DEVELOPMENT OF STEMMING ALGORITHM FOR GURAGEGNA TEXT

A Thesis Presented

By

Mehbub Ebrahim Abdella

to

The Faculty of Informatics

of

St. Mary's University

In Partial Fulfillment of the Requirements

for the Degree of Master of Science

in

Computer Science

June, 2023

Addis Abeba Ethiopia

ACCEPTANCE

DEVELOPMENT OF STEMMING ALGORITHM FOR GURAGEGNA TEXT

By

Mehbub Ebrahim Abdela

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

Internal Examiner

Full Name	**Signature**	**Date**
Dr. Mulugeta Adbaru	_____	_____

External Examiner

Full Name	**Signature**	**Date**
Dr Mesfin Abeba	_____	_____

Dean, Faculty of Informatics

Full Name	**Signature**	**Date**
Dr Alembante Mulu	_____	_____

June, 2023

DECLARATION

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

Mehbub Ebrahim Abdela

_____

Signature

Addis Ababa

Ethiopia

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

Dr Alembante Mulu Kumlign

_____

Signature

Addis Ababa

Ethiopia

June, 28 2023

ACKNOWLEDGMENT

Befor everything I wanna say Allahamdulilah, for everything Allah has made for me

I would like to use this opportunity to offer my sincere appreciation to everyone who has supported and encouraged me while I worked on my thesis. My adviser Ph.D. Alemebante Mulu is someone to whom I owe a big debt of appreciation. Without his supervision and his exciting and insightful thoughts, remarks, and recommendations, my thesis would not have been possible. I sincerely appreciate it.

I would like to express my gratitude to my beloved family for their support

Additionally, I would like to express my profound appreciation to the Gurage Zone Education Department, in particular Ato kebede demsee, for his assistance in supplying all the essential materials for the text corpus, including Guragegna dictionary and primary school textbooks.

Mehbub Ebrahim Abdela

June, 2023

Table of Contents

List of Figure

List of Appendix

List of Tables

List of Acronyms

CV	Consonant - Vowel sequence
CC	Consonant - Consonant sequence
1PL	1st Person Plural
1SG	1st Person Singular
2PL	2nd Person Plural
2PL/HON	2nd Person Plural/Honorary
2SG	2nd Person Singular
3F/PL	3rd Person Feminine/3rd Person Plural
3HON	3rd Person Honorary
3M	3rd Person Masculine
3MO	3rd Person Masculine Object
SW	Stemming weight
UI	Under stemming index
OI	Over stemming index

Abstract

The process of stemming involves stripping a word of its inflectional and derived variations. It is crucial for many applications of natural language processing. When analyzing the importance of page for user query which only specifies one form, the varied word structures used in searching and indexing should be anticipated. Conflation methods can help improve the efficiency of an IR system by condensing variant phrases into a single form. In order to standardize as many similar phrases and word patterns as possible. That may be utilized in the retrieval procedure, stemmers are employed in information retrieval.

During this type of research work, a solid awareness of the Guragegna grammar in addition an examination of the language's inflectional and derivational affix was required. The Gurage language generates several word forms using stems by use of affixation and reduplication (final, total, and frequentative). Prefix, suffix, and infix are frequently used affixations. Gurage often concatenates affixes, which can lead to almost large words with a lot of semantic content.

This study introduces the first stemming algorithm that conflates Guragegna phrase variants. Python programming was used in the creation of the Gurage stemmer. The researcher created little rule sets for related affixes in an attempt to follow an algorithm with a straightforward structure. In order to develop the stemmer, a list of stop words and the Experimental text document were both acquired from various sources along with a research article that covers the morphology of the Gurage language.

The iterative, context-sensitive, and recoding methods used in this study's stemmer eliminate prefix, suffix, and reduplicated letters that are final, total, and frequentative reduplicates. Prefix, suffix, and then letter reduplication were applied as part of this experiment's removing technique. in the evaluation process is contained in the Data set. The experiment text has 1,933 words, of which 1266 resulted from the stemming procedure, out of a total of 1266.The number of words successfully stemmed is 1097, achieving an accuracy of 86.65%. 13.34% of the stemmed words were wrongly stemmed. Over stemming accounts for 7.97% (101) of the terms, while under stemming accounts for 5.37% .

Keywords: stemming algorithm; Guragegna stemmer; context-sensitive stemmer; iterative stemmer; Guragegna language

Chapter 1

INTRODUCTION

1.1 Background of The Study

With improvements in information technology, documents and information in electronic form are rapidly increasing. Organizing, managing, and retrieving relevant documents from such a collection of database sit becomes difficult and time-consuming [1]. To overcome the above problems many information processing systems have been developed Including management information system, database information system, data retrieval system and information retrieval system [2]. Information retrieval aims to retrieve all relevant documents for a user query using indexed terms in natural language text [3] , Text can be unstructured and ambiguous. In order to meet the search needs of the user, user requests must be translated into queries that can be processed by the information search system [4].

By removing each word's derivational and inflectional suffixes, a process known as stemming produces a form that unifies all words with the same root. [5] . Because natural languages are distinguished by morphological variations of phrase which can take on multiple forms due to insertion of distinct affixes, stemming is necessary. Stemming algorithms' main objective is to eliminate all affixe from the phrase (text) in order to leave only the stem. Additionally, stemming contributes to the regularization of an IR system's lexicon, which has benefits that are difficult to measure using conventional Information Retrieval experiments [6].

The majority of stemming research has been influenced by information retrieval. Information retrieval places an emphasis on the standardization of the representation, storage, organization, and access of data objects. The representation and structure of the data objects should provide users with simple access to the documents they're looking for or are interested in, in particular. For efficient and effective information retrieval, system translation, and word summarization morphological processing is frequently utilized. [7]

The development of various applications for natural language processing, such as text classification, text categorization, and morphological analyzer, will benefited from designing a stemming method for the Guragegna language.

A word can take on several forms in inflectional morphology without affecting which part of speech it belongs to. The variances are typically brought about by adjustments to the person, number, tense, and gender. According to [8]. such changes have no bearing on a word's class, hence a verb remains a verb even if its tense form is changed., "ሸም" (want), "ይሸም" (wants), "ቲሸም" (wanting), "ሸምንመ" (wanted). And derivational morphology leads in an alteration in the class of a word. For example, an affix transforms a word from adjective to noun, verb to noun, noun to verb, and so on.; like "አበጐዳ" (friend፣ ጓደኛ), "አበጐዳነ" (friendly) and "አበጐድነት" (friendship፣ ጓደኝነት)

The "Guragegna" are the languages spoken by the Gurage people. Gurage people live in the Gurage zone in Southern Nation Nationalities and Peoples state, which is bounded to the south by the Rift Valley lakes in the East [9], the River Awash in the north, and the River Gibe in the west and southwest of Addis Abeba. Geographically, Gurage is located about 100 kilometers south of Addis Abeba. Gurage is a catch-all term for Semitic-speaking peoples who live in the south of Addis Abeba.. Gurage zone is currently divided into fifteen administrative Wereda, which is encircled by the Cushitic-speaking groups of Oromo and Hadiya. [10]

different linguists classified Gurage language varieties in Leslau [11] the three dialect Groups of the Gurage languages as follows: Selte, Wolane, and Zay are connected to Harari in the East; Chaha, Ezha, Ennemor, Endegegn, Gyeta, Goggot, Muher, and Masqan are connected to Amharic in the West; and Soddo and Gurage are considered to be a single language in the North (East). Moreover, Hetzron [12] likewise creates a typological unit for the outer South Ethio-Semitic languages.

There have been numerous stemming algorithms proposed for various languages. The design of those stemmers ranges from the most basic approach, which involves the elimination of suffixes, to the use of a list of common suffixes, so as a result, a stemmer's overall functionality and efficacy in NLP applications vary depending on the language. In the majority of cases, the layout of stemmers is language specific and requires a significant amount of linguistic information within the language as well as an understanding of the needs of writing systems for that language.

1.2 Motivation of The Study

We are in the era of where everything is connected to computer. which will lead do digitalization of everything in the future every books, journals and magazines will be in digital. At this time human languages are being used to communicate with machine it's because of Artificial intelligence and it will grow more and more with in decade.

Finding unstructured content that satisfies information needs from vast volumes of data is a technique known as information retrieval. One method for overcoming the vocabulary mismatch issue in information retrieval (IR) is stemming. In order to decrease the size of index files and increase the retrieval efficiency, information retrieval uses the process of stemming, which reduces words to their stem. searching and indexing of word. Before assigning a term to an index term, any suffixes and prefixes are typically removed.

So, in order to retrieve correct or more meaningful data for vast collection information it need stemming algorithm to get in to root word for better prediction and as described above it is era of computer in order preserve the language it must be in digital form. As native of Gurage community it is my aim to preserve and help to grow the language in digital word. So my algorithm will help to achieve both of them

1.3 Statement of The Problem

Guragegna is the native language for the Gurage people. In the Gurage Zone, it is one of the most widely used languages. It is also spoken in various areas of Ethiopia, particularly in areas where the Gurage people have lived. [13]. Guragegna is morphologically very complex and a highly inflected language. It uses both kinds of morphologies, inflectional and derivational morphologies. Each inflectional and derivational morphologies in Guragegna bring about very huge numbers of variants for a single phrase.

As Guragegna is morphologically complex language It must be automated processes that can shrink a word to a tolerable size, boost retrieval efficiency, and capture the close connections between the language's many word forms. By eliminating inflectional and derivational affixes, stemming is a technique for distinguishing a word stem from a whole phrase, and there has been a lot of interest in designing algorithms for this purpose. Ethiopia has over 86 different languages. However, few studies have been conducted to develop a stemmers for local languages such as

Afan Oromo [16] , Ge’ez [17], Tigrigna [15], Amharic. [14] , Wolayta [18] ,Kambaata [19] and Slite [20] to make the issue of the respective languages' morphological complexity easier

According to early research, there are a lot of electronic papers in Guragegna's schools and government agencies, and their number keeps growing. However, there isn't a tool or system that enables businesses to obtain useful data for making decisions and other kinds of problem-solving As far as the researcher is aware, however, the Guragegna language has incredibly little linguistic resources, and there have been no computational efforts to computerize or automate the language. The fundamental information retrieval technologies must be created and implemented to enable the necessary access to this wealth of information and to support its growth. The creation of a stemming algorithm for the Gurage text may potentially open the door to the creation of other forms of natural language processing, including text categorization, text summarization, machine translation, and morphological analyzers.

1.4 Research Question

1. What are the characteristics and word-formation patterns in Guragegna language?
2. What are difficulties or problems in designing stemmers for Guragegna texts?
3. Using the Experimental document as a benchmark, does the designed algorithm report good performance?

1.5 Objectives of the study

1.5.1 General Objective

The main objective of this research is to develop stemming algorithm for Guragegna language text

1.5.2 Specific Objectives

To achieve the above general objective the following specific objectives will be attempted in the research work

- To analyze and understand various stemming techniques that have been created for other languages.
- To study the morphology of the Guragegna language and know how can stemming be achieved.

- To investigate, modify, and establish guidelines for stemming Guragegna texts
- To develop a stemmer for Guragegna language.
- To construct a list of stop words and affixes used in corpus
- To evaluate how well the stemmer performs on the chosen Experimental text

1.6 Scope and Limitation of The Study

The design and development of a stemmer for the Guragegna language is the goal of this research. The current research mainly examined the language's in both inflectional and derivational word varieties gurage. Prefix removal, suffix removal, and letter reduplication all have three distinct the stemmer also includes context-sensitive and recoding rules. The first type of reduplication is frequentative, the second is total reduplication, and the third is total reduplication removal. By putting the prefix letters to the list of prefix and suffix letters to list of suffix, prefix-suffix removing can be prevented from being included in the stemmer. Compound and irregular words as well as the infix stripping technique were not included in this study, which is a drawback of the research due to its complexity and time limits.

1.7 Methodology

1.7.1 Literature Review

Evaluation of documents is done in order to learn the language's features. A literature review was conducted to obtain data and comprehend the language because understanding the language's morphology is a crucial part of the study. A review of language-related works is conducted by looking through a variety of sources, including books, textbooks, journals, etc, in order to comprehend the morphology of Guragegna.

1.7.2 Data Source

One of the fundamental resources needed for research on natural language processing is a text corpus. A large body of material can accurately depict a language's morphological behavior. Therefore, choosing the right text is essential to developing a stemmer. The text will be used to gather prefixes, suffixes, and stop words. Additionally, the method will be tested using the text

corpus. School materials, fiction, other literature, and other sources will all be used for this investigation.

1.7.3 Programming Techniques

The stemmer algorithm for Guragegna text will be created using the Python computer language. This is due to the fact that Python makes manipulating text very simple, and also because the researcher has some programming experience with Python.

1.7.4 Genral approach

Design Science Research technique is the overall research methodology chosen for this study, and it is used to create the algorithm. In order to conduct design science research, an original, useful artifact must be made for a specific issue area. Problem discovery, solution formulation, development, testing, and conclusion are all steps in this research process.

Problem identification: Reading about NLP research issues in Ethiopia, notably the research gaps, led to the identification of the research issue in this study. Reading about the study gaps in the area therefore gave the researcher the chance to become aware of the limitations of stemming research and made it simple to determine which languages have not been explored in this context.

Suggestion: After identifying the Problem, a study plan was created with the intention of doing fresh research for gurage language while utilizing the underlying information already at hand.

Development: A rule-based stemming approach was chosen as part of this procedure, and a suitable algorithm was created for the Guragegna language based on a thorough analysis of its morphology. The stemming algorithm, the study's final artifact, is created and put into use using the Python programming language employing context-sensitive and iterative methods.

Evaluation: Following algorithm development, the stemmer was assessed using error counting techniques. The evaluation's findings were expressed in terms of appropriately, excessively, and inadequately stemmed words.

Conclusion: Conclusions have been drawn from the primary research findings at the completion of the study procedure. In this phase, the difficulties encountered when creating a stemmer for the Gurage language are also explained, as well as the artifact's summary behavior.

1.8 Significance of The Study

A word in Guragegna might have a lot of different variants, and combining these variants improves information retrieval performance. Creating a stemming algorithm for the conflated Guragegna variant words. The creation of tools like document summarizers, indexers, thesauri, word frequency counters, and spell checkers can be beneficial from developed stemmer Additionally, it can be applied to minimize word variations and the overall quantity of files.

1.9 Organization of The Thesis

There are five chapters in the thesis statement. The beginning, which is the first chapter, includes background information, a statement of the problem, the objective, the methodology, the scope and limitations, and the significance and applications of the research. The concepts and methods of stemming algorithms are explained in the second chapter. The various stemming procedures are covered in detail. Review of stemmers created for native and foreign languages are related works is also done.

Chapter three examines the morphology of the Guragegna language. This chapter's main interests are the language's both inflectional and derivational morphology. The chapter will also go into detail on the verb, adjective, and noun word construction procedures in Guragegna.

The development and evaluation of the stemming algorithm for texts in Guragegna are covered in the fourth chapter. This chapter includes lists of stop words and affixes. The development process for the stemmer and the selection criteria are also discussed in this chapter. Conclusions drawn from the data are presented in the last chapter, chapter five, along with suggestions for additional research.

Chapter 2

Literature Review and Related Works

2.1 Conflation Techniques

The presence of various variants for a phrase resulted from its diverse morphology. Some words could have several forms that call for some sort of processing to return them to their base word.. Consequently, the main issue with indexing and retrieval systems is the morphological variance of a phrase. To ensure that the gender of the subjects and the verbs correspond, variety of tenses, person, temper, or voice, phrases may alter within a sentence. [20].

Morphemes can be derivational or inflectional, which means they can create new words or give current words new inflection. Derivational morphemes are those that alter the word's part of speech. For instance, wonder- wonderful. A word becomes an adjective as a result. The word that has a derivational morpheme added to it is referred to as a derivate. The new word will differ from the old one in meaning.

Inflectional morphemes are suffixes that are attached to a word to give it linguistic value. It has the ability to give a tense, number, comparison, or possession. For example Plural: ‘‘Bike’’ ,‘Bikes’, ‘‘Car’’, ‘‘Cars’’.

Word conflation is the technique of reducing a bunch of words with a common meaning to a single term, or in other words, finding morphological variants of phrases with similar concepts and representing them through their root phrases. [20]. It works by grouping together comparable words with similar meanings in common forms.by utilizing conflation techniques to perform the conflation. There are manual conflation techniques and automatic conflation techniques. At search time, manual conflation is conducted using right-hand truncation. It is applied to query terms but not to documents. Some conventional online systems, such as ERIC (Education Resources Information Center), a virtual digital library system, allow users to truncate query terms by employing wildcard letters, such as an asterisk (*) [21]. For instance, if the first search word is abbreviated to FOUND*, more results on the subject FOUNDATION will be returned. However, people are frequently unfamiliar with the truncation method. [20]. emphasized that the manual right-hand truncation causes two key issues. the first is Over truncation refers to the final stem of

a phrase being stripped off too soon and the second Under truncation results in retrieval of too few relevant related phrases.

For instance, if a user over shortens "publication" to "public" any terms associated with "data" and "publication" in addition to being unrelated words can be obtained. however, when the word has been under truncated. A user is going to find a very small number of relevant phrases, if any. For instance, any important papers relating to "COMPUTERS" and "COMPUTATIONAL" will not be recovered if the word has been reduced to "COMPUTER" [22].

