QUERY EXPANSION FOR SIDAMA LANGUAGE INFORMATION RETRIEVAL SYSTEM

MSc THESIS

AEMRO NOKOLA MASALE

ADVISOR: Dr. ANITHA VARGANTHAM

HAWASSA UNIVERSITY INSTITUTE OF TECHNOLOGY FACULTY OF INFORMATICS, DEPARTMENT OF COMPUTER SCIENCE

MAY, 2024

HAWASSA, ETHIOPIA

QUERY EXPANSION FOR SIDAMA LANGUAGE INFORMATION RETRIEVAL SYSTEM

AEMRO NOKOLA MASALE

ADVISOR

ANITHA VARGANTHAM

THESIS SUBMITTED TO HAWASSA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE INSTITUTE OF TECHNOLOGY, SCHOOL OF GRADUATE STUDIES, HAWASSA UNIVERSITY, HAWASSA, ETHIOPIA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTERS OF SCIENCE IN COMPUTER SCIENCE

MAY, 2024

DECLARATION

I hereby declare that this MSc Specialty or equivalent thesis is my original work

And has not been presented for a degree in Hawassa University, and all sources of

Material used for this thesis/dissertation have been duly acknowledged.

Name: - AEMRO NOKOLA MASALE

Signature: _____

This MSc Specialty or equivalent thesis has been submitted for examination with

My approval as thesis advisor.

Name: _____

Place and date of submission: _____

SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

ADVISORS' APPROVAL SHEET

This is to certify that the thesis entitled "Query expansion for Sidama language information retrieval system" was submitted in partial fulfillment of the requirements for the degree of Master's Degree with specialization in Computer Science for, the Graduate Program of the Department Computer Science, and has been carried out by

AEMRO NOKOLA (ID NO.: GPCoScR/0002/14), is conducted under my supervision.

Therefore, I recommend that the student has fulfilled the requirements and hence hereby can submit the thesis to the Department.

Dr ANITHA VARGANTHAM.	_____	May 2024
Name of major advisor	Signature	date
_____	_____	May 2024
Name of major advisor	Signature	date

ACKNOWLEDGEMENTS

This thesis would not have been possible without the help and support of God. So, first and foremost, I would like to give plenty of gratitude, praise, and thanks to Almighty God for giving me strength, peace and security throughout my ups and downs on this research journey. I would like to express my profound and sincere gratitude to my research advisor, Dr. Anitha Vargantham for her beneficial comments, strong commitment and guidance to direct me to strongly do this thesis work using my ability. Her comments strengthen me. Dr. Anitha is a really gifted academic, who was passionately guide and shape researchers, understanding the ideas of the researchers by reviewing their documents line by line and word by word without being tired. Her vision, vitality, genuineness and motivation have profoundly inspired me. It was a countless integrity and pleasure to work and study under her guidance. I am very thankful for what she has suggest me. I would also like to thank her for her friendship, kindness and compassion during the research work. I would like plenty of gratitude to my relative wife Azrit Ashenaf and my lovely children Enjonkena, Alibray, Sinard and the newly coming baby Lexita for their endless love, caring prayers, support and encouragement, me to achieve an excellent accomplishment. In particular, I haven't worded to explain Azrit Ashenaf paid sacrifices for educating and preparing me for my future. Since it is difficult to mention her contribution to my achievements in words, it is better to say my heart recorded it forever. Additionally, I would like to extend my sincere gratitude to my mom Ayantu Kie, my elder brother Damtew Nan and his wife Esen KushaKer and special Asmamaw Alemayew he supports me by delivering different optimistic support. I would like to thank Hana Kayamo and Tatek Zerihun and his wife Gen. They support me by providing optimistic ideas. Last but not least, my great thanks go to Sidama Region national state vise president Mis Beyene Barasa. He provided me with all the necessary funds to achieve my MSc program, and he taught me to gain prolific knowledge and skills. Finally, I would like to thank all the people who gave me optimistic support to complete this research work directly or indirectly.

Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS (NOMENCLATURE)

Abbreviations	Full Representation
AQE	Automatic Query Expansion
CSA	Central Statistical Agency
CSM	Cosine Similarity Measure
DBMS	Data Base Management System
DM	Data Mining
DCM	Dice Coefficient Measure Similarity Measure
ED	Euclidian Distance
FBCH	Faith-comes-by-hearing
HCTE	Hawassa Collage of Teachers Education
IDF	Inverse Document Frequency
IQE	Interactive query expansion
IR	Information Retrieval
JSM	Jaccard Similarity Measure
ML	Machine Learning
MQE	Manual query expansion
MSc	Master of Science
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
PCC	Pearson Correlation Coefficient
PM	Probabilistic Model
QE	Query Expansion
SNNPR	Southern Nations Nationalities and Peoples Republic
SRSEB	Sidama Regional state Education Bureau
STVC	Sidama Television Corporation
SZEB	Sidama Zone Education Bureau
TF-IDF	Term Frequency Inverse Document Frequency
VSM	Vector Space Model

ABSTRACT

Information retrieval has become vital research topic in this computing era. Information retrieval is the process of searching and retrieving knowledge-based information from database. Wide range of the users are using some IR on an everyday activity. However, IR still have different challenges such as, short user query expression, ambiguity nature of the natural language and the vocabulary mismatch between query terms and relevant documents which are resulting decreased IR systems efficiency. The goal of this study was to design and develop manual-query expansion for Sidama language IR using Vector Space Model to improve Sidama language IR system by minimizing short query and query-document mismatch problems. To attain the abovementioned goal, we have studied several IR based literatures, get better understanding about IR, searching models, indexing, query expansion techniques and fundamental Sidama language morphology and language structure. We have designed the manual query expansion for Sidama IR in two subsections. The first subsection performs text preprocessing and indexing using *inverted file indexing* technique. The second subsection performs the comparison, query expansion, searching and ranking according to cosine similarity measurement. The implementation was done using Python programming language. In order to measure an implemented prototype system, we have collected 500 documents from different sources as document corpus and 20 initial queries. Query-document relevance judgement was done manually by domain experts and documents are categorized according to query. Two experimentations have done for each of twenty queries. First experimentation was done searching with initial user query (without query expansion). The second experiment has been done searching with expanded query (with query expansion). The performance measurements have been calculated using common efficiency measurement units such as precision, recall, and F-measure. During the first experimentation, we recorded the results for each of 20 initial queries and obtained the average precision values of 67.86%, average recall value of 66.53%, and average F-measure value of 65.66%. The second experimentation has been performed using manual query expansion and obtained results were 75.73% average precision, 96.08% average recall and 83.68% average F-measure. As we can see the average results of the two experimentations, a significant improvement has been recorded in the second experimentation (manual query expansion-based searching). An average precision, recall and f-measure values are increased by 7.87%, 29.55%, 18.02% respectively in the second experimentation than the searching results in the first experimentation. Greater improvement has been seen in recall value, which indicates that almost all relevant documents in the corpus have been successfully retrieved during query expansion searching. Finally, we can conclude that, the proposed manual query expansion-based searching results in greater improvement than searching without query expansion. But the lack of rule based stemming algorithm was the main issue that diminish the performance, in future need further studies.

Key Words: Information Retrieval (IR), Vector- Space- Model (VSM), Indexing, Query expansion, Searching.

CHAPTER ONE

1. INTRODUCTION

1.1. Background

There is massive amount of data existing on Internet because data are being producing now a day by different organizations, corporations, institutions, industries, groups or individuals in many different languages other than English. These produced data are used for different aims, but not all data which placed on computer are equally significant for user's need. Users need to choose the most significant document(s) for their planned use from large placed documents. Discovery useful information very quickly from massive collection of documents which stored for a long period of time in different language remains as challenge still now and needs series attention (Bilal A. A.-S., 2018).

Web-search does not yield relevant results to the user query; there are multiple reasons for this. **Frist** the query can be too short to express appropriately what the user is seeking. **Second,** the user is often not sure about what he is looking for until he sees the results. **Third**, though the user knows what he is seeking for, but he does not know how to express the proper query (Hiteshwar, 2019). **Fourth** an information retrieval system has several language dependent components which need IR integrated knowledge with natural language knowledge; such as stop word removal, stemming, lemmatizing and normalizing tools (Vera, Jaap, Christof, & Maarten, 2004).

Information retrieval system is necessary to search relevant documents from the internet expansion (Thompson, 2007). It also helps to find relevant documents from the internet, which are deposited in Sidama language for the Sidama language users to gratify their information need. Information retrieval is discovery the formless document that satisfies the need of users from within large document collection (Christopher D. Manning P. R., 2008). An information retrieval system is system that supplies and achieves information on documents and also allows users discovery the information they need. It yields documents that contain answer to users question rather than unambiguous answer to their information need. Retrieved documents most of the time gratify users 'information needs. Documents which gratify users 'information requirements are said to be **relevant**

documents, whereas documents which are not sustaining users 'information requirement are called **irrelevant documents** (Christopher D. Manning P. R., 2009)**.**

Information retrieval system includes three subsystems. Those are **indexing, weighting** and **searching**. **Indexing** is an offline process which denoting and establishing large document collection using indexing structure such as Inverted file, sequential files and signature file to save storage memory space and it supported out by the producers or authors to speed up searching of information from the total document as per user's query. **Weighing** is calculating the importance of each index term in document collection to the user query. Because all terms which found in the document(s) are not equally important. **Searching** is the process of checking documents to discovery relevant documents that matches to the user's query or information need. Searching is the process of connecting indexed terms to query terms and return relevant successes to users' query.

These three methods are interconnected to each other for attractive effectiveness and efficiency of IR systems. **Efficiency** is the process of improving calculating resource such as the storage space and time complexity, while **effectiveness** concerned with relevancy of document retrieved that gratifies users 'information requirement. Information retrieval system performance evaluation is stimulating task, because performance of information retrieval is evaluated depending on relevance of retrieved document toward users 'query and efficiency. But it is actually challenging to classify what is relevant from the irrelevant one, because human information demanding behavior is changeable. In order to remove or minimize such kinds of the problem using information retrieval models are necessary. Because Information retrieval models are responsible for determining the prediction of what is relevant and what is not relevant. The role of each information retrieval models are explained in next chapter two.

Even if information retrieval models were developed, each of IR model contains some limitations, which cause disproportion between relevant documents and documents returned by IR system. Information Retrieval (IR) problem has attracted attention, because document increase dramatically. For this problem, diverse researchers have been introducing diverse techniques to attain a good solution. One of the possibilities solutions is reconstruction of the query expansion (Abdelmgeid, 2008).

Sidama Language is one of regional language widely used in politics, education, governmental administration, religious, cultural and trade aspects. Large number of uninstructed documents are there, but almost no information retrieval researches have been undertaken so far. In this study, we are striving to apply manual query expansion technique to design IR system for Sidama language there by solving the word mismatch and short query formulation problems.

1.1. Query Expansion (QE)

Query expansion was first implied as a method for indexing and searching in automatic library system in 1960 by Moron et al (Hiteshwar, 2019). The goal of query expansion is to reduce this query or document disproportion by adding extra terms to the query term which has the same meaning. The objective of our query expansion method is to discovery an appropriate extra term to increase the chance of retrieving relevant documents to the user query and to minimize short user query and word mismatch problems. There are three types of query expansions. These are: manual, automatic, and interactive. Each of these expansions can be based on either of the two types namely based on search results and based on knowledge structures (Abdelmgeid, 2008).

