ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

AUTOMATIC THESAURUS CONSTRUCTION FOR TIGRIGNA TEXT RETRIEVAL

HAGOS HIETE

July 2011

ADDISABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

AUTOMATIC THESAURUS CONSTRUCTION FOR TIGRIGNA TEXT RETRIEVAL

A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Science

By

HAGOS HIETE

July 2011

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

AUTOMATIC THESAURUS CONSTRUCTION FOR TIGRIGNA TEXT RETRIEVAL

By

HAGOS HIETE

Name and signature of Members of the Examining Board

Name	**Title**	**Signature**	**Date**
_____	Chairperson	_____	_____
_____	Advisor(s),	_____	_____
_____	Examiner,	_____	_____

Dedication

The dedication of this thesis manuscript is to my family, for their love and devotion to the success of me throughout my life.

ACKNOWLEDGMENT

I would like to express my heartfelt gratitude and deep appreciation to my advisor, Mr. Ermias Abebe, for his guidance, patience, comments and encouragement throughout. This research would not have been possible without the guidance and technical support of my advisor. I am evenly grateful to all those who have contribution to my work for their valuable comments on my research. Thankful acknowledgement is also due to my fiancée W/t Rahel Hagos for her moral support. Also my appreciation goes to the School of Information Science in Addis Ababa University for giving me valuable knowledge throughout my study.

I wish to thank my family for their inspiration and encouragement during the entire period of my study. Also, thanks to all my friends who are everywhere and everybody who care for me, for their high spirit friendships and encouragement.

CONTENTS

List of Tables

List of Figures

Acronyms and Abbreviations

IR—Information Retrieval

NLP—Natural Language Processing

LSI—Latent Semantic Indexing

SVD—Singular Value Decomposition

SERA —System for Ethiopic Representation in ASCII

ABSTRACT

Thesaurus is a list of related terms, which helps to solve the vocabulary problem in information retrieval raised because authors and indexers use different terms for the same concept. Searchers may have no skill in selecting good search terms. They may use vocabulary for submitting a query that is different from the one indexed in the system. So, they may not get good results although there are some related documents in the collection. Therefore, it is reasonable to expand query terms with additional related terms drawn from a thesaurus.

Tigrigna is a language in the Ethio-Semitic family spoken mainly in Tigray region of Ethiopia and in Eritrea. Currently, the size of electronic documents in Tigrigna language is increasing significantly. Robust retrieval tools would therefore be needed in order to use these documents. As thesaurus is an important component of information retrieval, studies have been conducted on automatic thesaurus construction for Tigrigna Information Retrieval.

Even though automatic thesaurus construction has its own drawback, it is better than the alternative manual construction. In this thesis, an automatic approach to Tigrigna thesaurus construction from document collection based on term to term co-occurrence matrix is introduced. An encouraging result is obtained in the experimentation of the system on Tigrigna documents. The result on a random sample of terms shows that the system has accuracy of 75.28%.

CHAPTER ONE

INTRODUCTION

1.1 Background

A thesaurus (plural: thesauri) is a set of terms that are semantically related (Maziar, A. 2008). It helps to improve the quality of retrieval by guiding indexers and searchers about which terms to use. Thesaurus is a structure which supports the automatic indexing and retrieval, and it is a structured dictionary, which is focused on the representation of a limited set of semantic relations between different concepts (Robert, M. 2009). It is a valuable tool in Information Retrieval (IR), both in the indexing process and in the searching process, used as a controlled vocabulary and as a means for expanding or altering queries (Monica, L. 2002).

Tigrigna is a language in the Ethio-Semitic family spoken by 5-6 million people in the Tigray region of Ethiopia and in Eritrea. In Ethiopia, Tigrigna is one of the regional government office languages. It is significant for its level of development in literature, literacy, standardization, etc.. Tigrigna is the principal literal language and medium of instruction in primary schools (1 to 8) and regional government offices of the Tigray region. Nowadays, Addis Ababa University and Mekelle University has started to teach students Tigrigna Language in degree program.

Tigrigna is written in the Ge'ez script (Ethiopic script) common to other languages in the family. In spite of the relatively large number of speakers and its power of expressiveness for things, Tigrigna was a language for which very few computational linguistic resources had been developed for many years, and very little had been done in terms of making useful higher level Internet or computer based applications available to those who only speak Tigrigna. But now because of the rise of desktop publishing along with the arrival of the internet, web publications, the introduction of Unicode and its inclusion of the Ethiopic writing system, the writing system which Tigrigna uses, have resulted huge collection of electronic language resources. The volume of electronic documents is

increasing significantly from time to time. The issue at this time is how the information need matches with the information stored in information retrieval systems. Manually searching information from a huge collection of documents is tiresome, time consuming and difficult task. The solution for this problem is designing automatic Information Retrieval System (Baeza-Yates and Ribeiro-Neto, 1999).

Automatic information retrieval (IR) systems were originally developed to help manage the huge scientific literature that has developed since the 1940s. Many University, corporate and public libraries are now using IR systems to provide access to books, journals, and other documents (Kanyarat, L. 2003). A fundamental problem for information retrieval is mismatch of query (information need) of IR system users with the authors of the documents stored in IR systems due to use of different words to refer the same concept (Jinxi, X. 1997).

A thesaurus provides a precise and controlled vocabulary that serve to enhance retrieval performance in an IR system both in document indexing and document searching (Andargachew, M. 2009). With very large collection of documents, we have to reduce the set of representative index terms in order to reduce the computational costs. This can be accomplished through text operations. Text operations include five document processing procedures. These are Lexical analysis, elimination of stopwords, stemming, selection of representative terms and term categorization through thesaurus (Baeza-Yates and Ribeiro-Neto, 1999).

This study aimed to examine existing automatic thesaurus generation techniques and adapt them to the Tigrigna language. This is necessary because there are no papers in English that detail the construction of Tigrigna thesauri. With the thesaurus, a set of experiments were performed to measure its applicability and quality.

1.2 Motivation

A motivation to this research has come from the problem of searching resource using search engines because of terminological difference between searchers and content developers when browsing certain materials.

The concern of Information Retrieval (IR) is locating documents that are relevant for a user's information need or query from a collection of documents. Word difference or mismatch is the basic problem for information retrieval. A query is usually a short and incomplete description of the underlying information need. The users of Information Retrieval systems and the authors of the documents often use different terminology to the same concepts (Jinxi, X. 1997).

Similarly, recently there is huge number of different documents in Tigrigna language such as Tigrigna books, magazines, journals, official documents etc.. To better access those Tigrigna documents, from Information Retrieval systems, thesaurus generation is a vital solution.

1.3 Statement of the problem

Thesauri are significant and valuable tools of IR systems for enhanced term classification during indexing, and query expansions during searching. Thesauri are used in web applications, but the majority of them are manually constructed (Jos´e, R. and Lourdes, A. 2006). Manual approaches often suffer from the problem of low coverage (limits its availability to some particular topics) and high expense. In addition, a thesaurus normally requires to be periodically updated to include new terminology, in particular in modern terms, such as those related to Technology (Jos´e, R. and Lourdes, A. 2006).

One of the major problems in modern Information Retrieval is the vocabulary problem that concerns the representativeness between terms used for representing documents and the terms used by the searchers to describe their information need (Hayel, K. *et al*., (n.d.)). Besides the information overload, another challenge in information processing is

the vocabulary mismatch problem, referring to the fact that people tend to use different terms to describe a concept. Due to their different backgrounds and expertise, the chance that two people describe a concept using the same term is pretty low, this even may happen with a same person, he or she may use different terms to describe the same concept at different times because of the learning process and the evolution of concepts (Libo, C. 2006). One way of handling the vocabulary problem is using a thesaurus that shows the relationships among terms.

"Thesauri are used to standardize terminology and to provide alternative and preferred terms for any application" (Jos´e, R. and Lourdes, A. 2006). Professionals or content developers in the same field may employ different terminologies for the same thing. At this time thesaurus helps in making the terminology standard.

Therefore, the purpose of this research work is to conduct automatic thesaurus for Tigrigna text retrieval to assist in the development of an effective Tigrigna IR systems.

1.4. Objective of the study

1.4.1 General objective

The general objective of this research is to develop automatic thesaurus for Tigrigna text retrieval from document collection.

1.4.2 Specific objectives

To achieve the above general objective the researcher has the following specific objectives:

- Study literatures, techniques and tools applicable to thesaurus construction
- Investigate the terms of the language that pertain to thesaurus construction
- Compile a corpus for thesaurus generation
- Evaluate the performance of the thesaurus generation system using test datasets
- Provide concluding remarks and recommendations for future research.

1.5 Methodology

To achieve the goals of this study, the researcher followed the following methodology.

1.5.1 Literature Review

Review of literatures such as books, articles, journals and the Internet are conducted to investigate the principles/theories of the various approaches, techniques and tools that are employed in different areas and if relevant to this research work. Furthermore, features of thesauri and approaches to automatic thesaurus construction had studied. Based on the Literatures planned to study features and approaches had investigated and used for automatic thesaurus construction for Tigrigna text retrieval.

1.5.2 Development Tools

The Python programming language was used as a development tool because it is suitable for text manipulation. Further it is also available with GNV lisence which made it accessible for the research.

1.5.3 Data Preparation

A corpus was developed by gathering Tigrigna documents. In order to get the documents and some important information, domain experts are consulted and linguistic literature in the Tigrigna language had been reviewed.

1.6 Scope and Limitations of the study

There are several types of thesaurus construction methods suggested by different researchers for different languages. However, the scope of this research is construction of automatic similarity thesaurus for Tigrigna documents based on term-term co-occurrence value from document collection. It has included some stemming and stopwords elimination. The lack of a freely available electronic corpora, standard stemmer and complexities of Tigrigna orthography were some of the challenges to this research. Other limitations are the thesaurus terms are composed of single word terms and does not

included documents which have encoding different from Unicode encoding. In this research only affixations (specifically prefix and suffix) are dealt with, infix and other morphologies are not considered.

1.7 Significance of the Study

It is believed that the work will:

- Contribute to enhance Tigrigna Information Retrieval search engines.
- Make easy the interface between Tigrigna language users with content developers.
- Use for other researchers as a frame work for further research

1.8 Organization of the Thesis

This research is organized in to five chapters. Chapter one is an introduction to the work, where the problem statement, the motivation behind the research and the objectives are discussed. Chapter two presents concepts and approaches of thesaurus construction, including overview of thesauri, approaches, features, application of thesaurus, and related works. Features of the Tigrigna language and other related topics that are required for better understanding of the domain are also discussed in this chapter. Techniques and algorithms particularly applied in this work are presented in chapter three. Chapter four discusses the experiment conducted and results thereof. In chapter five conclusions and recommendation are provided

.

CHAPTER TWO

LITERATURE REVIEW

The literature review section deals with, purposes and uses of thesauri, features of automatic thesauri, approaches to automatic thesaurus construction, construction of vocabulary, stemming, conceptual models, similarity computation, and Tigrigna language with its writing system.

2.1 Overview of thesauri

A number of researchers have conducted thesaurus research on different languages and discussed on the uses, features and approaches of thesaurus. There are many different definitions of thesauri, varying from quite modest definitions that focus on the relations between words without stating which kinds of relations that are meant, to such definitions that state more exactly which relations that are concerned (Monica, L. 2002). Scholars (Karen, S. and Peter, W. 1997) defined thesaurus in two ways. The first is in terms of its function, a thesaurus is a terminological control tool used in translating from the natural language of documents, indexers or users (searchers) in to a "more constrained system language (documentation language, information language)". The second is in terms of structure, a thesaurus is a controlled and dynamic vocabulary of semantically related terms which covers a specific domain of knowledge.

Thesauri are intellectual creations and they are rich semantic networks of vocabulary terms. Thesauri as unique tools facilitate the society and access of information. Studying the current and potential application of thesauri is important; particularly as people increasingly search information systems from their comfort such as from own home, wireless network connections and other places without assistance from information professional (Jane, G. 2004).

A thesaurus is a structured collection of concepts and terms for the purpose of improving the retrieval of information. A thesaurus should help the searcher to find preferable search terms, whether the terms be descriptors (keywords) from a controlled vocabulary

or the multiple terms needed for a comprehensive free-text search all the different terms that are used in texts to express the search concept. Most thesauri establish a controlled vocabulary, a standardized terminology, in which each concept is represented by one term, a descriptor, that is used in indexing and can thus be used with confidence in searching; in such a system the thesaurus must support the indexer in identifying all descriptors that should be assigned to a document or other object in light of the questions that are likely to be asked (Dagobert, S. (n.d)).

A well-constructed thesaurus has been recognized as a valuable tool in the effective operation of an information retrieval system as expansion of queries with related terms using thesaurus can improve performance (Hazra, I. and Aditi, S. 2009). A good thesaurus provides, through its hierarchy better terms by associative relationships between concepts. It is a semantic road map and guidance for searchers, indexers and anybody else interested in an orderly grasp of a domain specific field (Dagobert, S. (n.d)). There are different approaches of indexing One of which is user-oriented indexing. In the approach of request-oriented indexing (or user-oriented indexing) the concepts to be included in the thesaurus are collected from actual and expected search requests. They are then organized into an easily grasped structure that serves as a framework or checklist for the indexer in analyzing objects or documents. The users have told the thesaurus builder what they are interested in and the thesaurus builder has organized these interests into a logical framework that communicates user interests to the indexer. The indexer can now consider these interests in analyzing documents, making sure that an object or document will be assigned all descriptors under which a user may want to find them. Request-oriented indexing requires a well-structured thesaurus; it depends on the semantic road map provided by the thesaurus. Request-oriented indexing starts with a hierarchical display, using the alphabetical display only for augmentation (Dagobert, S. (n.d)).

Thesaurus construction approaches are typically divided into manual approaches and automatic approaches. Though, some form of manual thesaurus construction is mandatory due to the relational complexities, semantic ambiguities, and dynamics, inherent in languages, manual construction and maintenance is complex and time consuming (Jesper, W. 2004). It is prone to contain errors and is hardly ever consistent;

furthermore, it must be kept up-to-date continually if it is to be of any use. A carefully crafted thesaurus can improve the effectiveness of an information retrieval system considerably and is therefore an important component. It can aid the researcher in formulating his queries more effectively and provide disambiguation of problematic terms. A thesaurus constructed automatically from a document collection is useful for two reasons: (1) it can include implicit knowledge about the domain contained in the documents; (2) it does not suffer from the problems of manually constructed thesauri. (Sebastian, M. K. 2004).

The attractive aspect of automatically constructing or extending lexical resources rests clearly on its time efficiency and effectiveness in contrast to the time consuming and outdated publication of manually compiled lexicons. Its application mainly includes constructing domain-oriented thesauri for automatic keyword indexing and document classification in Information Retrieval, Question Answering, Word Sense Disambiguation, and Word Sense Induction (Dongqiang, Y. and David, M. 2008).

A thesaurus is a prearranged list of terms, usually related to a particular domain of knowledge. Different professionals use different terminology to express the same thing. Thesauri are used to standardize terminology and to provide alternative and preferred terms for any application. They are especially useful in indexing and retrieving information processes, by providing the different forms which a concept can adopt. There are three basic relationships between thesaurus terms. The first is Terms related by the equivalence relationship have an equivalent meaning, in different senses (they are synonyms; one is the translation to the other, its archaic form, etc). The second one is in a set of equivalent terms, one of them, distinguished as the preferred one, and is used in the hierarchies and for indexing. Preferred terms are arranged into hierarchies with different numbers of levels. These levels go from the broadest type of term to the narrowest and most specific one. Finally, there can be associative relationships between terms which are not connected by a hierarchy, for example because they are narrower terms of different broad terms, but they still present some kind of relationship (Jos´e, R. and Lourdes, A. 2006).

Thesaurus classification in the development of information retrieval, including the rise of web search engines and the storage of Meta data, different kinds of lexicons or dictionaries were used to improve information retrieval approaches. One of this is the so called controlled vocabulary. A controlled vocabulary contains the normalized forms of all the words that are important for a specific domain. These collections of technical and well-defined concepts are normally created by specialists. In the opening of the process of enhancing a thesaurus it is mostly necessary to find concepts which are relevant for the thesaurus. The thesaurus could be improved by adding words from the document corpus. But obviously not every word that appears in the corpus has the same relevance for the thesaurus. A ranking method is needed to classify the most relevant concepts out of the documents (Robert, M. 2009).