According to [3], stemming is a technique will be used to conflate or to decrease the morphological differences of phrases to a single word known as stem. The stemming procedure "is an algorithm" that lessens all phrases with a single root word to a common form by removing those word's derivational and inflectional affixes. Prefixes, infixes, and other phrase extensions are also eliminated. Stemming methods are critical as they improve the performance of file retrieval system and decrease the quantity of index files by combining many morphological period variations into one stem.

In some case steaming may look like lemmatization but it's different from lemmatization let look the difference However, the purpose of both steaming and lemmatization is the same. to reduce a word's various inflected and occasionally related derivative forms to its basic form [23]. Stemmers are computer algorithms that remove morphological variations from a phrase to create stems, enabling automated phrase conflation. Automatic term conflation uses 4 different techniques.

1. Affix removal.
2. n-gram method;
3. successor variety;
4. table lookup;

The Figure 2.1 will show the variety types of conflation methods and stemmers.

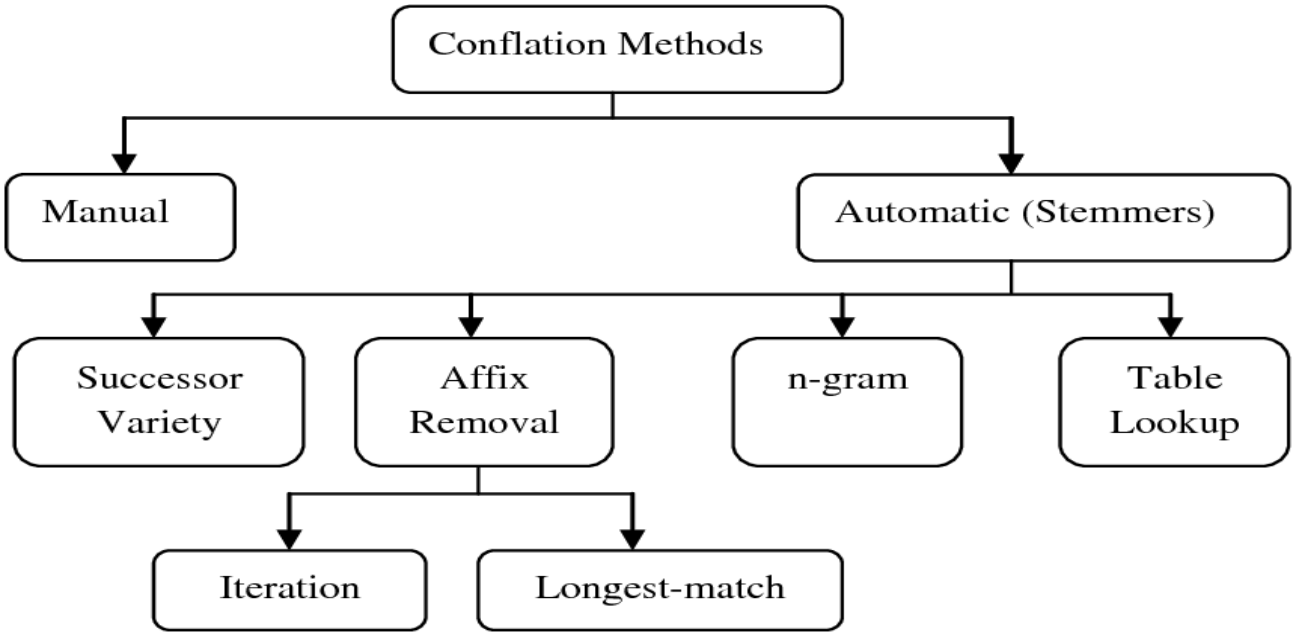

Figure 2.1 Methods for word or Phrase Conflation for stemming

2.2 Stemming Algorithms

Words are stripped down to their root form or basic form using the natural language processing approach known as stemming. To match word and retrieve relevant information or document from large database we will use query for evaluating the outcome or result of user interest. The Entered query may include different evaluation method which are Boolean operators that is, the operators AND, OR, and NOT are used to combine terms. In this model, keywords or index terms serve as placeholders for documents. [24]. In order to be retrieved from bunch of documents it must meet users query condition or it must have higher rank from other. For each type of query, there has to be a way to determine if a given query word matches a given phrase in a file, and the simplest way to do this is to allow actual matching.

The stemming procedure's primary purpose is to eliminate all potential affixes and thus minimize the phrase to its stem [25]. Search engines such as Google and Yahoo employ stemming to contest phrases with it suffixes and prefixes to the phrase stem, resulting in a search with numerous alternatives within the meaning that ensures the maximum diversity of relevant articles that appear in search results. Stemming also includes applications for gadget translation, file summarizing, categorization of text, glossary development, spelling checks, and text type. It is used in search engines for query expansion, indexing, and other natural language processing

concerns. The link between document and query in information retrieval is often determined by the variety and frequency of phrases that they share.

Unfortunately, phrases have several morphological variants that cannot be known by term-matching techniques deprived of some type of NLP. These versions can be viewed as equal by IR programs since they often have identical semantic meanings. Stemming is often measured (tested) by its impact on the query Because Stemming modifies the information received to answer the query it can change the quality of the answer set. Information retrieval experiments have been mixed, although selection Stemmer and stub can change the information matching the query, such that the average performance differs from reality, but is a small improvement [26].

But evaluating average efficiency might disguise major changes in individual searches, and the criteria used to analyze efficiency can make stemming results appear little even when the file sets retrieved have been drastically transformed. Rather than using changes in precision and recall as an indirect measure of accuracy, stemming algorithms can be examined by the correctness of the conflations they generate. This is the method used by [27].

The features of stemming algorithms can differ significantly depending on whether a stemming dictionary is utilized, which suffix is used, and what the stemmer is supposed to perform and The majority of algorithms are founded on ideas and approaches. This process involves deleting the longest single match or iterative deleting multiple suffix words. The reason behind the iterative Suffixes is eliminated from the stem in the order determined by derivational rules in the iterative technique. Stripping begins at the end of the word and works its way to the start in this approach. For instance, if the word 'wonderfulness' is reviewed, the suffix -ness will be deleted in the first iteration, and when 'wonderful' is reconsidered, the suffix -ful will be removed, leaving 'wonder' as the final stem. The class order Stripping rule will be defined by the programmer.

Other approaches, such as the iterative approach, are utilized by [5]. It operates through five steps, simulating the process of inflectional and derivational word formation by using five different classes of suffixes up to 60 in total. Certain rules don't remove suffixes but instead modify the ending stem to a different ending. For example, removing the suffixes "-ing" and "- ion" from "Transporting" produces "Transport" and "Transporta," each with distinct stems. However, employing the recoding method that converts "-rt" ends to "-rta," the two words are merged to the same stem, "Transport." [5]

Lovins stemmer uses longest-match principle. the problem using the longest matching method requires a much larger affix list This requires filling and processing list as well as storage costs. storage costs have fallen over the previous decade, so keeping a huge list is no longer an issue, but creating a list can be tough, particularly for languages with complicated vocabulary.

Stemming algorithms were created for a variety of languages, including English [5] [3] are the two most common English stemmers. most stemming algorithms comply with the concepts and strategies hired by the following two English stemmers, Porter and Lovis. Most algorithms' bases are shaped by algorithms that use suffix stripping.

2.2.1 Successor Variety

Hafer and Weiss [28] developed the successor variety, which bases stemming on the frequency of letter sequences within a body of text. As more characters are added, the number of substrings of a phrase will decrease till the segment's limitation is reached. The work in structural linguistics that tried to determine texts and morpheme boundaries based on the distribution of phonemes in a large body of text served as the foundation for successor range stemmers. The stemming technique derived from this work states words with letters rather than phonemes. After this process is finished, the number of substrings that can be formed from a word will decrease as more characters are added until a segment limitation is reached. The successor variety will thereafter significantly grow.

2.2.2 Statistical Approach

Statistical Approach or in another name N- gram approach employs statistical techniques via an inference process, and rules for word generation are developed based on a corpus. Several of the techniques used are N-gram [29] and Hidden Markov Models (HMM) [30]

Single N-gram stemming tries to avoid this difficulty. The idea is to look into how each N-gram is split over the entire file. Unique root words are going to appear significantly less frequently than variant portions (common prefixes and suffixes) because of the morphological invariants. Inverse document frequency (IDF), a common statistic, is used to find them. In a few European languages, the authors evaluated the algorithm successfully. By evaluating specific types of sentences, such as record titles, the n-gram stemming technique produced positive results, especially in topic areas like chemistry. examples include Adamson and Boreham [31] discovered that the n-gram approach

created measures of similarity and successfully grouped record titles from Chemical Titles. However, in order to cluster any statistical dictionary, the n-gram approach's implementation necessitates a tremendous amount of work.

HMMs, or Hidden Markov Models. It doesn't require any prior linguistic expertise or specially-made training data. Instead, it employs unsupervised training, that can be carried out at the time of indexing. HMMs are finite-state automata with probability-based transitions. instead, it employs unsupervised training that can be carried out at indexing time. HMMs are finite-state automata with probabilistic functions describing transitions. Every letter that makes up a sentence is regarded as a state. The authors formed groups from all potential states.

1. **Initial**: consider to be roots only
2. **Final:** consider to roots or suffixes

The splitting point, or transition from roots to suffixes, is produced by the word-building process, which involves transitions between stages. By employing Porter's technique as a baseline, HMM tended to over stem the words in its checks.

2.2.3 Dictionary-Based Technique

This approach displays every phrase that exists in a chosen language together with any variants that have been made by adding new words to the root phrase. To stem a word, the table is called upon to find a phrase that matches. If two instances of a phrase are found, the associated root form is returned. Then, using data lookup, words from inquiries and indexes might be stemmed. [32]. That dictionary is usually created by hand. Using a table lookup, words can be stemmed using this technique. The table's entries are arranged alphabetically. The search can be made more efficient by using a hash table or binary search list. despite the fact that such techniques are correct [32] .

The biggest issue with this approach is maintaining dictionary every time the language updated it dictionary. It also has its own advantage most of the time it produces good stems. but, the technique is limited to retrieving only those phrases which have previously been saved. However, the method is only capable of recovering phrases that have already been saved. As the corpus grows, more space is needed for storage, which could make the search method in effective.

This straight approach seems effective, yet it is insufficient to deal with the "unlimited" phrases and their construction., Notably in languages with better morphological structures that are

inflected. His evaluation of algorithmic stemmers was primarily motivated by this, which led him to draw the conclusion that "in spite of the faults they can be seen to make, they still deliver right functional outcomes". Not only will a dictionary designed to aid in stemming today likely need major upgrading in a few years, but a dictionary currently in use for this purpose may already be several years out of date. In Krovitz's experiments with dictionaries [3]

2.2.4 Affix Removal Algorithms

Rule-based algorithms are used for affix removal. The most popular method is called "affix removal," which removes prefixes and suffixes from a word to create a standard stem form. The longest match is one of the standards or rules that the affix removal method is built on. [33]. yet subject to the morphology of the intended tongue. Based on a set of principles or guidelines, an iterative largest match stemmer removes the longest string of characters from a phrase. This process is repeated until no additional characters can be eliminated. Even when characters were eliminated, stems might not be effectively combined.

A given word is stripped of its suffixes via an algorithm for removing affixes, leaving only the stem. Despite the algorithm's drawbacks. Owing to its great precision and recall, it is the one most widely acknowledged. Although Lovin's stemmer uses the same rule-based methodology as Porter's algorithm, it is more cautious because its rules are not applied iteratively. [4].

2.3 Stemming Algorithms for Local Languages

there are some researches done in Ethiopian languages for steaming text which will be reviewed and presented as follow.

2.3.1 Stemmer for Amharic

Amharic has a fairly rich morphology. Guragegna and Amharic are both Semitic languages with a comparable morphological system in which word inflection and derivation follow a comparable structure. reviewing Amharic stemmer will help to develop Guragegna stemmer.

Nega Alemayehu and Peter Willett [14] names will be always mentioned when we review Amharic stemmer because the build The he first Amharic stemming algorithm, The aim was to investigated the effect of stemming in information retrieval for Amharic language. This stemmer has been developed according to stemmer of Slovene [34]. Which they change the stemmer of

Slovene to be used for Amharic language. The algorithm first finds a group of stop-words, followed by a set of affixes associated to the final content-bearing words. They exploited the features of the generated affixes to steer the stemmer's progress. The stemmer removes affixes using iterative approaches that specify a minimum stem length, recoding, and context-sensitive rules, with prefixes deleted before suffixes. After acquiring the stem of the word, the root is obtained by removing all leftover vowels.

Alemu and Lars Asker [35]developed the other Amharic stemmer. The stemmer will look for all potential divisions of a given word that are coherent with the morphology rules of the chosen language, followed by the prefix and suffix of the word will be chosen based on corpus statics. After eliminating the prefix and suffix from the term, it will seek up the getting word in the dictionary to validate the stem word. When its tested on a limited text of 50 sentences, the stemmer obtained an accuracy of 85%. 805words. A statistical examination of a 3.5-million-word Amharic news corpus was used to determine the distribution and frequency of prefixes and suffixes over Amharic nouns. [36]

2.3.2 Oromo Stemmers

There has been many Oromo stemmers but M. Wakshum [16] is the first to develop stemmers for Afaan Oromo language. The stemmer uses a suffix table in conjunction with rules to eliminate suffixes from provided words by browsing up the longest match suffix from the suffix list of 342 suffixes that are automatically built by counting and ranking the most common endings. Other linguistically appropriate suffixes have been manually added. It applies the longest-match, context-sensitive technique, and rules that eliminate prefixes and suffixes, yet when tested on 1061 words, the stemmer achieves an accuracy of 92.52%. The evaluation was carried out by counting stemming mistakes and reducing dictionary size.

The second Afaan Oromo stemmer was created by [37]. In order to fix some problems which are encountered by previous Wakshum M. the new developed stemmer included some features which are not included in Wakshum. The idea is derived from Porter stemmers, who deal with measuring, organizing rules into clusters, and assessing word creation depending on the nature of their ends. The rules only apply under particular situations, such as the final stem must have a specified minimum length, and the stemmer was tested on a collection of 5000 words, with the researcher claiming that out of 2458 words, 90 (3.66%) were under stemmed and 15 (0.61%) were over

stemmed. The stemmer's accuracy was 95.73%, while it produced 105 words (4.27%) of stemming error.

2.3.3 Stemmer for Silt'e

M. Kedir in 2012 [38] developed the only stemmer for slite language. According to the research, he used iterative, context-sensitive, and recoding methods to eliminate prefix, suffix, and reduplication of letters. In the study article, the iterative but longest match stemmer is used first, and the lists of affixes are tested against the word. Following that, it will remove affixes repeatedly until all affixes have been eliminated. There are three steps for removing affixes. The first phase eliminates prefixes, the second removes suffixes, and the last step removes reduplication of letters from the word. The stemmer was evaluated on a data set of 1486 words and demonstrated a precision of 85.71% while reducing dictionary size by 34.99%.

2.3.4 Stemmer for Wolaytta

The initial Wolaytta language stemmer was created by L. Lessa [18]. The Wolaytta language depends on suffixation to create diverse word forms. This iterative, rule-based stemmer is context-sensitive. The algorithm used an iterative development process to create the stemmer.

Lemma, the researcher, compiled the list of potential suffixes using a semi-automatic method. The stemmer first receives a word to be stemmed, and then it determines if a suffix from the list has been added to the word. The final word is then taken into consideration as a stem once the suffixes have been iteratively removed from the word and the relevant conditions have been applied. In a sample of 884 words, the stemmer's performance was tested and found to be 86.9% effective.

2.3.5 Stemmer for Kambaata

Jonathan Samuel Sumamo [19] has created a stemmer for Kambaata text that uses the language's terms' longest-match, context-sensitive stemming algorithm. Kambaata is a purely suffixing language with a complex morphology that primarily uses suffixation to convey word content.

The context-sensitive, single-pass, longest-matching stemming algorithm was created by adopting the rule-based stemming methodology, and the error counting method was used to assess the stemmer's efficacy. 138 words (5.69%) over stemmed and 16 words (0.66%) under stemmed, but the errors of over stemming and under stemming were reduced to 2.60% (63 words) and 0.54%

(13 words), the researcher claims. The stemmer was evaluated using test sets of 1385 and 1040 distinct words. respectively performance of the stemmer has been upgraded to 96.87%

2.3.6 Stemmer for Tigrigna

G. Berhe [15]has developed the first Tigrigna stemmer. The Porter approaches for defining the rules are used by the Tigrigna stemmer, which employs an iterative approach but eliminates the longest affix when two affixes fit the word. Five stage guidelines were utilized by the stemmer. The first phase is removing double letter repetition by using the word to be stemmed as an input. Prefix-suffix pairs are removed in the second stage. Prefixes are removed in the third phase, which also takes the results of prefix-suffix stripping. When deleting a prefix, the length of the string after the prefix has been removed is measured and the prefix list is checked for matches. By accepting the result from the previous stem and determining if the word has any matches from the list of suffixes, the fourth step eliminates suffixes. The suffix is deleted from the word if there is a match and the remaining string is longer than the required length. The method stops redundancy of a single letter in the last stage.