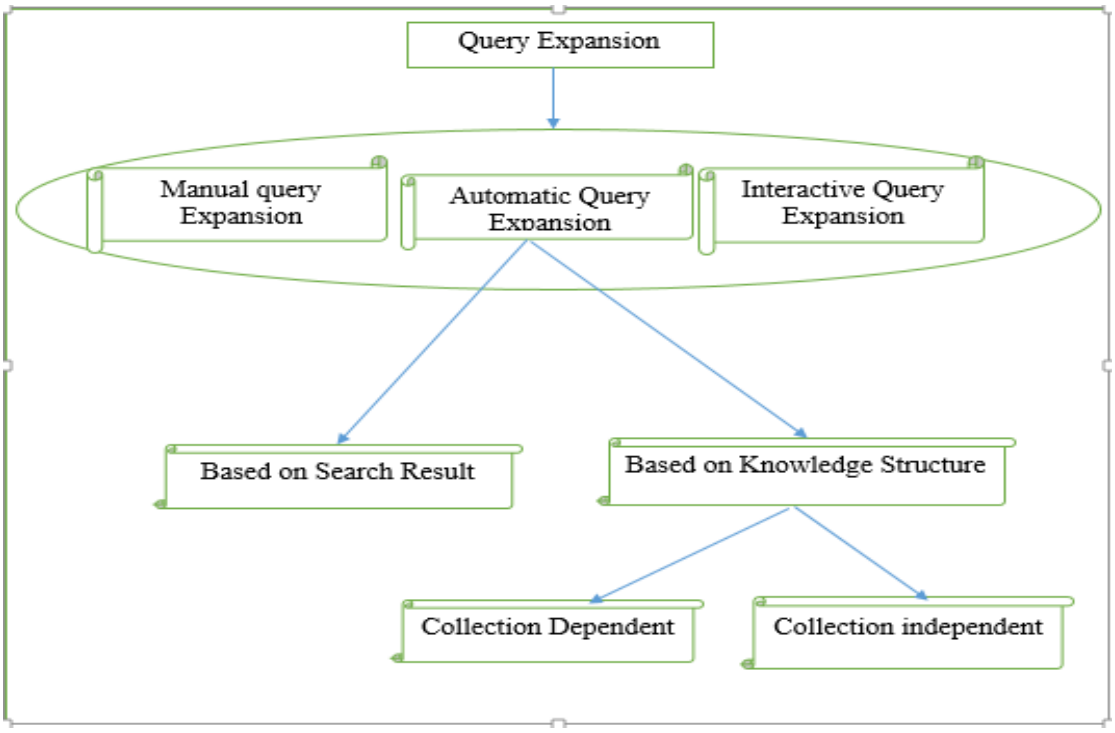

Figure 1. 1: Query Expansion

1.1.1. Manual query expansion (MQE)

Manual query expansion takes place when the user improves the query by adding or deleting search terms without the support of the information retrieval system. New search terms may be known by revising previous retrieval outcomes. Conclusions about the meaning of terms are equal to the users themselves and are dependent on the knowledge of the users with the search system. (In-Ho & GilChang, 2003). Manual query expansion is related with Boolean online and CD-ROM searching. The most significant and vital part to online search is the search plan development. Good search plan development depends on the use of one's knowledge about online searching systems, indexing vocabularies and database structure, it also needs a good realization of the method of the matching model of information retrieval and how this is applied in the system searched (Biruduraju, 2011).

1.1.2. Automatic query expansion (AQE)

In information retrieval system automatic query expansion has extensive history as it has been recommended by Moron and Kuhins in 1960. When the volume of data has dramatically increased while automatic query expansion has been reestablished. In automatic query expansion the system selects and adds terms to the user's query without any user intervention. Automatic query expansion procedure is hidden in the general retrieval procedure where in systems service weighted retrieval methods. Slight increase the number of the query, then the synonymy problem has become even more serious to handling, then which needs to increase the scope of Automatic query expansion (Carpineto C. a., 2012). A massive amount of Automatic query expansion methods has been existing using a variety of methods that influence on numerous data bases and service sophisticated approaches for discovery new structures associated with the query terms in the previous years. Today, there are firmer theoretical foundations and a better understanding of the utility and limitations of Automatic query expansion. In fact, Automatic query expansion has regained much popularity thanks to the evaluation results obtained at the Text retrieval Conference series (TREC) 2, where most participants have made use of this technique, reporting clear improvements in retrieval performance.

1.1.3. Interactive query expansion

When we have been comparing the performance of the interactive query expansion result with automatic query expansion strategy, it can be seen that interactive query expansion has the potential to be the most stable technique overall. Interactive query expansion is effective method because it has greatest potential to gives the highest average precision general and also interactive query expansion technic need to be applied carefully to be effective. Interactive query expansion is query that executed when needed and are usually not repeated on a pre-defined cadence.

Interactive query expansion as contrasting to automatic query expansion there are two things accountable for defining and choosing terms for expansion (Biruduraju, 2011). **First** interactive query expansion retrieval system is similar with automatic query expansion because it designed to choose terms and then weigh and rank them consequently. **Second** user who is presented ranked list of terms has to decide which terms to be added to the search. As it is based entirely on the user's preferences, it is the user's responsibility to determine the final terms and it becomes increasingly difficult to point out the reasons for achievement or letdown because of the irrepressible variables.

Many IR studies have done for developed languages such as English, Arabic, Chinese, French, Spanish, and Japanese. For even small European languages such as Dutch and Finnish, an interesting IR works have done and many language technologies are maintained. (Vera, Jaap, Christof, & Maarten, 2004), (Gezehagn G. E., 2012) . Hence there is a need to design an information retrieval for Ethiopian languages such as Sidama language.

1.2. Motivation

According to the Ethiopia Central Statistics Agency (Samia, 2007) Sidama nation had more than five million Sidama peoples. Hawassa is the capital city of Sidama regional state, therefore large numbers of people they come in and out from Hawassa speak Sidama language as fine. And also Sidama language spoken by several bordering national groups such as: Arsi, Bale, Kanbata, Hadya, Oromo, Gedeo, Wilayat, Alaba, Gurage, etc. who live around Sidama regional state and in Hawassa city Administration (Indrias, Sileshi, Yohannis, & Desalegn, Sidama-amharic-English Dictionary, 2007). According to Ethiopia peoples number sequence Sidama

people are the fifth largest ethnic group next to Oromo, Amhara, Somali, and Tigray peoples. Sidama language is being used as a medium of communication in Sidama regional state and other bordering nations. It is working language in Sidama regional state and the language also given as a subject from the junior up to preparatory schools in Sidama region. Sidama Language is also being given as a field of study of literature and folklores as well as it gives as course in Hawassa University since 2012. There are also a number of Television and radio programs giving the distribution services in the language. Due to these reasons, there are massive groups of documents existing on several desktops in the form of books, magazines, newspapers, Bible, officials and legal documents, etc. Retrieving relevant information from those several warehoused documents now days became a great pain. Manually seeking the essential document from massive warehoused documents is very boring getting worse more effort and time. Massive number of documents written in several languages but IR systems are limited to some well-known languages and which being used frequently by computer experts, researchers, and that language family. Which indicates systems left out many non-well-known languages, so that people who speak only non-famous language are in front of complications in accessing documents written in their mother tongue language. Among these non-well-known languages Sidama language is one in which no IR researches have been done before. According to Bilal, 2018 (Bilal A. A.-S., 2018), it is really important to have an information retrieval system for all languages that donates a lot for the development of once language. So that, this was also another issue that inspired us to design IR for Sidama language as well. Multiuse IR systems such as search engines should help numerous languages in order to plan with documents and user queries written in many diverse languages. Information retrieval systems have several language dependent processes which requires combination of the knowledge of information retrieval systems with natural language knowledge with respect to the specific language. Maximum of IR systems are developed and practical for English Language which donates a lot for the progression of the language. But it continuously wants a great effort to apply algorithms developed for English language to other languages. This is because; every single natural language has its own and extremely diverse morphological appearances, word creation, syntactic rules and features (Senbeto K. D., 2019). We inspire to conduct this research is for different reasons. **The first reason** that motivates us to conduct research on information retrieval for Sidama language is that there has been no information retrieval developed for Sidama language. But, a massive amount of electronic data is processed every day then the language is used in schools, offices, and other media as mentioned above. When processing electronic data, IR has the potential to retrieve relevant document from the source. Due to the

absence of IR for Sidama language, the texts have ability to interruption the message of the writer. The researcher is one of the community members of this language speaker and needs to contribute something to the language to make it more respected in his discipline. **The second reason** was to solve IR problems resulted by short and ambiguous (unclear) user query, poorly verbalized query, and vocabulary mismatch. **The third reason** was to design IR system in order that fit for Sidaama Language. It helps the speakers of the language to obtain information of their interest.

1.3. Statement of the problem

There are more than 7,000 languages in the world organized from which about 2,000 (28.6%) languages spoken in Africa (William R. L., 2016). Among them Ethiopia has more than 83 different languages with up to 200 different dialects spoken (Demewoz, Automatic thesaurus construction from Wolayita text, 2013). The Ethiopian languages are divided into four main language clusters. These are Semitic, Cushitic, Omotic, and Nilo-Saharan (Demewoz, Automatic thesaurus construction from Wolayita text, 2013). Sidama language is classified under Cushitic language family. Sidama language is one of the widely spoken and official working languages in Sidama National Regional State starting from 2020. Sidama language plays a crucial role in social, political, religious and economic activities for people speaking the language. Now a day, large numbers of documents are available written in Sidama language in the form of text books, magazines, newspapers, news, and in social Medias and on the websites. Which indicate Sidama language is available in electronic format both on the Internet and on offline sources. The increment of electronic documents creates some difficulty on the task of IR systems. The language speakers want to find out the most relevant document for their information need. Selecting the most relevant document which satisfies users need from big document collection become difficult task. IR system requires optimal query construction which is adequate enough to represent the information need of the user. There is the shortage of properly studied language specific resource and research that helps to develop information retrieval system and also no NLP applications attempted so far for this language like information retrieval, text summarization, machine translator, morphological analyzer, spell checker, grammar checker, etc. The absence of these language specific works for one language may hinder the development of language technology. The problems may occur when retrieving the document; the information retrievals use the keyword for

retrieving the necessary documents. But rather than giving whole words as a keyword, it is better to breakdown words into morphemes before giving words to search engine. Clearly, these extensive inflectional and derivational features of the language are presenting several challenges for text processing and information retrieval tasks in Sidama language. For this reason, developing the query expansion for Sidama language information retrieval system is the ground work for future development of other NLP applications. Also, developing the query expansion for Sidama language information retrieval system is one of the contributions for development of the language. Query expansion for Sidama language information retrieval system plays a vital role in NLP systems. This means, the output of information retrieval is assisted as an input for other NLP applications. From related works reviewed, we observed that there is no far attempt made to design and develop IR systems for Sidama language. Also, techniques used in other languages may not be directly applied to Sidama language information retrieval because of the morphological structure of the Sidama language was complex. Hence, the purpose of this study is to explore the possibility of producing Sidama language words that are supportive for developing other NLP applications by developing the query expansion for Sidama language information retrieval system. Applying query expansion for Sidama language information retrieval system enables Sidama language speakers to access information that gratifies their information needs and fills the identified knowledge gaps and contributes to the development of the language.

Information retrieval systems have different challenges such as, *short user query expression*, *ambiguity nature of the natural language* and *the vocabulary mismatch* between query terms and relevant documents. But good information retrieval system to some extent eases the above challenges to retrieve best results. This research word strives to minimize these problems and increase recall value.