2.2.1 Purposes and uses of thesauri

Information Retrieval systems are specially designed for the purpose of searching relevant items with respect to an information need of a user. This would be ideally realized by a system that understands the inquiry of the user as well as the content of the documents in the document collection. The search is usually based on the searchable items which are usually terms. The process of mapping documents to term representations is called indexing. In most retrieval systems, the index terms are all words in the documents with the exception of stopwords like, determiners, prepositions and conjunctions. A search request (query) then consists of terms, and the documents in the result set of the retrieval process are those that contain these query terms. Most of the works in retrieval research in the previous decades have been concentrated on refining this term based search method. One of these refinements is to use a thesaurus. A thesaurus in the field of information and documentation is an ordered completion of concepts which serves for indexing and retrieval in one specific domain. A central point is not only to define terms but also relations between terms (Gerda, R. (n.d)). In Information Retrieval thesaurus describes a certain knowledge domain by listing all its main concepts and semantic relations between them. In their simplest form thesauri consist of a list of important terms and semantic relations between them. Thesauri have been used in documentation management projects for years. Manual thesauri were even

used by libraries and documentation centers long before the computer era. This long tradition and the more recent success of the thesaurus based information systems has led to adoption of thesaurus-based techniques by industries and to the development of international standards. The purpose of thesaurus is to provide help for users in time of searching and to improve the building of the query. In fact, this can improve the perception of the system answer to the user.

The selection of search terms for query formulation and expansion is a challenging task within the information search and retrieval process. Two general approaches have been adopted in studies on search term selection which are system-centred and user-centred. The system-centred approach is represented by work on algorithms and evaluation based on the traditional IR model, a model that fundamentally ignores the users and their interaction with the system. In contrast, the user-centred approach focuses on the cognitive, interactive, and contextual aspects of IR and considers users, use, situations, context, and interaction with system (Ali, A. S., *et al*., (n.d.).

A thesaurus is a sort of terminological base: it is a collection of terms, plus a set of relations among them and designed to support the user in creatively selecting vocabulary. This thesaurus is presented to the user so the user can choose terms among it and use to retrieve potentially relevant documents from large collections. If these terms are extracted from the actual corpus, they can be used into the query. Structuring the thesaurus can help the user finding the right term in a given domain (Jean-Pierre, C. (n.d.)). Thesaurus mainly uses in time of Indexing and Searching. In Indexing, thesaurus uses to select the most appropriate thesaurus entries for representing the document. In Searching, it uses to design the most appropriate search strategy. Which means if the search does not retrieve enough documents, the thesaurus can be used to expand the query; if the search retrieves too many items, the thesaurus can suggest more specific search vocabulary.

Designed for vocabulary control and knowledge representation, a concept structure is a systematic organization of important vocabularies, which usually contains a finite set of carefully chosen concepts and terminologies, and some kinds of relationships between

these concepts and terminologies. According to the relationships, the vocabularies may be further organized in structures of different forms for easier access (Libo, C. 2006).

A thesaurus is used to standardize terminology and, as a result, allows the recording of information in a consistent and retrievable way. It provides the user with a single preferred term to use where there is a choice of terms with the same or similar meaning. By standardizing information that is entered onto a database it is easier to search records and retrieve the required output. The use of a thesaurus allows the retrieval of information created by someone else, it also allows users to access and retrieve data. The simplest way to ensure that the information is consistent is to use a wordlist. This is simply an alphabetical list of representative terms used to control the information in the collection of documents (corpus). A wordlist by itself does not allow the user to create relationships between the terms. Using a thesaurus can greatly aid the indexing and retrieval of information. Even though its development is complex, a thesaurus offers many advantages in comparison with a list of keywords. For instance, the efficiency of the selection of terms is improved and recurrent data are eliminated by the hierarchical and associative structure. A thesaurus also allows a group of users to make use of a similar system of indexation whatever the level of precision their research may require (Phil, C. 2011).

Thesaurus is one of the major ways of providing searchers with terminological support in the query formulation and expansion. A thesaurus consists typically of controlled vocabulary, which represents the semantic relations between the terms. A searching thesaurus is developed for supporting users in query formulation and expansion by suggesting additional terms, synonyms and narrower terms (Anne, S. and Pertti, V. 2004).

Information retrieval is the key technology for knowledge management which guarantees access to large corpora of unstructured data. Modern information retrieval systems use a range of statistical and linguistic tools to scale up the effectiveness of searching textual documents. One of the main problems that characterize natural language texts is the fact that a single word may occur in some (or many) different forms, as determined by the

language's inflectional and derivational morphologies. This is a problem for IR systems even in a language like English that has a relatively simple morphology; it is a much greater problem for languages with more complex morphologies that can yield hundreds or even thousands of variants from a single word (Nega, A. and Peter, W. 2003). From this point of view thesaurus construction has a vital role to scale up the effectiveness of searching textual documents.

In general (Karen, S. and Peter, W. 1997) have grouped the major purposes of thesaurus in to seven.

1. To present a map of a given field of knowledge, including how concepts or ideas about concepts are related to one another, which helps an indexer or a searcher to understand the structure of the field.
2. To provide a standard vocabulary for a given subject field which will ensure that indexers are consistent when they are making index entries to an information storage and retrieval system.
3. To make available a system of references between terms which will ensure that only one term from a set of synonyms is used for indexing one concept, and that indexers and searchers are told which of the set is the one chosen; and to provide guides to terms which are related to any index term in other ways, either by classification structure or otherwise in the literature.
4. To give a guide for users of a system so that they choose the correct term for a subject search; this stresses the importance of cross-references. If an indexer uses more than one synonym in the same index- for example, "abroad", "foreign", and "overseas"- then documents are liable to be indexed randomly under all of these; a searcher who chooses one and finds documents indexed there will assume that he has found the correct term and will stop his search without knowing that there are other useful documents indexed under the other synonyms.
5. To set new concepts in a scheme of relationships with existing concepts in a way which makes sense to users of the system.

6. To supply classified hierarchies so that a search can be broadened or narrowed systematically, if the first choice of the search terms produces either too few or too many references to the material in the store.
7. A desirable purpose, but which it would be premature to say is being achieved, is to provide a means by which the use of terms in a given subject field may be standardized.

2.2 Features of automatic thesauri

2.3.1 Co-occurrence analysis

Term co-occurrence analysis is an approach used in IR researches for forming multi-phrase terms. Lexical co-occurrence analysis is closely related to Latent Semantic Indexing (LSI). In LSI, document-by-word matrices are created and processed by Singular Value Composition (SVD) instead of word-by-word matrices. The nature of differences between lexical co-occurrence and LSI are of technical and conceptual nature. Technical nature, deals with time complexity. LSI analysis is more time consuming than SVD, LSI analysis does not have strong relation between time complexity and corpus size as Lexical co-occurrence analysis. Conceptual nature, concerns with the fact that the lexical co-occurrence approach uses term representations independently, whereas in LSI term representations are only used to compute document representations (Monica, L. 2002).

2.3.2. The concept space approach

A concept space is defined as a network of terms and their respected weighted associations which can represent concepts (terms) and their associations. The association value between two terms is a quantity between 0 and 1; this value is Computed from the co-occurrence of terms from a given document collection and it represents the strength of similarity between the terms. The association is minimum 0 and maximum 1. Zero association between two terms means the terms have no similarity. It is because the terms never co-exist or co-occurred in a document. If there is nearest to 1 association value

between term one and term two, based on the idea of concept space, the terms are highly related in the document collection (Felix, C., *et al*., 2002), (Monica, L. 2002).

2.4 Approaches to automatic thesaurus construction

Thesaurus can build all types of relationship that exist between words, such as hierarchic, synonym, and morphological relationships (Nurazzah, A., *et al*., 2010). It is a set of concepts in which each concept is represented with at least synonymous terms, broader concepts, narrower concepts, and related concepts (Yousef, A. and Fernand, V. 2002). In other words thesaurus is a kind of dictionary that defines semantic relationship among words. Automated thesaurus dictionary construction (machine-understandable) is one of the most difficult issues, though its effectiveness is widely proved by different research areas such as natural language processing (NLP) and information retrieval (IR). Thousands of contributors have spent much time to construct high quality thesaurus dictionaries in the past. However, since it is difficult to maintain such huge scale thesauri, they do not support new concepts in most cases. Therefore, a large number of studies have been made on automated thesaurus construction based on NLP. However, issues due to complexity of natural language, for instance the ambiguous/synonym term problems still remain. There is still, a need for an effective method to construct a high-quality thesaurus automatically avoiding these problems (Kotaro, N., *et al*., 2007). In general, building manual thesauri requires a lot of human labor from linguists or domain experts and they are expensive to build and there is high demand of automatic thesaurus construction.

Automatic thesaurus construction is an extensively studied area in information retrieval. The original motivation behind automatic thesaurus construction is to find an economic alternative to manual thesauri. Almost all automatic thesauri are based on the so called association hypothesis, which states that words related in a corpus (collection of documents) tend to co-occurrence in a corpus (Jinxi, X. 1997). There are three approaches to construct automatic thesaurus. The first approach is designing a thesaurus from document collection, which uses a collection of documents as the source for thesaurus construction. Second is merging existing thesauri, when two or more thesauri

for a given subject exist, that need to be merged into a single unit, and the third is based on tools from expert systems, thesauri are built using information obtained from users (Yousef, A. and Fernand, V. 2002), (Nurazzah, A., *et al*., 2010).

2.4.1 Thesaurus from document collection

Automatic thesaurus from document collection is generated by computing the co-occurrence values between domain-specific terms found in a document collection. These co-occurrence values are derived from the term frequency and document frequency of the terms (Schubert, F. Siu, C., *et al*., 2000). (Yousef, A. and Fernand, V. 2002) This is a standard to designing a thesaurus from document collection. In this approach the idea is to use a collection of documents as the source for thesaurus construction. Designing a thesaurus from document collection assumes that a representative body of text is available. Using statistical or linguistic procedures, it is possible to identify vital terms as well as their significant relationships.

2.4.2 Merging existing thesauri

Merging existing thesauri is appropriate when two or more thesauri for a given subject exist, that need to be merged into a single component. If two or more existing thesauri are available, merging is likely to be more efficient than producing the thesaurus from scratch or clean sheet (Yousef, A. and Fernand, V. 2002).

2.4.3 Thesaurus based on tools from expert systems

Thesaurus based on tools from expert systems approach is used to build automatic thesauri using information obtained from users. For example, if two terms of query of a user used to retrieve systems are combined by OR these terms are probably synonyms. Most of the term pairs may either morphologically similar like “net” and “network” or translations of each other (Yousef, A. and Fernand, V. 2002).

2.5 Construction of Vocabulary

In Information Retrieval, often distinction is made between controlled and uncontrolled vocabularies. Uncontrolled vocabularies allow for every token in a document to be a potential index term, without giving consideration to word form and other linguistic features. Controlled vocabularies on the other hand, have rules that regulate which words that are allowed to be index terms, as well as the word forms and other specific features of those terms. A thesaurus is a controlled vocabulary that shows relations (e.g. semantic) between terms, which can aid searchers in finding related terms to expand queries (Monica, L. 2002).

The aim of vocabulary Construction is to identify the most representative terms (words and phrases) for the thesaurus vocabulary from document collections. The first step is to identify an appropriate document collection. Next step is to determine the required specificity for the thesaurus. Then the vocabulary terms are now ready for normalization. The simplest and most common normalization procedure is to eliminate very trivial words such as prepositions and conjunctions. The next standard normalization procedure is to stem the vocabulary. Stemming reduces each word into its root form. In vocabulary construction the relevant aspect is term selection for the thesaurus from the document collections (Schubert, F. Siu, C., *et al*., 2000).

Controlled Vocabulary (CV) construction by automatic or semi-automatic methods can be categorized into statistical and linguistic approaches. In the statistical approach, terms are extracted from a document by IDF (inverse document frequency). Adapted to the controlled vocabulary construction problem, the assumption is that frequently co-occurring words with a text window (sentence, paragraph or whole text) point to some semantic cohesiveness. The co-occurrence approach needs human intervention before terms can be used for controlled vocabulary creations. In the linguistic approach, terms and their relations are based on the distributional context of syntactic unit (subject and object) and the grammatical surrounding function these unit. Suppose we have two terms “Electronic business" and “Electronic industry". These two terms can be semantically

mapped. The substituted words are semantically close (i.e. business and industry) (Ahsan-ul, M. and Margherita, S. (n.d.)).

2.6 Stemming

Stemming is a computational process removing inflectional and derivational affixes and returning a word base, not necessarily a real word. The main difference to morphological normalizing is that normalizing turns the word to lexical full base form. The negative effect of stemming and normalization as processes in information retrieval is that they may produce noise as unrelated word forms are sometimes conflated to a single form. (Turid, H. 2003).

Natural language texts have Variations in word forms. The reasons for such variants include requirements of grammar usage, antonyms, transliteration, abbreviation, and spelling errors. In natural language text the main source of word variation is morphology, natural language text with suffixing and prefixing being the most common ways of creating a word variant, both inflectional and derivational morphologies can result in very large numbers of variants for a single word depending on the morphological complexity of a language. Morphological complexity can have a strong impact on the effectiveness of information retrieval (IR) systems. Therefore, there is a need for automated procedures that can reduce the size of a lexicon to a controllable level, and also capture the strong relationships between different word forms (morphology). Morphological analysis, the identification of a word-stem from a full word form, helps in conflating semantically related words to the same form (Nega, A. and Peter, W. 2002).

Stemming is used for reducing different morphological variants of a word into a common form. It is widely used in IR, with the assumption that morphological variants represent similar meaning. It is applied during indexing and is used to reduce the vocabulary size, and it is used during query processing in order to ensure similar representation as that of the document collection. For morphologically less complex languages like English or Swedish, this usually involves removal of suffixes. For languages like Amharic or Arabic, that have a much richer morphology, this process also involves dealing with prefixes, infixes and derivatives in addition to the suffixes (Atelach, A. and Lars, A.

2007), (Nega, A. and Peter, W. 2002). Likewise Tigrigna language has a rich morphology and it deals with prefixes, suffixes, infixes and derivatives.

2.7 Similarity Computation

Similarity computation deals with deriving a relationship between pairs of terms. Two common computations are the cosine and dice coefficients (Schubert, F. Siu, C., *et al*., 2000), (McGill, *et al*., 1979) After the significant thesaurus vocabulary has been identified, determining the statistical similarity between pairs of terms is followed. There are a number of similarity measures available in the literature. Two of these are Cosine and Dice. The first calculates the number of documents associated with both terms divided by the square root of the product of the number of documents associated with the first term and the number of documents associated with the second. And the second computes the number of documents associated with both terms divided by the sum of the number of documents associated with one term and the number associated with the other.

The similarity between any two documents (or between a query and a document) can subsequently be determined by the distance between vectors in a high dimensional space. Zero distance between words indicates similarity and the most common similarity measure is the cosine coefficient. Cosine coefficient defines the similarity between two documents by the cosine of the angle between their two vectors. It resembles the normalized inner product of the two vectors, means inner product divided by the products of the vector lengths (square root of the sums of squares)(Katy, B. Chaornei, C., *et al*., (n.d.)). The following figure illustrates sample similarity thesaurus output.