Y. Fisseha in 2011 [32] developed the other Tigrigna stemmer was. Y. Fiseha has proposed the algorithm to reduce the gap in the previously mentioned G. Berhe algorithm and to produce a more potent Tigrigna stemmer. Based on G. Berhe's earlier study findings, the researcher developed a technique that, in order to increase the efficacy of the stemmer, eliminates the affixes of the Tigrigna words in a comparably bigger corpus by utilizing particular grammatical rules. However, this technique can only handle prefixes and suffixes. In accordance with the report, reduplication, compounding, and irregular words were not addressed in this study. The algorithm correctly stems the words 86.1% of the time.

2.4 Stemming Algorithms for Foreign Languages

2.4.1 English Language Stemmers

For the English language, several stemming algorithms have been created. These methods range in complexity from the straightforward database lookup to the complex iterative longest matching.

2.4.1.1 Lovins Stemming Algorithm

Julie Beth Lovins created The Lovins Stemmer, a one pass, context-sensitive, longest-match stemmer [3] . A word being stemmed has an ending with a satisfying condition located and deleted using a lookup on a table containing 294 endings, 29 conditions, and 35 conversion rules. This stage mostly focuses on handling doubled consonants and irregular plurals. This stemmer is attempting to handle both IR and linguistics, but he is unsuccessful in both.

The method does not produce acceptable linguistic results since some suffixes cannot be stemmed because they are not included in the rule list because it is not complicated enough. The transformation of words is fraught with issues. The stems are reformed into phrases using the receding parameters in this technique to ensure that they match the stems of other related meaning phrases. The fundamental issue with this method is that it is being shown to be largely inaccurate and frequently fails to synthesize phrases from the stems or pair the stems of words with similar meanings. It is also unsatisfactory from an IR perspective since its lengthy rule set and fading stage slow down its execution. Due to the nature of the single pass technique used by this Stemmer, it can only remove one suffix from a word at most. Using a list of roughly 250 possible suffixes, it eliminates the longest suffix from the word till the stem, which is at least three characters long after the suffix has been removed.

2.4.1.2 Dawson Stemming Algorithm

With a list of more than a thousand English suffixes, this stemmer expands on the Lovins stemmer's methodology. It employs longest match approval iteratively. This stemmer makes use of a list with 260 English suffixes with corresponding elimination condition codes that Lovins generated following [25] 's revision. the 1200 suffixes are updated. It read the suffixes and the state of the code numbers backward to prevent storage and processing time issues. To handle stems, Dawson extended the partial matching method.

The key concept of Dawson's algorithm is that two stems are of identical form if they match up to a particular character threshold and the remaining characters for each stem fall under the same stem ending class.

2.4.1.3 Porter Stemming Algorithm

Martin Porter [5] The Porter Stemmer, a conflation Stemmer, was created at Cambridge University. The Stemmer is based on the concept that most of the about 1200 suffixes in the English language are composed of a collection of smaller and simpler suffixes. It consists of five phases, with rules being applied at each step. If a suffix rule matches a word inside a step, the conditions related to that rule are then evaluated on the stem that would arise if the suffix were deleted according to the rule's definition. A rule fires, the suffix is deleted, and control shifts to the following step after it has met all of its requirements and been approved. When a rule from one step fires and control transfers to the next, or when there are no more rules in that step, manages moves to the next step, if the rule is rejected. If the rule is rejected, the following rule in the step is then tested. The resulting stem is returned by the stemmer once control has been transferred from step five in this procedure, which lasts for all five phases.

There are several condition criteria that make up the Porter algorithm. There are three categories of circumstances: rules-related conditions, suffix-related conditions, and stem-related conditions. Porter's method is efficient in terms of storage and processing time since it employs a vocabulary of roughly 60 suffixes and only includes a few context-sensitive and recording rules.

2.4.1.4 Krovetz Stemming Algorithm

The Krovetz Stemmer was developed by [7]. The stemming algorithm changes the past tense from past to present and the plural to a single form. Frequently, this is combined with another algorithm. nevertheless, cannot handle massive papers on my own. shown to not deliver accurate findings in a consistent manner. The elimination of "-ing," the change from the plural to the singular form (e.g., "-ies," "-es," or "-s," respectively), and the change from the past tense to the present tense. The method of converting first eliminates the suffix, and then after checking a dictionary for any recoding, it restores the word's stem.

2.4.2 Arabic Stemming Algorithms

Because Arabic is a highly inflected language and has a complicated morphology, it is particularly challenging to design natural language processing tools for Arabic information retrieval. There are a number of stemming techniques for the Arabic language that are also quite effective, both derivationally and inflectionally. In comparison to English text samples of equivalent size, Arabic text includes more terms that appear only once and more different words, and the spelling of Arabic

also contributes to variability that can be confusing to information retrieval systems. Problems don't arise from the letters' right to left ordering or from how they appear in different contexts.

The light and root-based stemmers are the techniques that are most frequently utilized in Arabic stemming. The goal of root-based stemming is to identify the Arabic top word's root by eliminating all prefixes and suffixes that are currently attached. For Arabic, a number of morphological analyzers have been created., e.g. [39][47]. In order to eliminate the most common suffixes and prefixes from an Arabic word form, various light stemmers have been created. These light stemmers are all based on suffix and prefix removal and normalization. Illustrations of light stemmers: Aljlayl & Frieder's Stemmer Darwish's Al-Stem, and Larkey et al.'s U Mass Stemmer [40]. After modest normalization, the light stemmers had several stop word lists with Arabic pronouns, particles, and the like eliminated. Depending on the stemmer tests, it was determined that the light stemmer performed better than the root-based method because it eliminates sense ambiguity by classifying words with similar semantic properties.

2.5 Evaluation Methods for Stemmers

The accuracy of stemmers will be evaluated using different methods. The manual approach, vocabulary reducing, and Paice's method are the three most well-known techniques. In the manual technique, the review process is carried out by a person who chooses the appropriate stem for each word. We employ three assessment metrics: the total number of mistakes due to excess stemming, the number of errors due to under stemming, and the number of accurate outcomes. The goal of stemmers is to reduce the amount of the vocabulary for indexing reasons, and any repeats will be deleted or omitted from the vocabulary reduction acquired by reducing the number of words in the corpus by the number of stems created.

Paice's approach Stemmers are rated based on errors that happened during the stemming procedure or during error counting. How well a stemmer performs outside of the context of retrieval is determined by measurements of under-stemming and over-stemming., In order to compare various stemmers qualitatively, three evaluations are made: the over-stemming index (OI), the under-stemming index (UI), and the stemming weight (SW). This method involves phrase sampling rather than repetition, with the words being 28 semantically and morphologically linked thematic categories.

The OI/UI ratio provides the SW. [27] An ideal stemmer would stem all the words in a set to the same stem. Paice has analyzed many English stemming algorithms apart from the context of an IR system, and he did not employ conventional precision and recall settings. A stemmed cluster has under stemming error if it has more than one distinct stem. This correlates to a detrimental impact on recall in an IR system. If a stem from one stemmed group appears in additional stemmed groups, the stemmer made over stemming mistakes, which decreases accuracy. Therefore, a competent stemmer should make the fewest under- and over-stemming errors feasible.

2.6 Related Work

As I have reviewd related works in above chapter which are local stemming algorithms developed for Ethiopian languages like Amharic, Afaan Oromo, Kembatta, Tigrigna, Silt'e, and Wolaytta I have clarified what are out come researcher and their problem with possible future work.we can look up 2.3 Stemming Algorithms for Local Languages

The literature review led the researcher to the conclusion that many local languages have unique stemming algorithms. One of the languages in which stemming studies have been conducted is Amharic. Other languages include Afaan Oromo, Kembatta, Tigrigna, Silt'e, and Wolaytta. But there is no research has been done for developing a stemming algorithm for texts in the Gurage language. Because of this, the researcher used the chance to do research on creating a stemming algorithm for Gurage words.

Chapter 3

Morphology of Gurage Language

3.1 Overview of The Gurage Language

Gurage is the name of a region in the Southern Nation, Nationalities, and People's Regional State (SNNPRS), known as the Gurage Zone. Wolkite, the seat of the Gurage Zone, approximately 150 kilometers to the southwest of Ethiopia's capital Addis Ababa. Gurage also describes the residents of the Gurage Zone. Guragina is the name given to Gurage by its local speakers. Chaha, Ezha, Endegegn, Ener, Inor, Gumer, Gura, Geyto, Meskan, Muhir (with sub-varieties "di-bet and an-bet"), Dobi also known as Gogot, and Kistane also known as Sodo or Aymelel are the thirteen dialect clusters that make up the language. [13].

Gurage is a collection of three mutually incomprehensible groups of Southern Ethiopian Semitic languages: Northern Guragc, Eastern Guraqe, and Western Guragc.Mesqan, Enncr, Endcqaqn, Enncmor, Ceyto, Chaha, Ezha, Gumer, and Gura are members of the western Gurage group. Together with a few other minor dialects, these four, which are from Central Eastern Gurage, make up the "Sabat bet" (Seven House) languages. [12].

The majority of the Cheha ethnic community, which is a component of Sebat Bet Gurage, resides in Chaha worda. The Wored has an overall population of 115,918 people, 106,933 of whom are in rural areas. Emdibir serves as the werda's capital city, and its area stretches between the Magiacha and Wengia rivers in the north and south, respectively. Its northern and eastern borders are Eza, while its southern and western borders are Geto and Ennemor.

3.2 The Writing System of Gurage Language

Gurage uses Ethiopian national writing system Ge'ez, The Ge'ez script is used to write all Guragigna languages and dialects. This script's Gurage subgroup has 44 unique glyphs.. This Ge’ez script is mainly used for Amharic language, Geez and Tigrigna language.

3.2.1 Vowels and Consonants of Gurage Language

Gurage Chaha's grammatical structure is closely connected to that of Amharic and other Ethio-Semetic languages, as stated in Leslau [11] and Ford [41] Gurage contains distinct morphophonemic characteristics, seven vowels, and thirty-seven consonants.

3.2.2 Consonants

The consonant sounds [s] and [p] are used for loan words from Amharic. For example, one can find these phonemes in the loan words like *məs'haf* „ {መጽሀፍ}book ‟ *s'ələt*„ {ጸሎት}prayer‟ *p'ent'ə {ፈንጠ}*„protestant‟ and *p'ap'as* „{ጳጳስ}bishop [42].

Loan words from Amharic employ the consonant sounds [s] and [p]. These phonemes, for instance, may be found in loanwords like ms'haf {መጽሀፍ} "book," s'lot "prayer," {ጸሎት} p'ent' "protestant," *{ፈንጠ}*and {ጳጳስ} p'ap'as "bishop". [42].

Table 3. 1: table shows the consonants of gurage

		Labial		**Dental**	**Post-alveolar**	**Palatal**	**Velar**		**Glottal**
		plain	**round**				**plain**	**round**	
Nasal		m	m^w	n					
Plosive/ Affricate	**voiced**	b	b^w	d	d͡ʒ ⟨ǧ⟩	ɟ ⟨g^y⟩	g	g^w	
	voiceless	p	p^w	t	t͡ʃ ⟨č⟩	c ⟨k^y⟩	k	k^w	
	ejective			t’ ⟨ṭ⟩	t͡ʃ’ ⟨č̣⟩	c’ ⟨ḳ^y⟩	k’ ⟨ḳ⟩	k’^{w} ⟨ḳ^w⟩	
Fricative	**voiced**			z	ʒ ⟨ž⟩				
	voiceless	f	f^w	s	ʃ ⟨š⟩	ç ⟨x^y⟩	x	x^w	h
Approximant		β		l		j ⟨y⟩		w	
Rhotic				r					

Table 3. 2: table shows the consonants words of gurage

ኽ	ኾ	ለ	መ	ረ	ሰ	ሸ	ቀ	ቐ	በ	ቨ	ተ	ቸ	ነ	ኘ	ኧ	ከ	ኸ	ወ	ዘ	ዠ	የ	ደ	ጀ	ገ	ጘ	ጠ	ጨ	ጰ	ፀ	ፈ	ፐ
x	x y	l	m	r	s	š	ḳ	ḳ y	b	β	t	č	n	ñ	ʾ	k	k y	w	z	ž	y	d	ğ	g	g y	ṭ	č̣	p̣	ṣ	f	p

3.2.3 Vowels

As I mentioned above Gurage has seven vowels and each Based on the location of the tongue, vowels are divided into three groups: front vowels 'i, e', central vowels ',ɨ, a', and rear vowels 'u, o'. When the center vowel 'ə' follows the voiceless velar fricatives 'x', an allophone 'ɛ' is produced. Bahire Araya [42]

Table 3. 3: table shows the Vowel of gurage

	Front	Central	Back
High	i	ɨ̶	u
Mid	e	ə	o
Low		a	

3.3 Morphology

The study of morphology explains how words are generated in a particular language. and minimal linguistic unit of a language is called morpheme but it must have meaning. There are two types of morphology. The initial is When word stems are coupled with grammatical indicators for things like person, gender, and number, the process is known as inflectional morphology. In this instance, the parts of speech won't alter. However, Derivational Morphology is concerned with modifications that will alter the components of speech. For instance, a verb can be used to create a noun or an adjective. [43].

Free morphemes and bound morphemes are the two different forms of morphemes. When compared to bound morphemes, which cannot occur on their own as words, free morphemes may stand alone as words and don't require any assistance to have meaning with other words. for example free morphemes of cheha Gurage "ቸነ"[መጣ]፣{məṭə}"came", "መካ"[ቸገረ]"problem" this word have meaning on there own. But when we came to bound morpheme they can't stand by themselves. For instance, when we add "ች" on suffix of "ቸነ" it will change noun, gender so its bound morpheme because of it will not have meaning with out the added suffix [9].

3.3.1 Word Formation in Gurage

The Gurage language uses a variety of word-formation techniques. which are changing vowel patterns, compounding and Affixing which means that adding prefixes suffix, suffix and infixes to words it may be at the beginning or end.

Sometimes we may add suffix and prefix at time but its not wildly seen. So the common ways of word formation is adding of suffix, prefix and infix to phoneme. For instance "ኤማ",[መንገድ], road and when suffix 'ነ' is added it became "ኤመነ",[መንገደኛ], passenger. "አዅ"፣[አየ] watch and and when suffix 'ቲ' is added it became "ቲአዅ", [አያየ], watching [43]. Therefore, one guragegna word can give very large number of variants because of complex morphological structure. Compounding is the one way of word formation, when two or more words are compounded ant they must have different meaning from the first. For instance, "ሜናቶት" menatot [worker or employee] but when we dived the word in to two it will give us "ሜና","mena" [job] and "ቶት","tote",[do].

3.4 Derivational and Inflectional Morphology of Gurage

Inflectional affixes are created by joining word stems with grammatical markers for things like person, gender, number, tense, and case. In gurage language we have five parts speech which: propositions, adjectives, nouns, verbs, and adverbs. Conjunctions and prepositions are unhelpful for IR or other natural language processing objectives therefore the focus of the the study of derivational and inflectional morphology.

3.4.1 Inflectional And Derivational Verbs

Verbs make up an important portion of the dictionary and display various morphosyntactic characteristics depending on how the consonant-vowel sequence is arranged. The perfect tense

third person masculine singular is the most basic form of the verb [43]. The majority of words with triliteral roots are listed with the verb in the third person masculine singular. Each of the three conjugation classes A, B, and C belongs to a root with three consonants.

A class. These do not germinate in the citation form (perfect), but the vowel "e" is present between both sets of consonants. For instance, "አዥ",aeze [አየ], 'he watch', "መሰ",mese,[ነቀለ]'he took off' and The gemination of the second consonant in all forms sets the B class apart. Example "ምረቐ"',mereqe[መለመለ], 'Recruited' ," ሰከከ" Sekeke,[ሰካ], 'Plug in '. The small number of members who belong to the C class pronounce the vowel a between the first and second consonants. For instance, 'ባነረ' banere,[አፈረሰ], demolish 'ታከረ' takere,[አቀና] he fixes [42].

3.4.1.1 Aspect of Verbs

In Gurage language morphology verbs have four types of stems which are perfective stem, imperfective stem, Imperative stems and the last one is Jussive stems.

Perfective stems

Most of time The perfective stem is used to form simple past, present perfect and past perfect tenses

For Example, 'ታገደ'[tagede] "he is prisoned", when we changed to Present perfect '

ታገደም' [tagedem] "he has prisoned".

Imperfective stem

The imperfective stem will be used for the formation of the present/future and past-imperfective tenses. They described by the insertion of the vowel /ə/ between the first and penultimate consonant radical roots canonically

Example: 'ይታገደ' [yetaged] "he will prisoned".

Jussive stems

Depending on the characteristic of the root morpheme in the canonical triliteral roots, stems are created by inserting either the vowel /ə/ or /ɨ/.

Example: 'የምበር', [yember] "may live"

‘የስርፍ’ yesrəf ‘may fear‟

Imperative stems

The subject marking affixes are what set apart the jussive stem template slots. It is clear that the imperative forms of verbs are used to communicate orders and directions in the second person singular and plural of either gender.

Example: ‘ብራ’, [bera] “you eat‟

ኔኽ’ yesrəf ‘come on‟

3.4.2 Verb Inflection

The sentence elements other than the subject can potentially be marked in the verb's morphology, as is the case in other Ethio-Semitic languages, as shown by the benefactive (ben), detrimental/instrumental (detr), and complement person suffix (cps) categories above.. The benefactive and detrimental/instrumental markers are not necessary for the complement person suffixes to exist, although these last two types do.