At the end, this research tries to reply the research questions as follow:

- Which experimentation techniques are more suitable to handle the challenges of word mismatch?
- To what extent does the Vector Space Model with query expansion is effective for Sidama Language information retrieval than basic Vector Space Model?
- How much the manual query expansion increases the recall value of Sidama language IR?

1.4. Objectives of the study

1.4.1. General objective

The general objective of this study is to design and evaluate query expansion for Sidama language information retrieval system by using vector space model.

1.4.2. Specific objectives

To attain our general objective, we have defined the following specific objectives:

- To review literature on Information Retrieval
- To study the basics and word formation of Sidama language to perform text operation?
- To collect Sidama text corpus and prepare queries
- To design an architecture of Information Retrieval Systems for Sidama language text.
- To apply text operations for text corpus and queries to produce index terms
- To evaluate the performance of the system

1.5. Methodology

To fulfill our formulated objectives specified above, we have employed the following methods and procedures.

1.5.1. Literature Review

To capture the IR basics, IR models, text operations, techniques and language features, we have undergone literature review. A review of literature was also be conducted to get improved with the basic Sidama language text features in relation to the topic and the objectives of the study. For that reason, different journal, books, articles, and former related research work as well as electronic resources on the Web have been reviewed.

1.5.2. Methods of Data collection and preparation

Sidama language does not have standardized text corpus prepared so far for information retrieval. In order to minimize such problem and to measure the effectiveness of IR systems we have collected different data set from different sources such as Ethiopia Press Agency/Sidama language, Sidama language Holly Bible, Bakkalcho newspaper crawled from the Faith Comes by Hearing (FBCH), Sidama Media Network (SMN), Dayye FM Radio program, Sidama Regional state Culture, Tourism and Sport bureau, Hawassa University Sidaamigna Department and Sidama Regional state Education Bureau

(SRSEB) using document analysis data collection method. The collected data organized and cleaned manually by language expert and saved on NotePad in UTF-8 format ready for Python Programming.

1.5.3. Implementation Tools

We used the Python Programming Language to implement the designed prototype system. We preferred Python because of its simplicity, unlicensed, convenience for big data analysis, and has dynamic data types for processing linguistic data (Steven, et al., 2009). It also has extensive built-in help functions, having NLTK extension which provides basic classes, modules, function and easy to-use interfaces for processing language specific texts. We have applied a widely used Vector Space model for searching and matching user query terms with documents in the corpus. To compute weights values of the terms in the vector we will use TF-IDF.

1.5.4. Evaluation Techniques

With the intention of measure the performance of the system. First, we have organized the Sidama text corpus and then prepared 20 queries for performance measurement. The queries are formulated from the organized dataset (corpus). Then after, we have applied the relevance judgment by language experts to evaluate the effectiveness of the prototype system towards the test queries and compared with the results obtained by the implemented system. To enumerate the effectiveness of our prototype IR, we have used common IR systems evaluation techniques such as: precision (the fraction of retrieved documents that are relevant), recall (the fraction of relevant document that retrieved), and their harmonic mean (F-measure). Finally, we will make the comparison between the relevance judgment by language experts and prototype systems results.

1.6. Scope and Limitation of the Study

The goal of this study was to design and develop Manual Query-based query expansion for Sidama language IR to minimizing short, unclear and poorly designed user query problems and query-document mismatch problems.

We have collected the textual data type from different data sources for our study. We have applied text preprocessing operations such as: tokenization, stop words removal,

normalization, and stemming. To produce index terms, we have employed inverted index file method and for similarity measuring and matching we have used popular Vector Space Model (VSM). We have used suffix trimming stemmer preparing list of suffixes in Sidama language.

For query expansion we applied the manual query expansion method to solve the problem of document missing due to synonym terms mismatch. We have chosen manual-query expansion technique because of the absence of language dependent recourses such as: WordNet, Thesaurus, and Ontology developed so far in Sidama language, which by themselves need further studies.
Additionally, thew absence of rule-based stemmer in the language was one of the challenges that affects the result.

1.7. Significance of the Study

Now a day we can find researches undertaken in specific languages of the world. So many researches have been done on the languages which are internationally used for trade and innovation. When we come to local languages like Sidama Language in Ethiopia, only little researches have been tried, on few topics. In IR research area, almost no research has been done yet. The word formation, morphology and structure of one language are far different from that of the other language. Since, IR systems are language dependent and it requires developing language specific algorithms for text preprocessing, and indexing. Our study helps the speakers of Sidama language to fulfill their information needs by developing algorithms for query and text preprocessing, indexing, searching and ranking. Sidama Language speakers will be able to use it for searching and retrieving the information of their interest from large number of stored documents written in the language. This study also contributes as step stone for the Sidama language development in terms of Natural Language Processing, Machine Learning, Big Data Analysis, Google search, E-library systems, etc. This study also helps interested researchers to conduct further studies on this area and in developing advanced systems on Sidama language such as search engines, text categorization, data mining, Machine translation, mobile applications development on Sidama language, Question answering, and for dictionary development. This study is also useful for the development of cross-language retrieval

systems, which is able to retrieve documents written in other language(s) by combining the translation models and retrieval models into one unifying framework to satisfy the information need of users. The study will assist Universities, Schools, public Libraries, Mass Media Agencies, and other search and retrieve News, Official Letters, and other stored documents prepares in the language. The study also will benefit me to get enough knowledge in the area of information retrieval, Natural Language Processing and also to fulfill my MSc program graduation requirement.

1.8. Organization of the Thesis

The thesis has been organized into six chapters. The **first chapter** introduces the general background of the research including the query expansion for Sidama language information retrieval system, statement of the problem, objectives, scope and limitation, research methodology and significance of the study. **Chapter 2** discusses the literature reviewed. The general overview of IR, and its basic processes are discussed. And also, IR models are discussed. In **chapter three** Sidama language morphology and writing system are discussed. In **chapter four** System design and Architecture presents. In **chapter five** the experimentation and evaluation of the proposed system are discussed. From the both experimentations obtained result and confront challenges also presented in it. The experimentation and result discussion are made. Dataset and implementation of proposed work and the evaluation result are presented. Additionally, major research findings and challenges are discussed. Finally, in **chapter six** conclusion, contributions and forwards recommendations for future work

CHAPTER TWO

2. LITERATURE REVIEW

2.1. Overview of IR Systems

Searching digital information on the Web or from document collection has long become part of the human daily life. Today, information retrieval has gained much care due to the increasing of digital data and the necessity of accessing relevant information from massive corpus rapidly and exactly (Tilahun Yeshambel1, 2020). Information retrieval systems are computer programs that supply and achieve information in documents and supports users in retrieving the information of their attention (Senbeto K. D., 2019). Information retrieval system discoveries essential information from documents and return to the user with its position. Those documents that gratify the user requirement are called relevant documents. The core goal of information retrieval systems is to deliver easy access of information that gratifies the user attention. Because very fast increase of the massive amount of unstructured data and growing number of documents on every single desktop need effective techniques of information retrieval system. Information retrieval system includes *three significant functions* for effective document retrieval. These are indexing text, searching useful documents in a collection based on user queries and developing ranking algorithms to improve the performance (Shweta J.patil, 2017).

1.1.1. Information Retrieval Vs Data Base Management System

Information retrieval and Data Base Management Systems are the technologically advanced to represent and store data and information retrieve effectively. There are significant differences between them even if they have similarity in storing and retrieving data and information. The data base management system handles highly structured information, but information retrieval systems can handle both structured as well as unstructured data. Their major differences are specified in the following table

Table 2. 1: The difference Between IR And DBMS

IR	**DBMS**
Handle unstructured natural language texts	Handle highly structured and well-structured data only in database rows and columns
Supports user-based query terms with concepts	Only supports SQL and relational Algebra queries without concept
User without the knowledge of IR can use it	Users without having basic knowledge about SQL queries cannot use it
Provides ranked documents based on similarity measurement	Does not provide ranked documents
IR systems can return documents even if the exact user query terms are not existing in the document, because it considers derivational morphology of words	It can return only exact match between query terms and document terms, because it does not consider the **derivational morphologies of words**
Locate documents in terms of ***relevance*** and ***preference*** of user interest	Does not provide information-based relevance
Indexing is term or word based, involving NLP processes for all words	Indexing is ***key*** based for certain *field* common for the entire records

1.1.2. Information Retrieval Vs Natural Language Processing

Natural Language Processing is a computer-based linguistic technology aimed to understand and manipulate human languages in order to communicate with computer. So, the Natural Language Processing computationally analyze texts and learn the language. Information retrieval aimed to systematically organize, store and retrieve useful information from huge collection of documents of unstructured nature. For effective retrieval of information, it should pass through text pre-processing stages where Natural Language Processing comes to play the major linguistic role. Information retrieval is the broader field of study which involves not only textual data, but also image, audio and video data as well. The Natural Language Processing helps to analyze and understand the text (query and document) so as to

improve the retrieval process. The Natural Language Processing enables the searching algorithms to search the entire texts rather than titles and abstracts unlike desktop search.

1.1.3. Information Retrieval vs Data Mining

Both information retrieval and data mining are computer services to holder massive amount of data to use them effectively for once attention. Both of them pass through common data preprocessing steps such as tokenization, normalization, stop word removal and stemming. But data mining covers the wider study, which even uses information retrieval as its sub-stage together with other advanced functions such as data integrations, information discovery, characterization, classification, clustering, and trend/deviation and outlier analysis. The information retrieval analyses the data source to deliver known information and documents that gratifies the information needs of user from stored data collection; whereas data mining analyses data source to see hidden data, and design from those collection of data.

2.2. Fundamental Information Retrieval Processes

To discovery useful information from the Web is surely a boring and difficult task. Specially, for native users the problems become tiredly interrupt all their effort lost on searching for information of their interest. The main difficulty is the absence of well-defined underlying data model for the Web; this indicates that structure and information definition of knowledge designed in low quality. These difficulties have attracted researcher's concentration in information retrieval and its technique as solution. Then-after the area of information retrieval take attention as other technology. An information retrieval system assists users as a bridge between their information need and enormous amount of stored information. Even though the task of information retrieval systems is retrieving relevant information, unfortunately there is no such system which can retrieve relevant and only relevant documents as per user's query. Information retrieval system involves three mechanisms such as document source, query preparation and its evaluation for effective retrieval. These components work in three basic phases (Senbeto K. D., 2019). The **first process** is the representation of documents and user query. In information retrieval systems; documents and user search declarations should be first denoted in a suitable encoding technique is said to be indexing. Both document and query pass through numerous text processing actions during the indexing process. The **second process** is the

comparison of similarity between the user query and all existing documents in the source and then retrieves more associated documents to the user query. The **third process** in information retrieval systems is ***ranking*** process which include ordering the retrieved documents according to their level of importance to the user query. Each of these processes is discussed figure 2.1 below.

