0	100	100	100	100	100
foon	N5	39	39	39	39		0	0	0	0	100	100	100	100	100
gamna	N6	27	25	27	25	25	2	0	2	2	92.59	100	92.59	92.59	94.44
dhukkuba	N6	27	26	26	27	27	1	1	0	0	96.30	96.30	100	100	98.15
														Total Average Accuracy	97.46

As can be seen from the tables 7.2 and 7.3, the algorithms of the synthesizer are tested with 10 verb and 14 noun test stems and the result of each test was checked by linguists for the validity of generated words.

As it can be seen from Table 7.2, the results of the experiments for verbs show an accuracy of 96.28% correctly generated words on average. This means that out of about 230 generated words on average, 212 of them are correctly generated by the system. If the average accuracy of synthesized words for each class is considered, we can see that the accuracies are 96.40, 96.68, 97.38, 96.56, 96.80, 95.98, 95.46, 96, 94.46, 97.14 for class1,..., class10 respectively.

From Table 7.3, we can observe that the average accuracy of generated words for nouns is 97.46% on average i.e. from the total average of 25 generated words, almost 24 on average is correctly synthesized. The average accuracies of each noun class are 96.98, 97.16, 98.87, 92.92, 100, 96.30, 100 for class1, ..., class7 nouns.

The experiment has revealed that noun morphology is extremely simpler compared to verb morphology. For instance, a single verb can take on average 230 surface forms whereas a noun can inflect for only number and cases taking about 25 surface forms in average. For both verbs and nouns, as the feedback of linguists' evaluation shows, the incorrect results obtained from this test were due to the fact that some suffixes are incorrectly classified for the wrong class of stems. The missing of rules has also contributed to the wrong generation of words. The incorrectly added suffixes and the missing of rules, for instance assimilation, brought overgeneration of word forms. It is also apparent that suffixes might have been missed during manual compilation from various sources. The missed suffixes might bring more number of generated words. For example, the excluded double causative suffixes such as **'-siisis-**' as identified by linguists definitely decreases the number of word forms generated from a single stem as it can combine with inflectional suffixes, and also appear in reduplications. Among the other the errors, improper handling of syllables within a class is worth mentioning. Some of the generated words were correct both grammatically and semantically, but not used in day-to-day conversation. For example, most of the words ending in suffixes starting with '**-am-**' such as barat**amoo,** barat**amne**, …, can possibly be used, but not common. These are called possible words i.e.

conform to the word-formation rules of a language. We considered them as correctly generated words. So they should be included into the dictionary of the language, and shouldn't be considered wrong.

It is also worthy to analyze the result of the above experimentation to assure that Afaan Oromoo is inflectional language as indicated on literatures interms of the total number of inflectionally and derivationally generated words from verbal stems.

Table 7.4: Comparison of inflectionally and derivationally formed words for selected verbal stems

Stem Name	**Class**	**Number of Words Generated**	**Number of words generated by inflections**	**Number of words generated by derivations**
beek	V1	222	208	14
fedh	V2	286	276	10
eeg	V3	200	196	14
barat	V4	254	241	13
barreess	V5	250	241	9
awwaal	V6	224	218	6
abaar	V7	226	219	7
cuuph	V8	200	192	8
bushaaw	V9	230	221	9
ajaj	V10	210	206	4
Average		**230**	**221**	**9**

A close look at Table 7.4 shows that among the 230 words generated on average, about 221(96%) of them on average are formed by inflections while 9(4%) of them on average are formed by derivations. This brings us toward the conclusion that Afaan Oromoo is inflectional language as already forwarded by the linguists, and inflection is the dominant word form synthesis.

Generally, the evidence taken from other morphologically rich and complex languages indicates that Afaan Oromoo is morphologically complex language. For example, the number of words that can be generated from a single stem in Telugu is 17 for nouns and 130 for verbs where Afaan Oromoo far exceeds Telugu especially in the number of words generated from verbal stem. It has also been stated in different literatures that the language is structurally complex.

During the experiment, the eight linguists who evaluated the outputs of the system have identified the following points.