3.4.2.1 Perfective

The perfect nature of verbs is used in Gurage verbs to generate the past and complete acts. These verbs serve as the foundation for other forms that could inflect their base. The fundamental form of the perfect tense, that has a stem and a subject marking, usually indicates the past tense. In the perfect tense of a verb, a subject marker and a pronoun suffix may be present.. The name, gender, and subject number are all indicated by a subject marker. The person, gender, and subject number all contribute to determining the shape of a subject marker.. The suffixes attached are ም/m /, ሽ/Sh /, ሮ/ro/, ኒ/ni/, ሁ/hu/, /. With all sorts of verbs, subject-marker and pronoun indicator suffixes are frequently employed without any change. [43].

Table 3. 4: Inflections perfect tense

Verb Variations	**Person**			**Gender**		**Number**	
	1	2	3	**M**	**F**	**Singular**	**Plural**
ሴተረ	+			+	+	+	
ሴተረም		+		+		+	
ሴተረሽ		+			+	+	
ይሴተረ		+		+		+	
ትሴተረሽ		+			+	+	
ሴተሮ፣			+	+	+		+
ሴተሮሽ		+			+		+
ሴተረሁ		+		+	+		+
ሴተረኒ			+	+			+

3.4.2.2 Object agreement markers

The non-subject argument that is the object is stated via bound affixes. These morphemes can be observed joining verbs that are preceded by applicative objects. The object and object applicative affixes are related to two fundamental allomorphs, "light" and "heavy." These object morphemes represent the person, number, and gender.

The voiced velar (-ኣክ-) has an allomorph in the second person object marking (-ኣሀ-). Therefore, using the second person plural or the heavy set the voiceless fricative changes to a voiced velar sound. the third person, of the voiced palatal approximant -የ has allomorphs in the alveolar voiced nasal. But, unlike the first person singular, where the light and heavy allomorphic sets are indicated by the morphemes -ኣ/ይ and -ነ, respectively, there is no conditional change in the first-person plural [43].

The syntactic roles of direct object markers can be specified by possessive pronouns that can agree with the pronominal affixes allowed on verbs..

Table 3. 5:Object agreement markers gurage

Person	**Light**	**Heavy**
1PS	-ኤ	-ን
1PL	-አንዳ	-አንዳ
2MS	-ኣሀ	-ክ
2FS	-ኣሂ	-ኪ
2MPL	-ኣሁ	-ኣኩ
2FPL	-ኣሂማ	-ኣኩማ
3MS	-ነ	-ዊ
3FS	-ና	-ያ
3MPL	-ኖ	-ዮ
3FPL	-ነማ	-jəma የማ

3.4.2.3 Imperfective

In Guragegna, the imperfect verb is used to describe non-past behavior. There are prefixes and/or suffixes. As is typical for Semitic languages, several of the person identities used with imperfective stems combine prefixes and suffixes..

Table 3. 6: the imperfective stems non-past of gurage

present/future	‘Gloss’
እየት	‘I work
ትየት	‘You (m) work
ትየች	‘You (f) work

ይቶት	‘He works
ትቶት	‘She works
ንቶችነ	‘We work
ይቶቹ	‘You (pl) work
ትቶትሁ	‘They (pl) work

The above Table 3. 7 shows how suffixes and prefixes are added for the root verb “ቶት”

Table 3. 8:the present/future form of gurage

Person	Singular	Plural
1st person	እቶት (`I-chot)	ንቶት (n- chot)
2nd person-masculine	ትቶት (t-chot)	ትቶቱ (t- chot -u)
2nd person- feminine	ትቶች (t-chot-i)	ትቶች (t- chot)
3rd person-masculine	ይቶት (y-gebr)	ትቶች (y- chot -u)
3rd person-feminine	ትቶች (t-gebr)	ትቶች (y- chot -a)

Table 3. 9: the Person suffixes form of gurage

Person	**suffixes**	**Example**	**Gloss**
1PS	ኣ	ኣዶሩ ኣ ኣዶታ	„my mother“
1PL	-ንዳ	ኣዶሩ ንዳ ኣዶተንዳ	„our mother“
2PMS	-ሓ	ኣዶሩ ሓ ኣዶተሓ	„your mother“

2PMPL	-ሑ	ኣዶተ ኣሑ ኣዶታሑ	„your mother“
2FS	-ሒ	ኣዶተ ሒ ኣዶታሒ	„your mother“
2FPL	-ኣሒማ	ኣዶተ ኣሒማ ኣዶታሒማ	„your mother“
3MS	- ኦ	ኣዶተ ኦ ኣዶተኦ	„his mother“
3MPL	- ሖኖ	ኣዶተ ሖኖ ኣዶተሖኖ	„their mother“
3FS	-ሖኣ	ኣዶተ ሖኣ ኣዶተሖኣ	„her mother“
3FPL	- ሖነማ	ኣዶተ ሖነማ ኣዶተሖነማ	„their mother“

3.4.2.4 agreement markers

Additionally, subject agreement indications are present in imperfective verb tenses. Imperfective agreement markers are prefixed to base morphemes in a manner comparable to subject agreement markers in the perfective tense (see Table 3.9), and in certain situations, they are also suffixed onto verbs. Consider the table below.

Table 3. 10 markers of Subject agreement in imperfective

Person	**Subject marker**
1PS	ኧ-

1PL	ኧ-ን
2MS	ተ-ኣ
2FS	ተ-ኣ
2MPL	ተ- ኡ
2FPM	ተ-ኣም
3FS	ቲ
3MS	ዮ-ኣ
3MPL	የ-ኣም
3FPL	ዮ-ኣም

The imperfective verb conjugations are morphologically connected to the imperfective subject agreement indicators. These affixes subject agreement morphemes are always agreed to by the sentences' optional objects in terms of number, gender, and person. Think about the information below [44]

Table 3. 11: the Person suffixes markers in perfective

Singular	**Examples**	**Plural**	**Examples**
1PS	ረከሰ	1PL	ኧ--ረከሰ-ን
	ረከሰ		ረከሰን
	I bite/am biting		We bite/ are biting
2MS	ተረከሰ	2MPL	ተ-ረከሰ-ኡ
	ተረከሰ		ተረከሱ
	You bite or are biting		You bite or are biting

2FS	ተረከሰ-ኢ	2FPL	ቲ-ረከሰ-ም
	ተረከሲ		ቲረከሰም
	„You bite/are biting“		„You bite/are biting“
3FS	ተ-ረከሰ	3FPL	ዮ-ረከሰ-ኦም
	ተረከሰ		ዮረከሰም
	„She bites/is biting“		„They bite/are biting“
3MS	ዮ-ረከሰ	3MPL	የ -ረከሰ-ኦ
	የረከሰ		የረከሶ
	„He bites/is biting“		„They bite/ are biting “

3.4.2.5 Negative

Main verbs that are negatively imperfective are simply distinguished by the inclusion of a morpheme. The added negative morpheme has no effect on the imperfective verb's accentual structure.

For example. ቶት means ‘I work’ but when we add ‘አት’

አት+ቶት አትቶት It give us don’t work

‘ሴተረ‘ means hide when we add ‘አት’

‘አት + ሴተረ’ አትሴተረ It give us don’t hide

Table 3.12: negatively imperfective markers of gurage

Verb Types	Affirmative	Negative
Type A	ሰበረ He broke.	ኣንሰበረ He did not break.

	ከተፈ „He chopped.“	ኣንከተፈ „He did not chop.
Type B	ሸገረ „He changed.“	ኣንሸገረ *„He did not change.“*
Type C	ባነረ „He destroyed.“	ኣንባነረ „He did not destroy.“
	ቃቀሰ „He beckoned.“	ኣንቃቀሰ „He did not beckon.“

3.4.2.6 Imperative And Jussive

The imperative simply utilizes suffixes; the jussive forms combine prefixes and suffixes as well. The suffixes used with the imperfective stem are the same for both imperative and jussive.

For Example,

ንሻደ [neshade] let me divide ሻደ [shade] divide

ትሻደ [neshade] she divide ይሻደ [shade] let him divide

ንሻዶ [neshad0] let them divide ንሻደም [neshad0] let us divide

3.4.3 Derivation of Verbs

Because the semantic qualities frequently overlap and are erratic, dealing with derived stems seems better from a morphological perspective. The following derivational procedures have been identified. [43] Prefixation of አት thematic change from አ to ኤ . It carries a causative connotation. Other than አ , thematic vowels are prolonged. The prefix is exclusively added to verbs with lengthy thematic vowels.

Examples:

በተረ |betere| ቀደመ front አትበተረ |betere|, ቀደመ cause to be front

በከረ |bekere| ተቸገረ Troubled አትበከረ |bekere|, ተቸገረ cause to be Troubled

ተራሐ |terahe| ተላከ sent አትራ |terahe| ተላከ Cause to be sent

ተሟተ |tmuate| ታገለ Struggle አትሟተ |tmuate| ታገለ Cause to Struggle

ተሳረ |tesare| ጠየቀ ask አትሳረ |tesare| ጠየቀ Cause to Ask

3.5 Morphological Reduplication

produces verbal stems. The penultimate root morphemes are repeated to indicate the verbal semantic range.

Reduplication: There are three different types of reduplication in gurage. These include Total, final, and frequentative reduplication [43].

3.5.1 Frequentative

Transitive triradicals and vocoid-second quadrilaterals replicate their penultimate radical to create the frequentative. As its shown below in 3,12 table Reduplication involves manipulating labialized or palatalized consonants as a single unit

Table 3. 13: the Reduplicated Frequentative stems form of gurage

Perfective stems	**Gloss**	**Reduplicated stems**	**Gloss**
ቅጥር	„ kill“	ቅጥጥር	„kill again and again“
መጠረ	„separate“	መጠጠረ	„separate again and again
ቃጠረ	„tie“	ቂቃጠረ	„tie again and again “

3.5.2 final reduplication

final reduplication occurs when copying is done edge-in, as opposed to when copying is done edge-only and certain bases reduplicate their final radical.

Table 3. 14: The final Reduplicated stems form of gurage

stems		final Reduplicated	
ለካ	Measure	ለካካ	Measure Well
ቆና	Roast	ቆናና	Roast Well
ቸጠ	Tire	ቸጣጣ	Tire Well

3.5.3 Total reduplication

In total reduplication the singular stem will be remove again in order to create the point who is the presented.

Table 3. 15: Total reduplication of gurage words

Singular		Plural	
ገፈ	long/tall	ገፈገፈ	„long ones“
ኦር	short	ኦርኦር	„short ones“
ብሻ	red	ብሻብሻ	„red ones“

3.6 Noun’s Inflection

A noun is in the nominative case when it is the subject of a verb, the accusative case when it is the object of a preposition, and the genitive case when it is the object of a preposition. A Guragegna noun's form is determined by the gender, number, and grammatical case of the word. Affixes are used to create nouns from other word classes, mainly verbs. They can also be created by joining two or more basic nouns to form a larger semantic phrase. Furthermore, formative vowels are inserted into the C-slot in nouns, which are derived from other word classes via root and pattern. Complex nouns have a complicated morphological derivation or word development [43]. Degif Petros Banksira In Gurage, nouns are morphologically inflected for case, number, gender, and definiteness. With reference to the examples given below, each of these concepts is discussed.

3.6.1 Number

The bulk of words having plural numbers is kinship terms and names of domesticated animals. Additionally, there is suppressive plural, which is commonly referred to as fragmented plural and is produced by separate morphological processes. However, the majority of the time, the plural numbers are hinted at in the verbs and also expressed through the third-person pronoun

The plural form of a number is commonly represented by a cardinal number. In contrast to their counterexamples, which are single nouns that are added to all countable nouns to indicate the plural notion, which may be regarded as "noun plus they" as in the data below, third person masculine(ሁኖ) plural is used to imply the plural countable nouns.

Example a. እምር እምርሁኖ

“ድንጋይ” “ድንጋዬች”

Imer Imerxno

stone stones

b. እማር እማርሁኖ’

Imar imar

‘አህያ imar

Donkey donkeys

3.6.2 Gender

Grammar is not the main factor determining gender; unpredictability is. As a result, gender is determined by sex for both people and animals. Feminine and masculine genders are indicated by internal modification, and these nouns mostly refer to kinship words and domesticated animals

Because they lack inherent gender indicators, Gurage also usually assigns gender to inanimate words [43] Since all inanimate nouns are often indicated as male depending on the socio-cultural milieu of the language variety, inanimate nouns are normally given their gender based on vocal conjugations. Think about the information below:

Table 3. 16: the Gender form of gurage

Masculine		**Feminine**	
'ማሸ"	„husband“	*ማሸት*	„wife“
ኣርቾ	„boy“	*ገረዱ*	„girl“
ኣባ	„father“	*ኣዶት*	„moth

3.7 Noun's Derivation

There are main and secondary gueragegna nouns. They are derived if they share root consonants with verbs, adjectives, or other nouns, or if their meaning is comparable. Otherwise, they are crucial. Nouns are created through affixation and intercalation from other nouns, adjectives, roots, stems, and the verb's infinitive form. A morpheme

3.7.1 Abstract nouns

In this morphological process, the condition of concrete nouns is changed into the equivalent state of abstraction concepts [44]. In Gurage, adding the limit suffix - -ነት to nouns or adjectives results in semantically new and unitary abstract nouns. The examples below illustrate the morphological operation.

Table 3. 17:Abstract nouns of gurage

Concrete noun		**Abstract noun**	
ፈንገየ	theif	*ፈንገዮት*	theft
ትከ	child	*ትኩት*	childhood
ሰብ	man	*ሰውት*	personality

Vowel deletion takes place when the abstract noun marker is attached to nouns with vowel finals. Therefore, as one can determine from the data given above, the epenthetic vowel (ɨ) is added in lieu of the other vowels if it attempts to generate more than two sequences of consonants.

Adjectives also get the suffix morpheme (--ነት), which serves the same morphological purpose. As an outcome, adjective-class words are changed into abstract nouns to produce abstract nouns [43]

3.7.2 Gerundive nouns

The (-ኣት) suffix is used to join gerundive or infinitive verbal stems to jussive verbal stems in Gurage. The jussive verbal stems receive the bound morpheme (-ኣት) as a suffix, which results in gerundive nouns

Table 3. 18: the Gerundive nouns form of gurage

Jussive stems		**Infinitve/gerundive**	
ጠንኪር	be strong	ጠንኪር-ኣት ጠንኪሮት	„to be strong/being strong ‟
ሲብር	break	ሲብር-ኣት ሲበሮት	„to break/breaking‟
ብራ	eat	ብራ+ኣት ብሮት	„to eat/eating

3.7.3 Nouns of Group identity

Nouns that designate the individuals who make up a certain grouping are derived morphologically. In Gura, nouns that function as group identifiers are formed using the bound morpheme (-ኣነ); when vowel ending words are connected to this noun former morpheme, the initial vowel must be eliminated. When this morpheme is linearly added as a suffix to simple nouns, the complex nouns are derived.

Table 3. 19: the Group identity nouns form of gurage

Noun		**Group identity**	
ኤማ	road	ኤማ-(-አነ) > ኤማነ	pedestrian
ተስፋ	hope	ተስፋ-(-አነ) > ተስፈነ	optimist
fərəz ፈረዝ	„horse“	ፈረዝ-(-አነ)> ፈረዝነ	„horse man“

3.8 Adjective Inflection

In Ethio-semitic languages, adjectives can have both simple and complex forms, and they share some morphosyntactic traits with nouns and pronouns. Compound adjectives are affixes, whereas simple adjectives are citation forms. Even though adjectives differ in number and kind among languages, they are conceived in a language using semantic principles. [9].

Despite being fewer in number, Gurage's adjectives share morphological and syntactic possibilities with nouns and pronouns; as a result, as well as to the conceptual sets of semantic bases, they can also be inflected and derived for distinct morphological usages.

3.8.1 Number

Solitary numbers are not denoted in Gurage, although multiple numbers are physically apparent. As a result, stem reduplication is used to describe plural adjectives, and they are also inadvertently expressed by the usage of pronouns, most often third person masculine plural (-ችኖ), which is realized as -no in informal conversations.

Some adjectives duplicate to produce their plurals. The adjectives are entirely repeated when they convey the plural forms of their counter examples. The examples that follow show how to duplicate plural adjectives.

These derived plural adjectives do not need connectors while reduplicating; instead, they are concatenated entirely by themselves. As a result, Gurage adjectives derived their plural

counterexamples by complete reduplication, but there is no partial reduplication in my fieldwork data.

Additionally, the morpheme (ኽኖ) is post-modified adjectives to express morphological plural numbers, even if their counterexamples (single form) are not noted. Adjectives are inflected indirectly for plural number. The adjectives formed and derived from this morpheme (ኽኖ) are commonly suffixed. [42].

Table 3. 20: the adjective inflection Number form of gurage

Singular		**Plural**	
መርካማ	beautiful	መርካማኽኖ	beautiful ones
ዮሬትየ	sleepy	ዮሬትየኽኖ	sleepily ones
ጠረቅ	dry	ጠረቆኽኖ	dry ones

This bound morpheme affects a variety of adjectives as a consequence. It is symmetrically concatenated to the generic adjectives, as was demonstrated by the talks that came before it.

3.8.2 Derivation of adjectives

Gurage also derives adjectives from morphological derivation. Adjectives are therefore produced from other fundamental word classes using affixes like ዮ- , …ዮ, -ኣማ , -ኣነ "-," and others as detailed below. Common nouns can be transformed into superior adjectives by adding the suffix -ዮ. Simple nouns are linearly suffixed by the morpheme (-ዮ) in this instance, as opposed to the derivational process in [43].