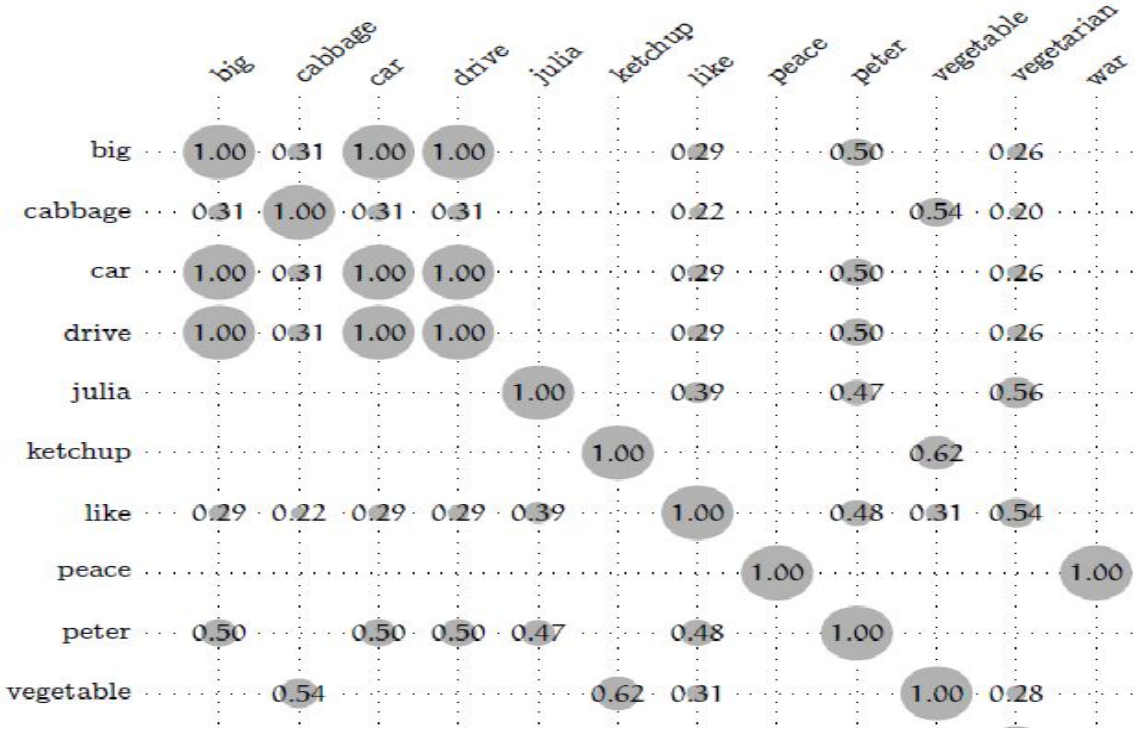

Figure 2.1sample similarity thesaurus (Sebastian, M. K. 2004)

The information structure that we call a similarity thesaurus is a matrix that consists of term-term similarities. A similarity thesaurus is based on how the terms of the probabilities of the documents representing the meaning of the terms (Kanyarat, L. 2003).

Word meaning can be regarded as a function of word distribution within different contexts in the form of co-occurrence frequencies, where similar words share similar contexts. Word similarity depends on to what extent they are interchangeable across different context settings. The flexibility of one word or phrase substituting another indicates its extent to be synonymous provided that the alternation of meaning in discourse is acceptable (Dongqiang, Y. and David, M. P. 2008).

Many researchers have used term co-occurrence in IR to identify semantic relationships that exist among terms. To classify relevant and non relevant documents query terms are valuable, and then their associated terms will also be useful, and can be added to the original query. A number of coefficients have been used to calculate the degree of relationship between two terms.

Three well-known coefficients are Angel, F. Carlos, G., *et al*., (n.d.):

$$\text{Tanimoto}\ (t_i, t_j) = \frac{C_{ij}}{C_i + C_j - C_{ij}}$$

$$\text{Cosine}\ (t_i, t_j) = \frac{C_{ij}}{\sqrt{C_i . C_j}}$$

$$\text{Dice}\ (t_i, t_j) = \frac{2.C_{ij}}{C_I + C_j}$$

where C_i and C_j are the number of documents in which terms t_i and t_j occur, respectively, and C_{ij} is the number of documents in which t_i and t_j co-occur. The coefficients have values between 0 and 1: if two terms occur only in the same documents, the associated coefficient is 1. If there is no document in which they co-occur, the value is 0. According to the above equations:

Tanimoto:-

compute the number of documents associated with both terms divided by the sum of the number of documents associated with one term and the number of documents associated with the other minus the number of documents associated with both terms

Cosine:-

compute the number of documents associated with both terms divided by the square root of the product of the number of documents associated with the first term and the number of documents associated with the second term.

Dice:-

compute the number of documents associated with both terms divided by the sum of the

number of documents associated with one term and the number of documents associated with the other.

2.8 Tigrigna Language

There are criteria which qualify a language as a major language in a country. Three criteria posits, one being a language is spoken by over a million speakers or by 25% of the population; second being an official language; and third is being the language is medium of education of over 50% of secondary school graduates. In 1970 Amharic, Oromo and Tigrigna had been meeting criterion and now there are five indigenous languages which qualify are meeting criterion. Amharic and Tigrigna have substantial written traditions and belong to the Semitic language family. The Ethiopic script is central to the culture of Amharic and Tigrigna, and has long played a part in the culture of other ethnic groups, not least in their contribution to discussions of change. The special status of Geez as a religious and literary resource is intimately bound up with the script in which it is written. Tigrigna as a mother tongue is predominant in Eritrea and northern Ethiopia Tigray region. When Eritrea is a separate state, according to National Office of Population 1993, estimated number of Tigrigna language mother-tongue speakers in Ethiopia has fallen, further behind numerically in Ethiopia remains third. Tigrigna is significant for its level of development in literature, literacy, standardization, etc. (Thomas, B. and Wondwosen, T. 1996).

Tigrigna language has morphological complexity (Michael, G. 2009). As far as the knowledge of the student researcher there are many people who speak only Tigrigna language in the Tigray region in particular in the rural areas of the region. Medium of education at primary level (1-8) is Tigrigna, Amharic and English are given as course; in high school English is a medium of instruction, (9-10) Tigrigna is given as course and 11-12 it is an elective course of (Tigrigna and Amharic) languages. Since 1988 Tigrigna language is offered as a minor field of study hosted in the department of Amharic and degree program has started from 2006 academic year in Mekelle University, (Mekelle University, 2010). Recently, AAU has started degree program in Tigrigna.

Tigrigna which is a member of Ethio-Semitic family is spoken by 5-6 million people in the Tigray region of northern Ethiopia and in Eritrea. Tigrigna is written in the Ge'ez script the same as other languages in the Ethio-Semitic family (Michael, G. 2009). It is a member of the Ethiopic branch of Semitic languages and it is written with a version of the Ge'ez script and first appeared in writing during the 13th century"(Omniglot, (n.d)).

2.8.1 The Tigrigna Writing System

All of the Ethio-Semitic languages of Ethiopia and Eritrea use the Ethiopic script (yacob, D. 2005). In origin, The Ethiopic writing system is a South Semitic script, probably derived from the Sabean system; which can be found in writing from the fourth century, was leaved by" Giiz" to Amharic and Tigrigna. It essentially uses one character per syllable, and consists of 275 symbols. The system works on the basis of seven 'orders' for each basic consonant symbol, representing seven vowels; in addition there are some additional variants for "labialized" consonants (with vowel) (Thomas, B. and Wondwosen, T. 1996).

A complexity that enters into Tigrigna spelling are the presence of Ge'ez loan words and words derived from a Ge'ez root. Ge'ez is the ancient language of Ethiopia. Ge'ez had a richer phonemic range and required additional letters for its orthography. Unlike Arabic and Hebrew, Ethiopic is written from left to right. There is no case difference of letters or distinction between upper and lower case.

The most salient graphical units in this writing system represent a consonant followed by a vowel. Characters representing the same consonant followed by different vowels are similar in shape. For example, here are the characters representing: /he/, /hu/, /hi/, /ha/, /hie/, /h/ and /ho/:

ሀ ሁ ሂ ሃ ሄ ህ ሆ

All of them have a sort of U-shape. Here are: /le/, /lu/, /li/, /la/, /lie/, /l/ and /lo/.

ለ ሉ ሊ ላ ሌ ል ሎ

As a result, the writing system is usually displayed as a two-dimensional matrix in which the rows contain units beginning with the same consonant and the columns contain units ending in the same vowel. The columns are traditionally known as "orders". The first order, in Tigrigna, represents the vowel /e/, the second the vowel /u/, the third the vowel /i/, the fourth the vowel /a/, the fifth the vowel (really diphthong) /ie/, and the seventh the vowel /o/. The sixth order represents the consonant alone means the sixth order in the letters simply represents the consonant by itself.

English alphabets are represented the Ge'ez letters more or less as per windows seven Operating system representation.

If we count the ge'ez symbols (Appendix2) there are 275 symbols, as church scholars saying, the five letters with five symbols are known as derived letters (ዲቃላ ሆህያት) and most of the rest 35 letters have derived letters from their fourth order letter. For example, ባ, ቱ, ሚ e.t.c. The Tigrigna language also has punctuations for demarcation of words, phrases and sentences. Some of the punctuations are:-

፡ ። ፤ ፣ ፥

colon full stop semi-colon Comma preface colon

List of Words and phrases in Tigrigna Language are separated by (፣), (።) uses to close a sentence or as sentence boundary. For two or more sentences to use them together (፤) uses to separate them from each other. (፡) was used as word boundary but these days space is used instead.

Tigrigna uses two ways of numeric systems. These are the Arabic numbers and the "Geez" or Ethiopic numbers. The following are the Ethiopic numbers. The Ethiopic number system does not use a zero.

፩ ፪ ፫ ፬ ፭ ፮ ፯ ፰ ፱ ፲ ፳ ፴ ፵ ፶ ፷ ፸ ፹ ፺ ፻ ፼

But currently almost all of the users are often using Arabic digits.

2.8.2 Tigrigna word formation and Challenges

Words in Tigrigna are formulated by integration of letters from its alphabets**.** A word may have a single meaning or multiple meanings and the same meaning can be denoted in different forms; this will be because of symbol redundancy (yacob, D. 2005), forms of spelling and in consistency in writing (Andargachew, M. 2009), and stressing of the word.

The first, symbol redundancy rises from same or approaching sound of distinct syllables. For example, **ሀ** and **ሐ**, **አ** and **ዐ**, **ፀ ጸ** and **ጠ**, **ቀ** and **ቐ**, pairs have approaching sound; **ጸ** and **ፀ** pairs have the same sound. Words from these pairs of letters can be formed as **መቀለ** and **መቐለ** the same meaning which means Mekelle; **ጠባይ**, **ጸባይ** and **ፀባይ** the same meaning which means (conduct or behavior); **ጸባ** and **ፀባ** also have the same meaning milk. The second, various forms of spelling for the same word as in '**ፀሓይ**' Vs '**ፅሓይ**' , '**ማርያም**' Vs '**ማሪያም**' and '**ኮምፒተር**' Vs '**ኮምፒዩተር**'. There is also inconsistency in writing, as an example, '**ደቂኣንእስትዮ**' Vs '**ደቂ ኣንእስትዮ**' can be written as one word or phrase. The third, some words can be sound by stressing or not. Here words change their meaning. For example, the word '**መረረ**' has two meanings, when it is not stressed it has the meaning bitter and when it is stressed it has the meaning blame. The same to this there are words with different meaning such as '**መተረ**' and '**በሪሑ**' which have two meanings each.

2.8.3 Stemming for Tigrigna

Tigrigna uses affixation to drive different forms from stems. Common affixations in Tigrigna are prefix, suffix, prefix-suffix pair and duplication. Because of its complexity in morphological structure, a single Tigrigna word can have thousand of variants. (Girma, B. 2001) has developed stemming algorithm for Tigrigna language text documents. As reported the developed stemmer is an interactive system and uses context sensitive rules that removes prefix, suffix, prefix-suffix pair, duplication of single and

double letters. This explores the possibility of developing a stemmer to conflate variant words of Tigrigna language for use in IR of the language.

The stemmer developed by (Girma, B. 2001), reported an accuracy of 84% to test file of 1568 words selected randomly from sample texts. Stemmers developed for other languages could not be applied for Tigrigna language because of the morphological complexity and difference in features of the language. This implies that developing an effective and efficient Tigrigna stemmer is very challenging. For example the word '**በልዐ**' (he eats) has many affixation and meanings because of its affixations, ' **በሊዑ**', '**በልዐት**', '**በሊዓ**', '**ተበሊዑ**', '**ተበሊዓ**', '**በሊዐ**', '**በሊዕና**', '**በሊዓቶም**', '**በሊዕዎም**', '**በሊዐንኡ**', '**በሊዕወን**', '**በሊያማ**', '**በሊዐንኡ**', '**ተበላልዐ**', '**ተበላሊዑ**' e.t.c

2.9 Related works

Related works to this research are studied in the following sections. The first section discusses Tigrigna thesaurus and the rest deals with non-Tigrigna thesaurus.

2.9.1 Tigrigna thesaurus

As far as the knowledge of the researcher, there is no full-pledge Tigrigna thesaurus. But (Mulu, G. 2001) has tried to develop bilingual thesaurus related to Tigrigna language. (Mulu, G. 2001) has developed an English-Tigrigna bilingual thesaurus based on the structure of Japanese Electronic Dictionary Research (EDR).In the process of generating the bilingual thesaurus, the tools and techniques used for developing the EDR dictionary are used. The thesaurus or dictionary has been developed based on the noun forms and functions of English and Tigrigna languages. As reported the complexity of Tigrigna can be resolved by applying the techniques applied for irregular plural forms in English. As reported the researcher shows the possibility to develop thesaurus and other electronic materials for Tigrigna language and he has developed the system prototype using Visual C++ for 500 head words.

2.9.2 Non- Tigrigna thesaurus

In the next sub sections thesaurus works on Semitic languages Amharic and Arabic, and non-Semitic languages English and Chinese are discussed.

2.9.2.1 Amharic automatic thesaurus

An automatic thesaurus generation for Amharic Text retrieval has developed by (Andargachew, M. 2009). The endeavor was to develop thesaurus automatically for text retrieval so as to help the development of effective and efficient Amharic retrieval system. The developed system for automatic thesaurus generation for Amharic text retrieval is domain independent. This means the system can be trained and used in different domains without any modification as long as the documents collection has a UTF-8 Unicode encoding. His experimentation of the system has made on Amharic Bible documents. In the research WORD SPACE model is used for the automatic thesaurus generation system and the WORD SPACE model was derived from the inverted file index by using random projection algorithm for dimensionality reduction and nearest clustering algorithm to generate thesaurus from the word space model developed.

The thesaurus developed by (Andargachew, M. 2009), which is reported has 58% accuracy and has an outstanding feature in its efficiency in terms of its response time. It takes only 12 minutes and 8 seconds to create 20,315 term vectors in the WORDSPACE. The thesaurus was integrated to an IR system for query expansion in order to further investigate its applicability Amharic information retrieval. The two most frequent and basic measures for information retrieval effectiveness (precision and recall) were used before and after using the thesaurus for query expansion. Stemming and stop word elimination components of the system has play vital role in the thesaurus generation and are evaluated.

Andargachew has discussed related works to his work, features and approaches of thesaurus construction. In addition he discussed Amharic scripts and the Ethiopic writing system with its Unicode representation in computer; ambiguities and inconsistencies in Amharic writing system. A corpora based automatic thesaurus generation approach is used to construct thesaurus for Amharic document collection. To develop the work, he

has used Java programming language for the programming task and the semantic vectors API for developing WORD SPACE model for document collections.

The researcher concluded that thesauri are among the components of IR systems and plays a significant role for enhancement of recall. Manual thesaurus construction is time consuming, labor intensive and suffers low coverage. It is costly, requires highly skilled experts in a subject domain, highly conceptual and knowledge intensive task. To solve these problems automatic generate thesaurus is required.

2.9.2.2 Arabic Monolingual thesaurus

In Arabic language, an automatic thesauri has been Designed and build using term-term similarity and association techniques by (Hayel, K., *et al*., (n.d.)). The thesauri can be used in any special field or domain to improve the expansion process and to get more relevant documents for user's query using Arabic language. The researchers use the definition for thesaurus as Merrian Webster Dictionary definition, thesaurus is i: book of words or of information about a particular field or set of concepts; Especially, a book of words and their synonyms ii: a list of subject headings or descriptors usually with a cross-reference system to be used in the organization of a collection of documents for reference and retrieval. Thesaurus contains creative synonyms and related phrases that allow authors to enhance their vocabulary.