- In the insertion process, we have only considered to insert 'i' between consonant endings of stem and consonant start of suffixes when three consecutive consonants come together. But, for some words whose sound preceding the last consonant are long, such as **beek** 'to know' germination of the last consonant of the stem is needed before adding the suffix '-**ach-** and –**adh-** ' . The linguists suggested inserting the last consonant of the stem before the attachment of these suffixes to correct the errors so that words such as **beekachuu** become beek**k**achuu.
- Suffixes are compiled for particular stem classes, which are in turn grouped by their endings. Stems in the same class behave similarly in surface word synthesis for the majority of suffixes. But some suffixes need to be applied according to the sound structure of the stem. For instance, the suffix **'-ni'** in the class1 of verbal stem works perfect for stems that have short sound before the last consonant like **dhuf-+-ni=dhufni** 'to come', but it gives no sense for stems having long sound before the last consonant like **deem-+-ni=deemni** 'to go', rather it should be **deemsi** or **deemichi**. This problem can be stated as the concatenation of a suffix for wrong stem class.

- An important property of reduplication that has not been mentioned in any literature was suggested by the linguists. If the second consonant of verbal stem is double, then reduplication of the stem takes the form CVCV, where C=consonant and V=vowel, pattern rather than CVCCV pattern unusually. For example, **barreesse** 'he wrote' becomes **babarreesse** 'he wrote several time' rather than **babbarreesse**.

- The linguists suggested that more words can be generated by considering double causatives, but we used only single causatives in this thesis.

- Afaan Oromoo has many dialects. So if all of them are considered, even tremendous number of words can be generated from a single stem as suggested by the domain experts.

Generally, from the feedbacks and suggestions of the linguist who evaluated the system, we observed that the performance of the system can improve if the recommendations are included. Moreover, the researcher observed that Afaan Oromoo is morphologically complex, but at its infacy in the area of morphology and phonology, and needs to be studied linguistically to promote the computational aspect.

CHAPTER EIGHT: CONCLUSIONS AND FUTURE WORK

8.1. CONCLUSIONS

The purpose of this study was to design a morphological synthesizer for Afaan Oromoo particularly for verbs and nouns. To develop word generation algorithm, the knowledge of language morphology is necessarily required. Accordingly, the morphological properties of Afaan Oromoo have been studied to develop the synthesizer. Then, various techniques to morphological synthesis are reviewed. Rule based approach has been employed as development method because it takes the morphological properties of the language into account. A database consisting of stems as knowledge base, rules and suffixes used has been designed. The boundary change rules such as deletion, epenthesis and assimilation were considered. Words in the language are largely formed by affixation process, suffixation being dominant. The reduplication and pre-fixation processes also share some roles in the generation process. Most of the algorithms are designed from scratch as there are no previously designed algorithms for this purpose based on the morphological properties of the language to generate verb and noun forms from an input stem. Some algorithms have also been adapted from other languages.

The stem is a basic lexical unit of the language for both verbs and nouns. These stems are categorized into classes according to their morphological properties. The words are generated from stems through affixation. Therefore, morphological analyzer can reduce produced words into some stem paradigms. This enables the language to be processed in rule based approach and algorithmically. Finally, a prototype morphological synthesizer was developed to evaluate the performance of the designed algorithms. The analysis of the total number words generated from single stem especially verbal stem shows that Afaan Oromoo is morphologically complex language.

In order to test the accuracy of the algorithms developed the stems from the existing classes were considered. From the experiment, it is possible to say that the performance (96.28% for verbs and 97.46% for nouns) of the prototype is acceptable. The study has indicated developing the synthesizer for Afaan Oromoo in rule based approach for verbs and nouns. It is pretty easy to extend the system for other parts of speech with minimum effort. As it stands, the prototype can handle word generation in adjectives as their properties are similar with nouns.

We hope that this research paves a way for a full fledged Afaan Oromoo morphological synthesizer/analyzer for those who want to pursue conducting research in natural language processing in the language. This work plays important role in the spell checker as suggestion list generator, machine translation, information retrieval and so on.

8.2. CONTRIBUTION OF THE WORK

The main contributions of this thesis work are summarized as follows:

- The lexicons suitable for Afaan Oromoo synthesizer have been designed
- The study has developed many new, and some adapted algorithms for Afaan Oromoo Morphological Synthesizer.

8.3. FUTURE WORK

Morphology needs deep understanding of the features and rules of language under consideration. Hence, there are a number of holes for improvement and modification for Morphological synthesizer of Afaan Oromoo. Below are some of the recommendations we propose for future work.

- There are different dialects spoken in Afaan Oromoo. Though it is difficult to handle all dialects, the more dialects considered the more number of words generated from single stem. Therefore, dialect consideration is crucial to have more generated words.