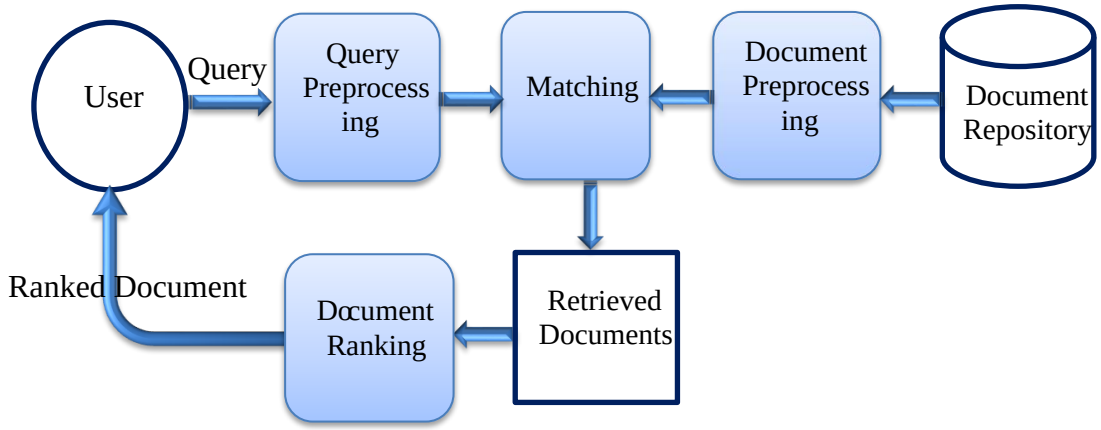

Figure 2. 1: General information retrieval System Architecture (Croft, 1993)

2.2.1. Document indexing

Document and query indexing is a text preprocessing step in the information retrieval systems development process. During indexing process texts are collected, analyzed and stored to simplify fast and correct text retrieval. indexing is an offline process that used to mining content bearing terms from document group and establish them using indexing structure to increase searching speed. It is a language dependent process which contains diverse text preprocessing processes such as: normalization, tokenization, stop word removal and stemming. The language precise text passes through these processes in order to screen content bearing terms and then extract these content-bearing terms and creates data structure in order to locate each term to increase the efficiency of retrieval systems. An Index term mostly contains nouns and verbs because they have meaning by themselves. But connectives, adjectives and adverbs are less beneficial to transport issue, so that most of the time they had eliminated as the stop words. In order to indexing and searching goal information Retrieval systems use inverted indexes. The reason why the need of indexing is a single word may occur in several morphological variations in a query and document. Thus, increasing the size of the index and decreasing

retrieval performance. The need of indexing for decreasing such kind of problem. But all index terms are not equally important in in the documents.

2.2.2. Similarity Measurement and Matching

When the query and document are processed and indexed, then the following stage is similarity measurement and matching. Similarity measurement includes comparing the degree of similarity and relatedness between user query and documents in the source. Similarity measurement calculates the occurrence of indexed-term frequency between query and documents. The outcome of similarity measurement is relevant documents that are matched with the user's query. There are different similarity measuring methods (K. Pradeep Reddy1 & Vardhan4, 2018). These are: - Euclidian Distance (ED): Cosine Similarity Measure (CSM): Jaccard Similarity Measure (JSM): Dice Coefficient Measure (DCM): Pearson Correlation Coefficient (PCC).

2.2.2.1. Euclidian Distance (ED)

Euclidian distance measure is regular distance measure to calculate the distance between two points in two- and three-dimensional space. This measure is used in document grouping to group the documents into similar groups based on the distance between the documents (Pradeep, Raghunadha, Apparao, & Vishnu, 2018). Euclidian Distance measure is denoted in equation (1).

$$ED(dx,d= \sqrt{\sum_{i=1}^{m}(w(ti,dx) - w(ti,dy))2} \quad \text{.....1}$$

2.2.2.2.Cosine Similarity Measure (CSM)

In vector space model, the sets of documents and queries are seen as vectors. Cosine similarity measure is a common technique for manipulative the similarity value between the vectors. With document and queries being denoted as vectors, similarity indicates the closeness between the two vectors. Cosine similarity measure calculates similarity as a purpose of the angle made by the vectors. If two vectors are nearby, the angle formed between them would be small and if the two vectors are distant, the angle formed between them would be large. The cosine value differs from +1 to -1 for angles ranging from 0 to 180 degrees respectively. A score of 1 estimate to the angle being 0o, which means the document are similar. While a score of 0 estimates to the angle being 90o, which means

the documents are completely not similar. The cosine weighting measure is applied on length-controlled vectors for making their weights similar. Equation (2).

$$CSIM(q,dj)=\frac{\sum_{i=1}^{m} w(ti,q)x\, w(ti,dj)}{\sqrt{\sum_{i=1}^{m} w(ti,q)2\, x}\ \sqrt{\sum_{i=1}^{m} w(ti,dj)2}} \quad \ldots\ldots 2$$

2.2.2.3.Jaccard Similarity Measure (JSM)

Jaccard Similarity measure is additional measure for manipulative the similarity between the queries and documents (Pradeep, Raghunadha, Apparao, & Vishnu, 2018). In Jaccard measure, the index starts with completely dissimilar and goes to completely similar which means a minimum value of 0 and goes to a maximum value of 1. Jaccard similarity measure similarity between the documents and user query. The value is between 0 and 1. 0 display that documents are dissimilar and 1 displays those documents are identical with each other. Value between 0 and 1 display the probability of similarity between the documents. Equation (3) denotes the Jaccard Similarity measure.

$$JSIM=\frac{\sum_{i=1}^{m} w(ti,q)x\, w(ti,dj)}{\sum_{i=1}^{m} w(ti,q)2+\sum_{i=1}^{m} w(ti,dj)2-\sum_{i=1}^{m} w(ti,q)x\, w(ti,dj)} \quad \ldots\ldots 3$$

2.2.2.4.Dice Coefficient Measure

Dice Coefficient measure is used to compare the similarity between two samples of text (Pradeep, Raghunadha, Apparao, & Vishnu, 2018). Equation 4 demonstrate the Dice Coefficient measure.

$$DSIM(q, dj) = \frac{\sum_{i=1}^{m} w(ti,q)x\, w(ti,dj)}{\alpha \sum_{i=1}^{m} w(ti,q)2+(1-\alpha)\sum_{i=1}^{m} w(ti,dj)2} \quad \ldots\ldots 4$$

Where, w (ti, q), w (ti, dj) are the weights of the term ti in query q and document dj similarly. α Limit range is from 0 to 1. α control the degree of consequences of false negative versus false positive errors. In general, Alpha value is 0.5. if $\alpha > 0.5$, DCM measure gives more meaning to precision and if $\alpha < 0.5$, this measure gives more meaning to recall.

2.2.2.5.Pearson Correlation Coefficient

Pearson Correlation Coefficient is used to calculate the linear relationship among two documents dx and dy (Pradeep, Raghunadha, Apparao, & Vishnu, 2018). Pearson Correlation Coefficient measure gives a similarity value between -1 and +1, where -1 is total negative linear relationship, 0 is no linear relationship, and +1 is total positive linear relationship. Equation (5) is used to calculate the Pearson Relationship Coefficient.

$$PCSIM(dx, dy)=\frac{m\sum_{i=1}^{m} w(ti,dx)\ x\ w(ti,dy)-\sum_{i=1}^{m} w(ti,dx)\ x\ \sum_{i=1}^{m} w(ti,dy)}{\sqrt{\mathrm{m}(\sum_{i=1}^{m} w(ti,dx)2-(\sum_{i=1}^{m} w(ti,dx)2))\ x\ \mathrm{m}(\sum_{i=1}^{m} w(ti,dy)2-(\sum_{i=1}^{m} w(ti,dy)2))}} \ldots.5$$

Where, w (ti, dx), w (ti, dx) are the weights of the term ti in documents dx and dy respectively.

2.2.3. Document representation and term weighting

We can represent the document in the different techniques. Techniques which we used to represent document help us to give diverse weight for diverse terms with respect to given query. Official it supports us if document is either relevant or not with respect to user's query. In this work tf*idf term weighting is used for defining relevance of the terms.

There are different techniques which used to transfer weight to terms.

A. **Binary Weights**: To represent the presence of the term in the vector we used (1) but to represent the absence of the term in the vector we used (0).
B. **Raw term frequency:** To represent the presence of the term in the vector we used (1) but to represent the absence of the term in the vector we used (0).
C. **tf * idf:** tf * idf **equal** Term Frequency *Inverse Document Frequency

There are three diverse techniques in this study which has been used to calculating term weight tf *idf. The motive why tf *idf weight is designated because it is normal technique of calculating weight. In Information Retrieval tf*idf, is a well know technique to estimate how word is significant in a document. tf*idf is also a very inspiring method to change the textual illustration of information into a Vector Space Model. The value of tf*idf upsurges with proportion to the number of times a word seems in the document and reductions as the term happen often in the document. The good thing about tf-idf is that it depends on both term frequency (tf) and inverse document frequency (idf). This brands humble to decrease rank of common terms throughout the entire document corpus, but

upsurge in rank of terms happen in fewer documents more often. The term frequency (tf) is simply the number of given terms appears in the document. This count is frequently normalized to avoid a partiality towards longer documents to give a measure of rank of the term within specific document.

$$Tfij=\frac{fij}{max(fij)} \quad \text{..... 6}$$

The inverse document frequency (idf) is measure whether the term is infrequent transversely all documents. It is gained by dividing the whole number of documents by the number of documents containing the term, then taking the logarithm and quotient. Higher idf value is gained for infrequent terms whereas lower value for frequent terms. It is mainly used to differentiate importance of term throughout the collection.

$$IDF=\log_2 \frac{N}{dfi} \quad \text{.....7}$$

Document frequency (df) is the number of documents that containing in the given term. N is the total number of the given documents & tf*idf is the term weight.

$$Tf*idf = tfij*\log_2 \frac{N}{dfi} \quad \text{.....8}$$

2.2.4. Ranking and Highlighting

In modern information retrieval systems ranking the document is the most important point because it improves the quality of the retrieval system. A ranking part accepts retrieved sets of documents from similar process and delivers well-arranged set of the documents according to their level of relevance to the user query in descending order. The document ranking process takes place depends on the value of similarity measure calculated using information retrieval models, VSM in our case. The outcome of ranking process is ranked list of retrieved documents according to the level of relevance.

2.3. Expansion And Relevance Feedback

2.3.1. Query Expansion

Query expansion is the technique of re-constructing a user query by combining additional related terms in order to improve the performance of retrieval systems (Senbeto, 2019). In query expansion two or more different words convey the same meaning. Query expansion helps to modify the user query to re-constructing the right query in order to retrieve the right

information that user really needs. Most of the time, user queries are not representative enough, not well verbalized, short and ambiguous. Researchers understood that there is a struggle for users to express respectable search query during information retrieval. Most of users express feeble queries without full knowledge of the document collection and the retrieval environment according to these researchers, this kind of queries lead information retrieval systems to poor coverage of desired result. For the reason that of the above struggle, most users' query requests reformulation to get the outcomes of their interest. In other word, initial users query requests reformulating users query terms with similar meaning of words. The main reason for developing query expansion is users query assumed to be unclear. Users query requests to be expanded in order to solve the unclearness nature of query terms. This improvement of user's query may help IR systems to retrieve relevant document and few non-relevant as considerable as possible. Query processing is the key responsibilities in information retrieval which done to improve the performance of retrieval systems. This is because some query terms have better significance to locate the right document required by the user than the other, so that it is much loved idea to use several alternatives (expanded) terms in query to find the required document instead of confining in only limited terms. Query expansion is the optimal technique to solve these problems by including additional and related terms to the initial query for query re-construction. The aim of query expansion is to develop query that address the intention of user during information retrieval. There are three types of the query expansion. These are manual, automatic and interactive. And each query expansions can be classified based on either search results and based on knowledge structures in chapter one. In chapter one we have discussed the types of the query expansions. In this chapter we will see the approaches of the query expansion.