Table 3. 21: the adjective Derivation Number form of gurage

Noun		**Adjective**	
ወሬት	sleep	ወሬትየ	sleepy
ጉነር	hair	ጉነርየ	hairy

ቅጠር	leaf	ቅጠርየ	ever green

Simple nouns are transformed into adjectives of color and quality using the morpheme -ኣማ. This morpheme is used to construct new, semantically connected adjectives. Think about the information below

Table 3. 22: the adjective Derivation Number form of gurage

Noun		**Adjective**	
መርክ	appearance	መርክ --ኣማ > መርካማ	beautiful
ጉኽር	load	ጉኽር -ኣማ > ጉኽራማ	strong
ቅጠር	rumor	ቅጠር -ኣማ > ቅጠራማ	Talkative

Suffix (-ኣነ) This morpheme is used in Gurage to create adjectives that describe a noun's status or quality. In this case, the words' last vowels are changing. Think on the info below [41].

Table 3. 23: the adjective inflection Number form of gurage

Noun		**Adjective**	
መዛ	injury	መዛ--ኣነ> መዛነ	Injured
ቀልብ	„mind“	ቀልብ-ተ-ኣነ> ቀልብተነ	„wise“

Chapter 4

Design and Implementation of The Stemmer

4.1 Introduction

In chapter three, a morphology aspect of the Gurage language has been described (reviewed). It has been established that the primary method of word creation in Gurage is affixation. As previously mentioned, the components of affix are: - reduplication, prefix, suffix, and pair of prefix-suffixes for Gurage language Words are inflected and derived using these affixes. For definiteness, number, and person, nouns are inflected. For gender, number, person, aspect, and tense, verbs are inflected. Gurage, like other Semitic languages, has a complicated morphological structure that led to a lot of word variants. A stemmer's primary function is to reduce various word variations to their fundamental form (root). Therefore, the primary goal of this chapter is to create a language-specific stemming algorithm. As a result, the next section describes the creation of a language-specific stemming algorithm. Along with the examination of the stemmer, the collection of stop words and affixes has also been presented.

4.2 Corpus

For the experiments and the construction of the stemmer, the researcher used samples of text from several sources. The researcher assembled a collection of documents using records from several sources. There are 4,202 unique words among the 10,721 tokens that make up the corpus. The document was used to collect stop word lists, gather affixes, and test the algorithm. The test data was randomly gathered from the document to evaluate the stemmer from various perspectives of word types. Word distribution in sample text documents of a language may be seen using word-ratio, which is helpful for examining a language's behavior. This makes it easier to see how a document's words are dispersed morphologically.

Table 4. 1 the word ratio of sample document for Gurage language.

Name	Description	Word tokens	Word types	Word ratio (distinct to total word)
Books	Guragena books fiction, history, dictionary	10,721	4,202	39.19 %

4.3 Normalization

In NLP applications like stemming, normalization and tokenization are crucial data preprocessing procedures. During the preparation step, file formats, character sets, and variant forms are constantly updated to guarantee that all content, regardless of its source, is in the same format. Preprocessing is required to give the source text to the stemming program in a way that is suitable for it. All punctuation, control characters, numerals, and special characters are stripped from the text before the data is processed. Gurage employs 26 distinct Ethiopic alphabets, as compared to Amharic and Tigrigna (in the Gurage language writing system, different letters that have similar sound, like (ሠ,ሀ,ጸ,ዐ,ጎ,) are not used; only ሐ,ፀ,ኧ,ሰ are used. However, humans have written these similar sounding characters employed in the Tigrigna and Amharic writing systems in various documents from actual life.

4.4 Tokenization

For this study, words serve as tokens. All punctuation, control characters, numerals, and special characters are removed from the text before the data is processed. Space is used to demarcate words because all punctuation has been replaced by spaces. Consequently, a character string is recognized as a word if a space is added after it. In tokenization operations, a string of valid characters identified as a word was recognized.

4.5 Stop Word List Creation

By gathering the words that appeared the most frequently throughout the Sample paper, the stop word list was created. The document's word frequency was created using a Python algorithm. Stop words are words that have no function in natural language processing (NLP) applications yet are often used in writing texts. These stop word lists are created for two main reasons: These words could impair stemming performance because no NLP application will benefit from the usage of stop words in stemming. Second, eliminating stop words aids in file size reduction. Stopping is the process of getting rid of words from texts that don't add much to the substance.

Pronouns, prepositions, particles, and articles were used to create a Guragegna stop word list for the sake of this study using a Python software. Guragegna's broad stop word list was created. The word forms are first arranged in the collection of Guragegna documents according to how frequently they appear. The terms that occur most frequently are taken. Second, all verbs, nouns, and adjectives more or less directly related to the primary subjects of the underlying collections were carefully removed from this list.

Table 4. 2: Sample prefixes of Guragegna

NO	Word	Frequency
1.	'ቃር'	198
2.	'ታነ'	125
3.	'ሰብ'	104
4.	'ባረም'	100
5.	'ባረችም'	70
6.	'ም'	66
7.	'ቈን'	59
8.	'ባሬም'	57
9.	'ምሽት'	56
10.	'ናማጋ'	54

11.	'ምር'	53
12.	'እያ'	51
13.	'ቤት'	44
14.	'መደር'	42
15.	'ዘንጋ'	41
16.	'ምስ'	40
17.	'ሸማ'	39
18.	'ዳር'	36
19.	'ጋመ'	35
20.	'የምር'	34
21.	'እኳ'	34
22.	'አርብ'	34
23.	'ሸም'	32
24.	'በር'	32
25.	'ዝርጋት'	31
26.	'ዛሀ'	30
27.	'ቄሸ'	28
28.	'ቄም'	27
29.	'ብር'	27
30.	'ኧኒ'	27

4.5.1 Compilation Of Prefixes

A list of prefixes that are used to build the algorithm is compiled from several sources depending on the grammatical functions of the affixes and their frequency among the Guragegna words found

in the document collection. The list was compiled from different Guragena research's [43] Degif Petros Almayehu [9] Bahire Araya [42]

Table 4. 3: Sample prefixes of Guragegna

prefixes	**prefixes**
አት	ኣን
አሁ	በ-
ይ	እን
ን	እም
አን	ኣት
እ	ባን
ት	አስ

4.5.2 Compilation of Suffixes

A list of Suffixes that are used to build the algorithm is compiled from several sources depending on the grammatical functions of the affixes and their frequency among the Guragegna words found in the document collection. The list was compiled from different Guragena research's [43] Degif Petros Almayehu [9] Bahire Araya [42]

Table 4. 4: Sample suffixes of Guragegna

suffixes	suffixes
ነት	አሁ
ኦት	ሂነማ
አነ	አሂማ
ህኖ	ቹ

አማ	አሀ
አነ	ወት
ሁሞ	ተነ
ጉዳ	ነማ
ሂኖ	የ

4.6 The Proposed Architecture

There are numerous techniques developed for English that have also been studied for Semitic languages like Amharic, Tigrigna, Silt'e, and Geez. The Gurage stemmer was created using some of the methods employed in these algorithms. The procedure is iterative, although it starts by removing the longest affix. The stemmer accepts directly Unicode data that has been represented in Unicode. As a result, when creating the algorithm, the length is employed to indicate the size of the word. A Gurage word must have at least two letters to be meaningful. The algorithms based on this idea have therefore employed the minimum word size.

Gurage uses one or a combination of the methods to carry out the inflection and derivation operations. affixing externally without altering the stem just removing the internal and exterior affixes with a few modifications may have been the beginning of the end., Middle or beginning which may lead to insertion or deletion of consonant and vowels or pattern exchange

For the goal of reducing affixes, the stemmer was designed in two steps. Prefixes are removed in the first phase. In order to remove a prefix, it is necessary to check for matches in the stop word list, prefix list, word length, and context-sensitive conditions. The prefix will not be removed from the word if it matches the prefix list and meets the context-sensitive criteria; otherwise, the prefix will be stripped out. The suffix is removed in the second stage. The phrase is initially contrasted with a list of stop words. If there isn't a matching in the stop word list, the total amount of the word will be considered. The suffix list file will be opened and the suffix and word matching will be checked if the phrase is more than two. If a match exists for the phrase but it doesn't satisfy the context-sensitive requirements, the suffix stripping method is applied. Otherwise, no suffix is going to be dropped.

The stemmer first obtains the word before determining whether it is in stop word list . If an exact match is found, the stemmer continues and takes the next word if the file hasn't finished; otherwise, the phrase will be delivered for the next phase, the length of phrase will be recorded. If the word is less than three letters long, it will not be stemmed and will instead be recorded to a stemmed file; if not, the phrase will undergo an affix elimination procedure in order to remove the required affixes. The program then compares each word to the stop word after removing the prefix and suffix. Recoding and context-sensitive checks will be performed as necessary, and the appropriate action will be done, depending on the situation. architecture

4.6.1 Context Sensitive Rules

A context-free stemmer eliminates strings without regard for the remaining stem, allowing it to remove strings that appear to be valid affixes but are not. For examples removing “re-“from “response” or “-al” from “fatal” and it produces similarly unsatisfactory outcomes when context free is used for Gurage. For instance, ጥራርወት {terarwt} {ጥላ} umbrella , ስንወት {senwt} {ጥርስ መፋቂያ} ‘brush’, have a leading string ወት {wt} which is in the list of prefix removing this leading string result in ጥራር {terar}, ስን {sen} has nothing to do with the right stem. The researcher reached the conclusion that affix removal should be controlled by two action codes and three criteria and that a context-sensitive method should be adopted.

Two categories of context-sensitive actions are employed:

Action 1: do not remove any affixes

Action 2: preform remove affix

4.6.2 Recoding Ruls

- If the word stars with the letter “ተ” and ends with ‘ም’ do not remove the first word. only remove the last word
- If the phrase ends with ‘ንዳ’ do not remove the suffix and change the word to six letter formats

4.6.3 Context Sensitive Conditions

- To maintain the language's minimal stem length, a stem must be at least two characters long.
- If the characters at the second place of the stem and the characters at third place match with the first, the rules are applied. This is done to avoid the removal of non-genuine affixes. Take action 1 when we handle reduplication
- If the word can be spliced in to two and the spliced word has the same charterers remove the second splice

4.7 Compilation of Affix

Prior to using this technique, some frequent list of stop words from the Experimental txt file are eliminated by comparing them against lists of stop word. The length of each word is also examined during affix removals. Every time a term is longer than two words, the stemming process is applied. This is due to Gurage's two-word minimum word length. Following the removal of affixes, the stemmed word is checked against a list of stop words; if it is included, it is not subject to the affix removal procedure and is not included in the stemmed file. In essence, the above-described stemming techniques involve two steps. First do not remove any affix and the second will be in order to use the two action we must state the condition which will tell us when to use action or affixation. The conditions are described in context sensitive conditions.

4.7.1 Prefix, Suffix Striping And Letter Reduplication

The procedure accepts a word and checks for the presence of a true prefix, removing it when the requirement is met. If there are more than two words remaining, the prefix will be stripped after evaluating the context sensitive rule. If no condition is met, the prefix is stripped; otherwise, the term is returned and the process suffixes examines a word to see whether any suffixes in the suffix list match it. If a match is made, the context-sensitive conditions will be checked, and based on the results, the relevant action will be executed. The process is essentially similar with the prefix striping; the only difference is that the elimination and affixes are on the right side of the phrase..

The first letter of the word is duplicated in this process. The output of the prior procedure is used as an input in this procedure. When a word has a reduplication and the first letter is not in the

alphabet's initial order, the reduplication is detected and the first letter of the word is removed. If not, the operation returns the word in its original form. The second procedure It looks for reduplication and, if there, strips the second letter from the word before returning it intact. If not, the function just outputs the word as is.

Table 4. 5 prefix and suffix striping stemming algorithm

```
1 Get Phrase
2 find Phrase length and check if <= 3 and check if string
        If word is less than 3 or not string
              Go to step 5
        Else
              Continue
3 Read the stop text and see whether the phrases match.
                If phrases exist in the list stop words then
                Go to step 5
4 Check to see if the temporary suffix matches the suffix list file.:
 If match found check conditions:
                Remove suffix
                Check length of temporary suffix
                If length of new stem word is greater than 4:
                   If a recoding or context-sensitive rule is followed:
                     do recoding
                print stemmed word, assign variable to new stemmed word
                and continue
        4.1 Check to see if the temporary Prefix matches the Prefix list file.::
            If match found then check conditions:
                Remove Prefix
                Check length of temporary Prefix
                If length of new stem word is greater than 4:
                        Continue
                   If a recoding or context-sensitive rule is followed:
                  print stemmed word, assign variable to new stemmed word
                and Go to step 4
        else:
                Go to step 5

5 Check If end of file not reached Go to step 1
Else:
Stop processing
```

Table 4. 6 : Final reduplication removal stemming algorithm

```
1 Get  Phrase
2 find  Phrase length and check if <= 3 and check if string
If word is less than 3 or not string
      Go to step 5
Else:
      Continue
3  Read the stop text and see whether the phrases match
     If  phrases exist in the  list of stop word then
     Go to step 5
Else:
4: 4. If the second and the third letters of word are same or in the in same alphabet:
       If length of new stem word is greater than 2:
              Eliminate the second letter
               return the  Remove word
                Go to step 5
          Else:
          Go to step 5
5:  Check If end of file not reached Go to step 1
Else : End processing
```

Table 4. 7 : final reduplication of word removing two pairs

```
1 Get Phrase
2 find  Phrase length and check if <= 3 and check if string
If word is less than 3 or not string
      Go to step 6
Else:
      Continue
3 Read the stop text and see whether the phrases match
     If  phrases exist in the  list of stop word then
     Go to step 5
Else:
4. check if the word can split in to two part equally:
          4.1: If the first parts of letters in Phrase are the same to the second parts of the letter of the
          Phrase are the same:
                If length of new stem Phrase is greater than 2:
                      Remove the second parts of the latter
                       return the Phrase word
                        Go to step 4
                   Else:
                   Go to step 5
5: Check If end of file not reached Go to step 1
Else: End processing
```

Table 4. 8 : frequentative reduplication of word removing last ordered

```
1 Get Phrase
2 find Phrase length and check if <= 3 and check if string
If word is less than 3 or not string
      Go to step 5
Else:
      Continue
3 Read the stop text and see whether the phrases match
     If  phrases exist in the  list of stop word then
     Go to step 5
Else:
4: If the last [-1] and last [-2] letters of word are the same or last [-2] and last [-3] letters of the word are
the same:
       If length of new stem word is greater than 2:
             Remove the last [-2] letter
              return the Remove word
               Go to step 4
         Else:
         Go to step 5
5: If end of file not reached Go to step 1
Else :   End processing
```

4.8 Implementation of The Stemmer

Python is used to create a sequential program that implements the algorithm's rules. The lists of affixes are verified against the word according to the implemented algorithm, which is the longest match first. The affixes that Guragegna speakers employ to create various word variants have been gathered by the researcher. The correct affixes are chosen from all conceivable combinations using these affixes to generate the rules.

Table 4. 9: Sample of prefix striping

NO	Prefixes to be removed	The word to be stemmed	The word after stemmed
1.	'ተ '	ተሻደም	ሻደም
2.	'ይ'	ይቶት	ቶት
3.	'እን'	እንግፈር	ግፈር

4.	'በ'	በነሲብ	ነሲብ
5.	‘አት’	አትበካ	በካ

Table 4. 10: Sample of Suffixes striping

NO	Suffixes to be removed	The word to be stemmed	The word after stemmed
1.	'ነት'	አበንደነት	አበንደ
2.	'ንዳ'	ኣዶትንዳ	ኣዶት
3.	'ኣነ'	ኤማነ	ኤማ
4.	'የ'	ቅጠርየ	ቅጠር
5.	‘ሁኖ’	እማርሁኖ	እማርሁኖ

Table 4. 11: Sample of Reduplication striping

NO	Reduplication to be removed	The word to be stemmed	The word after stemmed
1.	'ጣ	ቆጣጣ	ቆጣ
2.	'ጠ'	መጠጠረ	መጠረ
3.	'ጥ'	ቅጥጥር	ቅጥር
4.	'ካ'	ለካካ	ለካ
5.	ና’	ቆናና	ቆናና

4.9 Evaluation of The Stemmer

Correctness, retrieval efficiency, and compression performance are three factors that are taken into account while rating stemmers. Over stemming and under stemming are two examples of faulty stemming. A phrase is over stemmed when an excessive amount of it is eliminated. Over stemming can lead to the confounding of unrelated terms. The elimination of too little of a phrase is known as under stemming. Under stemming will stop words from being confused with one another. The stemmer's performance was assessed using manual counting technique. This makes it easier to compare the amount of incorrectly confused mistakes with the right ones.

The experiment text, which was chosen at random from the sample text document, was used to test the stemmer. 1,933 experiment text words. Two different types of faults are seen throughout the stemming process.

The experiment text's stemmed result is 1266. Out of these terms, 5.37% (68) were under stemmed and 5.37% (101) were over stemmed. The stemmer yields 13.34% (169 words) stemming mistake in total. As a result, the stemmer's accuracy increases to 86.65%.