- This thesis basically records the processes involved in developing a computational framework for the verbs and nouns of Afaan Oromoo. The **HORSIISAA** program developed thus involves many things starting from linguistic study to developing a lexicon for the program to work on to developing a user interface and the prototype. As the ambition of this research is to develop a computational framework for the verbs and nouns morphology of the language, the program does not aspire to account for all the grammatical categories in the language under discussion. Enhancing the system to include all the other parts of speech to make full fledged system is also an interest in future works.

- The morphological processes considered in this thesis are inflections and derivations. Further linguistic study should be considered to include the third process, compoundation.

- It is also important to develop Afaan Oromoo morphological Synthesizer using machine learning or hybrid and compare the performance with the rule based approach used in this thesis.

- The synthesizer is lower level input for other applications like spell checker, dictionary compilation, POS tagger, machine translation and so on. Therefore, further studies should also focus on implementing application that can use the outputs of word synthesizers, and integrating the components together should be considered.

- There are a lot of holes in the linguistic study of the language in general, and in morphology and phonology in particular. Linguists should give due consideration to intensively study the language structure and make it available for use in developing computational models.

- Afaan Oromoo is spoken widely in different regions of Oromia with different dialects. So, standardization in written Afaan Oromoo language is essential, for instance, as to which suffix is appropriate for particular noun.

- The prefixes and suffixes are almost exhaustive. There may be some prefixes, suffixes and rules missing during manual compilation, and they can be added in the lexicon to give additional surface forms.

REFERENCES

[1] Allen J., Natural Language Understanding. 2nd Ed. California: Redwood, Benjamin/Cummings Publishing Company, Inc, 1996.

[2] Daniel Jurafsky & James H., Speech and Language Processing: An introduction to speech recognition, natural language processing, and computational linguistics, Prentice-Hall, Inc, 2000.

[3] Diriba Megersa, "Thesis: An automatic sentence parser for Oromo language using Supervised learning technique", Department of Information Science, AAU, 2002.

[4] Hawine Alemayehu, "Thesis: The structures of nominal clauses in Afan Oromo A minimalist approach", Department of Linguistics, AAU, 2007.

[5] Wakshum Mekonnen, "Thesis: Development of a Stemming Algorithm for Afan Oromoo Text", Department of Information Science, AAU, 2000.

[6] Kula Kekeba**, "**Experimentation Report: Evaluation of Oromo-English Cross- Language Information Retrieval", Language Technologies Research Centre IIIT, In IJCAI 2007 Workshop on CLIA, Hyderabad, 2008

[7] Kula K. T., Vasudeva Varma and Prasad Pingali. Evaluation of Oromo-English Cross-Language Information Retrieval, In IJCAI 2007 Workshop on CLIA, Hyderabad (India), 2007

[8] Mao Y., "Natural Language Processing Module (Part of Speech Tagging and Sentence Parsing) Laboratory Manual" Available at http://www.csic.cornell.edu/201/natural_language/, 1997.

[9] Mekonnen Hundie, "Thesis: Lexical Standardization In Oromo", Department of Linguistics, AAU, 2002.

[10] Mesifin Getachew, "Thesis: Automatic Part of Speech Tagging for Amharic Language: An Experiment Using Stochastic Hidden Markov (HMM) Approach", Department of Information Science, AAU, 2001.

[11] Salton G., Natural Language Processing. Introduction to Modern Information Retrieva***l***. New York: McGraw-Hill, 1983.

[12] Uibo H., On Using the Two-Level Model as the Basis of Morphological Analysis and Synthesis of Estonian. Available at www.ut.ee/~heli_u/art/LREC02-Uibo.pdf, 2001

[13] Gumii Qormaata Afaan Oromoo. Caasluga Afaan Oromoo, Jildii – 1. Komishinii Aadaaf Turizmii Oromiyaa. Finfinnee, Ethiopia, 1995 E.C.