2.3.2. Approaches to Query Expansion

During information retrieval, system developers have been used different query expansion techniques. These query expansion techniques are relevance feedback based (like explicitly, implicitly and pseudo) query expansion techniques, External resource based (like WordNet based, thesaurus based) query expansion techniques, Query log based and Wikipedia based query expansion techniques (Ashish, 2015). These techniques are broadly categorized into two. these are Corpus based (local analysis) and External resource based (global analysis). Among all of them we want to use relevance feedback-based query expansion techniques.

2.3.2.1. Relevance Feedback Based Query Expansion (Corpus-Based Expansion)

Relevance feedback is one of the query expansion techniques which increases user query based on the exploration outcome of initial test supported by user desired. The user delivers extra information to the system by linking on the retrieved document in the initial exploration. The system will rebuild new query using terms in the designated documents and re-compute term-weight in the query. This method will use statistically computed (top ranked) terms in the initial attempt to rebuild the new query. In generally, relevance feedback has not only advantages but it also has the disadvantages. Because most of the time relevance feedback dependent on the search result of previous test (Beaza-Yates & Ribeiro-Neto, 2011). By that reason we can say relevance feedback method is effective if the initial user query is good sufficient to denote the essential information. But if the first user query is feeble to indicate the interest of the user, then the renewed query can also be feeble because it is dependent on the top ranked retrieved documents of the first query. The second problem is that; even if the relevance feedback expands the user query by using statistically top ranked outcome set, it still misses the synonym terms and misspelled words. Thirdly, the relevance feedback has high calculating price and very long reply time as query expanded and became long which in turn subsequent in additional overload for the system (Christopher D. Manning P. R., 2009)). Moreover, users sometimes became opposed and interruption to think more and to provide extra feedback terms for query expansion. In relevance feedback there are different techniques. These are explicit feedback; implicit feedback and Pseudo relevance feedback. We will explain each of them one by one in the following section.

2.3.2.2. Explicit Feedback

In an explicit feedback user prearrange information for query modification. This means in explicit feedback approach, the user participating in the query rebuilding process by delivering extra information. And also, in explicit relevance feedback approach, the relevance judgment is provided directly by the users or by a group of human assistants. Explicit feedback is an interactive approach in which the initial retrieved documents are existing to user and the user is requested to choose the relevant documents. These models are not much useful because users assume the system to be independent and retrieve the outcomes for the user. User would finally get irritated by frequent communication required from him for each search. These types of models can be used for testing search engines where developers are ready to interact with the system. In general, explicit

feedback is the type of feedback where query modification information is directly given by the user, which means user reconstruct query by providing extra information.

2.3.2.3. Implicit Feedback

Implicit relevance feedback requires the smallest spread of effort from the user to improve the retrieval performance. Which means in implicit feedback technique, user do not include in deliver extra information for query modernization, but the system by itself will originate query information either from statistically top ranked outcome set (local analysis) or from external sources such as a thesaurus, dictionary and WordNet, and Ontology which is usually termed as the global analysis. Query expansion algorithms at first evaluate given query on collection of documents and then select from relevant documents suitable terms. The original users query extended with such selected terms from the first ranked document. The reformulated query is used to retrieve new set of relevant documents. If the first set of relevant documents in the whole document collection is known, one can calculate the query vector that can fit the whole of the relevant documents in the corpus. The best query vector technique used for differentiating the relevant documents from the non-relevant is stated by the Rocchio's algorithm which is called query expanding algorithm based on vector space model. This equation is given Equation 1.

$$\vec{q}_{opt} = \frac{1}{|C_r|}\sum_{\forall \vec{d}_j \in C_r}\vec{d}_j - \frac{1}{N-|C_r|}\sum_{\forall \vec{d}_j \in C_n}\vec{d}_j \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (1)$$

Where **Cr** are set of relevant documents, **Cn** is the set of non-relevant document, **N** is the total number of documents in the total collection and **dj** is any document in the total document collection, which sometimes can be relevants or non-relevant. The concept holds an assumption that, term-weight vectors of relevant and non-relevant documents are dissimilar. Hence the core idea to reformulate the query is such that it gets closer to the term-weight vector space of the relevant documents and in turn away from the non-relevant once. As (Beaza-Yates & Ribeiro-Neto, 2011). wrote that, the problem is the relevant documents are not known or it is an unrealistic situation. The way to avoid this problem is formulating an initial query **q** to **qm** and to incrementally change the initial query vector. This incremental change is accomplished by restricting the computation to the documents known to be relevant according to the user's relevance judgment. There are three classic and similar ways to calculate the modified query **qm** as follows [2].

Standard Rocchio:

$$\vec{q}_m = \alpha\vec{q} + \frac{\beta}{N_r}\sum_{\forall \vec{d}_j \in D_r}\vec{d}_j - \frac{\gamma}{N_n}\sum_{\forall \vec{d}_j \in D_n}\vec{d}_j \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

Ide Regular:

$$\vec{q}_m = \alpha\vec{q} + \beta\sum_{\forall \vec{d}_j \in D_r}\vec{d}_j - \gamma\sum_{\forall \vec{d}_j \in D_n}\vec{d}_j \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

Ide_Dec_Hi:

$$\vec{q}_m = \alpha\vec{q} + \beta\sum_{\forall \vec{d}_j \in D_r}\vec{d}_j - \gamma \max\_rank(D_n) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

There is an understanding of these three techniques have similar results. These techniques are carried out, based on multiple executions of their algorithm, until it is close enough to the ideal query proposed in Equation 1 which is called query expanding method based on vector space model. The main advantage of the relevance feedback is introduced better results and simplicity but optimality (i.e. the number of iterations to produce vector m) remains a problem. The relevance feedback model for probabilistic model which is dynamically ranks documents similar to a query q according the probabilistic ranking principle. Retrieved documents can also be made more relevant than initially retrieved once, using the similarity of document dj to a query q in the probabilistic model. The formula for similarity in probabilistic model is given in Equation 5.

$$\text{sim}(\text{dj}, \text{q}) = \sum_{ki \in \text{q} \wedge \text{ki} \in \text{d}j} \log\left(\frac{p(\text{ki}/\text{R})}{1 - \text{P}(\text{ki}/\text{R})}\right) + \log\left(\frac{1 - \text{P}(\text{ki}/\bar{\text{R}})}{p(\text{ki}/\bar{\text{R}})}\right) \quad \ldots\ldots\ldots 5$$

Where P ($\frac{Ki}{R}$) and P ($\frac{\overline{Ki}}{R}$), are the probability of term existing in the relevant document set, and in the non-relevant document set respectively. Unlike the vector space model-based relevance feedback, the method based on probabilistic model doesn't expand queries rather it adjusts the weights of the query terms as they are. The problem encountered in this model is the values for P ($\frac{Ki}{R}$) and P ($\frac{\overline{Ki}}{R}$), is not known, and thus initial assumptions are taken as 0.5 and ni, N, respectively. Where, ni is the number of documents that term Ki is found, and N is the total number of documents in the corpus. After retrieving documents based on the assumed values, a user judges the documents by marking them as relevant or not. Based on this judgment, the values of ($\frac{Ki}{R}$) and P ($\frac{\overline{Ki}}{R}$), are changed to

$\frac{|Dr,i|}{|Dr|}$ and, $\frac{ni-|Dr,i|}{N-|Dr|}$ respectively, where |Dr, i | is the number of documents where term Ki exists in, and |Dr, i| is the number of relevant documents as judged by the users. Although using this model enhances precision, it has three problems. First, document term weights are not considered during the feedback loop. Second, weights of terms in the previous query formulations are ignored. And third, no query expansion is used (the same set of index terms in the original query is reweighted over and over again).

2.3.2.4. Pseudo Feedback

Pseudo relevance feedback is one of the query reformulation approaches which doesn't need the user's intrusion on the expanding process. In Pseudo relevance feedback first retrieve some ranked documents from original user's query, then users query modifies to the reformulated query to retrieve final results for users by evaluating the first retrieved document. In Pseudo Relevance Feedback based models initial query is powerful and highest k outcomes are gotten. Pseudo relevance feedback captures the significant terms only based on co-occurrence. But only co-occurrence is not sufficient for accuracy of outcomes. Pseudo Relevance Feedback also called as Blind Feedback programs the physical part of relevance feedback and has the advantage that evaluators are not essential (Ashish, 2015). Pseudo Relevance Feedback has been fruitfully practical in several IR contexts and has been shown to improve precision and recall of search engines.

2.4. Performance Evaluation

Performance evaluations metrics are used to measure the mark of some objectives specified to attain by the systems. All systems are developed to encounter some targeted functionality. These functionalities are estimated to make sure that, either the system performing as planned to encounter the specified objectives. The estimation metrics used for IR systems is known as retrieval performance evaluation. There are several retrieval performance evaluation methods. We have been used recall, precision and F-measure research methods.

2.4.1. Precision (P)

Precision is ratio of relevant documents retrieved by the system to the entire number of documents retrieved. In other word it is the fraction of the documents retrieved that are relevant to the user's information need and is given by Equation 13:

$$\text{Precision} = \frac{Relevant\ document\ retrieved}{Reterieved\ document} \text{………………………………}13$$

2.4.2. Recall (R)

Recall and precision are arithmetical capacities which used to estimate the performance of information retrieval systems. IR systems can be estimated based on in what way the system can retrieve relevant documents and few irrelevant documents as considerable as possible. To estimate recall and precision of a system first relevant documents prerequisite to be prepare according to relevance decision. And also, we will be computing recall and precision as follows:

Recall is number of relevant documents retrieved by the system divided by the total number of existing relevant documents that should have been retrieved. It is the fraction of the documents that are relevant to the query that are successfully retrieved, and given by Equation 14

$$\text{Recall}= \frac{Relevant\ document\ retrieved}{Relevant\ document} \text{…………………………………}14$$

2.4.3. F-measure

F-measure is average combines of both recall and precision and is given in Eq15.

$$\text{F-measure}=\frac{2\ x\ P\ x\ R}{P+R}\text{………………………………………………...}15$$

2.5. Information Retrieval Model

Information retrieval model needs the particulars document illustration, query illustration, and the similar function. IR models provide conceptual bases for predicting relevant documents for the user need. There are two core motives for having retrieval models. First information retrieval models guide research and deliver the income for academic argument. The second motive why we need retrieval model is because it helps as a blue print to implement an actual retrieval system. In information retrieval systems, there are three classical IR models. These

are: - Boolean, Vector Space and Probabilistic models (Manwar, Hemant, Chinchkhede, & Vinay, 2012).

2.5.1. Boolean Model

Boolean model is the oldest information retrieval model that matches document terms with the query terms by using Boolean operators. In the Boolean model there are three basic logical operators. These are AND, OR and NOT. AND is reasonable product, OR is reasonable sum and NOT is reasonable difference. AND is used to cluster set of terms in to single statement. For example, 'computer AND science' are two queries terms which joint by operator 'AND'. If terms in the user query are connected by operator OR, documented with either of terms or all terms will be retrieved. For example, if query is Information OR Technology, document covering Information, or Technology, or Information Technology will be retrieved. What makes Boolean model good model is that it creates a sense of control to user over the system. In it user deciding what should or should not be retrieved. Query reconstruction is also humble. Boolean model retrieves when all documents are matching with query term unless otherwise it may not retrieve anything if there is no matching document. So, there is no relevance judgment and ranking mechanism (Gezehagn G. E., 2012) .

Problems, missing partial matching. It does not consider the inflectional and derivational variations of words.