Table 4. 12 : Over stemming and under stemming stems

NO	Stem	Expected stem	Resulted stem	Error type
1.	ተራከበም	ራከበ	ተራከበ	Over stemming
2.	ኣምቢቸም	ኣምቢቸ	ኣምቢ	Over stemming
3.	የቐጫዳነ	ቐጫ	ቐጫዳ	under stemming
4.	እንሰረሰረኑም	ሰረሰረ	ሰረሰረኑ	under stemming
5.	ፈጠርቅና	ፈጠር	ፈጠርቅ	under stemming
6.	ገትራክብነሽ	ትራክብ	ገትራክብነ	under stemming
7.	ትታጫውድን	ጫውድ	ታጫውድ	under stemming
8.	ፈዘዘም	ፈዘዘ	ፈዘ	Over stemming
9.	ኣሰዊም	ኣሰዊ	ኣሰ	Over stemming
10.	ብትብሬ	ትብሬ	ብሬ	Over stemming

4.9.1 The Results

The experiment text used in the evaluation process is contained in the Data set. The experiment text has 1,933 words, of which 1266 resulted from the stemming procedure, out of a total of 1266.The number of words successfully stemmed is 1097, achieving an accuracy of 86.65%. 13.34% (169) of the stemmed words were wrongly stemmed. Over stemming accounts for 7.97% (101) of the terms, while under stemming accounts for 5.37% (68). The following factors may be responsible for the inaccuracies that were detected by the stemmer. According to a thorough analysis of the language's morphology, more context-sensitive rules are primarily necessary, and the second will be Due to the language's complexity, it was challenging to produce an exhaustive list of affixes at once.

Table 4. 13: accuracy of the first stemmer

Test Set	**Total Word Count**	**Correctly Stemmed Words**	**Over Stemmed Words**	**Under Stemmed Words**	**Stemmer Accuracy**	**Error Rate**
EXP	1266	1097	101	68	86.65%	13.34%

4.9.2 Word Compression Ratio

The word compression rate of the stemmer is also assessed. The formula is used to calculate the word compression rate (C), or dictionary reduction [45]

Which is : C = 100 * (W - S)/W

Where **W** number of total words

S number of stemmed words from W

C is the compression value (in percentage)

Experiment text has 1,933 words. The number words of the test data after stemming also counted and Stemmed words are 1266

Table 4. 14: word compression ratio of total words

Test set	Stem (S)	Word (W)	Compression Ratio (C)
EXP	1266	1,933	34.50%

4.9.3 Finding of The Study

In this research, a context-sensitive and longest match stemmer that conflates word variations to their respective roots is created for stemming Guragegna literature using a rule-based approach. When the steaming procedure is finished, over- and under-stemming issues are noted, or the experiment uses 1,933 words that were chosen at random from the available sources of text content. The data are used to evaluate the stemmer's performance, and the results reveal 86.65% accuracy. 34.50% of the terms in this word data set were stemmed words. The experiment's overall errors were 13.34% in total. Conclusions of the research are presented in the following chapter, along with recommendations for further study.

Chapter 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

There are many Semitic language in Ethiopia. One of the Semitic languages is Guragegna. The root pattern structure serves as the foundation for the shared grammar of these languages. Affixation is the primary method used in Guragegna to generate words. Prefix, infix, and suffix are used in Guragegna. It also repeats words in sentences Guragegna words can result in a huge number of alternatives, therefore, effective conflation methods are necessary if the excellent recall is to be reached in searches of Guragegna text data sets. This study looked into the viability of creating a language-specific stemming algorithm. The following conclusions are offered based on the tests run for this study and the outcomes obtained.

- In order to assess the stemmer created in this study, 1,933 test words of varying sizes were randomly chosen from the Experimental text. The outcomes of trial demonstrated indicated that the dictionary size was reduced by 34.50% for stems while the stemmer performed at an accuracy of 86.65%.
- The stemmer's context-sensitive rules are designed to operate well for groups of words.
- Only inflectional and derivational affixes are combined by the stemmer and It does not conflate irregular forms with compounding.
- Due to the language's morphological complexity, there are significant over and under stemming errors.
- The tests were conducted on a tiny collection so it is unknown how the stemmer may affect a larger data collection.
- The guragegna language had many branches that may differ one from other the researcher main focus was on Cheha (one breach of gurge language)

- Finding a rule that is effective for the majority of or a large group of words is the most challenging aspect of researching stemming rules for Guragegna words.

5.2 Recommendations

This study demonstrated the potential for creating a stemmer to combine word variations in the Gurage language, which has a complicated morphology and allows for the existence of several word variants. However, due to time constraints, the study's sample size was constrained, and it was not tested in an IR setting. Although this study activity has yielded positive results. For more work to be done in order to make the output useable, the following recommendations are made.

- Other NLP tools like morphological analyzers, machine translators, word frequency counters, and automatic text summarizers can also be designed using the stemmer as one of their component parts.
- Analyzing the stemmer on a vast text collection gathered from many sources. This is so that larger samples, compared to smaller samples, can more accurately capture the characteristics of the language.
- measuring the stemmer's performance in a real retrieval session while evaluating it in an IR environment if it developed in the future
- This stemmer's accuracy could be improved by adding more context-sensitive and recoding rules.
- Additionally, by conducting additional language study, this activity can contribute to understanding of the language because of the researcher main focus was on Cheha (one branch of gurge language) part of the language

Bibliography

[1] G. Salton, Automatic text processing The Transformation, Analysis, and Retrieval of, Arlington Street, Suite 300 Boston, MAUnited States: Addison-Wesley Longman Publishing Co., Inc., 1989.

[2] S. a. Thambidurai, "Algorithm for Root Word Stemming, Information Technology," *Jornal 5(4),* pp. 685-688, 2006.

[3] J. B. Lovins, "development of a stemming algorithm, Mechanical Translation and," in *development of a stemming algorithm, Mechanical Translation and*, 1968, pp. 22-45.

[4] D. Sharma, "Stemming Algorithms: A Comparative Study and their Analysis," *International Journal of Applied Information,* pp. 2-6, 2012.

[5] M. Porter, An algorithm for suxffix stripping Program, 1980.

[6] F. W. R. Baeza-Yates, formation Retrieval: Data Structures & Algorithms, NJ: Prentice-Hall Englewood Cliffs, 1992.

[7] Krovetz R, "Viewing Morphology as an Inference Process," *Annual International ACM SIGIR Conference on Research and Development in,* pp. 191-202, 1993.

[8] G. T.Stump, Inflectional and Derivational morphology, Kentucky, USA, Kentucky: Cambridge University, 2001.

[9] A. Alemayehu, Relative clauses in Chaha, Addis Ababa : University of AddisAbaba (MA thesis), 1990.

[10] E. Assefa, The Structure of a noun phrase in Ezha, Ethio-Semitic, adiss abeba : University adiss abeba , 2011.

[11] Leslau, Ethiopic Documents: Gurage. Popular Interpretation of Bird Sounds in Ethiopia, New York, 1952.

[12] R. Hetzron, The two futures in central and peripheral western Gurage, Hudson: Harrassowitz Verlag, 1996.

[13] F. Menuta, The Morpheme Gurage Morpho-Syntax Department of Language and Literature, Hawassa : Hawassa University, College of Social Sciences, 2019.

[14] N. A. a. P. Willett, "The Effectiveness of Stemming for Information Retrieval in Amharic for Information Retrieval," *Electronic Library and Information Systems*, pp. 254-259, 2003.

[15] G. Berhe, temming Algorithm Development for Tigrigna Language Text Document, Addis Abeba: Addis Abeba Unversity, 2001.

[16] W. Mekonen, development of Stemming algorithm for Afaan Oromo Text, Addis abeba : Addis abeba University, 2000.

[17] A. Belay, Designing A Stemmer For Ge'ez Text Using Rule Based Approach, Addis Abeba : Addis Abeba Unversity, 2010.

[18] L. Lessa, Development of Stemming Algorithm to Wolytta Text, Addis Abeba : Addis Abeba Univerity, 2003.

[19] J. S. Sumamo, Designing A Stemming Algorithm For Kambaata Text: A Rule Based Approach, Addis Abeba : saint Mary Univerity, 2019.

[20] M. P. a. P. Willett, "the Effectiveness of Stemming for Natural-Language Access to Slovene Textual Data," *In Journal of American Society for Information*, pp. 360-390, 1992.

[21] W. Peter, Automatic indexing of documents and queries," Document retrieval systems, 1988.

[22] P. Ahlgren, The Effects Of Indexing Strategy-Query Term Combination On Retrieval Effectiveness In A Swedish Full Text Database, Gothenburg: University of Gothenburg, 2004.

[23] D. Christopher, An Introduction to Information Retrieval, England:Cambridge : Cambridge University, 2009.

[24] S. Kodimala, Study Of Stemming Algorithm, Las Vegas: University Of Nevada, 2008.

[25] Dawson, Suffix removal and word conflation, ALLC bulletin, 1974.

[26] D. Harman, "How effective is suffixing," *Journal of the American Society for Information Science*, pp. 7-15, 1991.

[27] C. D. Paice, "An evaluation method for stemming algorithms," *Croft and van*, p. 42:50, 2001.

[28] M. H. a. S. Weiss, "word Segmentation by Letter Successor Varieties," *Information Storage and Retrieval*, pp. 341-395, 1974.

[29] M. J. a. McNamee, "Single N-gram stemming," *international ACM SIGIR conference on research and development in information retrieval*, pp. 312-314, 2003.

[30] M. M. a. N. Orio, the Measurement of Term Importance in Automatic, 2003.

[31] G. A. a. J. Boreham, The Use of an Association Measure Based on Character Structure to Identify Semantically Related Pairs of Words and Document Titles, 1974.

[32] Y. Fisseha, Development of Stemming Algorism for Tigrigna Text, Addis Ababa : Addis Ababa University, 2012.

[33] D. Sharma, "temming Algorithms: A Comparative Study and their Analysis," *International Journal of Applied Information Systems*, p. 2:6, 2012.

[34] P. M. &. Willett, "Processing of documents and queries in a Slovene language free text retrieval system," *Literature and linguistics computing,* pp. 173-189.

[35] Argaw A.A, "Reducing words to their citation forms," *proceedings of the 45th annual meeting of the association for computational Linguistics,* pp. 104-97, 2007.

[36] W. Kraaij, "Viewing stemming as recall enhancement," *In Hans-Peter Frei, Donna Harman, Peter Schauble and Ross Wilkinson(editors),* pp. 40-48, 1996.

[37] T. D. a. E. Abebe, "Designing a Rule Based Stemmer for Afaan Oromo Text," *International journal of computational linguistics,* 2010.

[38] M. Kedir, "Designing a Stemming Algorithm for Silt'e Language," Addis Ababa University, Addis Ababa, 2012.

[39] K. S. a. Garside, Stemming Arabic text, Lancaster: University, Lancaster, 1999.

[40] K. Darwish, CLIR experiments at Maryland for TREC 2002, Maryland, 2002.

[41] C. Ford, "Notes on the Phonology and Grammar of Chaha-Gurage," 1994.

[42] B. A. Keleta, "Gura Documentation and Description of Morphology and Syntax," Addis Abeba University, Addis Ababa, Ethiopia , 2020 .

[43] D. P. Banksira, SOUND MUTATIONS "THE MORPHOPHONOLOGY OF CHAHA, Philadelphia/Amsterdam , 1984.

[44] B. Wakjira, "Morphology and verb construction types of Kistaniniya," NTNU-Trondheim, Norway:-Trondheim, 2010.

[45] Kaplan, A method for tokenizing text," Inquiries into words constraints and contexts, 2005.

APPENDIXES

Appendix i: Suffix words collected for the stemmer

ሽ	ቄ	ነ	ና	ማ	ችም	ኒ	ናም	ም	ዊም	
ንም			ታ	ነት	አት	ኣነ	ህኖ	የ	ኣማ	አነ
ሁኖ			ኛ	ንዳ	ሂኖ	አሁ	ሂነማ	አሂማ	ቹ	ኣህ
ኤ		ሂማ	ሁ	አ	አማ	ናህ	ናሂማ	ነማ	ኖ	
ና		ተና								

Appendix ii Prefix words collected for the stemmer

ያ	በ	ቲ	ብት	ባነ
ብ	የ	አን	እም	እ
እን	አት	ይ	ይት	ት
ተ	ን	ኣት	አስ	ባን

Appendix iii: Guragegna Phrases before stem

አትያነችም ንሸያም ዋርኹምሽ አትቁርስ ይስጭኔ ተረሣናም በመሰሮም ንኽነሼ ስረተተም ኣምጣጣም ብታተረ ቀነስናም ጠበጠም ቀነስኹም ተባናም ያቅቀረብሁና ኣትጉዘገዘም እንዲመገቡ ናጐረም ያቀነፈ ይፈርሑፑ ተቓጨምዌ መሠረም አፍጣጠረችም ይስብር ኣርገነገም ፍሬሽ የበርዴ ኣዝራበጠም አምባሁ የገሳሙና ያርዋኽታ የዋራኝ ኣዳራሽ የዋሪነታ ኧበርሽ አንሸውሽ ታናጠፉም ዛንዜተነ ብሸክልሊት ተግመገመም የፈትኸና ዋነሶም አጠሳችም የታጠፍ አድሁም ተረከበም እንቖሰም የባርጠማ ላዴሽ ኣበራም አወረወም ተዝዋጠረም በርማ የሬፐሪች የደናኽ የሙገመነ ጎበናም ትራማሽ ይስጭየው የጉራማ ተሙገማ ኣሐበደም ይረሕየ ባንክሬ ይትዐሪ አትባጀትኡ ኣማነሽ አንገተሸታ ብታኻን ገፖንም ኣዛገገም አጮሩንም አህማ ነገውም አትያነም የቀለጠች ተጣነረም ይረግም የህርም የማደድ ጠርጠርተነ ባሪታ ዋርሁም ተበበርሁም ገፓችም አጎዴሽ ይበዛቢ ያትዋናኖ በቀናችም እንቕብንየ ነቀመም ጥንቆቅማ ኣንግራሳም የአሰራሽ አሸቼም ይደውርፑም የረፐራ የሰጡትን ተኖችም ትትጫወዶ ኣበሰረም ብሕብር የዘርማ ተዳናም አትነቖም ኣዥባፐረም ደቦሐረም ፍጨጨም ኡ_ኣቖም መሰረንም መሰስመታ ሙትንያዌ አገኈም በጎቆታ ንጨጨም የትከኣዌት ተሣረናም ትጠራሽ በብረዌ ኢደገችም ሸኽኖም ምፍመኝታ ዴንቅነት ፈለቀም ሰተበም ካርታ ጣፍሁምመጣፈታ ኣደፈፈም ተድረነገም የድምድጓድ ጀግጀባረም ጣሰማ ሔንደም የኣጅየት ኤስማ ኣቁራጠበም የድርስየ አሸችናም ገፈሪም መደሬታ ኣድዋነም እትቖነወችሬም ኣኽማ ያርዋሴ ተዛሮክንም ዳረንም ኣጠዊም የጉራጌም መሰረም አትቅጥ ዝፈዜም ባንትÓባበረ አትባቶም ያቀረቹ ነገሰም ወበረኽታ ኤፈሽ አሸነትውሽ ተጫረም ሙትትባረም ተረመደም ትሠራኒ ቀነሶም ቖጠረም ጉርማስነት ኤጠመም ከሰረም ንከከሰም ተራጨም ሁትሽ ገገታ ጥባነረችም ተረፈቀም ግረተመም ይወጥቅና ደበረም በኻቆታ ይቀርሴ ተቄቄረም ተራሐም ኤርቀሩሽ እየቋጠብሽ ገግመታ _ኣከሙሽ እርኻም ኤወጣኒ የሜናዥቴ ብረሸም ብሎአቸ ተስሓነም የውርምየ ይረብር አርዌ ድረከረም ኣጨነዘም ተሳደደም ኣትመመረም ነፑሪም ኣወረም ያንዛከሸ ወገበታ _አለም ተሕማቸም ብዓዝረታ ይህርሸ የትዕክት ረዘታ የዌድማም መጨቀም ጋዋኣመነም እዳገሬ ኣበታም ደረርባረም ቅየጓደምዌ ባረቼም ባጎፊሸ አፍጣጥርም ቀጠዌ አትኽርወ ጐዛርመታ ኣማነደም ንማድት ውኻትተናካሸ ትጉየዋ የኩርታዊ ትሸታ ኣትቖነበም ዝፐረራም አትሣሰው የባረቅ ኣፈቀረም ጠነቀም ያርዋኽ አትባተርኽም ኣትክባበሰም ቴለማ ጠናማ ተበግዳና ዘንገረነ ኑባሬም ተጨጨም ይወርድ ይትረገፍየ ሰላድየ ተጥበጠበም ገነታ ዣንዠኤፐም ያትቅነህ ኣትሸበም ኢያማ የሁሌት ቅራመደም ይወርክ እግረናው የቄምሁጉ ቀነሰም ይወደረሞ ይፍትፍቱ ሸረኛ ሙኽማ የሰረች ኣድፋፈረም በትፈጠርሁ ጡሪነተና ወሸለም ኣተከሰም ዝጉባረም እቖነትሞ ተኛኽተም ሃብኝም እነዝህነው አረነጠም ጉነርቃጥሮት እንኽፈኽፈም ባርኽም ኽረቖም ከቹም ኹችም እንደሕሬቾ አሰናኝም ትኽርኽ ይውርም የድረኒ ከድልኒ ተሣሬም ኤፋጀሽ ቢጠርቅና ይሸርፉ እሞበናነ ይትቖማመጥየ አትድብረማጐ ተበንጠም ኣሴቀም በፋዘታ ትራማታ ቁናታ ያጋታ ኣሑራመጠም ተከፓም ኣብሳካም ይብርኽ ድረተረም ወኽትማ ሸርበሣህ በርዝናባማና ተራፉተም የፓችም ተሳነፉም በበርዛና ጥለየንም ዝረበጠም ባነቦጐ ያነበንጦ ቦርጭማ ኣትራታም ተቓጠም ተቾነሽ የሬነውያ ተጉመመጠም ስቀሰቀም ተቀፑሪም ጨዋዳህ የሞራወ የሚታኝ ባናቅቆነውነ ቅርጥናሽ ብትገፍርጦ ለጀታ ይኾሹትን ኻርሁንም ኽረምቃማ ኣገኼም እንድህ እመታ ጎፑረረም ያወርድሽ የተነፈ ኣዌዘሪም ከጠላኝታ ወንደም የነጌሮ የስክትን ገግመኽነማ የጠፋጦ ኣትየጠረም ያፈርወንቀርቅር የምስጉን ሶሬታ ለሰማችህ የዘንጋ ሻባፐርሁም ምርታዊህ ኤንግድኹሽ ኣትረሳም ያናችን በፖችህ ኣትኼታም እንፉር ተደፐወሽ ተዛመደም ይድምድሁም ፈፍትም የኑሩሣ ጫርኝም አትቃው