[14] Kettunen, "Dissertation:Reductive and Generative Approaches to Morphological Variations of Keywords in Monolingual Information Retrieval", Department of Information Studies, University of Tampere , 2007

[15] Ann Copestake, "Lecture notes: Natural Language Processing, University of Cambridge,Computer Laboratory", 2003

[16] Sanat Kumar Bista and Birendra Keshari,"project:Nepali Lexicon Development"**,** Kathmandu University,Information and Language Processing Research Lab, 2003

[17] Marry Ann Tan and Rachel Roxas, FIMOLE: Filipino Morphological Learner, De La Salle University, College of Computer Studies, MSc thesis, 2004

[18] Trost H., Computational Morphology, http://www.coli.uni-saarland.de/~schulte/Teaching/ESSLLI-06/Referenzen/Morphology/trost-2003.pdf, 2000, last accessed on May 3, 2010

[19] Koskenniemi, Kimmo, Two-Level Morphology: A general Computational Model for Word-form recognition and production. *Publication 11,* Universityof Helsinki, Department of General Linguistics, Helsinki, 1983

[20] Abara Nefa. Long Vowels in Afaan Oromo: A Generative Approach, M.A. Thesis, School of Graduate Studies, Addis Ababa University, 1988.

[21] Oromo Language, http://en.wikipedia.org/wiki/Oromo_language, last accessed on jan 1, 2010

[22] Carol Peters and Paraic Sheridan, Multilingual information access, In ESSIR '00: Proceedings of the Third European Summer-School on Lectures on Information Retrieval-Revised Lectures, Springer-Verlag, 2001

[23] Karttunen L., Constructing Lexical Transducers, In the Proceedings of 15th International Conference of Computational Linguistics. COLING-94 pp. 406 – 4111983, 1994

[24] Kibur Lisanu "Thesis: design and development of automatic morphological synthesizer for Amharic perfective verb forms", Department of Information Science, AAU, 2002.

[25] Samit B. et al, Inflectional Morphology Synthesis for Bengali Noun, Pronoun and Verb Systems, Proc. of the National Conference on Computer Processing of Bangla (NCCPB 05), Dhaka, Bangladesh, March, 2005

[26] Pullman et al, Computational Morphology of English. Linguistic 26, 545 – 560, 1988.

[27] Antworth E., Morphological Parsing with Unification Based Word Grammar. North Texas Natural Language Processing Workshop, University of Texas, 1994, http://www.sil.org/pckimmo/ntn1p94.html, last accessed on December 20, 2009.

[28] Ingo Plag, Word-formation in English, Cambridge University Press, 2002

[29] Tom M. Mitchell, Machine Learning, McGraw-Hill, 1997

[30] Andrew Roberts , Machine Learning in Natural Language Processing, http://andy-roberts.net/misc/latex/sessions/bibtex/bib_example_nat.pdf, last visited on Feb, 2009

[31] Assefa W/Mariam, "Thesis: Development of Morphological Analyzer for Afaan Oromoo text", Department of Information Science, AAU, 2005.

[32] Chenda N. and Wataru K, Hybrid Approach for Khmer Unknown Word POS Guessing, In The 2007 IEEE International Conference on Information Reuse and Integration, August 2007

[33] Dr.Abdulsamad M., Seerluga Afaan Oromoo,Caffee Oromiyaa,Finfinnee, 1994

[34] Tilahun G., Seera Afaan Oromoo, Finfinnee, 1995

[35] Habte B., "thesis: Analysis of Tone in Oromo", Institute of Language Studies, AAU, 2003

[36] Hana S., Czech Morphological Lexicon, In Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology, Madrid, 41– 47. ACL., 1997

[37] Abebe K., "Thesis: Case System in Oromo", Addis Ababa University, 2002

[38] Madhavi G. and Lori L., TelMore: Morphological Generator for Telugu Nouns and Verbs, Proc. Second International Conference on Universal Digital Library, Vol Alexandria, Egypt, Nov 17-19, 2006

[39] Violetta C. et el, Arabic Morphology Generation using concatenation strategy, In Proceedings of the 6th Applied Natural Language Processing Conference (ANLP 2000), Seattle, Washington, USA, 2000

[40] Andrew Carstairs-McCarthy, An Introduction to English Morphology: Words and Their Structure, Edinburgh University Press, 2002

[41] Getachew Mamo, "Thesis: Part-Of-Speech Tagging For Afaan Oromo Language Using Transformational Error Driven Learning (Tel) Approach", Department of Information Science, AAU, 2009.

[42] Gobinda G. Chowdhury, Natural Language Processing: Department of Computer and Information Sciences, University of Strathclyde, Glasgow G1 1XH, UK, http://www.cis.strath.ac.uk/cis/research/publications/ papers/strathcis_publication_320.pdf, last accessed on February 8, 2010

[43] Morka Mekonnen, Text to speech system for Afaan Oromo, MSc Thesis, School of Graduate Studies, Addis Ababa University, 2001.