2.5.2. Vector Space Model (VSM)

The VSM model shows the documents and query as row and column vector. VSM calculate vector values and cosine values between documents and query to measure their similarity to retrieval the most relevant documents for user query. The cosine measures how much documents are close to the query terms in vector space (Senbeto K. , 2019). Vector space model is a popular model now it used by many researchers for information retrieval systems development. This is because; it reflects term-frequency measures, inverse document frequency measures, index-term weights for document corpus, which achieves main relevancy in retrieving the most relevant documents in information retrieval. The index-term weight enables to calculate the level of similarity between query terms and each document and to rank the retrieved documents according to the degree of relevance. In vector space model documents and terms are denoted as vectors in a n-dimension space. The word "terms" is not essential in the model, but terms are typically words and phrases. The values given to elements in the vector

space describe the degree of importance of term in representing document. Sometimes a given term may receive a diverse weight in each document in which it occurs. If the term is not appearing in a given document the weight of that term will be zero in that document B. R., Ribeiro Neto, "Modern Information Retrieval". For a given document (dj) the weight of the terms in it can be stated as the coordinates of dj in the document space. A document collection containing a total of documents (di) described by terms (tj) is denoted as term by document matrix (T x D). Each row and column of this matrix is a term vector and document vector respectively. The element at row i, column j, is the weight of term j in document I (Christopher D. Manning P. R., 2009)

Table 2. 2: Example of Term Document Matrix

Term list	**Do1**	**Do2**	**Doc3**	**Do4**	**Doc5….**	**Docn**
Term1	W11	W12	W13	W14	W15….	W1n
Term2	W21	W22	W23	W24	W25….	W2n
Term3	W31	W32	W33	W34	W35….	W3n
Term4	W41	W42	W43	W44	W45….	W4n
Term5	W51	W52	W53	W54	W55….	W5n
Term6….	W61	W62	W63	W64	W65….	W6n
Term k	Wk1	Wk2	Wk3	Wk4	Wk5….	Wkn

The weight of terms in the documents or in the queries given by using term frequency (TF) and inverse document frequency (TF*IDF) system. The term frequency is the number of existences of the given term within the given document. Term frequency (TF) tries to measure the importance degree of the term within the given document. Inverse document frequency (IDF) is a number of a given term within whole collection of documents. The similarity of document and query in vector space model is determined by link between the vectors dj and vector q. The link is quantified by the associative coefficients based on the inner product (dot product) of the document and query vector. Though, the greatest extensively used by vector space model and popular similarity measure is the cosine coefficient, which measures the angle between the document vector and the query vector. According to (Beaza-Yates & Ribeiro-Neto, 2011), Vector space model (VSM) have several advantages. First, it improves retrieval performance using its term-weighting system. Second, the incomplete matching plan of vector space model permits retrieval of document estimated the query state. Third, it sorts and rank the documents

according to their degree of similarity to the query using cosine similarity measurement. Finally, it is simple to and fast to implement. However, the model has also several disadvantages. If there are no common words are shared between the query and documents in text collection, the similarity value will be zero and no document will be retrieved. This is usually happened when users and authors desire to use different words which have the same meaning B. R., Ribeiro Neto, "Modern Information Retrieval. Like vector space model, the probabilistic model uses the statistical results of term frequencies to decide the relevance of the documents towards the user query and to rank them. Salton and Buckley proved that the VSM achieves better result than Probabilistic Model in general collection (Salton & Buckley, 1988). The term-weighting and similarity measuring property of VSM made it the popular information retrieval model. Most search engines are developed using the similarity measure of VSM to rank retrieved documents (Jitendra & Sanjay, Analysis of Vector Space Model in Information Retrieval, 2012). Besides, the VSM is simple and fast model with lighter workload (Yegosh, Ashish, & Sexena, 2013). Due to these reasons, the researcher preferred and employed the VSM to undertake this study.

2.5.3. Probabilistic Model

Unlike VSM, which consider the ***similarity*** of documents with user query, the probabilistic model considers the probability of the relevance of a document to the user query and the ***distribution*** of query terms in documents to determine the relevance of documents (Yegosh et.al., 2013).

The central idea of probability model is the pre-estimation of the probability of relevance of documents to the user query (Beaza-Yates & Ribeiro-Neto, 2011), (Asnakew, 2018). The probability model mostly uses the distribution of terms in the user query as well as documents in the corpus to determine the probability of relevance to the given query. The model performs the initial guess to retrieve the first set of relevant documents from which the user will select relevant documents and provides feedback. The system will utilize additional information provided by the user to refine the query and repeats the process until more relevant documents are retrieved.

Term weighting in this model is based on probability of relevance using binary independence model (BMI) and its value is binary (Arjun & Sindhu, 2014). The ranking of retrieved documents is based on the decreasing probability score.

2.5.4. Semantic Model

The Semantic web is the addition of a present web that describes information in well-defined meaning and supports the machines to realize information. The Web has become an object of our daily life because multimedia information such as graphics, audio or video has become a main part of the web's information traffic. When we compere semantic web with traditional search, semantic web executes information that links proper meaning with the content but traditional search engines retrieve the information based on the occurrence of words in a document. The attention of semantic web is to make the web as machine- realizable by announcing comments, where automated agents will be able to realize the content on the web, start the connection between them and take logical conclusions to complete the complex task with least human interaction. Ontology is usually defined as formal vocabulary and is considered as one of the main pillars of the semantic web. This is used to define the world by stating the concepts and their connections in the clear way. The Semantic Web search engines goal is to fetch the reliable, precise, relevant information what actually user needs without taking much time to traverse the irrelevant pages what traditional search engine does (K.Ezhilarasi, Literature Survey: Analysis on Semantic Web Information Retrieval Methodologies, 2018).

2.6. Comparison of the IR Models

Sakthi Murugan and Dr. G. Aghila (Sakthi Murugan & Aghila, 2013) have evaluated various IR models based on five important criteria such as: the real time applications, precision, recall, pre-processing and retrieval time complexity. According to their study, vector space and semantic similarity indexing models have higher precision and recall value and most real time application now a days are developed by these models. The probabilistic model has low precision value and takes long preprocessing as well as retrieval time compared with VSM and Semantic models. The summary of their evaluation results of the three major IR models against five basic benchmarks are as follows. The **greater number of dollar sign** ($) indicates that, the model is **more important** based on the evaluation criterions.

Table 2. 3: IR model comparison

IR Models	Comparison Criterions				
	Real time Applications	**Precision**	**Recall**	**Preprocessing Time**	**Retrieval Time**
Vector Space	$$$$	$$$	$$$	$$$$	$
Probabilistic	$	$	$$	$$$$$	$$$$
Semantic	$$$	$$$	$$$	$$$$	$$$$

2.7. Review of Related Studies

2.7.1. Related Works on Non-Ethiopian Languages

Hiteshwar Kumar Azad and Akshay Deepak were done optimistic study on Query Expansion Techniques for Information Retrieval. Researchers show why frequently, a web-search does not produce relevant outcome by providing different reasons. Among Researcher reflect reasons user query relation with multiple topics, the shortage of query to express what user in search of, user understanding what he is seeking before he realizes result considered as a challenge why web-search does not produce relevant outcome. Researcher see query expansion as a solution for these above reflect problems. Then researchers classify each type of the query expansion and explain the important of the query expansion. Researchers reflects query expansion techniques from four key parts. These are data sources, applications, working methodology and core approaches. Researchers also explained Synonymous and polysemous words as case for the reduction of recall and precision value.

Abdelmgeid Amin Aly (2008) was done hopeful study by using a query expansion technique to improve document retrieval. The objective of researcher was to finding suitable extra term by using query expansion technique. Researcher list some information retrieval models and explain their limitation, which case mismatch between document and user query. Researcher provides query expansion as a solution for the above limitation and classify query expansion in to three types like: manual, automatic and interactive. Researcher used two query expansion technique in sequence to reconstruct query. These similarity thesaurus-based expansion and local feedback techniques. Researcher used three standard test collections and evaluate the

retrieval performance by using average precision measure. Researcher showed that the improvement increases with the size of the collection and with extra search terms.

2.7.2. Related Works on Ethiopian Languages

Query Expansion Using Local Analysis

A lot of works has been done on Ethiopian languages in IR expected to develop the performance of the retrieval process. Maximum of the studies started in Ethiopian languages are based on Local Analysis: using the arithmetical calculation of the terms within the documents included in data set.

In 2012 Gezehagn was designed and developed hopeful information retrieval work for Afaan Oromo text. To make Afaan Oromo language document information retrieval effective research applay and evaluat vector space model approaches. Researcher have been collected different data from different sources to organized 100 documents and 9 queries for expermentation. The researcher did the total implementation of the information retrieval system in two main Element. These are Indexing and Searching. In indexing part, the researcher applied text pre-processing techniques for both documents and queries to make document and query ready for the similar process. To save memory space and to increse the speed of searching process researcher applied inverted file indexing structure for every non stop words. To represent document and query as vector researcher used vector space model in searching part. Researcher to measure the rank of document terms to the user query terms applied TF*IDF term weighting method. In order to measure the similarity of documents with user query as well as to ranking the retrieved documents lastly researcher used Cosine Similarity approach. Researcher had gote the experimental result for precision 57.5 % and for Recall 62.64% . Even if, improved outcome was achieved in precision and recall values, the essential performance of the information retrieval system was not achieved. The key difficulties were word mismatch due to synonymous and polysemy terms according to the researcher. As the researcher recommended thesaurus will handle the difficulties of synonymous terms in order to improve the performance of the information retrieval systems.

Talented work was done by Senbeto Kayimo(2019), who designed thesaurus-based query expansion for Sidama information retrieval. Researcher applyied the vector space model to Sidama language text retrieval system to make sidama language document retrieval effective. Researcher prepared 250 documents as dataset and 10 queries for prototype evaluation and

stored as UTF-8 encoding format which is suitable to support several languages. Researcher have been collected different documents from different sources for dataset preparation such as a Sidama Television Corporation, Sidama Zone Culture, Tourism and Sport bureau, Hawassa Teachers Education Collage, Hawassa University Sidama language Department, Sidama Zone Education Bureau (SZEB),Dayye FM and Radio program agency. The researcher implemented the IR system in three main parts. These are Indexing, query expansion and Searching. Indexing is the first part according to researcher arrengement, then researcher applied text pre-processing methods such as tokenization, normalization, stop word removal and stemming for document and query to make both are ready for the similary process. Query expansion is the second component as the researcher organized which receives stemmed query terms from text pre-processing function and expands the user query by adding synonymous terms from the thesaurus. For each query terms the expansion function adds one or more related terms from thesaurus without the participation of user in delivering extra information for query expansion. Searching is the third part as the researcher arranged sequence. Researcher used vector space model in searching part to denoted both documents and user query as vector. And also, researcher applied TF*IDF term weighting approach to measure the importance of document terms to the user query. In order to measure the similarity of documents with user query as well as to ranking the retrieved documents lastly researcher used Cosine Similarity approach. Researcher was done experimentations in two techniques. These are baseline vector space model without the query expansion and using vector space model with thesaurus-based query expansion. Researcher was recorded for each of the 10 queries Precision, recall and F-measure average values. In the first experiment researcher evaluate average performance results of the system without query expansion in terms of precision was 55.13%, in terms of recall was 77.91% and in terms of F-measure was 63.19% respectively. In the second experiment researcher evaluate average performance results of the system with the thesaurus-based query expansion in terms of precision was 58.18%, in terms of recall was 90.99% and in terms of F-measure was 69.67% respectively. The experimentation results show that, the IR systems with thesaurus-based query expansion outperforms the standard VSM model by 6.48% F-measure. As this research shows the second experimentation was more preferable for the effective Sidama information retrieval system than the first one. But during the study, the absence of rule based stemming algorithm and uniform thesaurus were the major problem that also reduced performance of retrieval efficiency.