They use three phases in their thesaurus development. The first phase is about document preparation that means removing stop words, tokenization (deleting punctuation marks, commas, special signs, and numbers), normalization and stemming. Elimination of stop words reduces the size of the indexing structure and thus increases the performance of the system and enables it to retrieve more relevant documents. As reported the researchers use the stemming algorithm of (Smeaton, A.F., *et al.,* 1983), with a little bit modification and they Use Vector Space Model to put the text of documents and the query in vectors. In the second phase, they select index terms. Terms selected for index term are terms which are repeated two to seven times in the text. This indicates the researchers ignore the terms that appear in most documents in the collection (i.e., have high frequencies),

and the terms that appear only once in a document (i.e., terms that have low frequencies). Their expectation was use of a controlled vocabulary leads to an improvement in retrieval performance. Creating the inverted file based on the stemmed words of each document. The stemmed words technique which was used, was suffix prefix removal. They have used C# and a SQL database to implement their work, by computing term frequency (tf), inverted document frequency (idf), and tf*idf. In the third phase, to build the thesaurus they made comparison among the laws (Inner product, Cosine, Jaccard or Dice) used in finding 'Term Similarity'/'Term relationship' between the different terms. AS a result, Cosine equation was used in building the similarity thesaurus because of its commonality. The computation of term frequency (tf), inverted document frequency (idf), and tf*idf is conducted. In addition, they calculate the weight or tf*idf of each term in a document by multiplying the normalized term frequencies with inverse document frequencies. After these steps, the inverted file that contains index terms (i.e., words) and terms frequency and the weight of each term in a document retrieved.

The researchers use 242 documents that were presented in the Saudi Arabian National Computer Conference, for each run 59 queries entered automatically .They have designed and built an automatic information retrieval system from scratch to handle Arabic text. To achieve this goal, they have constructed an automatic stemmed words and full word index using inverted file technique. Depending on these indexing words, the researchers have built three information retrieval systems. Information retrieval systems are developed using a term frequency-inverse document frequency (tf*idf) for index term weights. In addition the researchers used Similarity Thesaurus by using Vector Space Model with four similarity-measurements (Cosine, Dice, Inner product and jaccard) and compared between the similarity measurements to find out the best that will be used in building the Similarity thesaurus. At last, the researchers used an Association thesaurus by Applying Fuzzy model for index term weights. They implemented their system in C# language, and Runs on IBM/PCs and compatible microcomputer. The results they get were analyzed using the Recall and Precision criteria by applying 59 queries.

AS the researchers' explanation, their experiment showed that the Jaccard and Dice similarity measurements are the same for the VSM model, while the Cosine and Inner

similarity measurements are the same as well, but they are a little bit better than Jaccard and Dice measures. Their study points, that the best case for development of information retrieval system is using association thesaurus with stemmed words.

2.9.2.3 English monolingual thesaurus

(Olena, M. Ian, H. 2006) and (Dongqiang, Y. and David, M. P. 2008) have developed English thesaurus.

(Olena, M. Ian, H. 2006), developed a domain specific thesaurus based automatic "Keyphrase" Indexing. "Keyphrases" represents a brief but precise summary of documents. The domain of the study used on UN Food and Agricultural Organization (FAO) by randomly downloading 200 full text documents from (www.fao.org/documents/) for the training and evaluation material. There are Keyphrase extraction and term assigning existing approaches. In Keyphrase extraction, the phrases occurring in the document are analyzed to identify apparently significant terms, on the basis of properties such as frequency and length. In term assignment, term assignments are chosen from a controlled vocabulary of terms, and documents are classified according to their content into classes that correspond to units of the vocabulary.

AS reported, (Olena, M. Ian, H. 2006), uses keyphrase indexing, an intermediate approach between keyphrase extraction and term assignment that combines the advantages of both and avoids their limitation. Documents in the collection are pre-processed. That means, White space and punctuation are used to segment each document into individual tokens; Elimination of stop words, stemming remaining terms and sorting them into alphabetical order are conducted. For semantic term conflation, non-informative terms are replaced by their equivalent representative terms and for each training documents, candidate terms are identified and their feature value are calculated. Four features turned out to be useful in their experiments. These are the TF*IDF score, the position of the first occurrence of a phrase, the length of a candidate phrase in words and the node degree or the number of thesaurus links that connect the term to other candidate phrases. Index terms were assigned to the documents by professional indexers.

Three semantic relations bi-directional links between related terms (RT) and Inverse links between broader terms (BT) and narrower terms (NT) were defined.

(Dongqiang, Y. and David, M. P. 2008)**,** developed an automatic thesaurus for English language. As ground of their automatic thesaurus construction, distributional similarity as often calculated in the high dimensional vector space model (VSM). They proposed that to first categorize contexts in terms of grammatical relations, and then overlapped the top n similar words yielded in each context to generate automatic thesauri. As they explained in their report, the hypothesis was word meaning can be regarded as a function of word distribution within different contexts in the form of co occurrence frequencies, where similar words share similar contexts.

The researchers were first employed an English syntactic parser based on Link Grammar to construct a syntactically constrained VSM to automate thesauri construction and the word space consists of four major syntactic dependency sets that are widely adopted in the current English language research on distributional similarity. After parsing 100 million-words from British National Corpus (BNC): and filtering out non-content words and morphology analysis, they separately extracted the relationships to construct four parallel matrixes or co-occurrence sets. The dependency sets that consider were RV: verbs with all verb-modifying adverbs and the head nouns in the prepositional phrases; AN: nouns with noun-modifiers including adjective use and pre/post-modification; SV: grammatical subjects and their predicates; VO: predicates and their objects. Following the reduction of dimensionality on the dependency sets, they created the latent semantic representation of words through which distributional similarity can be measured so that thesaurus items can be retrieved. (Dongqiang, Y. and David, M. P. 2008), employed cosine similarity to compute similarity of word vectors.

The cosine of the angle θ between vectors $\mathbf{x}$ and $\mathbf{y}$ in the *n*-dimensional space is defined as:

$$\text{Cos}\theta = \frac{x.y}{|x||y|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2}}$$

where the length of x and y is |x| and |y|.

As the researchers explanation Semantic similarity is often regarded as a special case of semantic relatedness, while semantic relatedness also contains word association. Distributional similarity consists of both semantic similarity and word association between a seed word and candidate words in its thesaurus items, except for the 'noisy' words (due to the parsing or statistical errors) that hold no possible relationships with the seed.

2.9.2.4 Chinese monolingual thesaurus

(Schubert, F. Siu, C., *et al*., 2000) developed an automatic Chinese thesaurus that can be used to provide related terms to users queries to enhance retrieval effectiveness. The thesaurus is developed by computing the co-occurrence values between domain specific terms found in a document collection. Term frequency and document frequency of terms is used to compute the co-occurrence values.

Since, Chinese texts has no word delimiters, an extra word processing called word segmentation is used. Because there were not papers in English that detail the construction of Chinese thesauri, the automatic thesaurus developed by examining existing automatic thesaurus generation techniques and adapts them to the Chinese language.

The system is trained using an economics domain; this is not because the thesaurus is domain specific rather to derive a generic process for an automatic thesaurus generation that can be applied to other subject domain. The researchers used the Economics Terminology Dictionary, to identify and extract terms from the full text of each document found in their corpus. At the time of implementation the researchers are not develop Chinese IR system from scratch rather they modified an existing English language-based

IR system, called, the mg(managing gigabytes) to support the Chinese IR and automatic thesaurus generation. The corpus used in the research compromise three months Economics news (documents).

Finally, to measure the effectiveness of the system set of queries were used. A total of 30 queries and the relevant documents for each query were derived and identified after manual reading through the complete corpus. Results obtained from the experiments conducted, ensures the automatic generated thesaurus is able to improve the retrieval effectiveness of a Chinese IR system.

CHAPTER THREE

METHODOLOGY

3.1 Introduction

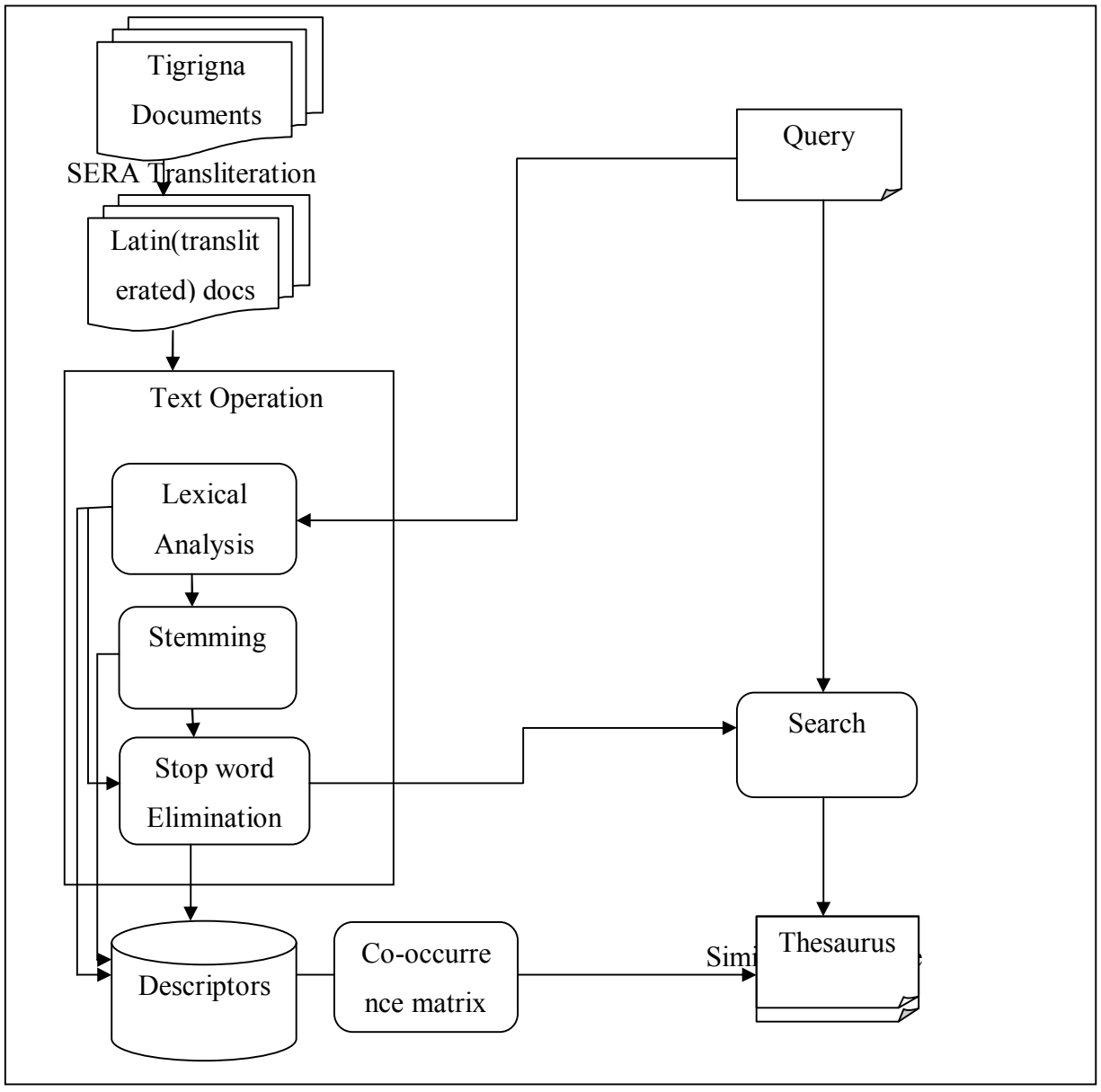

Figure 3.1 System Architecture of the thesaurus generation system.

Figure 3.1 depicts the model followed in the development of automatic Tigrigna thesaurus generation. In the model the main tasks that are done for preprocessing of Tigrigna text which are Transliteration, tokenization, stop words and number removal, stemming, co-occurrence manipulation, and similarity.

3.2 Preprocessing

Data preprocessing describes any type of processing performed on raw data to prepare it for another procedure. It transforms the data into a format that will be more easily and effectively processed for the purpose of the user (Christopher, D. Manning, *et al*., 2008).

In this study, preprocessing is the process of making the data ready for further processing. Since the collected documents are unstructured they are converted in to appropriate way for thesaurus construction. It includes, transliteration, tokenization, removal of stopwords and punctuations, stemming, and normalization processes. This helps to identify representative or descriptive terms which are manageable for thesaurus construction processes.

3.2.1 Transliteration

The Tigrigna documents are first transliterated to Latin using the System for Ethiopic Representation in ASCII (SERA) transliteration scheme. This is because the converted Latin scripts are convenient for stemming and internal processing of the documents. In Tigrigna characters encapsulate consonants and vowels and this makes stemming using the script difficult. For example, in the word **መምህሮም** (their teacher), the suffix is **ኦም** and the stem is **መምህር** here removing the suffix is difficult but after transliteration to Latin script removing the suffix “om” from the word “memhrom” to achieve the stem “memhr” is easier.

3.2.2 Tokenization

In NLP, tokenization can be defined as the task of splitting a stream of characters into words. However, very often it is associated with lower or upper level processes (Habert et al (n.d)). Lexical analysis is concerned about converting a given document to individual words or tokens by the process known as tokenization. It deals with the identification of term separators, accents, spacing etc. It also involves a decision on considering or disregarding special characters like hyphens, digits, punctuation marks, spaces, letter cases, etc. Usually punctuation marks, numbers, etc are ignored and white space is used as word delimiter for processing. In Tigrigna language, words are usually surrounded by whitespace and optionally preceded and followed by punctuation marks, parentheses, or quotes. A simple tokenization rule can therefore be stated as follows: Split the character sequence at whitespace positions and cut off punctuation marks, parentheses, and quotes at both ends of the fragments to obtain the sequence of tokens. This simple rule is quite accurate because whitespace and punctuation are fairly reliable indicators of word boundaries in the language.

The stream of characters in a natural language text must be broken up into distinct meaningful units (or tokens) before any language processing beyond the character level can be performed. If languages were perfectly punctuated, this would be an easy to do: a simple program could separate the text into word and punctuation tokens simply by breaking it up at white-space. But real languages are not perfectly punctuated, and the situation is always more complicated (Worku, K. 2009).

In this study, words are taken as tokens. All punctuation marks are removed and white space is used as a word demarcation. Hence, if a sequence of characters is followed by space, that sequence is identified as a word. Given a character sequence and a defined document unit, tokenization is the task of identifying it up into pieces, called token; perhaps at the same time throwing away certain characters, such as punctuation marks, digits etc. Here is an example of Tigrigna text tokenization:

Input: **ኣሕዋተይ፣ መሓዙተይን ደቂዓደይን ዓድና ብተፈጥሮ ሃፍቲ ዝተዓደለት እያ።።**

Output:

አሕዋተይ	መሓዙተይን	ደቂዓደይን	ዓድና	ብተፈጥሮ	ሃፍቲ	ዝተዓደለት	እያ

There are eight tokens; seven of the tokens are useful for indexing after stemming consideration. The set of index terms could be entirely distinct from the tokens. For instance, they could be semantic identifiers in taxonomy, but in practice in modern IR systems they are strongly related to the tokens in the document. However, rather than being exactly the tokens that appear in the document, they are usually derived from them by various normalization processes (Christopher, D. Manning, *et al*., 2008).

The major question of the tokenization phase is what are the correct tokens to use? In this example, it looks fairly trivial: the researcher uses whitespace to slice and throw away or strip punctuation characters.

Algorithm 3.1: Tokenization

```
do
    Read the contents of a file
    Split the content of the file in to list of words by white
    space
    For punc in punctuation mark list
        If a word starts with punc then
            Strip punc
        end if
        If a word ends with punc then
            Strip punc
        end if
    end for
while end of file
```

3.2.3 Stopword and Number Removal

Stopwords are non-content bearing words in a document. They are words which occur frequently and have less power in discriminating one document from another. Some are used for syntactic purpose; for instance, articles, prepositions, conjunctions, etc. (Christopher, D. Manning, *et al*., 2008). Stop words are normally removed from a document to facilitate effective representation of a document. In information retrieval system, stopwords have little or no contribution for retrieval process. So these terms should be removed. Every language has its own list of stopwords. In the case of the Tigrigna writing system these stopwords exist in different formats. A list of stopwords is compiled from Tigrigna literatures and some other list of stopwords is also added to the list by consulting with domain experts. For example, the following are some Tigrigna stopwords occurring in different formats (እዩ, ነይሩ, እዩነይሩ, እንታይ, እንታይነት, እንተይኮነስ, እንተዘይ, እንተዘይኮነ, እንተዘይኮይኑ). So these variant of words are needed to be stemmed to their root words and the stopword removal is performed on it.