[44] Alemayehu Dumessa, Word Formation In Diddessa Mao, Msc Thesis, School Of Graduate Studies, Addis Ababa University, 2007.

[45] A.G. Menon, Amrita Morph analyzer and Synthesizer for Tamil: A Rule Based Apprroach, Amrita University, 2009

[46] Nizar Habash, Large Scale Lexeme Based Arabic Morphological Generation, Large scale lexeme based Arabic morphological generation. In *Proceedings of Traite- ment Automatique du Langage Naturel (TALN-04)*, Fez, Morocco, 2004

[47] http://www.silinternational.com/pckimmo/v2/doc/overview.html, last accesed on May 27, 2010

[48] Guido Minneny, John Carroll and Darren Pearce,Applied Morphological Processing of English, Cognitive And Computing Sciences, University Of Sussex, Brighton Bn1 9qh, Uk, Cambridge University Press, 2001

[49] http:// www.csa.gov.et, last accessed on May 30, 2010

[50] Tony McEnery&Andrew Wilson, Corpus Linguistic, Edinburgh University, 2001.

[51] McShane, Marjorie, and Sergei Nirenberg. 2003. Blasting open a choice space: Learnininflectional morphology for NLP. *Computational Intelligence* 19(2):111–135

[52] http://webcache.googleusercontent.com, last accessed on june 28, 2010

[53] M.G. khayat et al, An Arabic Morphological Analyzer/Synthesizer, in the journal of *KFUPM, 2009*

APPENDICES

Appendix A: List of verb suffixes

a	adhee	amtu	atte	dani	ine	itani	nnu	ta	uuttan
aa	adhu	amtuu	atti	de	inu	ite	nu	tani	uutti
aaf	adhuu	amu	attu	dha	is	iti	oofna	taniittu	xa
aas	ama	amuu	atu	dhe	isa	itu	oofta	te	xani
aat	amaa	amuudhaa	chiisa	dhu	isan	ja	ooftan	teetta	xe
aatii	aman	amuudhaaf	chiisan	di	ise	jani	oofte	teetti	xi
aatu	amani	amuuf	chiise	du	isisa	je	oofti	ti	xu
achisa	amanii	amuun	chiisna	duu	isise	ju	ra	tu	xuu
achisan	ame	ani	chiisne	e	isisna	la	re	tuu	
achise	amne	aniiru	chiiste	eera	isista	le	ru	u	
achisna	amni	anna	chisiisa	ees	isistan	lu	se	ullee	
achistan	amoo	anne	chisiisan	eet	isiste	na	sisa	umsa	
achiste	amta	annu	chisiista	eeti	isna	naan	sisan	uu	
achuu	amtan	ata	chisiistan	i	istan	ne	sise	uuf	
achuuf	amtani	atani	chisiiste	ifna	iste	neerra	sisna	uufan	
adha	amte	ate	chisiistu	ifte	isu	ni	sisne	uufi	
adhe	amti	atini	da	ina	ita	nna	siste	uufii	

Appendix B: List of noun suffixes

aa	eeyyii	iinis	irratti	llee	ooliin	s	umallee
aaf	f	iis	irrattillee	n	ooliiwwan	tii	umarratti
an	I	illee	irrattis	ni	ooma	tiin	umarrattis
aniif	icha	irraa	irrattuu	oolee	oota	tu	umatti
aniin	ichi	irraahillee	itti	ooleedhaan	ootaaf	uma	umattillee
arraa	ii	irraahis	ittii	ooleef	ootaan	umaa	umattis
atti	iif	irraahuu	ittiin	ooleen	ootadhaan	umaaf	umattuu
dhaa	iifis	irraan	ittillee	ooleewwan	ootawwan	umaafillee	ummaa
dhaaf	iifuu	irraanillee	ittis	oolii	ootni	umaanillee	ummaan
dhaan	iin	irraanis	ittuu	ooliidhaan	oottan	umaanis	wwan
een	iinillee	irraanuu	lee	ooliif	rra	umaanuu	yyuu

Appendix C: Sample questionnaire for checking the validity of generated words

Automatic Afaan Oromoo Morphological Synthesizer: Master's thesis at Department of Computer Science, Addis Ababa University

QUESTIONNAIRE FOR KEY INFORMANTS

Dear Respondent,

Morphological synthesis or generation is a process of returning one or more surface forms from a sequence of morpheme glosses. The morphological synthesizer in this case will enable one to generate the surface form (e.g. cries) from its constituent distinct parts called morphemes (e.g. /cry / and /s/). That means /cry/ + /s/= /cries/. In Afaan Oromoo, from the stem *deem*-'to go', for instance, one can generate **deemuu, deemna, deeman, deemtu, deemte, deemsa, deemuuf, deemsa, deemaniiru, deemneerra, etc**. The process entails inflections (number, tense, and person), derivations and reduplication for verbs, and inflections and derivations for nouns.