In 2017 Tsadu Zeray was design hopefully query Expansion for Tigrigna Information Retrieval. Researcher had been collected different data from the different data sources such as health, education, Religion and philosophy, social, politics, sport and art related areas to prepared 300 Tigrigna document corpus and 10 user queries for experimentation purpose. The researcher did the whole implementation of the information retrieval system in three main Element. First, researcher perform morphological analysis in order to get the accurate stem form of words. Second, researcher used WordNet as reference to realize exact sense of ambiguous terms Third researcher was reformulate query to expand the query with the known sense using word sense disambiguation from the WordNet. Lastly, the query expansion module is integrated with information retrieval system to meet the planned goal of improvement of Tigrigna information system performances. The searching and the prototype system have been developed using PHP programming language with MySQL database and JSON files based on the architectural. Researcher had got the experimental result for precision 78.8% %, for Recall 0.91% and for F-measure 0.75 when using morphological analysis. Performance increase in percentage of precision 9.7 %, recall1.6 %, and F-measure 8%. This experiment shows good improvements on the result. After query expansion and morphological analysis applied the experimentation result of the system turn out to be precision 81%, recall 94% and F-measure 84%. The key goal of this study is to increase the performance of the probabilistic information retrieval system. To achieve his objective, researcher use query expansion model and query terms expanded using synonymy, to increase the retrieval of relevant documents and decrease the retrieval of non- relevant document. This study shown improvement by 12% precision, 4% recall and 10% F-measure in overall performance.

Another work has been done by Abey Bruck and Tulu Tilahun in 2015 on Bi-gram based query expansion model for Amharic information retrieval system. The goal of researchers is to upsurge precision of an Amharic information retrieval system though protective the original recall. Researchers were tested system by using two approaches. These are using bi-gram mode without query expansion and using bi-gram model with query expansion. The test outcome presented that bi-gram based model outperformed the original query-based retrieval, and scored 8% development in entire F-measure. This outcome was encouraging researchers to design an appropriate search engine, for Amharic language. Researchers have been prepared 21,000 steamed terms, 930 documents and Nine queries with polysemous terms are formulated for testing from Amharic Bible Old Testament. Researcher had got the experimental result for Recall 67%, for precision 57% and for F-measure 61% when Systems using bi-gram model

with query expansion and for Recall 83%, for precision 39% and for F-measure 53% when Systems using bi-gram model without query expansion.

Table 2. 4: Summary of IR related works on Ethiopian languages

Research Title	**Author and Year**	**QE Approach Used**	**IR Model**	**Corpus and Query**	**Effectiveness**
Thesaurus-based query expansion for Sidaama information retrieval	Senbeto K. (2019)	Thesaurus (Global Analysis)	VSM	• 250 Docs • 10 Queries	69.6% F-measure
Query Expansion for Tigrigna Information Retrieval	Tsadu Zeray (2017)	Word sense disambiguation from the WordNet(GA)	Probabilistic	• 300 Docs • 10 Queries	84% F-measure
Bi-gram based query expansion model for Amharic information retrieval system	Abey Bruck & Tulu Tilahun (2015)	Pseudo relevance feedback	Bi-gram Model	• 930 Docs • 9 Queries	61% F-measure
A Probabilistic Information Retrieval System for Tigrinya	Atalay Luel, 2014	Relevance Feedback	Probabilistic Model	• 300 Docs • 10 Queries	74.4% F-measure

Many promising studies have been tried as described above in many local and foreign languages but only few attempts have been made on the Sidama language. Document and query term-mismatch problem still remain unsolved, limited documents and queries are used to test the efficiency of the system.

According to the morphological complexity of a languages, inflectional and derivational morphologies can result in very large numbers of variants for even a single word. As a result, variations in word formation can have a strong impact on the effectiveness of information retrieval (IR) systems and on morphological analysis tools Alemayehu and Willet (2003). Subsequently, language specific IR Systems required to be designed that efficiently fit the morphology, word formation and all the features of certain language. The aim of this study is to design a prototype Query Expansion for Sidama Language IR.

CHAPTER THREE

3. SIDAMA LANGUAGE WRITING SYSTEM

3.1. The Overview of Sidama Writing System

In this chapter we were discussed the morphological characteristics of the Sidama language. Section 3.1.1 gives a brief explanation of the Sidama language writing system. Section 3.1.2 discusses the Inflectional and Derivational Morphology of the Sidama language. In section 3.1.3 shortly explain about Morphology of Nouns. Section 3.1.4 discusses about the morphology of verbs. Lastly, section 3.1.5 discussed about the morphology of Adjectives

Sidama language is one of the official working languages in Sidama Regional State. It mostly spoken in the Southern part of Ethiopia in Sidama National Regional State and peoples they live around Sidama National Regional State. Sidama language is the biggest language next to Afan Oromo and Somali, under Cushitic category with more than 5 million speakers of the language as we have discussed in previous chapter. It nearly related to Hadiyyisa (53%), Gedeuffa (40%), Kambaatisa (62%), Burji (41%), (Alaba) (64%), and Kabeena (64%), (Desta L. L., 2020). As Anbessa and Tafesse G/Mariam (Anbessa & Tafesse, A Linguistic Analysis of Personal Names in Sidaama, 2019) stated, Sidama language is observing rapid calibration and progress in the last two decades starting from the late 1990s. The language began to give as subject from first cycle up to preparatory in Sidama National Regional State. It has also a department at Hawassa university. In the 1933 the first Sidama language document was seem (Anbessa T. , 1987). Gospel of Mark book was also translated into Sidama text using Latin script during that time by Sudan Interior Religious Missionaries. In 1937 African-Italian chief political director Minister Mariyo Martino was wrote the first Sidama Language grammar. The Provisional Military Government of Ethiopia and Adult Education Department of the Literacy Campaign translated several texts into the Sidama language Following 1976, (Senbeto K. , 2019). In 1983 Armido Gasparini was develop the first Sidama-English Dictionary. Starting from 1970 to 1993, most of Sidama texts were written in Ethiopic Script. The Latin Script was assumed as Sidama Script in 1993 because during that time Federal Democratic Republic of Ethiopian government stimulated the dialect languages to be the language of administration and medium of primary education. During that time the Sidama Language

was officially stated as the medium of work and instruction. Sidama language shares many structures of the English writing system with some changes. Therefore, letters in English or Latin alphabets are also found in Sidama language, except for the method they are combined in phonetic alphabets and the styles in which they are spoken. However, the linguistic structure is completely different. In Sidama language, the creation of the sentences is **Subject-Object-Verb.** But in English, the creation of the sentence is **Subject-Verb-Object** (Mekonen M. W., 2022).

3.1.1. Sidama Language Letters and Sounds

Sidama scrip uses the total number of 34 Latin letters including Apostrophe mark among them 27 alphabets are single symbols letter [A,B, C, D, E,F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, ‘] and 7 alphabets are the two other letters combined to form one Sidama Latin alphabets, they are said to be “Siwiilu Fidalla” such as [**CH, DH, NY, PH, SH, TS, ZH]**. Apostrophe mark (‘) is used as one of letter in Sidama writing system. Similar to English letters Sidama letters has both small and capital symbols. From 34 letters 29 are consonants and 5 are vowels. Vowels are A, E, I, O, U which can be used in small and capital forms to form words. **From total of 29** consonant letters five letters are categorized as borrowed letters (“Argete Fidalla”), which includes **P, TS, V, Z, ZH.** There are no Sidama language words that can be formed by using these borrowed consonant letters. But these letters are used to form adopted words from other languages. For example the consonant p is used to form adopted words like “Pappaayya” meaning papaya.By using TS, words like “Tsaloote” meaning Pray in English can be formed. By using v, words like, “Vayiirese”, in English virus “Vayiittaammine” in English vitamin etc can be formed. With consonant Z, words like Zayiite, in English which means oil” etc can be formed. By using borrowed letter zh, words like “Zhaanxila” in English which means an umbrella.

3.1.2. Vowel phonemes

There are five short vowels [A, E, I, O, U] and five long equal vowels [AA, EE, II, OO, UU]. They are like to those in English, but they are spoken differently. Each short or long vowel is pronounced in the same technique during its usage in any Sidama language literature. When in Sidama language vowel shortening and lengthening, then the meaning of the word can be changed. There are several Sidama language words whose meaning is changed by the vowel shortening and lengthening. For example, ***Sinna*** *in English which*

means **Branchs,** while ***Siinna*** *in English which* means **coffee cups**. In the above words, the shortening and lengthening of the vowel /i/ make the meaning difference between the two words. All vowels have the same character in the word. In table 3.1 & 3.2 shows some examples of how to differentiate the meanings of words in the case of vowel shortening and lengthening in the word.

Table 3. 1: An example of the shorten of the vowel

Vowels	***Vowel with short sound***	***Meaning***
A	***Jawa***	*'Great, old'*
E	*De'a*	*'To neglect'*
I	*Dina*	*'To limp'*
O	*Dora*	*'Clay soil'*
U	*Kula*	*'to tell'*

Table 3. 2: An example of the lengthening of the vowel

Vowel	**Vowel with long sound**	**Meaning**
AA	**Jaawa**	'to become thin'
EE	Dee'a	'to have diarrhea'
II	Diina	'Enemy'
OO	Doora	'To elect'
UU	Kuula	'Blackish blue'

Vowel can also be shortened and lengthened at the beginning, middle and end of the word. For instance,

At beginning

/uula/, 'in English which means Earth',

At middle

/Kaasa/, 'in English which means to plant'

At end

/Waa/, in English which means 'water'