The purpose of identifying stop words is, to remove insignificant words from the list of index terms. Index terms are believed to represent documents or discriminate one document item from the others; whereas, stopwords are not. Hence, using those words in the list of index terms is unimportant. That is why their exclusion from index term list is vital. The challenge here is some of the Tigrigna words are exist in different format. So, it is very difficult to fully collect and remove the stopwords for the system development. If a standard list of stopwords is available in the area, the performance of the system would improve significantly. Tigrigna Stop words are listed at appendix 4.

Algorithm 3.2: Stop Word Removal

```
for stopw  in stop list
if stopw  in token list then
        Remove stopw
end if
end for
```

In most cases numbers are less discriminating among documents (Baeza-Yates, R. and Ribeiro-Neto, B. 1999). In this study also, numbers are not considered as index terms. So, index terms list of the study does not contain any number. In Tigrigna languages Ge'ez numbers (Listed at Appendix 1) and Arabic numbers are possible to use in writing. These numbers are removed during tokenization. There are possible cases to use numbers to express things. For example, ቦይንግ777 (name of company), 20ዓዲ (name of certain place), 70እንደርታ (name of certain place) etc.. But in this study it is not considered. Simply numbers are removed. So, index terms list does not contain any number.

Algorithm 3.3: Number Removal

```
For number is in number list
    If number in token list then
        Remove number
    end if
    If token starts with number then
        strip number
    end if
    If token ends with number then
        strip number
    end if
end for
```

3.2.4 Stemming

In Tigrigna a word decline for number, person, gender etc. and stemming therefore, deals about converting these varying forms of the same word into one-root form. Stemming is

one of the preprocessing made on automatic Tigrigna thesaurus construction. To stem word tokens to their root terms, since there is no available stemmer for Tigrigna a simple stemmer that can remove common Tigrigna prefixes and suffixes was developed based on the concept of the algorithm developed by (Girma, B. 2001).

The Tigrigna language is morphologically rich language. This is the main reason to incorporate the stemming part in the thesaurus construction process of the system. In this process words are stemmed to the root word. In the Tigrigna writing system affixes are added in many positions of the stem. The stemmer developed removes prefixes and suffixes only, other morphologies are not considered.

Stemming process is conducted after transliteration of the Tigrigna script using SERA transliteration scheme into alphabetic, i.e. vowel and consonants are written separately. The suffix and prefix removing routines assume that vowels are represented in writings. In the stemmer, in most of the words after suffix removal the last character of the remaining word is changed to "*sades*" and this makes some difference in meaning of the root words. For example, the words: - **ኢትዮጵያ፣ ኢትዮጵያዊ፣ ኢትዮጵያዊት፣ ኢትዮጵያን፣ ኢትዮጵያውያን፣ ኬንያ፣ ኬንያዊ፣ ኬንያዊት፣ ኬንያን፣ ኬንያውያን**, after transliteration the words become (Ethiopya, Ethiopyawi, Ethiopyawit, Ethiopyan, Ethiopyawyan, Kenya, Kenyawi, Kenyawit, Kenyawi, Kenyawyan. To achieve the stems of these 10 words, the suffixes, awi, awit, awyan should be removed and ya ("*rabe*") changed to y("*sads*"). Then the stems are two words (Ethiopy, Keney). This result is different from the root (Ethiopya, Kenya) and it is common in suffix removal of many Tigrigna words.

The rules to remove prefix or suffix from a given word may not always hold true. For instance, if we remove the prefix **ብ** ('b') and the suffixes of the words **ብኢትዮጵያ፣ ብኢትዮጵያዊ፣ ብኢትዮጵያዊት፣ ብኢትዮጵያውያን** the correct stem would be acquired. But, this does not work for all Tigrigna words; removing '**ብ**' ('b') from the word '**ብርሃን**' ('light') would give '**ርሃን**' ('rhan'), which is meaningless; and removing suffix '**ና**' ('na') from '**ዋና**' ('main') gives '**ዋ**' ('wa'),which does not represent the original meaning. Therefore, to handle such problems exception list was prepared for

which affix removal rules do not apply. The stemmer developed takes words as an input and removes prefix and suffix of the word if the word is not present in the exception list.

Table 3.2 shows an example of the prefixes and suffices removed and an example under each affix; these are not the only affixes removed, list of all affixes considered for stemming purpose are presented in Appendix 3.

Table 3.1 prefix and suffix removed during stemming

Type	Affix	Example	
		Word	Translated to
Prefix	ብ	ብስራሕ	ስራሕ
	ን	ንስራሕ	ስራሕ
	ብዘይ	ብዘይስራሕ	ስራሕ
	ምስ	ምስስራሕ	ስራሕ
	ክም	ክምስራሕ	ስራሕ
Suffix	ን	ስራሕን	ስራሕ
	ና	ስራሕና	ስራሕ
	ኺ	ስራሕኺ	ስራሕ
	ኦም	ስራሖም	ስራሕ
	ኩም	ስራሕኩም	ስራሕ

Algorithm 3.4: Stemming

```
For token not in exceptional list
    If prefix in prefix list
        If token starts with prefix then
            L=length of prefix
            token = token[L:]
        end if
    end if
    If suffix in suffix list
        If token ends with suffix then
            L=length of suffix
            token=token[:L]
        end if
    end if
end for
```

3.2.4 Normalization

This topic deals about words in the Tigrigna writing system which can be written in different forms. In the Tigrigna writing system there are different words used with spelling inconsistency. The need to handle various forms of words with the same meaning is, to treat the same word with such different words as one word. In Tigrigna language, there is various form usage of words and there is no rule where to use those varying words. In the corpus collected for thesaurus construction of the current study, inconsistent usage of such words is very common. If those words are not replaced with

one common word, the same concept words are treated differently. As a result, we obtain more than one index terms for the same word. The word '**ኣምንታ**' ('positive'), for instance, is found in the data written in three forms as '**ኣውንታ**, **ኣወንታ**' and '**ኣምንታ**'; which means, there are three index terms for the word '**ኣምንታ**' ('positive'), even if all the three forms have the same meaning, 'positive'. Such kinds of things increase computational complexity (decrease efficiency) and decrease effectiveness (accuracy). Therefore, all words with varying forms but having the same meaning are converted to one common form. For example, inconsistency usage of sample terms in the Tigrigna writing system and translation of them in the compiled corpus for this study are depicted in table 3.2.

Table 3.2 Sample word normalization

Terms	normalized to	transliteration	meaning
ዕቤት፣ ዕብየት	**ዕቤት**	Ebiet	development
ዕዳጋ፣ ዕደጋ	**ዕደጋ**	Edega	market
ኣድላይ፣ ኣድላዪ	**ኣድላዪ**	adlayi	needed
ኣውንታ፣ ኣምንታ፣ ኣወንታ	**ኣምንታ**	awonta	positive
እንድስትሪ፣ እንዱስትሪ	**እንዱስትሪ**	endustri	industry
ኩሎ'ኹም፣ ኩሉ'ኹም	**ኩሉ'ኹም**	kuluKum	All of you
ዊዳበ፣ ውዳበ	**ውዳበ**	wdabe	association
ኣራኣእያ፣ ኣርኣእያ፣ ኣረኣእያ	**ኣረኣእያ**	areaeya	thinking

More than 45 Tigrigna words are treated in the normalization process. Other painful thing in the corpus processing work is the difference in spelling consistency and short form

word representation. For example, many writers use “sads” and “sals” means sixth and third sequence letters interchangeably, though, there is a big difference in between. This issue is treated with normalization. Tigrigna users use short form representation for single words and compound terms. For example, **ት/ቲ** is the short form for **ትምህርቲ** (education), **ኣ.ኣ** is the short form for **ኣዲስ ኣባባ** (Addis Ababa). This is also treated with normalization.

3.3 Vocabulary Construction

A thesaurus is a structured list of terms, usually related to a particular domain of knowledge. Thesauri are used to manage the complexities of terminology and to provide conceptual relationship of terms for any application. They are specially, useful in indexing and retrieving information processes, in other words, they may specify descriptors or keywords authorized for indexing and searching by providing the different forms which a concept can adopt. These descriptors form a controlled vocabulary (authority list, index language) (Jos´e, R. and Lourdes, A. 2006). In this study the vocabularies are the terms acquired after preprocessing is conducted.

Vocabulary construction is the process of identifying the most representative or informative terms for the thesaurus vocabulary from document collection. The main stated criterion in vocabulary construction is the collection should be sizable and representative of the subject area; and next determine the required specificity for the thesaurus (Kanyarat, L. 2003).

3.4 Index Term Weight

All index terms are not equally important in representing and discriminating a document; the degree of the terms in representing a document is different from one to the other. It is thus, required to measure how important a term is with regard to representation and discrimination of a document (Worku, K. 2009).

Term discrimination considerations imply that the best terms for document content identification are those that are able to distinguish certain individual documents from the

remainder of the collection. (Gerard, S. and Christopher, B. 1988). In this study, index terms are identified and co-occurrence matrix is developed which is useful for similarity construction.

3.5 Term-term co-occurrence matrix for automatic thesaurus construction

Thesaurus is able to disclose the conceptual knowledge of a document collection. The basic idea of the automatically constructed thesaurus is to calculate the similarities between two terms on the basis of their co-occurrence in a document collection (Wei S., Chenghua L., *et al*., 2006).The similarity measure can be done based on different similarity measures, such as cosine, dice etc.

In automatic thesaurus construction, the entire document collection is analyzed and co-occurrence relationships between terms are used to build a matrix of term-term relationships. Usually, term-term matrices of this type contain weights which are a measure of how related one term is with another term. These matrices are large and computationally expensive to compute. The matrices are used to cluster terms based on their co-occurrence data in the hope that terms that are closer together in this term-space are synonymous. Conceptually, what is underlined is the role of documents and terms are interchanged in the retrieval model. In essence, documents become the features of the term. Thus, two terms that appear in the same document are indexed by a similar feature and are deemed to have some type of synonymous relationship. Many formulae have been proposed to measure the association between two terms using co-occurrence data. The similarity between two terms k_i and k_j can be determined by evaluating the difference between the two-vectors vector $k_i = (d_{i1}, d_{i2}, ..., d_{in})$ and vector $k_j = (d_{j1}, d_{j2}, ..., d_{jn})$ in the document vector space. A simple binary weighting on these document weights would lead to the following cosine formulation of similarity between two terms:

$$\text{Cos}\,(k_i,k_j)=\frac{df(k_i,k_j)}{\sqrt{df(k_i)df(k_j)}}$$

where $df(t_i, t_j)$ is the number of documents in which both k_i and k_j co-occur, $df(k_i)$ is the number of documents in which k_i occurs and $df(k_j)$ is the number of documents in which k_j occurs. There are many variations of such formulae which aim to accurately find the best synonyms for a term. (Ronan, C. and Colm, O. (n.d)). While various techniques are in the literature, computing the co-occurrence values between domain-specific terms found in a document collection appears to be the standard technique.

Automatic thesaurus from corpus is generated by computing the co-occurrence values between domain-specific terms found in a document collection. These co-occurrence values are derived from term frequency and document frequency of the terms (Angel, F. Carlos, G., *et al*., (n.d.)).

The key to automatic thesaurus generation is representing each term as a vector. The terms are then compared using a similarity coefficient that measures the Euclidean distance, or angle between the two vectors. To form a thesaurus for a given term t, related terms for t are all those terms u such that Sim(t,u) is above a given threshold (David, A.G. and ophir, F. 1998). Co-occurrence matrix is a matrix that describes how often or in how many documents terms occur together with other terms. This type of matrix may be called an association cluster. An association cluster is based on the frequency of co-occurrence of pairs of stems inside relevant documents that are retrieved. When simple association cluster of terms is generated from the document collection then the association matrix S (an association matrix) is said to be unnormalized. An alternative is to normalize the correlation factor. For instance, if $C_{u,v}$ is adopted,

$S_{u,v}=\frac{C_{u,v}}{C_{u,u}+C_{v,v}-C_{u,v}}$ association matrix S is said to be normalized(Kanyarat, L. 2003)

An example of co-occurrence term to term matrix in three sample documents

Table 3.3 Sample document collection

Id	content
D1	use of thesaurus for information retrieval
D2	thesaurus for query expansion
D3	query for information retrieval

In the example presented in table 3.4, no stop words removal or stemming is considered. The co-occurrence of term i with term j is computed as follows.

Table 3.4 Co-occurrence of terms

terms	expansion	for	information	of	query	retrieval	thesaurus	use
expansion	1	1	0	0	1	0	1	0
for	1	3	2	1	1	2	2	1
information	0	2	2	1	1	2	1	1
of	0	1	1	1	0	1	1	1
query	1	1	1	0	2	1	1	0
retrieval	0	2	2	1	1	2	1	1
thesaurus	1	2	1	1	1	1	2	1
use	0	1	1	1	0	1	1	1

The number in each cell of the matrix is the number of times two terms have co-occurred. For example, information and retrieval co-occurred 2 times while information and thesaurus co-occurred 1 time.

One major approach for representing the meaning of a word in NLP is to treat it as a vector that codes the pattern of co-occurrence of that word with other expressions in a large corpus of language (Sahlgren, 2006), (Turney, and Pantel, 2010).

As we have seen above, association clusters are based on the frequency of co-occurrence of pairs of terms in documents and do not take into account where the terms occur in a document. The solution for such problem is metric cluster. Metric clusters are based on the correlation between two terms. Since two terms which occur in the same sentence seem more correlated than two terms which occur far apart in a document, it might be worthwhile to factor in the distance between two terms in the computation of their correlation factor. This can be expressed by $S_{u,v}=\frac{c_{u,v}}{||v(s_u)||X||v(s_v)||}$------(Kanyarat, L. 2003)

One other clustering technique is scalar clustering, which is a form of deriving a synonymy relationship between two local stems. The idea is that two stems with similar neighborhoods have some synonymy relationship. In this Let $S_u= (S_{u,1},S_{u,2}, \dots ,S_{u,n})$ and $S_v= (S_{v,1},S_{v,2}, \dots ,S_{v,n})$ be two vectors of correlation values for the stems S_u and S_v. Further, let S=[Su,v] be a scalar association matrix. Then, each $S_{u,v}$ can be define as

$$S_{u,v}=\frac{S_u.S_v}{||S_u||X||S_v||}$$----(Kanyarat, L. 2003)

Word co-occurrences or word associations can be categorized as first or second order co-occurrences, or direct or indirect co-occurrences respectively. First order term co-occurrence analysis measures the frequency of term t and term t_n tend to occur together in a text window. Second order co-occurrence analysis associate terms with similar context. Suppose that term X almost never occurs without term Y, and that term Z also tends not to occur without term Y, yet X and Z never co-occur in documents. From this

one may conclude that some relationship exists between X and Z, that is, they are related by the fact that each one co-occurs strongly with Y. In all probability, term X and term Z are synonymous or have relationship in this context. Synonyms tend not to occur with each other, yet the terms they co-occur with will be very similar. The assumption behind first order co-occurrence analysis is that semantically related terms tend to appear in some predefined context. Whereas the notion behind second order co-occurrence analysis is, that semantically similar words have a tendency to share similar contexts. This entails, that a first order co-occurrence analysis can generate an association profile (cluster) for a specific term. This profile contains other terms that frequently co-occur with the specifically chosen term. Subsequently, two different term association profiles can be compared through second order co-occurrence analysis. If they have similar association profiles, we can expect the two terms to be in a relationship (Jesper, W. (2004).).

3.6 Similarity computation

Similarity computation is concerned with deriving a relationship between pairs of terms. There are many similarity computation measures. Among the many similarity computation measures between pairs of terms, cosine similarity computation measure is the most commonly used one (Angel, F. Carlos, G., *et al*., (n.d.)). (Dongqiang, Y. and David, M. P. 2008). Cosine similarity measures computes the number of documents associated with both terms divided by the square root of the product of the number of documents associated with the first term and the number of documents associated with the second.

$$\text{Cosine}\,(t_i, t_j) = \frac{C_{i,j}}{\sqrt{C_i . C_j}}$$

where c_i and c_j are the number of documents in which terms t_i and t_j occur, respectively, and c_{ij} is the number of documents in which t_i and t_j co-occur. C_{ij} in the term-term matrix is non-zero if and only if there exists, a document in which both terms occur.