Automatic Afaan Oromoo Morphological Synthesizer is currently being developed at the Computer Science Department of the Addis Ababa University as master's thesis. This research is expected to provide valuable inputs for Afaan Oromoo as a subcomponent of natural language processing systems in applications like machine translation, dictionary (lexicon) development, speech recognition, parts of speech tagging, automatic sentence construction, spelling and grammar checking etc. In addition, it can be used in Oromoo teaching and learning, suggestion lists generation for spell checker and in Web search to automatically search for the inflected forms of the word even if the user only typed in the base form.

To this effect, I kindly request you to provide us the necessary and genuine information to the researcher by responding to the questions about the validity of the words generated by the system, on the attached questionnaire.

We would like to inform you that your responses will remain anonymous and will be used only for the above stated purpose.

Thanks in advance for your cooperation,

Kind regards,

The researcher
Department of Computer Science
Addis Ababa University

Section I: **Respondent Identification**

Position/Job title…….
Education Level…….

Section II: Answer the following questions based on the tables given below.

1. From the words listed in the following pages, put the sign (X) in front of the incorrectly generated word.
2. Could you specify the correct form?
3. Do you think more words can be generated? If yes, mention the mechanism (inflection, derivation, reduplication …). Please give example.
   _____
   _____
   _____
   _____
   _____
   _____
4. Provide general comment?