ወምበርሽ ይቆርጥን ተዘርጋት ይትሸሎ የትሣሩኖ ትትርኽ ብትብረሜ ትፈተፈም ተፍራተሐም ነበረም ፈፌታ አቅመሁና ተደንታንዳ ዮሮትም ይምበሮች ኣትረደፈም ጋሙመታ አፈኛደየ ሙትትንየ ቴቅሽነት ደርምሶት አስማኽማ ያትኝትር እንቅርሻ ኣጋማ ኧሰጦሽ ነፐሮም የቀጠረ እንፈዐ ወጣዓም ሔታሽ ተኟረም አወናችንም በራጫሙት ተጭማማም ኣከውየ ቀኖችም ዛበተም ፈዘታ ኤፈዘዌ ኩርስየ የባድራ እብራሽ ባፈናም ቲደርሸዮ ፍረነመም ይጠፍየ ይጠብጥ መርዛማ እወጣም ኣመሸም ኣጠናም ይኘራዌ የካርሁን ኣቸከረም አወናቐም እምበኖ ብምስን ኣንጥረጠረም ይጠቅር ተቃጠረም ትኽንዌ ተ኷መታ ይፍጥርሽ ይነቓነሶ ያገድኽን ተዘርኋት ቸናሁማ ተፈቆሬም አገኤም የዴሪው ቀነሙንንም ያኑቆንበ ያደግወ ፈንጠናኽም ያጐነን ንብረት የባረዊኣታ ክሰሰመም ቶሬተታ ኧ኷ዛም አርዋኽታ ንኽዌ ጠቆመችህ ያዘነረን ሴተረም የበርህማሽ ተቅመጠረም አመርÓወም ጠናቃም አወሬም ዋርጋም እቴገረ ትውነታ ኘኽረም ቲሸርት ያቤዛር ስፐረችም ቴክርም እማጥኽማን ተዝራነፈም ኣነበም ተቸሁኒ ኣትየጠበም በኘዋታ ብትሰና የባኼን ይቀርስ ነበርሽ ተገገዘም ኩረኰመም ተኟመም ኣገረም አማትሽ ያትቐቸውን እንቀያ ምክንያት ይረጥርየ ፍሬርቶት የጠረችኔ ዝፐሮም የአረቄ ተፈቀርዥፐንደም ሰክትኒ እስቀሰቀም ባንሰራ የታሪክታ ያኑራነስና ይጀፐራ ኣረደንም እሸታ የችም አሸችንምበስቀርጹንም ያነበት የስራሰረች የበናና ዴራረኽታ ፍጠፈጠም አትዋሽ ብትሰን ፈንደም እንም የሶሬታ ተስራመጠም ብታቱን ከስመታ ትንኾን ቲበርስ ሸካክተም ተዐቅኑጣም እሞብቤውን ቤቱሽ የሐራትየ እምቦያ ትረዝዝ ይስብረወ ስገረም መነገም ኣትሙጨጨም የአበራሽ ባኾምም ኣምራነገም ሙባረም የቀፐረ ያማራና ተድራረበም ይውርክ ይደቅቤቴ ቃሮንሁሽ ይብሮንቴ ይወርማ የሁኤችመንዳ ውረመህነማ ንግስት ሜራታ አነሰችም ጀባባረም ቲያድር ኣቴሸም እጥወተና ጉንጫኽማ ሀማይህ ኤጓድሁ ይትርዋሸፑ አየውሽ ለሸኺም ተርቃቀጠም የዝርማ ታቅኽርዌ ደመሰማት ባርጠማ ኽማውሽ አመሸም ምሱቄንየባረማ ይንገድሸሁና የቅረርየ ሓበራም ሰኽራም ያጓትረሳች እጅየት ዮባታ እንጐቻ ያባቱታ አንሽ ኣምጫመጨም ኣቴመረም ተቐፐረም ምሣሬታ አነረዌ አግደቼምሽ ነነወኝት እያተሁት ትባርሽ ቲያትያነወ ስናወም አቴብሽ አሸናም አፍርቶት ትዘፐርም የሬፐሬ ገማኽማ ገደደም ወገታ ይቄርሸፑ ቢያሻሽ ኣፍጣጠረም ናበዓታ ኣጭራነመም የዛታ የአሳንዶም ያናታ ታትጀፐርም ኣሓጨም አትቃርሸን ይዝከማ ይጠርቄ ኣቅረበበም አፈጠረም ተጥባነረም ኣመከረም ቶሬዮሁና ይጉርየ ትንብር ፍረቀረም ድረዘዘም ኖበመደረታ ኣጨገም ተቀጥረንም ኤሸንቃርሸ ጠረዘም ኣመሰተም ቀንጠችም ዝቀረም አትራነስኙሞ ነወነሽ ኽናችም ኣሐነቀም የክርኽ ጋባረም ጀኰረም ቀንቀንት ነግረነት የጐናቃ ብራሂም ትረሙድን ናጠችም ሃራቼም ምኅረተታ ፋትማ ምርትም ትትራክቦጭ የስማ በቤታህማ ጉኑርታ እኻርሽ የረፐነ ይትራመ ያትቋና የጋውነቱ ኣገረከረም ቆጥኘም ዝመረረም ወንዝብሽ ሥሩናኽም ገፓሁም ኢጋደኖሽ ድየተረሣችም አገፖኖም ሬሣናቱምዌ አገባሽም ትትገዳሮ ተሣርችናም ዝረኮም ነዌባም ተፈሰም አትብኒ' ኣገነፈም ኣንካካም እንፋጨ ኣደበተነ አሸችኖም ኣሕማ ኣደረም ያደቀዴ አረኸም ምላሽ ይሸነትን ትሞበርሽ ዝክሽ እንፉየ ቧሬንም ቅረጠመም አንኽሬም ውግሽ ኧራይኣኽታ የሚርመ ፈዘዘም ኣገነፈም ኣንካካም እንፋጨ ኣደበተነ አሸችኖም ኣሕማ ኣደረም ያደቀዴ አረኸም ምላሽ ይሸነትን ትሞበርሽ ዝክሽ እንፉየ ቧሬንም ቅረጠመም ብትስና ተቐባፐረም ኘውሽ ኣብረከሰም ተኟመደም ግዘነም ገነፈም አባመኝታ ዛህርታዊውየም ቀድራከትኹም ይረብኧታ ያገድዬ የሞበር ቅንበታ እኩምንየደ ናጥሁም ያቅሳብርሁ ኣትራነሰም ኣሳደደም ትፈጠርሁ እበርም ኣገፓም አህማም የሶሬሣ ስረግም የባሸጉየ ኣረካም ትክነት አጥዋሐረም ራውሬታ ያርብም ባነአኽ ፌጠጠም ይብጠብጤ ትራሽ ትሁንም እፍረጮኽዴ የትሸነሩን ዳሩሽ የዘብርዒ የጦሙት ኣፋነሰም ጠፋሁም አፍልቀረንም የርማጅነት የጦጨሞታ ቴጠጠም ጸመችም ማጣሪነት ይህርሸዌ ኧረጀታ ትኽሬሽ ኧግረታ ተበጠነም ይትፈቆሮኤ ፍራቀረም አረፍዌ ፓርላማ ባርጠማም ምርታዊአሁ የድነው የዘጓጋ በቤትመና ተደደቆም የብዒችት የጮኖያ ነፏጓም በጠነቀና ያጅየት ባኧማም በትከታ ያስብም

ዝረኽም ኣትሸሸደም ዱድቅየ ነመደም ኣትቀነሰም ጦናማ ተግረበጠም ተኰሺም ኣቐፓፐረም ያፎጦሪሽ ሰርሁም ንባህም ስነኽታ ተራመደም ባመሸ የሣሮቱ ተሸኰተንም ጠቀቀም ክናንሽ ይቐፐርየ ኻበረም ይኾነታ ኺፍረጥርቴዌ ጥረቄም አዘሰም እንኰረኰረም ኣሻረሐም ታትምረም ኣትስራናም ይደሞድሚደቱዝ አትርፍ ወሚደም ኢፎነም አትጋሙ ያገባይ ዝነፈም መሸታ አትቀተ ባሻኛ ይፍተታ እረብር ኣዳነገም ዴነሸሬም እንቐሶት ስቀረም የባርኖ ተድራጀም ፈጠመም ተኽረህ ገሬኖችም ባንኽሬሽ

Appendix iv: Guragegna Phrase words After stem

አትያነ ያነ አትያነች ያነች ያነችም ንሸያ ሸያ ሸያም ዋርኹም ቄርስ ስጭኔ ረሣናም በመሰሮ መሰሮ መሰሮም ኽነሼ ስረተተ ኣጣጣም ታተረ ቀነስ ቀነስና ጠበጠ ቀነስኹ ባናም ያቅቀረብሁ ቅቀረብሁ ያቅቀረብሁ ቅቀረብሁ ቅቀረብሁና ኣትጉዘገዘ ጉዘገዘ ጉዘገዘም ንዲመገቡ ዲመገቡ ናጐረ ቀነፈ ፈርሑፑ ተቻጨም ቻጨም ቻጨምዌ መሠረ አፍጣጠረ አፍጣጠረች ስብር ኣርገነገ ፍሬ በርዴ ኣዝራበጠ አምባ የገሳሙ ገሳሙ የገሳሙ ገሳሙ ገሳሙና ያርዋኽ ርዋኽ ርዋኽታ ዋራኝ ኣዳራ የዋሪነ ዋሪነ ዋሪነታ ኧበር አንሸው ሸው ሸውሸ ታናጠፉ ዛንዜተ ሸከልሊት ግመገመም የፈትኽ ፈትኽ የፈትኽ ፈትኽ ፈትኽና ዋነሶ ኣጠሳ ኣጠሳች ታጠፍ ኣድሁ ረከበም እንቐሰ ንቐሰ ቐሰ ንቐሰም ቐሰም የባርጠ ባርጠ ባርጠማ ላዴ ኣበራ አወረወ ዝዋጠረም በር ር ርማ ሬፐሪች ደናኽ የሙገመ ሙገመ ሙገመነ ጎበ ጎበና ትራማ ራማ ራማሽ ስጭየው የጉራ ጉራ ጉራማ ተሙገ ሙገ ሙገማ ኣሐበደ ይረሕ ረሕ ረሕየ ከሬ ትዐሪ ዐሪ ባጀትኡ ኣማነ አንገተሸ ገተሸ ገተሸታ ታኻን ገፖን ገፖ ኣዛገገ አፎሩን አፎሩ አህ ነገው አትያነ ያነ ያነም ቀለጠች ዣነረም ይረግ ረግ ረግም የህር ህር ህርም ማደድ ጠርጠርተ ባሪ ዋርሁ በበርሁም ገፓ ገፓች አጎዴ በዛቢ ያትዋና ትዋና ትዋናኖ በቀና ቀና በቀናች ቀናች ቀናችም እንቕብን ንቕብን ቕብን ንቕብንየ ቕብንየ ነቀመ ጥንቆቅ ኣንግራሳ የአሰራ አሰራ አሰራሽ አሸቼ ይደውርፑ ደውርፑ ደውርፑም ረፐራ ሰጡትን ኖችም ትጫወዶ ኣበሰረ ሕብር የዘር ዘር ዘርማ ዳናም አትነቕ ነቕ ነቕም ኣዥባፐረ ደቦሐረ ፍጨጨ ኢ_ኦቕ መሰረን መሰረ መሰስመ ሙትንያ አገጐ በጎቆ ጎቆ ጎቆታ ንጨጨ ጨጨ ጨጨም ትከኣዌት ሣረናም ትጠራ ጠራ ጠራሽ በብረ ብረ ብረዌ ኢደገ ኢደገች ሾኽኖ ምፍመኝ ዴንቅ ፌለቀ ሰተበ ካር ጣፍሁምመጣፈ ኣደፈፈ ድረነገም ድምድዳድ ጀግጀባረ ጣሰ ሔንደ ኦጅየት ኤስ ኣቁራጠበ ድርስየ ድርስየ አሸች አሸችና ገፈሪ መደሬ ኣድዋነ እትቕነወችሬ ትቕነወችሬ ትቕነወችሬም ኣኽ ርዋሴ ዛሮከንም ዳረን ዳረ ኣጠዊ ኣጠ የጉራጌ ጉራጌ ጉራጌም መሰረ ቅጥ ዝፈዜ ትዕባበረ አትባቶ ባቶ ባቶም ያቀረ ቀረ ቀረቹ ነገሰ ወበረኽ ኤፈ አሸነትው ጫረም ሙትትባረ ረመደም ትሠራ ሠራ ሠራኒ ቀነሶ ቐጠረ ጉርማስ ኤጠመ ከሰረ ንከከሰ ከከሰ ከከሰም ራጨም ሁት ገገ ጥባነረ ጥባነረች ረፈቀም ግረተመ ይወጥቅ ወጥቅ ይወጥቅ ወጥቅ ወጥቅና ደበረ በኻቆ ኻቆ ኻቆታ ቀርሴ ቄቄረም ራሐም ኤርቀኡ እየቋጠብ የቋጠብ የቋጠብሸ ገግመ _ኣከሙ እርኻ ርኻ ርኻም ኤወጣ ሜናዥቴ ብረሸ ረሸ ረሸም ሎአቸ ስሓነም ውርምየ ውርምየ ረብር አር ድረከረ ኣጨነዘ ሳደደም ኣትመመረ መመረ መመረም ነፑሪ ኣወረ ያንዛከ ንዛከ ንዛከሸ ወገበ _አለ ሕማቸም ብዓዝረ ዓዝረ ዓዝረታ ህርሸ ትዕክት ረዘ የቄድማ ዌድማ ዌድማም መጨቀ ጋዋኣመነ ዳገሬ ኣበታ ደረርባረ ቅየጓደም ባረቼ ባጎፌ አፍጣጥር ቀጠ ኽርወ ጐዛርመ ኣማነደ ማድት ውኻትተናካ ጐየዋ ኩርታዊ ትሸ ሸ ሸታ ኣትቕነበ ቕነበ ቕነበም ዝፐረራ ሣሰው ባረቅ ኣፈቀረ ጠነቀ ርዋኽ አትባተርኽ ባተርኽ ባተርኽም ኣትክባበሰ ክባበሰ ክባበሰም ቴለ ጠና ተበግዳ በግዳ ተበግዳ በግዳ በግዳና ዘንገረ ኑባሬ ጨጨም ወርድ ይትረገፍ ትረገፍ ረገፍ ትረገፍየ ረገፍየ ሰላድ ጥበጠበም ገጎ ዣንዣፑኤፐ ትቅነህ ኣትሸበ ሸበ ሸበም አያ ሁሌት ቅራመደ ወርከ