Many researchers have used term co-occurrence in IR to identify semantic relationships that exist among terms (Angel, F. Carlos, G., *et al*., (n.d.)).

CHAPTR FOUR

EXPERIMENT AND PERFORMANCE EVALUATION

Introduction

Automatic Tigrigna thesaurus generation system is implemented for different office documents (reports, notices and other office documents) using the Python programming language. This system prototype has been run on a laptop 2.10 GHz Intel core^{i3} processor and 4 GB RAM memory.

4.1Corpus Collection

A Corpus, plural corpora, is a collection of text which can be a flat text or a text with linguistic information. For Tigrigna language, there is no available standard corpus. Therefore, to compile the corpus for this study, office reports, notices and other office documents are collected. These documents were selected for corpus because they were easily accessible through friends. The compiled corpus is a promising corpus for thesaurus construction. Because it covers concepts of various fields such as Agriculture, Education, Industry etc.. All of the documents are saved as text files with a UTF-8 encoding. Then the Tigrigna documents in the collection are transliterated in to Latin script for preprocessing. A total of 80 documents which have a total of 24,401 words are collected; and the documents are classified into training dataset and testing dataset randomly. Both parts can be representatives of the Tigrigna corpus. 50 and 30 documents are used for training dataset and testing dataset respectively. The training dataset documents are used to generate the thesaurus and is composed of 5,182 vocabulary terms (or key words) extracted from the documents.

The basic assumptions in allotting training dataset and testing dataset is that both of them should be representative samples of the underlying domain, and the testing dataset must be independent instance that have played no part in generation of the thesaurus (Witten, I. H. and Frank, E. 1999).

4.2. Thesaurus Generation

4.2. 1 co-occurrence matrix

Thesaurus generation system is developed by building up a co-occurrence matrix based on index terms. The resulting co-occurrence matrix is then used to build a cosine similarity of terms.

The following figure shows sample co-occurrence matrix of documents Trained in the system.

	ጫፍ	ጭቡጥ	ጨቢጥ	ጭርሖ	ኤክስፖርት	ኤክስቴንሽ	ኔትዎርክ	ሕፅረ	ሕፅረት	ሕውይ	ሓፂር	ሓባር	ሓበሬ	ሓበሬት	ሓቢር
ጫፍ	1	0	2	0	0	0	0	0	0	0	0	2	0	0	0
ጭቡጥ	0	0	0	0	0	0	0	2	0	0	0	1	0	0	0
ጨቢጥ	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ጭርሖ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ኤክስፖርት	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
ኤክስቴንሽ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ኔትዎርክ	0	0	0	0	0	0	1	2	0	0	0	0	0	2	0
ሕፅረ	0	0	0	0	0	0	0	1	2	0	1	1	0	1	0
ሕፅረት	0	0	0	0	0	0	0	0	2	0	2	2	0	0	0
ሕውይ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ሓፂር	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0
ሓባር	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ሓበሬ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ሓበሬት	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
ሓቢር	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 4.1 term to term co-occurrence matrix

The similarity measure function returns a value of one if the term vectors are similar and a zero value if they are dissimilar. The following is sample similarity thesaurus which shows how the system is working.

	ጫፍ	ጭቡጥ	ጨቢጥ	ጭርሖ	ኤክስፖርት	ኤክስቴንሽ	ኔትዎርክ	ሕፅረ	ሕፅረት	ሕፁይ	ሓፂር	ሓባር	ሓበፌ	ሓበፌት	ሓቢር
ጫፍ	1	0.48	0.97	0.07	0.18	0.12	0.24	0.43	0.37	0.32	0.21	0.7	0.09	0.2	0.2
ጭቡጥ	0.48	1	0.47	0.1	0.12	0.09	0.17	0.65	0.37	0.35	0.23	0.63	0.12	0.16	0.14
ጨቢጥ	0.97	0.47	1	0.07	0.12	0.07	0.24	0.41	0.31	0.31	0.19	0.69	0.09	0.19	0.16
ጭርሖ	0.07	0.1	0.07	1	0.1	0.03	0.05	0.06	0.17	0.01	0.07	0.09	0.05	0.02	0.09
ኤክስፖርት	0.18	0.12	0.12	0.1	1	0.22	0.09	0.14	0.72	0.13	0.21	0.18	0.02	0.09	0.28
ኤክስቴንሽ	0.12	0.09	0.07	0.03	0.22	1	0.09	0.14	0.23	0.06	0.11	0.15	0.02	0.11	0.15
ኔትዎርክ	0.24	0.17	0.24	0.05	0.09	0.09	1	0.5	0.22	0.09	0.28	0.32	0.08	0.51	0.16
ሕፅረ	0.43	0.65	0.41	0.06	0.14	0.14	0.5	1	0.54	0.28	0.57	0.67	0.11	0.31	0.18
ሕፅረት	0.37	0.37	0.31	0.17	0.72	0.23	0.22	0.54	1	0.29	0.66	0.64	0.09	0.21	0.31
ሕፁይ	0.32	0.35	0.31	0.01	0.13	0.06	0.09	0.28	0.29	1	0.15	0.37	0.05	0.13	0.13
ሓፂር	0.21	0.23	0.19	0.07	0.21	0.11	0.28	0.57	0.66	0.15	1	0.7	0.07	0.18	0.25
ሓባር	0.7	0.63	0.69	0.09	0.18	0.15	0.32	0.67	0.64	0.37	0.7	1	0.13	0.25	0.21
ሓበፌ	0.09	0.12	0.09	0.05	0.02	0.02	0.08	0.11	0.09	0.05	0.07	0.13	1	0.05	0.02
ሓበፌት	0.2	0.16	0.19	0.02	0.09	0.11	0.51	0.31	0.21	0.13	0.18	0.25	0.05	1	0.13
ሓቢር	0.2	0.14	0.16	0.09	0.28	0.15	0.16	0.18	0.31	0.13	0.25	0.21	0.02	0.13	1

Figure 4.2 Sample Similarity thesaurus

4.3 System implementation

The Python programming language is used for the development of the system prototype. The Enthought Python Distribution (EPD) version 2.7.1 is used for Python installation, because The Enthought Python Distribution (EPD) is distribution of the Python™ Programming Language, including over 80 additional tools and libraries of which some are important for this study.

NumPy is an extension for Python that provides support for large, multi-dimensional matrices and arrays. It is also used for construction of the cosine similarity. The package's efficiency and user-friendly numeric routines make it an essential tool for Python data analysis and manipulation. NumPy is even useful outside of purely scientific applications as a generic data container to optimize processing capabilities. Among the features it contains are: a powerful N-dimensional array object, sophisticated broadcasting functions, basic linear algebra functions, sophisticated random number capabilities, tools for integrating with different programming language codes e.t.c. (Steven, F. Lott, 2009)

The developed system for automatic thesaurus generation for Tigrigna text retrieval is domain independent. That means, the system can be employed in a different domain. It

can be applied to other domains without extra modification. The system takes 14 minutes to preprocess the corpus and generate the result.

4.4 Discussion

In this study, Tigrigna words are stemmed to their root stem. The reason for stemming is for the words dimension redaction, speed of the system and memory management purpose. In the current study, there are some words which are not properly stemmed. This is because different Tigrigna words have different forms. As a result, they stemmed to the form different from their root stem. They violet the affixation rule used in this study. The following table shows sample not properly stemmed Tigrigna words in the system.

Table 4.1 Sample of not properly stemmed Tigrigna words

No	Words	Affixes	After stemming	Root
1	ሮቡ፡ሮቡ፡ሮቡ፡ጨቢጦም	ኣት፡ ኦም	ሮቡ፡ሮቡ፡ ጨቢጥ	ሮቡ፡ሮ
2	ዕሬት፡ ዕሩይ፡ ዕሩይ፡ዕሩያት	ኣት	ዕሬት፡ ዕሩይ	ዕሬት
3	ጥ'ቕሚ፡ ጥ'ቕማጥ'ቕሚ፡ ጥ'ቕሚታ፡ት፡ ጥ'ቕምታ፡ት፡ ጥ'ቕሞም፡ ጥ'ቕመ፡ ጥ'ቕማ ጠቓሚ፡ ጠቐምቲ	ታ፡ት፡ ኡ፡ ኣ፡ ኦም	ጥ'ቕሚ፡ ጥ'ቕማጥ'ቕሚ፡ ጥ'ቕሚ፡ ጥ'ቕም፡ ጠቓሚ፡ ጠቐምቲ	ጥ'ቕሚ
4	ሕታም፡ ሕትመት		ሕታም ሕትመት	
5	ሓካይም፡ ሓኪም		ሓካይም፡ ሓኪም	

The words in the above table are not properly stemmed because the algorithm used for this study does not perfectly handle them in its rules. For example, the plural form of the word, **ሓኪም** (Medical doctor) is **ሓካይም** (Medical doctors) which is different from the pluralization used for many of the Tigrigna words which is adding the suffix ("**ታ፡ት**"). Such problems had made decline directly the performance of the stemmer and indirectly

the accuracy of thesaurus generation. The stemmer evaluation is indicated in table 4.5. Another challenge in the stemmer part of the system is that it is not capable of removing infix, some of the prefixes and suffixes. The reason for the existence of such kind of problem is, it comes from difficulty of fully identifying all of the root words in the Tigrigna language.

In this study, stopword elimination was conducted after stemming. The basic reason to

Table 4.2 Evaluation of stop words after stemming is carried out

No	Stopword	Prefix	Suffix	After stemming	Properly stemmed
1	እንተነይሩ	እንተ	-	ነይሩ	Yes
2	እንተኮነግን	እንተ	-	ኮነግን	No
3	ከምዮም	ከም	ኦም	ት	No
4	ኣብኣዮም	-	ኦም	ኣብኣት	No
5	ከምዚኦም	ከም	ኦም	ዚ	No
6	ናትኪ	-	ኪ	ናት	No
7	ማለዮም	-	ኦም	ማለት	Yes
8	ምስዮም	ምስ	ኦም	ት	No
9	ስለዝኮነ	ስለ	-	ዝኮነ	Yes
10	ስለዙይ	ስለ	-	ዙይ	Yes
11	ከምናዮም	ከም	ኦም	ናት	No
12	እንተይኮነስ	እንተ	-	ይኮነስ	No
13	እቲኦም	-	ኦም	እቲ	Yes
14	እቲኣን	-	ኣን	እቲ	Yes
15	ብምኻኑ	ብ	-	ምኻኑ	Yes
Total	15				7
percent					46.67%

conduct stopword elimination after stemming is that some of the stopwords existed in different forms. But the first experiment showed elimination of stopwords before stemming has made stopwords over stemmed and under stemmed. This is evaluated by taking sample random terms from the system and the result achieved during the first experiment is depicted in table 4.2.

The first experiment shows 53.33% of the stopwords are not properly stemmed. Therefore, to handle this problem, the stop words list is included to the exception list in order not to apply affixation rule for the stopwords list. This helps us to control loosing form of stopwords because of stemming. The system has tested before and after the stopword list had included to the exception list. The system showed 48 keywords difference. This implies that many stopwords were stemmed inappropriately which are solved after the stopword list had included to the exception list.

Another challenge in removal of stopwords from the system was that some of the Tigrigna words exist in different format. So it was very difficult to fully collect and remove the stop words for the development of the system. If a standard list of stopwords is available for Tigrigna, the performance of the system may improve significantly and this is recommended for further research.

In the Tigrigna writing system there are different words that are written in abbreviation or short form. These words are separated from each other by periods (.), hyphen (-), slash (/). These representations are used for single word or a combination of words. For example, **ሕብረተሰብ፣ ሕ/ሰብ፣ ሕብረተ-ሰብ** representations are used to refer the same concept which means society. In this research such representations are compiled from Tigrigna literatures and normalized to one common form. Around 45 shorter forms of single and compound words are considered for this work. But this are not the only shortword form representations; this number would have been higher, if there was a comprehensive list of these words or other contextual mechanism is used in Tigrigna language. Another problem here is when the compound words are separated by whitespace; they are treated as different terms because the drawback of term-to-term co-occurrence measures is single-word terms are considered (Khurshid, A., *et al.,* (n.d). For

example, the compound word **ሕብረተ ሰብ**, is treated as two terms **ሕብረተ** and **ሰብ**. Additional problem is, some Tigrigna language users use informal short word representation in their writing.

4.5. Evaluation of Preprocessor

Table 4.3 Sample preprocessor evaluation

R.N	Stemmed terms	Properly stemmed	Stopword
1	**ጫፍ**	Yes	No
2	**ሓባር**	Yes	No
3	**ፀገም**	Yes	No
4	**ኣድል፣ ኣድላዪ፣ ኣድላይ፣ ኣድለይቲ**	No	No
5	**ኣለ·ት፣ ኣለ·ታ፣ ኣለ·ታዊ**	No	No
6	**ብቕዓት**	Yes	No
7	**ፍርዲ**	Yes	No
8	**እንራኽብ**	Yes	No
9	**ክኸው**	No	Yes
10	**ኸነ**	Yes	Yes
11	**ሕፅረት**	Yes	No
12	**ሕማም**	Yes	No
13	**ሓደጋ**	Yes	No
14	**ሓለዋ**	Yes	No
15	**ሓይሊ**	Yes	No
16	**ሕብረት**	Yes	No
17	**ሕጊ**	Yes	No
18	**ፅሩይ፣ ፅሩይቲ ፣ ፅርየት**	No	No
19	**መልሲ**	Yes	No
20	**ክኸውን፣ ክኸዉ፣ ክኾ፣ ክኾኑ፣ ክኾው**	No	Yes
total		15	3
percentage		75%	15%

Preprocessing component of the system is very crucial in the thesaurus generation process. The system is checked whether punctuations and numbers are removed and the evaluation of stemming and stopwords has been made by random sample selection of 20 terms from the system.

According to the sample 25% of the terms are not properly stemmed and 15% of the terms are stopwords. This means the accuracy of the stemmer is 75% and the accuracy of the stopwords removal is 85%. The reason for the inappropriate stemming of the terms is that composition of complete list of affixes (prefix and suffix) requires intensive manual intervension and demands team of domain experts to be engaged in the project work. This is also true for having standard stopword list (non content bearing terms).

4.6 Evaluation of the Thesaurus Terms

In this study, properly stemmed terms are selected and are used for the purpose of evaluation of thesaurus terms, for each of them using the system that is developed to undertake this operation. Table 4.4 shows the terms and their corresponding thesaurus terms. Therefore the accuracy of the thesaurus generation system is evaluated based on the sample of properly stemmed terms by checking them in the system. Here, top five and ten terms per term is used as a threshold. The top five and ten terms selected as per the number of similarity words generated. If the generated words were many, top five were selected for manageability purpose, otherwise, top ten were selected. Then the terms are evaluated to see if they are similar or not. The result of the finding shows 75.28% of the output has related concepts to the respective terms. The rest 24.72% has no related concepts. The improperly stemmed terms and the existence of some stopwords that are not detected in the document collection have resulted this irregularity. If the stemmer component is properly developed and the document properly stemmed and all the stop words removed from the document collection, a better output would have been expected.