_____
_____
_____

1. beekna
2. beekne
3. beekni
4. beeknu
5. beeka
6. beekkachuu
7. beekkachuuf
8. beekkadha
9. beekkadhe
10. beekkadhu
11. beekama
12. beekamaa
13. beekaman
14. beekamne
15. beekamoo
16. beekamta
17. beekamtan
18. beekamte
19. beekamti
20. beekamtuu
21. beekamuuf
22. beekani
23. beeke
24. beekeera
25. beekI
26. beekuu
27. beekuuf
28. beekneerra
29. beekaaf
30. beekaas
31. beekaat
32. beekaatu
33. beekuuttan
34. beekuutti
35. beekaa
36. beeknaan
37. beeku
38. beekeet
39. beekeeti
40. beekees
41. beekis
42. beekuufan
43. beekuufi
44. beekuufii
45. beekkadhuu
46. beekkadhee
47. beekamaan
48. beekamanii
49. beekamee
50. beekamni
51. beekamu
52. beekamtu
53. beekamtani
54. beekamuu
55. beekamuudhaa
56. beekamuudhaaf
57. beekamuun
58. beekullee
59. beekaatii
60. beekumsa
61. beekta
62. beektani
63. beektaniittu
64. beekte
65. beekti
66. beektu
67. beektuu
68. beekteetta
69. beekteetti
70. beeksisa
71. beeksisan
72. beeksise
73. beeksisna
74. beeksiste
75. bebbeekna
76. bebbeekne
77. bebbeekni
78. bebbeeknu
79. bebbeeka
80. bebbeekkachuu
81. bebbeekkachuuf
82. bebbeekkadha
83. bebbeekkadhe
84. bebbeekkadhu
85. bebbeekama
86. bebbeekamaa
87. bebbeekaman
88. bebbeekamne
89. bebbeekamoo
90. bebbeekamta
91. bebbeekamtan
92. bebbeekamte
93. bebbeekamti
94. bebbeekamtuu
95. bebbeekamuuf
96. bebbeekani
97. bebbeeke
98. bebbeekeera
99. bebbeekI
100. bebbeekuu
101. bebbeekuuf
102. bebbeekneerra
103. bebbeekaaf
104. bebbeekaas
105. bebbeekaat
106. bebbeekaatu
107. bebbeekuuttan
108. bebbeekuutti
109. bebbeekaa
110. bebbeeknaan
111. bebbeeku
112. bebbeekeet
113. bebbeekeeti
114. bebbeekees
115. bebbeekis
116. bebbeekuufan
117. bebbeekuufi
118. bebbeekuufii
119. bebbeekkadhuu
120. bebbeekkadhee
121. bebbeekamaan
122. bebbeekamanii
123. bebbeekamee
124. bebbeekamni
125. bebbeekamu
126. bebbeekamtu
127. bebbeekamtani
128. bebbeekamuu
129. bebbeekamuudhaa
130. bebbeekamuudhaaf
131. bebbeekamuun
132. bebbeekullee
133. bebbeekaatii
134. bebbeekumsa
135. bebbeekta
136. bebbeektani
137. bebbeektaniittu
138. bebbeekte
139. bebbeekti
140. bebbeektu
141. bebbeektuu
142. bebbeekteetta
143. bebbeekteetti
144. bebbeeksisa
145. bebbeeksisan
146. bebbeeksise
147. bebbeeksisna
148. bebbeeksiste
149. ni beekna
150. hin beekne
151. hin beeknu
152. ni beeka
153. ni beekkadha
154. ni beekkadhe
155. hin beekkadhu
156. ni beekama
157. hin beekaman
158. hin beekamne
159. ni beekamta
160. hin beekamtan
161. ni beekamte
162. ni beekamti
163. ni beeke
164. hin beeku
165. ni beekkadhee
166. hin beekamaan
167. ni beekamee
168. hin beekamu
169. hin beekamtu
170. hin beekamtani
171. ni beekullee
172. ni beekumsa
173. ni beekta
174. hin beektani
175. ni beekte
176. ni beekti
177. hin beektu
178. ni beeksisa
179. hin beeksisan
180. ni beeksise
181. ni beeksisna
182. ni beeksiste
183. ni bebbeekna
184. hin bebbeekne
185. hin bebbeeknu
186. ni bebbeeka
187. ni bebbeekkadha
188. ni bebbeekkadhe
189. hin bebbeekkadhu
190. ni bebbeekama
191. hin bebbeekaman
192. hin bebbeekamne
193. ni bebbeekamta
194. hin bebbeekamtan
195. ni bebbeekamte
196. ni bebbeekamti
197. ni bebbeeke
198. hin bebbeeku
199. ni bebbeekkadhee
200. hin bebbeekamaan
201. ni bebbeekamee
202. hin bebbeekamu
203. hin bebbeekamtu
204. hin bebbeekamtani
205. ni bebbeekullee
206. ni bebbeekumsa
207. ni bebbeekta
208. hin bebbeektani
209. ni bebbeekte
210. ni bebbeekti
211. hin bebbeektu
212. ni bebbeeksisa
213. hin bebbeeksisan
214. ni bebbeeksise
215. ni bebbeeksisna
216. ni bebbeeksiste

Appendix D: Sample Rules

Rules of morphotactics

1. The rule for inflected nouns: *Noun form= stem + number + case*
2. The rule for inflected verbs: *Verb form= prefix +stem + voice + aspect + number + person*

Rules of assimilation

Completely discussed in the body of this thesis

Rules of deletion

1. Rules of deletion for nouns: *stem=stem-the last vowel(s)*
2. Rules of deletion for verbs: *stem= stem- the last "dh/h/'"*

Rules of epenthesis

If verb forms contain the sequence *CCC*, where C=consonant and the last two is different from dh, ch, ny, sh, ph, ts, insert 'i' before the last consonant in the sequence i.e. *CCiC*

Rules of pluralization for classes 1-4,6

1. Stem+[suffix]=Stem[Suffix]
2. CVVCV=CVVCCan Where C=consonant and V=vowel
3. CVCV=CVCCeen Where C=consonant and V=vowel
4. Stem+eeyyii=stem-eessa/eettii...+eeyyii, where – stands for deletion
5. Stem+(o)ota/+(o)olee/+(o)olii= Stem-last vowel(s)+(o)ota/+(o)olee/+(o)olii, where – stands for deletion

Rules of derivation

1. Noun Stem +[-eenya, -ooma, -ummaa]=abstract noun
2. verb stem + [-eenya, -ina, -noo, -ii, -ee, -a, -aa, -iinsa, -iisa, -umsa, -maata, -tuu, -uu] =derived noun

Declaration

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

Declared by:

Name: _____

Signature: _____

Date: _____

Confirmed by advisor:

Name: _____

Signature: _____

Date: _____

Place and date of submission: Addis Ababa University, June 2010