ግረናው ቄምሁጉ ቀነሰ ወደረሞ ፍትፍቱ ሸረ ሙኽ ሰረች ኣድፋፈረ በትፈጠር ትፈጠር ትፈጠርሁ ጡሪነተ ጡሪነ ወሸለ ኣተከሰ ዝጉባረ ቐነትሞ ኘኸተም ሃብኘ ነዝህነው አረነጠ እንኽፈኸፈ ንኽፈኸፈ ኽፈኸፈ ንኽፈኸፈም ኽፈኸፈም ባርኸ ኽረቐ ከቺ ኼ ኼች ንደሕሬቸ ደሕሬቸ አሰናኘ ኸርኸ ይውር ውር ውርም የድረ ድረ ድረኒ ከድል ሣሬም ኤፉጀ ቢጠርቅ ቢጠርቅ ሸርፌ እሞበና ሞበና ሞበናነ ይትቐማመጥ ትቐማመጥ ቐማመጥ ትቐማመጥየ ቐማመጥየ ድብረማጐ በንጠም ኣሴቀ በፋዘ ፋዘ ፋዘታ ትራማ ራማ ራማታ ቁና ያጋ ጋ ጋታ ኣሑራመጠ ከፓም ኣብሳካ ብርኽ ድረተረ ወኽት በርዝባማና ርዝባማና በርዝባማና ርዝባማና ርዝናባማና ራፎተም የፓ ፓ የፓች ፓች ፓችም ሳነፉም በበርዛ በርዛ በበርዛ በርዛ በርዛና ጥለየን ጥለየ ዝረበጠ ቦጐ ነበንጦ ቦርጭ ኣትራታ ራታ ራታም ቓጠም ተቸነ ቸነ ቸነሸ ሬነውያ ጉመመጠም ስቀሰቀ ቀፑሪም ሞራወ ሚታኝ ባናቅቆውነ ቅርጥና ገፍርጦ ትገፍርጦ ለጀ ኘሸፑትን ኻርሁን ኻርሁ ኽረምቆ አገኼ ንድህ ድህ እመ መ መታ ጎፑረረ ያወርድ ወርድ ወርድሸ ተነፈ ኣቄዘሪ ከጠላኝ ወንደ ነጌሮ ስክትን ገግመኽነ ገግመኽ ጠፋጦ ኣትየጠረ የጠረ የጠረም ፈርወንቀርቅር ምስጉን ሶሬ ዘንጋ ሻባፐርሁ ኤንግድኹ ኣትረሳ ረሳ ረሳም ናችን ፖችህ ኣትኸታ ኸታ ኸታም ንፉር ፉር ተደፐወ ደፐወ ደፐወሸ ዛመደም ይድድሁም ድድሁም ድምድሁም ፈፍት ኑሩሣ ጫርኝ ቃው ወምበር ቆርጥን ዘርጋት ትሸሎ ሸሎ የትሣሩ ትሣሩ ትሣሩኖ ትርኽ ብረሜ ትብረሜ ትፈተፈ ፈተፈ ፈተፈም ፍራተሐም ነበረ ፈሬ አቅመሁ አቅመሁ ተደንታ ደንታ ደንታንዳ ዮሮት ምበሮች ኣትረደፈ ረደፈ ረደፈም ጋሙመ አፈኖደ ሙትትን ቴቅሸ አስኽማ ኽማ ማኽማ ትኘትር ንቅርሻ ቅርሻ ኣጋ ኧሰጦ ነፐሮ ቀጠረ ንፈዐ ፈዐ ወጣዓ ሔታ ኟረም አወናችን አወናች ራጫሙት ጭማማም ኣከው ቀኖ ቀኖች ዛበተ ፈዘ ኤፈዘ ኩርስ ባድራ ብራሸ ባፈ ባፈና ደርሸዮ ፍረነመ ይጠፍ ጠፍ ጠፍየ ጠብጥ መርዛ እወጣ ወጣ ወጣም ኣመሸ ኣጠ ኣጠና ይኘራ ኘራ ኘራዌ ካርሁን ኣቸከረ አወናቐ እምበ በ ምበ በኖ ምበኖ ምስን ኣንጥረጠረ ጠቅር ቃጠረም ትኽን ኽን ኽንዌ ተኧመ ኧመ ኧመታ ይፍጥር ፍጥር ፍጥርሸ ነቓነሶ ገድኽን ዘርኋት ቸናሁ ፈቆሬም አገኤ ዴሪው ቀነሙንን ቀነሙን ኑቆንቦ ደግወ ፈንጠናኽ ጐነን ብረት የባረዊኧ ባረዊኧ ባረዊኧታ ከሰሰመ ቶሬተ ኧኧዛ አርዋኽ ንኽ ኽ ኽዌ ዘነረን ሴተረ የበርህማ በርህማ በርህማሸ ቅመጠረም አመርዕወ ጠናቃ አወሬ ዋርጋ ቴገረ ትውነ ውነ ውነታ ኘኸረ ሸርት ቤዛር ስፐረ ስፐረች ቴክር ማጥኸማን ዝራነፈም ኣነበ ተቸሁ ቸሁ ቸሁኒ ኣትየጠበ የጠበ የጠበም በኘዋ ኘዋ ኘዋታ ብትሰ ሰ ትሰ ብትሰ ሰ ትሰ ሰና ትሰና ባኼን ቀርስ ነበር ገገዘም ኩረኬመ ኟመም ኣገረ አማት ትቐቸውን ንቀያ ቀያ ይረጥር ረጥር ረጥርየ ጠረችኔ ዣፐፐሮ አረቄ ፈቀርሾፐፐንደም ሰክት እስቀሰቀ ስቀሰቀ ስቀሰቀም ሰራ የሪክታ ሪክታ ታሪክታ ያኑራነስ ኑራነስ ያኑራነስ ኑራነስ ኑራነስና ጀፐራ ኣረደን ኣረደ እሸ ሸ ሸታ የ የች ችም አሸችንበስቀርጹንም አሸችበስቀርጹንም ነበት ስራሰረች የበና በና የበና በና በናና ዴራረኽ ፍጠፈጠ አትዋ ዋሸ ሰን ትሰን ፈንደ እን ን እ ንም ም የሶሬ ሶሬ ሶሬታ ስራመጠም ታቱን ከስመ ንችን በርስ ሸካክተ ዐቅኑጣም ሞብቤውን ቤቱ ሐራትየ ሐራትየ ቡያ ምቡያ ረዝዝ ስብረወ ስገረ መነገ ኣትሙጨጨ ሙጨጨ ሙጨጨም የአበራ አበራ አበራሸ ባኾም ኣራነገም ሙባረ ቀፐረ ያማራ ማራ ያማራ ማራ ማራና ድራረበም ውርክ ደቅቤቴ ቃሮንሁ ብሮንቴ ይወር ወር ወርማ የሁኤችመ ሁኤችመ ሁኤችመንዳ ውረመህነ ውረመህ ግስት ሜራ አነሰ አነሰች ጀባባረ ያድር ኣቴሸ እጥወተ ጥወተ እጥወተ ጥወተ እጥወ ጥወ ጥወተና ጉንጫኽ ኤጓድ ትርዋሸፑ ርዋሸፑ አየው ለሸኺ ርቃቀጠም የዝር ዝር ዝርማ ታቅኽር ባርጠ ኸማው አመሸ ምሱቄንየባረ ይንገድሸሁ ንገድሸሁ ይንገድሸሁ ንገድሸሁ ንገድሸሁና ቅረርየ ቅረርየ ሓበራ ሰኽራ ጓትረሳች ጅየት ዮባ ንጐቻ ጐቻ ያባቱ ባቱ ባቱታ አን ሸ ኣጫመጨም ኣቴመረ ቐፐረም ምሣሬ አነረ አግደቼም ያተሁት ትባር ባር ባርሸ ያትያነወ ስናወ አቴብ አሸ አሸና ትዘፐር ዘፐር ዘፐርም ሬፐሬ ገኸማ ገደደ

ወገ ቄርሽፔ ቢያሻ ኣፍጣጠረ ናበዓ ኣጭራነመ የዛ ዛታ የአሳንዶ አሳንዶ አሳንዶም ያና ና ናታ ታትጂፕር ኣሓጨ ቃርሸን ይዝከ ዝከ ዝከማ ጠርቄ ኣቅረበበ አፈጠረ ጥባነረም ኣመከረ ቶሬዮሁ ቶሬዮሁ ይዦር ዦር ዦርየ ንብር ፍረቀረ ድረዘዘ ኖበመደረ ኣጩገ ቀጥረንም ኤንቃርሸ ጠረዘ ኣመሰተ ቀንጠ ቀንጠች ዦቀረ ራነስኙሞ ነወነ ኸና ኸናች ኣሐነቀ ክርኽ ጋባረ ጀኰረ ነግረ ጐናቃ ብራሂ ራሂ ራሂም ረሙድን ናጠ ናጠች ሃራቼ ምጎረተ ፋት ርትም ትራክቦጮ የስ ስ ስማ በቤታህ ቤታህ ቤታህማ ጉኑር እኻር ኻር ኻርሸ የረፐ ረፐ ረፐነ ትራመ ራመ ያትቋ ትቋ ያትቋ ትቋ ትቋና ጋውነቱ ኣገረከረ ቆጥኘ ዝመረረ ወንዝብ ሠሩናኽ ገፓሁ ኢጋደኖ ድየተረሣ ድየተረሣች ኣገፖኖ ሬሣናቱም ኣገባሸ ትገዳሮ ሣርችናም ዝረኮ ነዌባ ፌሰም ብኒ' ኣገነፈ ኣንካካ ንፋጨ ፋጨ ኣደበተ ኣሾችኖ ኣሕ ኣደረ ደቀዴ ኣረኽ ምላ ሸነትን ሞበርሸ ዝከ እንፉ ንፉ ፉ ንፉየ ፉየ ቧሬን ቧሬ ቅረጠመ አንኽሬ ኽሬ ኽሬም ውግ ኧራይኣኽ ሚርመ ፈዘዘ ኣገነፈ ኣንካካ ንፋጨ ፋጨ ኣደበተ ኣሾችኖ ኣሕ ኣደረ ደቀዴ ኣረኽ ምላ ሸነትን ሞበርሸ ዝከ እንፉ ንፉ ንፉየ ፉየ ቧሬን ቧሬ ቅረጠመ ብትስ ስ ትስ ብትስ ስ ትስ ስና ትስና ቕባፐረም ኘው ኣብረከሰ ጋመደም ግዘነ ገነፈ አባመኝ ዛህርታዊውየ ቀድራከትኹ ይረብኧ ረብኧ ረብኧታ ገድዬ ሞበር ቅንበ ኩምንየደ ናጥሁ ያቅሳብር ቅሳብር ቅሳብርሁ ኣትራነሰ ራነሰ ራነሰም ኣሳደደ ትፈጠር ፈጠር ፈጠርሁ እበር በር በርም ኣገፓ አህማ ሶሬሣ ስረግ ባሸጉየ ባሸጉየ ኣረካ ትክ ክ ክነት ኣጥዋሐረ ራውሬ ያርብ ርብ ርብም አኽ ፌጠጠ ብጠብጤ ትራ ራ ራሽ ትሁን ሁን ትሁ ሁ ሁንም ፍረጮኽዬ ትሸነሩን ዳሩ ዘብርሂ ጦሙት ኣፋነሰ ጠፉሁ አፍልቀረን አፍልቀረ የርማጅ ርማጅ ርማጅነት የጦጨሞ ጦጨሞ ጦጨሞታ ቴጠጠ ጸመ ጸመች ማጣሪ ይህርሸ ህርሸ ህርሸዌ ኧረጆ ትኽሬ ኽሬ ኽሬሽ ኧግረ በጠነም ይትፈቆሮ ትፈቆሮ ፈቆሮ ትፈቆሮኤ ፈቆሮኤ ፍራቀረ አረፍ ፓርላ ባርጠማ ምርታዊ ምርታዊአ ድነው ዘጓጋ በቤትመ ቤትመ በቤትመ ቤትመ ቤትመና ደደቆም ብዒቾት ጮኖያ ነፏጓ በጠነቀ ጠነቀ በጠነቀ ጠነቀ ጠነቀና ጅየት ባኧማ በትከ ትከ ትከታ ያስብ ስብ ስብም ዝረኸ ኣትሸሸደ ሸሸደ ሸሸደም ዱድቅ ነመደ ኣትቀነሰ ቀነሰ ቀነሰም ጦና ግረበጠም ኩሺም ኣቾፓፐረ ያፎጦሪ ፎጦሪ ፎጦሪሽ ሰርሁ ንባህ ባህ ባህም ስነኽ ራመደም ባመ ሣሮቱ ሸኩተንም ጠቀቀ ክናን ይቾፐር ቾፐር ቾፐርየ ኻበረ ይቾነ ቾነ ቾነታ ኧፍረጥርቴ ጥረቄ አዘሰ እንኰረኰረ ንኰረኰረ ኰረኰረ ንኰረኰረም ኰረኰረም ኣሻረሐ ታትረም ኣትስራ ስራ ኣትስራና ስራና ስራናም ደሞድዒደቱዝ ርፍ ወዒደ ኢፎነ ጋሙ ገባይ ዝነፈ መሸ ቀተ ባሻ ይፍተ ፍተ ፍተታ ረብር ኣዳነገ ዴነሸሬ ንቖሶት ቖሶት ስቀረ የባር ባር ባርኖ ድራጀም ፈጠመ ኽረህ ገሬኖ ገሬኖች ባንኽሬ ኽሬ ኽሬሽ

Appendix v: Guragegna Cheha alphabet

	ä [ɐ]	u [u]	i [i]	a [a]	e [e/ɛ]	ë [ɨ]	o [o/ɔ]
h [h]	ሀ hä	ሁ hu	ሂ hi	ሃ ha	ሄ he	ህ hë	ሆ ho
l [l]	ለ lä	ሉ lu	ሊ li	ላ la	ሌ le		ሎ lo
m [m]	መ mä	ሙ mu	ሚ mi	ማ ma	ሜ me	ም më	ሞ mo
m [mʷ]	መዉ mwä		ማዊ mwi	ማ mwa	ማዌ mwe	ምዉ mwë	
r [r]	ረ rä	ሩ ru	ሪ ri	ራ ra	ሬ re	ር rë	ሮ ro
s [s]	ሰ sä	ሱ su	ሲ si	ሳ sa	ሴ se	ስ së	ሶ so
š [ʃ]	ሸ šä	ሹ šu	ሺ ši	ሻ ša	ሼ še	ሽ šë	ሾ šo
ḳ [k’]	ቀ ḳä	ቁ ḳu	ቂ ḳi	ቃ ḳa	ቄ ḳe	ቅ ḳë	ቆ ḳo
ḳy [cʲ]	ቇ ḳyä	ቈ ḳyu	ቈ ḳyi	ቃ ḳya	ቄ ḳye	ቅ ḳyë	ቈ ḳyo
ḳw [k’ʷ]	ቈ ḳwä		ቊ ḳwi	ቋ ḳwa	ቌ ḳwe	ቍ ḳwë	
b [b]	በ bä	ቡ bu	ቢ bi	ባ ba	ቤ be	ብ bë	ቦ bo
bw [bʷ]	በዉ bwä		ቡ bwi	ቧ bwa	ቧ bwe	ብዉ bwë	
v [β~β]	ሸ vä	ሹ vu	ሺ vi	ሻ va	ሼ ve	ሽ vë	ሾ vo
t [t’]	ተ tä	ቱ tu	ቲ ti	ታ ta	ቴ te	ት të	ቶ to
č [ʧ]	ቸ čä	ቹču	ቺ či	ቻ ča	ቼ če	ች čë	ቾ čo
x [x]	ኀ xä	ኁ xu	ኂ xi	ኃ xa	ኄ xe	ኅ xë	ኆ xo
xw [xʷ]	ኈ xwä		ኈ xwi	ኋ xwa	ኋ xwe	ኈ xwë	
n [n]	ነ nä	ኑ nu	ኒ ni	ና na	ኔ ne	ን në	ኖ no
’ -	አ ’ä	ኡ ’u	ኢ ’i	ኣ ’a	ኤ ’e	እ ’ë	ኦ ’o

	ä [ɐ]	u [u]	i [i]	a [a]	e [e/ɛ]	ë [ɨ]	o [o/ɔ]
k [k’]	ከ kä	ኩ ku	ኪ ki	ካ ka	ኬ ke	ክ kë	ኮ ko
ky [cʲ]	ኸ kyä	ኹ kyu	ኺ kyi	ኻ kya	ኼ kye	ኽ kyë	ኾ kyo
kw [kʷ]	ኰ kwä		ኵ kwi	ኳ kwa	ኴ kwe	ኸ kwë	
xy [ç]	ኸ xyä	ኹ xyu	ኺ xyi	ኻ xya	ኼ xye	ኽ xyë	ኾ xyo
w [w]	ወ wä	ዉ wu	ዊ wi	ዋ wa	ዌ we	ው wë	ዎ wo
z [z]	ዘ zä	ዙ zu	ዚ zi	ዛ za	ዜ ze	ዝ zë	ዞ zo
ž [ʒ]	ዠ žä	ዡ žu	ዢ ži	ዣ ža	ዤ že	ዥ žë	ዦ žo
y [j]	የ yä	ዩ yu	ዪ yi	ያ ya	ዬ ye	ይ yë	ዮ yo
d [d]	ደ dä	ዱ du	ዲ di	ዳ da	ዴ de	ድ dë	ዶ do
ǧ [ʤ]	ጀ jä	ጁ ju	ጂ ji	ጃ ja	ጄ je	ጅ jë	ጆ jo
g [g]	ገ gä	ጉ gu	ጊ gi	ጋ ga	ጌ ge	ግ gë	ጎ go
gy [ɟ]	ጘ gyä	ጙ gyu	ጚ gyi	ጛ gya	ጜ gye	ግ gyë	ጞ gyo
gw [gʷ]	ጐ gwä		ጒ gwi	ጓ gwa	ጔ gwe	ጉ gwë	
ṭ [t’]	ጠ ṭä	ጡ ṭu	ጢ ṭi	ጣ ṭa	ጤ ṭe	ጥ ṭë	ጦ ṭo
č̣ [ʧ’]	ጬ č̣ä	ጭ č̣u	ጪ č̣i	ጫ č̣a	ጬ č̣e	ጭ č̣ë	ጮ č̣o
f [f]	ፈ fä	ፉ fu	ፊ fi	ፋ fa	ፌ fe	ፍ fë	ፎ fo
fw [fʷ]	ፈዉ fwä		ፈዊ fwi	ፈዋ fwa	ፈዌ fwe	ፍዉ fwë	
p [p’]	ፐ pä	ፑ pu	ፒ pi	ፓ pa	ፔ pe	ፕ pë	ፖ po
pw [pʷ]	ፑዉ pwä		ፑዊ pwi	ፒዋ pwa	ፒዌ pwe	ፑዉ pwë	