Table 4.4 Sample thesaurus evaluation

Term	Thesaurus	
	Related concept	Non-related concept
ሓባር	በዝሒ፣ ዝተፈላለየ፣ ፓርቲ፣ ገምጋም፣ጥንካረ፣ ጥንኩር፣ ሓፋሽ፣ ሓዊስካ፣ሓይሊ፣ ዓቕሚ፣ ኣባል፣ ፀገም	ፃዕሪ፣ ፅቡቕ፣ ፅሑፍ፣ ጫፍ፣ ታሕቲ፣ ውሑድ፣ ታሕተው፣ ንሊህ
ሕፅረት	እዋን፣ ፃዕሪ፣ መብራህቲ፣ ቀረብ፣ ተዋሳእቲ	ሓደሽቲ፣ላዕለው
ፀገም	ሕፅረት፣ ዓቅም፣ ዓመፅ፣ አፈፃፀማ፣ኣረኣእያ፣ ገምጋም፣ ተሞክሮ፣ ግንዛበ፣ ምድላው፣ ፓርቲ	ምባል
ፍርዲ	ብህዝቢ፣ ዲሞክራሲ፣ ምጥፋእ፣ ተገንዚብ፣ ወሊቀ፣ ዝንባሌ	ይል፣ ሰፈር፣ ምልካ
ሓገዝ	ባንኪ፣ ልቓሕ፣ ምውጋድ፣ ተዋሂቡ፣ፕሮጀክት፣ ሸርፊ፣ ዘበርክት	ምምልማል፣ መንደር፣ መተወ፣ ክፃወት
ሓደጋ	ክለል፣ ላሕሚ፣ምብዛሕ፣ቀለብ	
ሓለዋ	ግንዛበ፣ ፖሊሲ፣ ሪፎርም፣ ዝሕግዝ	ዘል፣ ዘለ
ሓይሊ	ድልው፣ ስለጥ፣ ውሽጢ፣ ዝሕግዙ	
ሕብረት	ገጠር፣ ምርድዳእ፣ ዝድለ፣ ዝንባለ	
ሕጊ	ቐፅሪ፣ ጠቐሱ፣ ኣድለይቲ፣ ደንቢ፣ፍቓድ፣ ክረኽቦ፣ መፅናዕቲ፣ መዓቀኒ፣ መረጋገፂ፣	ኮሞሚ፣ ተዳልዩም
total	67	22
percentage	75.28	24.72

CHAPERT FIVE

CONCLUSION AND RECOMMENDATIONS

This chapter deals with conclusion of the study and recommendation for future study in the area.

5.1 Conclusion

A thesaurus can be constructed manually, automatically or semi-automatically. Manual and semi-automatically construction of thesaurus are time consuming, suffers low coverage, costly, labor intensive and maintenance complex. Even though, some form of manual thesaurus construction is mandatory due to the relational complexities and semantic ambiguities in languages. As a result, there is always a need to a system which can generate thesaurus automatically. The unique requirement of the system is that it should be domain independent, scalable to process huge collection of documents and fast response time in contrast to the manual construction. In this research a corpora based automatic thesaurus generation based on term-term co-occurrence approach is used, so as to construct thesaurus for Tigrigna document collections; and a remarkable output is achieved. The documents used have heterogeneous collection nature which makes identification of similarity of terms easy. The developed system is domain independent. The system can be tested by increasing the corpus size substantially and using documents from different domains.

In the generation of thesaurus terms preprocessing component of the system play very crucial role. More specifically to generate the thesaurus terms the document collection should be properly stemmed and non-content bearing words (stopwords) should be removed. The stemmer that is developed has been working with the assumption that Tigrigna writing system is alphabetic, not syllabic and because it utilizes SERA for representation of Tigrigna characters. This creates easy way of stemming for the varying form of Tigrigna words to their root stem. It can be concluded that transliteration to Latin

script is better than using without transliteration of the script for development of a system.

The evaluation for stemming and stopwords elimination has been done by taking simple random sample of 20 terms, and are examined carefully weather these terms are properly stemmed or are not stop words. The evaluation of this component of the system indicates that majority of the terms are properly stemmed and they are not stopwords. According to this evaluation 75% of the terms are properly stemmed and 85% of the stopwords are properly eliminated. Stop word removal is done after stemming is carried out. The compiled stopwords loose their original form because the affixation rule has applied on them. This over stemming or under stemming of stopwords can be controlled by adding the compiled stopwords list to exception list so as not to apply affixation rule for them. In order to improve the performance of automatic generation of the thesaurus terms, the stemmer and the stopwords removal part of the system should be improved and recommended for further research work in the area.

For the purpose of evaluating the accuracy of the thesaurus generation system a simple random sample of ten stemmed terms are selected and cross checked in the system. The result of the finding shows 75.28% of the output has related concepts to the respective terms. The rest 24.72% has no related concepts. The improperly stemmed terms and the existence of some stopwords that are not detected in the document collection may result for this irregularity. If the stemmer component is well doing (properly developed) and the document properly stemmed and all the stopwords removed during the preprocessing stage, a better output would have been expected.

5.2 Recommendation

Based on the finding of this research work, the following recommendations are proposed for future work:

- Construction of thesaurus by combination of manual and automatic methods
- Development of domain dependent automatic thesaurus generation.

- Development of efficient method of eliminating stopwords (non-content bearing terms).
- Development of standard Tigrigna stemmer.
- Development of Tigrigna spelling and inconsistency checker.
- Development of standard Tigrigna corpus that can be used for training and testing for systems development.
- A thesaurus that consider and control dialectical variations of Tigrigna language.

REFERENCES

Ahsan-ul, M. and Margherita, S. (n.d.). Creating and Aligning Controlled Vocabularies. University of Organization of the United Nations (FAO), Rome, Italy.

Ali, A. S. Crawford, R. Gobinda, C. (n.d.). Thesaurus-assisted search term selection and query expansion: A review of user-centred studies. University of Strathclyde, 26 Richmond Street, Glasgow G1 1XH Scotland, UK.

Andargachew, M. (2009). Automatic thesaurus construction for Amharic Text Retrieval. Masters Thesis, Addis Ababa University, Addis Ababa, Ethiopia.

Angel, F. Carlos, G. Figuerola, Jos´e, L. and Berrocal, A. (n.d.). Experiments in Term Expansion Using Thesauri in Spanish. Grupo de Recuperaci´on Automatizada de la Informaci´on (REINA)

Anne, S. and Pertti, V. (2004). Subject Knowledge, Thesaurus-assisted Query Expansion and Search Success, University of Tampere (FIN-33014), Finland.

Atelach, A. and Lars, A. (2007). An Amharic Stemmer: Reducing Words to their Citation Forms. Stockholm University/KTH, Sweden, Proceedings of the 5th Workshop on Important Unresolved Matters.

Baeza-Yates and Ribeiro-Neto, (1999). Modern Information Retrieval. ACM press.

Barbara, R. (2000). Latent Semantic Indexing: An overview. INFOSYS 240 Spring, Final Paper.

Christopher, D. M., Prabhakar, R. & Hinrich, S. (2009). An Introduction to Information Retrieval. Cambridge University Press (Online edition.). England.

Dagobert, S. (n.d). The Arts and Architecture Thesaurus (AAT) A critical appraisal. University of Maryland

David, A.G. and Ophir, F. (1998). Information retrieval: algorithm and heuristics, Springer, 2nd ed.

Dongqiang, Y. and David, M. (2008). Automatic Thesaurus Construction. *Conferences in Research and Practice in Information*. Flinders University of South Australia, Wollongong, Australia. Technology (CRPIT), Vol. 74.

Felix, C., Ben, K. David, C., and Chi-Yuen, N. (2002). An Efficient Algorithm for

Incremental Update of Concept Spaces, University of Hong Kong, Springer-Verlag Berlin Heidelberg.

George, T. and Vicky, P. (n.d.). A Generalized Vector Space Model for Text Retrieval Based on Semantic Relatedness. Athens, Greece.

Gerda, R. (n.d). Automatic Detection of thesaurus Relations for Information Retrieval Applications,Thechnical University of Munich.

Girma, B. (2001). A Steming Algorithm Development for Tigrigna language text documents. Masters Thesis, Addis Ababa University, Addis Ababa, Ethiopia.

Habert, Adda, Adda-Decker, Boula de Mar¨euil, Ferrari, O. Ferret, G. Illouz, Paroubek, (n.d). Towards Tokenization Evaluation. LIMSI-CNRS, F-91403 Orsay Cedex, France.

Hayel, K. Nidal ,Y. Ghassan, K. (n.d.). Automatic query expansion for Arabic text retrieval based on association and similarity thesaurus. Zarqa Private University, Zarqa, King Abdulaziz University, Jeddah, The Arab Academy for Banking and Financial Sciences, Amman.

Hazra, I. and Aditi, S. (2009). Thesaurus and Query Expansion: *International Journal of Computer science & Information Technology* (IJCSIT), Vol 1, No 2.

Jane, G. (2004). User Comprehension and Searching with Information Retrieval Thesauri. University of North Carolina at Chapel Hill, Haworth Press.

Jean-Pierre, C. (n.d.). Building Thesaurus from Manual Sources and Automatic Scanned Texts, Laboratoire CLIPS-IMAG, France.

Jesper, W. (2004). Verification of bibliometric methods' applicability for thesaurus Construction. PhD thesis, Royal School of Library and Information Science, Denmark.

Jinxi, X. (1997). Solving Word Mismatch Problem Through Automatic Text Analysis. Dissertation, University of Massachusetts Amherst, Chinese Academy of Science.

Jos´e, R. and Lourdes A., (2006). Query Expansion with an Automatically Generated Thesaurus. Departamento de Sistemas Inform´aticos Programaci´on.Universidad Complutense de Madrid. Spain.

Kanyarat, L. (2003). Automatic Thesaurus Construction with Term Context and Syntactic

Analysis for Thai Text Retrieval. Masters Thesis, Mahithodol University.

Karen, S. J. and Peter, W. (Eds). (1997). Readings in information retrieval, Morgan Kaufmann publisher, Inc., An Imprint of publisher, San Francisco, California.

Katy, B. Chaornei, C. Kevin, W. Boyack, (n.d.). Visualizing Knowledge Domains: Annual Review of Information Science and Technology, Chpter 5.

Khurshid, A. Mariam, T. Bogdan, V. and Chris, H. (n.d). Corpus-Based Thesaurus Construction for Image Retrieval in Specialist Domains. Universityof Surrey, Guildford, GU2 7XH, United Kingdom.

Kotaro, N. Takahiro, H. and Shojiro, N. (2007). Wikipedia Mining for an Association WebThesaurus Construction. Osaka University, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan.

Libo, C. (2006). Automatic Construction of Domain-Specific Concept Structures. Dissertation , Geboren In Peking, China.

Maziar, A. (2008). Dialectic Schemes in Thesaurus Creation, Library Philosophy and Practice. AJTKeshavarz Blvd, ISSN 1522-0222 ,Tehran, Iran.

Mekelle University, (2010). Brief Historical background. http://www.mu.edu.et/index.php?option=com_content&view=article&id=718&Itemid=706, accessed date 24/01/20011

Michael, G. (2009). Semitic Morphological Analysis and GenerationUsing Finite State Transducers with Feature Structures. Indiana University, USA, Proceedings of the 12th Conference of the European Chapter of the ACL, Association for Computational Linguistics, Athens, Greece.

Monica, L. (2002). Automatic Thesaurus Construction, Graduate School of Language Technology, Swedish School of Library and Information Science, University College of Borås.

Mulu, G. (2001). Development English-Tigrigna Machine readable bi lingual Dictionary: An input for Machion Translation. Masters Thesis, Addis Ababa University, Addis Ababa, Ethiopia.

Nega, A. and Peter, W. (2003). The Effectiveness Of Stemming For Information Retrieval In Amharic, Program: *electronic library and information systems,* Volume 37 ·

Nurazzah, A. Zainab, A. Tengku, M. (2010). Query Expansion using Thesaurus in Improving Malay Hadith Retrieval System. Universiti Teknologi MARA, Shah Alam, Malaysia, National Defense , University of Malaysia, Kuala Lumpur. IEEE.

Olena, M. Ian, H. Witten (2006). Thesaurus Based Automatic Keyphrase Indexing, University of Waikato, Hamilton, New Zealand.

Omniglot (n.d). Writing Systems and Languages of the World. http://www.omniglot.com/writing/tigrinya.htm accssed date 12/31/2010

Phil, C. (2011). Thesaurus Guidance Manual: HEREIN Thesaurus Editing Software, Final Draft. http://www.coe.int/t/dg4/cultureheritage/heritage/herein/THES-GManual-2011_en.pdf accessed date 15/3/2011.

Robert, M. (2009). Text-Mining for Semi-Automatic Thesaurus Enhancement, Diploma Thesis, University Mannheim, Matriculation Number 1020135.

Ronan, C. and Colm, O. (n.d). Evolving Co-occurrence Based Query Expansion Schemes. National University of Ireland, Galway, Ireland.

Salton, G. (1983). Introduction to Modern Information Retrieval, McGraw-Hill.

Schubert, F. Siu, C. Hui, Hong, K. Lim, L. (2000). Automatic thesaurus for enhanced Chinese Text Retrieval; *Library Review*, Volume 49, Number 5, China

Sebastian, M. K. (2004). Similarity Thesauri and Cross-Language Retrieval: Seminar 'Text Mining', Rheinische Friedrich-Wilhelms-Universität Bonn.

Thomas and Wondwosen, T. (1996). Issues in Ethiopian Language Policy and Education: *Journal of Multilingual and Multicultural Development*. Vol. 17, No. 5.

Tom, S. Bellomo, (2009). Morphological Analysis and Vocabulary Development: *Critical Criteria*,Volume 9, Number 1.

Turid, H. (2000). Dictionary-Based Cross-Language Information Retrieval, Principles,System Design and Evaluation. Academic Dissertation, University of Tampere, Acta Universitatis Tamperensis 962, Tampere.

Yousef, A. and Fernand, V. (2002). "ThesWB: A Tool for Thesaurus Construction from HTML Documents", Accepted in Workshop on Text Mining Held in Conjunction with the 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD-2002), Taipei, Taiwan.

Wei, S. and Chenghua, L. (2006). Intelligent Information Retrieval System Using Automatic Thesaurus Construction, (Received 20 July 2005; final version received 17 August 2006), China.

Witten, I. H. and Frank, E. (1999). Data Mining: Practical Machine Learning Tools and Technique, 1st edition, Morgan Kaufmann, San Francisco, CA.

Worku, K. (2009). Automatic Amharic Text News Classification: A Neural Networks Approach. Masters Thesis, Addis Ababa University, Addis Ababa, Ethiopia.

APPENDICES

Appendix1: Ethiopic numbers

፩	1
፪	2
፫	3
፬	4
፭	5
፮	6
፯	7
፰	8
፱	9
፲	10
፳	20
፴	30
፵	40
፶	50
፷	60
፸	70
፹	80
፺	90
፻	100
፼	1000

Appendix 2: Tigrigna letters

	e	**u**	**i**	**a**	**ie**	**0**	**o**
h	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
l	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
ḥ’	ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ
m	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
S’	ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ
r	ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ
s	ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ
x	ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ
q	ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ
kʷ'	ቈ		ቊ	ቋ	ቌ	ቍ	
q'	ቐ	ቑ	ቒ	ቓ	ቔ	ቕ	ቖ
xʷ'	ቘ		ቚ	ቛ	ቜ	ቝ	
b	በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ
v	ቨ	ቩ	ቪ	ቫ	ቬ	ቭ	ቮ
t	ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ
č	ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ
h	ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ
n	ነ	ኑ	ኒ	ና	ኔ	ን	ኖ
ñ	ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ
e	አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ
k	ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ
kʷ	ኰ		ኵ	ኳ	ኴ	ኸ	
K’	ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ
K’w	ዀ		ዂ	ዃ	ዄ	ዅ	
w	ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ
eʿ	ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ
z	ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ
zh	ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ
y	የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ
d	ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ
j	ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ
g	ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ
gʷ	ጐ		ጕ	ጓ	ጔ	ጕ	
t'	ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ
c'	ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ
p'	ጰ	ጱ	ጲ	ጳ	ጴ	ጵ	ጶ
ts	ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ
ts'	ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ
f	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ
p	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ

Appendix 3: Affixes

affixes								
prefix				suffix				
ን	ብ	ዘይ	እንት	ን	ና	ታት	ዖም	ኣ
በብዝ	ከከም	ምስ	ስለ	ኣዊት	ዎም	ዊ	ኪ	ኡ
ከካብ	ነናብ	ዝተ	ኣይነኒ	ኣት	ኩም	ውቲ	ላ	ሎም
በቢ	ነኒ	ብብ	ከኪ	ዋይ	ኡ	ዎ	ካን	ናት
ብመላእ	ከይ	እንተ	እንተዘይ	ዋ	ካ	ኣዊነት	ኩምን	ኩም
ከይን	ዘይን	ኣይ	እና	ኦም	ሎም	ዎ	ኩም	አን
እናተ	ከም	ነናብ	ለሊ	ኣዊያን	ኣዊ	ኩምን	ነቶም	ኺ
					ካን	ኻ	ኣይ	ኦት

Appendix4: Stopwords List

ኣብ
ኣነ
ኣበይ
ኣየን
እቲ
እታ
እዩ
እናንተ
እና
እዮም
እቶም
እሞ
ኣላ
እዙይእውን
ኣሎ
ኢና
ኣብዚ
እዚ
እያ
እያተን
እያቶም
እቶም
እየ
እየን
እውን
እዙይ
እምበር
እንተኮነውን
እንተኮነግን
እንተኮነግና
እንተይኮነስ
እስኪ
እምበር
ኢሎም
እምቢ
እሺ
እንደገና
ኢና
ኢሎም
ኢሉ
ኢልና
ኢለን
ኢልክን
ኢልኩም
ኢለ
ኢላተን
ኢላትኩም
እንተሎ
እንተለኩ
እንተለኪ
እንተለኩም
እንተለው
እንተለክን
እንተለና
እንተለካትኩም
እንትባሃል
እንተኮይኑ
እንተኮይነ
እንተኮይና
እንተኮንኩም
እንተኮይኖም
እንተኮይንካ
እንተኮይንኪ
እንተኮይነን
እንተኮይንና
እንታይ
እቶም
እቲ
እታ
እቲኣ
እቲኦም
እቲኣቶም
እቲኣተን
እቲአን
እቱይ
እዙይ
እዚኣ
እዞም
እዚኣተን
እዚኣቶም
እዚ
እዚኣ
እዚኦም
እዚአን
እንታዋይ
እንታወይቲ
እንታዮት
እከለ
እከሊት
እገለ
እገሊት
ኣባይ
ኣባኪ
ኣባካ
ኣባክን
ኣባኩም
ኣብአን
ኣብኦም
ኣባና
ኣብኣቶም
ኣብኣተን
ኣብዙይ
ኣብ'ዚ
እንትከውን
ኣነ
እዚ'ዩ
እንታይነት
ኣዝዩ
እንተድኣ
ኣነ
እወ
እንኮ
እውን
ኣበይ
ኣብኡ
ኣብዛ
ኣብዘሓ
ካብ
ካሊእ
ክንደይ
ክንዲ
ከም
ከከም
ክብል
ኸሊ
ከምቲ
ከምናቶም
ከምናታ
ከዓ
ካብዚ
ከምዛ
ከምቲ
ከምተን
ከምቶም
ከምታ
ከምቲኦም
ከምቲአን
ከቶ
ክሳብ
ክንዲ

ክስይ
ኩሎም
ኩልና
ኩልክን
ኩላትና
ኩላቶም
ኩለኩም
ኩለካትክን
ኩላተን
ከመይ
ካሊእ
ካልኦት
ከምዚ
ከምኡ
ከማይ
ከምኣ
ከማና
ከማኩም
ከማክን
ከምኦም
ከማካ
ከማኪ
ከምኣቶም
ከምኣተን
ክስቶ
ክስታይ
ከምዚኦም
ከምዚአን
ከምአን
ከማና
ከማካ
ከማኪ
ከምዙይ
ከምዚኣ
ከምዚኣቶም
ከምኡ
ኩልክን
ካብ'ታ
ከዓ
ኮነ
ካብ'ቶም
ክንደይ'ኳ
ክንደይ
ኩሎም
ክንድቲ
ነቲ
ናይ
ነቶም
ናብ
ነቱይ
ነታ
ነቶም
ነዚ
ናይቲ
ናቱ
ናተይ
ናታ
ናተን
ናትና
ናቶም
ነጀው
ናታተን
ናታቶም
ናትኪ
ናትካ
ናትኩም
ናታትኩም
ናታተን
ናትክን
ንስካ
ንሱ
ንሳ
ንሶም
ንሕና
ንሳቶም
ንሳተን
ንሳ
ንሱ
ንስካ
ንስኪ
ንሕና
ንስክን
ንሳቶም
ንሶም
ንሰን
ንሳተን
ንስኩም
ንምንታይ
ንመን
ናይመን
ነቲ ናታ
ነቶም
ነተን
ናፍቲ
ናፍቶም
ናፍተን
ናባይ
ናባኪ
ናባና
ናባካ
ናብኣ
ናብኡ
ናብኦም
ናብአን
ናባክን
ናባኩም
ናብኣቶም
ናብኣተን
ንሳ
ንሱ
ንስኻ
ንስኺ
ንሕና
ንዓይ
ንዓኪ
ንዓካ
ንዑኡ
ንዓኣ
ንዓና
ንዓኹም
ንዓኣቶም
ንዓካትክን
ንዐአን
ናብዚኣ
ናብዚኦም
ናብዚአን
ናብኣቶም
ነዛ
ነዚ
ነዙይ
ነዚኣ
ነዚኦም
ነዚአን
ነዚኣቶም
ነዚኣተን
ነዘን
ነዞም
ንዞም
ንዘን
ናይዚ
ናይዛ
ናይዞም

ናይዘን
ናብ'ቶም
ንቶም
ንተን
ናታትክን
ነናይ
ንብምሉኦም
ነናተን
ድሕሪ
ድማ
ደኣ
ምስ
ማለት
ማለታ
ማለቱ
ማለተይ
ማለትኪ
ማለትኩም
ማለተን
ማለትክን
ማለትና
ማለቶም
ማለትካ
መዓዝ
መን
ግን ገና
ግና
ግዳ ገለ
ገለ
ሕጂ
ሕዚ
ሓልሓሊፉ
ሕድሕድ
ሓደ'ኳ
ሓደሓደ
ስለዝኮነ
ስለዙይ
ስለዚ
ስለዚዝኾነ
የለን የላን
የለኩን
የለካን
ይኹን
የለናን
የለውን
የለዋን
ይኩንደኣምበር
ይኩንምበር
ምሳይ
ምሳኪ
ምሳካ
ምሳኩም
ምሳና
ምስኦም
ምስኣቶም
ምስአን
ምሳክን
ምሳካትክን
ምስመን
ምስኣ
ምስኡ
ምስቶም
ምስታ
ምስቲ
ምስተን
ምእንቲ'ዚ
ምስ'ቲ
ምስዚ ምስ
ማንም
ምስምስ
ምንታ'ይሲ
መዓዝ
ዘሎ
ዘላ ዘለዋ
ዘለዉ
ዘለካ
ዘለኩ
ዘለኩም
ዘለኪ
ዘለክን
ዘይብሉ
ዘይብላ
ዘይብለይ
ዘይብልካ
ዘይብልኪ
ዘይብሎም
ዘይብለን
ዘይብልና
ዘይብልኩም
ዘይብልክን
ዝያዳ
ብተወሳኪ
ብኣምሆይ
ብዘይ
ብዘይካ
በቲ
በታ
በቶም
በተን
ብቶም
ብተን
በቲኦም
በቲኣ
በቲአን
ብኣና
ብኣካ
ብኣኪ
ብኣይ
ብኣኩም
ብኣክን
ብእኦም
ብኣተን
ብአን
ብኣካትኩም
ብኣካትክን
ብኩሎም
ብኩልና
ብኩላኩም
ብኩልክን
በብሓደ
ብ'ሽም'ዚ
ብዘይካ ብኡ
ብሰንኪ
ውን
ወትሩ
ወይዉን
ወይ'ዉን
ወይከ ወይ
ወላእኳ
ዋላ
ጥራይ
ጥራሕ
ቅድም
ቅድሚት
ቅድሚ
ድሕሪ
ድሕሪሕዚ
ዓንተም
ዘለና
ኩሉ
እንተላ

Appendix 5: Tigrigna Script Translation to Latin Script for Preprocessing Purpose

ሀ	he	ደ	de	ኑ	nu	ሢ	si	ጨ	Ci
ለ	le	ጀ	je	ኙ	Nu	ሪ	ri	ጲ	Pi
ሐ	He	ገ	ge	ኡ	`u	ሲ	si	ጺ	Si
መ	me	ጠ	Te	ኩ	ku	ሺ	xi	ፂ	Si
ሠ	se	ጨ	Ce	ኹ	Qu	ቂ	qi	ፊ	fi
ረ	re	ጰ	Pe	ዉ	wu	ቒ	Qi	ፒ	pi
ሰ	se	ጸ	Se	ዑ	`u	ቢ	bi	ፃ	ha
ሸ	xe	ፀ	Se	ዙ	zu	ቪ	vi	ላ	la
ቀ	qe	ፈ	fe	ዠ	Zu	ቲ	ti	ሓ	Ha
ቐ	Qe	ፐ	pe	ዩ	yu	ቺ	ci	ማ	ma
በ	be	ሁ	hu	ዱ	du	ኒ	ni	ሣ	sa
ቨ	ve	ሉ	lu	ጁ	ju	ኚ	Ni	ራ	ra
ተ	te	ሑ	Hu	ጉ	gu	ኢ	`i	ሳ	sa
ቸ	ce	ሙ	mu	ጡ	Tu	ኪ	ki	ሻ	xa
ነ	ne	ሡ	su	ጩ	Cu	ኺ	Ki	ቃ	qa
ኘ	Ne	ሩ	ru	ጱ	Pu	ዊ	wi	ቋ	Qa
አ	`a	ሱ	su	ጹ	Su	ዒ	`i	ባ	ba
ከ	ke	ሹ	xu	ፁ	Su	ዚ	zi	ቫ	va
ኸ	Ke	ቁ	qu	ፉ	fu	ዠ	Zi	ታ	ta
ወ	we	ቈ	Qu	ፑ	pu	ዪ	yi	ቻ	ca
ዐ	`e	ቡ	bu	ሂ	hi	ዲ	di	ና	na
ዘ	ze	ቩ	vu	ሊ	li	ጂ	ji	ኛ	Na
ዠ	Ze	ቱ	tu	ሒ	Hi	ጊ	gi	ኣ	`a
የ	ye	ቹ	cu	ሚ	mi	ጢ	Ti	ካ	ka

ኻ	Ka	ቂ	qE	ፈ	fE	ዥ	Z	ቮ	vo
ዋ	wa	ቿ	QE	ፕ	pE	ይ	y	ቶ	to
ዓ	`a	በ	bE	ህ	h	ድ	d	ቾ	co
ዛ	za	ቨ	vE	ል	l	ጅ	j	ኖ	no
ዣ	Za	ቱ	tE	ሕ	H	ግ	g	ኞ	No
ያ	ya	ቼ	cE	ም	m	ጥ	T	ኦ	`o
ዳ	da	ኔ	nE	ሥ	s	ጭ	C	ኮ	ko
ጃ	ja	ኜ	NE	ር	r	ጵ	P	ኾ	Ko
ጋ	ga	ኤ	`E	ስ	s	ስ	s	ዎ	wo
ጣ	Ta	ኬ	kE	ሽ	x	ጽ	S	ዮ	`o
ጫ	Ca	ኼ	KE	ቅ	q	ፅ	S	ዞ	zo
ጳ	Pa	ዌ	wE	ቈ	Q	ፍ	f	ዠ	Zo
ጻ	Sa	ዔ	`E	ብ	b	ፕ	p	ዩ	yo
ፃ	Sa	ዜ	zE	ቭ	v	ሆ	ho	ዶ	do
ፋ	fa	ዤ	ZE	ት	t	ሎ	lo	ጆ	jo
ፓ	pa	ዬ	yE	ች	c	ሐ	Ho	ጎ	go
ሄ	hE	ዴ	dE	ን	n	ሞ	mo	ጠ	To
ሌ	lE	ጄ	jE	ኝ	N	ሦ	so	ፎ	Co
ሐ	HE	ጌ	gE	እ	`I	ሮ	ro	ጰ	Po
ሜ	mE	ጤ	TE	ክ	k	ሶ	so	ጸ	So
ሤ	sE	ጩ	CE	ኽ	K	ሾ	xo	ዾ	So
ሬ	rE	ጴ	PE	ው	w	ቆ	qo	ፎ	fo
ሴ	sE	ጼ	SE	ዕ	`I	ቾ	Qo	ፖ	po
ሼ	xE	ፄ	SE	ዝ	z	ቦ	bo	ኳ	lWa

ሓ	HWa	ጸ	SWa	፥	:	0	0
ሟ	mWa	ፏ	fWa	፡-	:-	1	1
ሣ	sWa	ፐ	pWa	፧	?	2	2
ሯ	rWa	ቈ	qWu	?	?	3	3
ሷ	sWa	ኈ	hWu	/	/	4	4
ኧ	xWa	ኰ	kWu	:	:	5	5
ቈ	qWe	ጐ	gWu	.	.	6	6
ቧ	bWa	ቊ	qWi	"	"	7	7
ኗ	vWa	ኈ	hWi	-	-	8	8
ቷ	tWa	ኰ	kWi	#	#	9	9
ቿ	cWa	ጐ	gWi	$	$		
ኈ	hWe	ቋ	qWa	%	%		
ኗ	nWa	ኋ	hWa	*	*		
ኟ	NWa	ኳ	kWa	(	(		
ኰ	kWe	ጓ	gWa	)	)		
ዟ	zWa	ቌ	qWE	_	_		
ዣ	ZWa	ኌ	hWE	+	+		
ዷ	dWa	ኴ	kWE	=	=		
ጇ	jWa	ጌ	gWE	\	\		
ጐ	gWe	ኧ	ea	{	{		
ጧ	TWa	።	.	}	}		
ፏ	CWa	፡	.	<	<		
ጳ	PWa	፣	,	>	>		
		፤	;	~	~		

Appendix 6. Sample exception list

ማይ	ዕደላ	ሓደጋ
ዕደጋ	ሓለዋ	ሒዝዋ
ዋና	ኔትዎርክ	እንጀራ
ጥዕና	ኣረዳድኣ	ኢንቨስትመት
ኣራኣእያ	ግልጋሎት	ኮሚሽን
ብዕራይ	መኪና	ብርቂ
ብልዒ	ዋና	ብርቱዕ
ስልኪ	ብራና	ብቕዓት
ብርሃን	ብልሓት	ባንኪ
ካዝና	ሑመራ	

Appendix 7: Sample code from the system

```
 import defaultdict      ## importing the Python module
import  numpy            ##importing  Numpy an extension of Python
“”” similarity matrix calculation”””
def similarity_matrix(vocabulary, d):  ## vocabulary -- a list of words derived from the keys of d
        cm = defaultdict(dict)
         vectors = get_vectors(d, vocabulary)
         for word1 in vocabulary:
                for word2 in vocabulary:
                        “””cm -- cosine similarity matrix “””
                                cm[word1][word2] = cosim(vectors[word1],
vectors[word2])
        return cm
def get_vectors(d, vocabulary):   ## adjust the document terms array
        vecs = {}
        for word1 in vocabulary:
                 v = []
                for word2 in vocabulary:
                        wA, wB = sorted([word1, word2])
                         v.append(d[wA][wB])
                vecs[word1] = array(v)
         return vecs
"""Cosine similarity between the two vectors vector1 and vector2."""
def cosim(vector1, vector2):
            numerator = dot(vector1, vector2)
```

```
        denominator = sqrt(dot(vector1, vector1)) * sqrt(dot(vector2, vector2))
        if denominator >0:
            return numerator/denominator
        else:
            return 0.0
def  format_matrix(vocabulary, m):
        s = ""
        sep = ""
        vocabtigrgna=[]
        col_width = 10
        for j in vocabulary:
            f=de.deTransliterate(j)          #  de transliterator call(latine to Ethiopic)
            vocabtigrigna.append(f)
       s += "  ".rjust(col_width) + sep.join(map((lambda x : x.rjust(col_width)),
     vocabtigrigna)) + "\n"
        for word1 in vocabulary:
            row = []
            row += [round(m[word1][word2], 2) for word2 in vocabulary]
             f = de.deTransliterate(word1)
            row.insert(0,f)
            s += sep.join(map((lambda x : unicode(x).rjust(col_width)), row)) + "\n"
      return s
  d = cooccurrencem.cooccurrence_matrix(corpus) # module cooccurrence calling
  “””” Sort the entire vocabulary (keys and keys of their value dictionaries).”””””
  vocab = cooccurrencem.get_sorted_vocab(d)
  cm = cosinec.cosine_similarity_matrix(vocabulary, d)
```

```
format_matrix(vocabulary, d)
“”"Cosine similarity matrix call"””
 format_matrix(vocabulary, cm)
```

Declaration

I declare that the thesis is my original work and has not been presented for a degree in any other university.

_____

Date

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

_____

Advisor