3.2. Sidama Language Morphology

Morphology is studying the internal structure of the word when during word formation. Word is formed by the collection of the morphemes. A morpheme is the building block from which a word is made up. Morphemes can be classified into two these are **free morphemes** and **bound morphemes**. Free morphemes have a lexical meaning and can stand alone. Free morpheme can be classified in to two: **lexical morpheme** and **functional morpheme** (Dr Israa Burhanuddin, 2019). While **bound morphemes** cannot stand alone and needs free morpheme to exist and change the grammatical meaning of the word. Bound morpheme can be classified into two: **inflection** and **derivation bound morpheme**. For instance, in the English language, the word **dog** is a free morpheme such that it can give full meaning to the reader. If the word changed to **dogs** the word contains both **free** and **bound** morpheme **dog + s**. as described above **dog** can stand alone because it is free morpheme but morpheme **-s** cannot stand alone because it is bound morpheme. The function of **-s** in the word is to show the plural number of the dog. When we take example by Sidama language the word **/iteemmo/,** is produced from /it/, and /–eemmo/. The word **/it/,** is a free morpheme such that it can give full meaning to the reader. While the word **/–eemmo/** is a bound morpheme such that it can't give full meaning to the reader. So, without free morpheme, bound morpheme has no meaning. There are two productive methods to form words from morphemes. These are **inflection** and **derivation**. **Inflection** is a process in which the identity and class of a word doesn't change, which means the word is still the same lexeme, belonging to the same class. So, a **verb is still a verb when inflected** for example if we take the word base form like **work** and when we put in the third person singular present indicative form it will become: **works**, when we put in the present participle form it will become **working**, when we put in the past participle form it will become: **worked**, **Nouns is still a noun, when inflected** for example if we take the word base form like **Apple,** the plural form is: **Apples** and also **adjective is still an adjective when inflected** for example if we take the word base form like **thick,** the comparative form is: **thicker** and the superlative form is: **thickest**. They do not change the part-of-speech category but the grammatical function is changed. Many inflectional features appear on words to express agreement (**person, number, and gender**) as well as to express **case, aspect, mood, and tense**. For example, in English, the word /play/ is a verb, and inflectional forms like /play/, /plays/, /playing/, and /played/ and in Sidama language the word /gani/ is a verb, and inflectional forms like /ganitu/, /gananni/, /ganino/

are produced by adding the 3rd person singular maker /-s//-itu(F.)/, the present continuous marker /-ing//-anni/ and the perfective /-ed//-ino/ respectively. These four-word forms of /play/ and /gani/ (i.e., /play, plays, playing/, and /playeded/ and /gani/, /ganitu/, /gananni/, and /ganino/) are all verbs and there is no change in the part-of-speech category due to the affixation. **Derivational morphology** is the type of morphology which study the formation of new words that differ from stem word either category or meaning. For example, do ➢undo (both are verbs, but they have opposite meanings) and also child➢ childhood (both are nouns, but they have different meanings) as well as when we take yellow ➢yellowish (both are adjectives, but they have slightly different meanings). And also, here are some examples of words which change class, but not the general meaning: central(adjective) ➢centralize (verb), deny (verb) ➢denial (noun), fog (noun) ➢foggy (adjective). In general, inflectional morphology is the study of the modification of words to fit into different grammatical contexts whereas the **Derivational morphology** is the type of morphology which study the formation of new words that differ from stem word either **category** or **meaning**. **Compounding morphology** is the process of forming a new word through combining of two or more words. Compounding is a process of word formation that combine two free morphemes into a single compound form. There are many options, but the most common ones are: noun + noun which give us noun: e.g. girl + friend ➢girlfriend, noun + verb which give us noun: e.g. guess + work➢guesswork, adjective + noun which give us noun: e.g. black + bird➢blackbird, noun + adjective which give us noun: e.g. sugar + free➢sugar-free. When we take as example compound words in English, /dogcatcher/ and in Sidama language /haqqiboce/ both dogcatcher and haqqiboce are compound words, because both /dog/ and /catcher/, and /haqqa/ and /boce/, are complete word forms in their own right before the compounding process has been applied, and are subsequently treated as one form.

In general, the language syllable's structure follows the (C)V(C)(C) general pattern for the creation of words, which means the greatest common syllable structure is CV, but some words begin with a VC syllable structure. Where C is the consonant and V is the vowel. Word had created in Sidama language by the combination of vowels and consonant. Vowels are said to be sound makers and can sound by themselves, while consonants are said to be sound receivers and cannot sound by themselves. In Sidama language word formation process both consonants and vowels can be shortened and

lengthened except **SH**. As it needs single **sh** in Sidama word "busha" to mean "bad or ugly" and double **shsh** in word "bushsha" to mean "soil". The highest number of consonants or vowels that can happen consecutively in Sidama language is two. Sometimes it appears to take more than two the same vowels with diverse words. At that time, it must take a double glottal stop or a single glottal stop to segment these vowels. For instance, ***'Tii'i****" in English which means* ***that one which used for female****,* it cannot have more than two successive consonants, one after the other. A single consonant and two consecutive consonants in a word will produce two diverse meanings of that word. The paired alphabets which explained above are used as a single letter. In the Sidama language SIWILU FIDALLA such as CH, DH, NY, PH, SH, TS, ZH are never double except SH**.** For example: "*AMANYOOTE*", in English which means discipline, DHABATA in English which means strong, **"**HAQQICHO in English which means 'tree, "*QUPHIICHO*" in English which means egg, "*TSALOOTE*" in English which means PRAYER, and "*SANXARAZHE*" in English which mean table. Among the paired letters, only "**S**H" have both hard and slight sound when we speak it. Solid sound is at all times denoted by double letter, while the slight sound denoted by a single existence of the paired letter. When "SH" come in word consecutively it has solid sound for example, "*BUSHSHA*" in English which mean "SOIL**"** and it has slight sound in word when it is single "*BUSHA*" in English which mean **"**BAD**"**. Due to the single and double consonants letters the meanings of the two words are different. The single and doubled characters are not the character of all consonants letters; because some consonants do not double at all, and some are infrequent at single. Consonants cannot be geminated(doubled) at the beginning and end of the word. Example, */BBAATTO/* instead of" BAATTO**"** - *earth*.

3.3. Morphological Categories of Sidama Language

Word shares a common grammatical property according to Anbessa Teferra said (Desta L. L., 2020). There are five-word classes in Sidama language. These are: **nominal, verb, adjective, adverb, postpositions**. But, Indrias Xa'miso et al specified **Conjunction** as additional sixth word class. Each word class has a specific morphological and syntactic property and can be classify its own subclass. Commonly, (Kawachi K. z., 2007) existing these word classes as open classes and closed classes. open classes contain Nouns, verbs, adjectives, and adverbs. Closed classes involve pronouns and related forms, clitics, and interjections. In our study we can't give more detain explanation about the difference of

both classes, rather we will discuss some of word classes i.e. verb, noun, adjective, and its inflections.

3.3.1. Nominal Based Morphology

The morphology of nominal is classified into three sub-classes. These are: nouns, derived nouns and pronouns. In Sidama language noun consisted by noun stem, that noun stem always ends with vowel according to Anbessa Teferra said (Desta L. L., 2020). These vowels they come at the end of the Sidama language noun stem are known as **/a/, /e/, /o/**. Nouns are marked for **number, gender, and case** grammatically. **Number** in Sidama language is singular and plural. Most singular nouns often take a singulative marker. In Sidama language Singulative is marked by /- cho/ for example /ger- cho/ in English 'one single old man'. For plural marker, some noun takes plural suffix /-uwa/ and /-ubba/ for example, Ballo, (singular)→Ball-uwa (Plural), uridde (singular)→uridd-ubba (Plural). **Gender** in Sidama language is famous in several ways. Gender marker can be marked for **feminine** and **masculine** suffixes. There is diverse gender-marking suffixes. These are: ➢ Some connection terms and common domestic animal names differentiate gender lexically. For example /qedhicha/(M) in English which means 'unmarred boy '→/seemo/ (F)) in English which means unmarred girl ', /lukicha/ (M) in English which means 'rooster'→ /lukitte/(F) in English which means 'hen', ➢ For some nouns, gender distinction is indicated via gender-marking suffixes such as: /-cha/ and /- eessa/ mark for masculine but, /-tte/ and /-eette/ mark for feminine. For example., /moot-i-cha/ (M) in English 'Lord', /moot-i-tte/ (F) in English 'master'. Plural nouns do not make gender distinctions. Thus, for instance, /og-eeyye/) in English which means 'bonesetters' is a plural form for both the **masculine** and **feminine**. In a couple of relationship nouns, feminine is marked by /-ee~e/ while the masculine is unmarked or basic in form. e.g., /ahaaho/(M) in English which meams grandfather →/ahaah-e/(F) in English which meams grandmother, /aroo/(M) in English which meams 'husband'→/ar-ee/(F) in English which meams 'wife'.

3.3.2. Consonants and vowels bunch

Sidama Language is very rich in **derivational** and **inflectional** morphologies (Kawachi K. z., 2007). Nearly all of its affixes belong to suffix. Sidama language has only negative "di-". prefixes. Every Sidama word ends with vowels. Sidama script has no prepositions

and definite articles (Kawachi K. , A grammary of Sidaama a cushitic language of Ethiopia, 2007). In addition to 34 Latin alphabets, the Sidama word formation use 5 glottalized consonants like (/'l/, /'m/, /'n/, /'r/, /'y/) which is always written apostrophe (') followed by consonants (l, m, n, r, and y). The following words shown in table as follow are glottalized words formed using glottalized consonants.

Table 3. 3: Glottalized words formed using glottalized consonants.

Glottal Consonants	**'l**	**'m**	**'n**	**'r**	**'y**
Words formed with glottalized consonants	'***kaa'lie'***	'***Su'minsa'***	**'Mi'na'**	'***Bu'ra***'	**Gerecho'ya**
Meaning of words in English	Help me	Their name	To build once house	Raw	My sheep

3.3.3. Inflectional and Derivational Morphology

In Sidama language Word formed by both **inflectional** and **derivational** morphologies. In inflectional morphology different words are formed from the root word without altering the class of words. In Sidama language word formation by inflectional morphologies maximum uses suffixes, and few are infixes. These by both inflectional and derivational morphologies formed words are **noun**, **verb**, **adjective**, **adverb** and soon. **Nouns** are any word that can be used to name or identify a place, country, object, animals or idea. There are two types of grammatical genders exist in Sidama language nouns. These are **masculine** and **feminine,** and the whole nouns of the language belong to one of these gender categories. Eg. '***ijaara'*** (noun) in English 'building', ***'ijaara-nke'*** (noun) in English 'our building', ***'ijaara-kki'*** (noun) in English 'your building', ***'ijaara-nsa'*** (noun) in English 'their building'. These words are in the same part of speech category but created by adding inflectional suffixes such as ***'-nke'***, ***'-kki'***, ***'--nsa'***, ***'oommo'***, ***'-eemmo'***, ***'ino'***, ***'-tino'***, e.t.c to mark gender, tense and person. **Verb:** In Sidama language verbs are also words that are used to indicate action, state of being, and event occurrences within time boundaries. Verb can be classified as modal, transitive, auxiliary and intransitive verbs. **Transitive verbs** are those verbs that transfer message to complement or object whereas, **intransitive verbs** do not transfer message to complement and hence, do not have complement or object. The following examples illustrate this fact. **Enjonsana *ijaara* hidhitu** in English which means **'Enjonsana bought building'**. The verb" hidhitu" in

English **'bought'** is transitive and a" ***ijaara***" in English '**building**' is the direct object because it receives the action (a ***ijaara*** -'buildig' is what is being bought). Similarly, Sidama language inflectional words such as '***ang-o'*** (verb) in English which means 'let us **drink'**, ***'ang-oommo'*** (verb) in English which means '**we drink'**, ***'ang-eemmo'*** (verb) in English which means '**we will drink'**, ***'ang-oonni'*** (verb) in English which means '**drank'**, ***'ag-i-no'*** (verb) in English which means '**he drink'**, ***'ag-gi-no'***(verb) in English which means '**she drink'**. By using derivational morphology, Sidama words can be created which are falling in different parts of speech. For example, words such as ***'egenna-amo'*** (adjective) in English which means 'knowledgeable', and ***'egenn-a'*** (Verb) in English which means 'to know' are derived from Sidama word ***'egenno'*** (noun) in English which means 'knowledge'. The table 3.4 below shows some Sidama words formed by derivational morphology.

Table 3. 4: Sample derivational words

Root word	Derived words
'Ito'(noun) 'eat	***'I-ta-nni-cho'*** (Advective) 'eater'
'*ago'*(noun) 'drink'	'***ag-aani-cho'***(Adjective) 'drinker'

Suffixes that derivate nouns from verbs

There are many bound morphemes suffixes that derive **nouns** from **verb** in Sidama language such as ***'-insho'***, ***'-ille'***, ***'-aancho '-ma'***, ***'o'***, ***'-atto'***, ***'-ano'***, ***'-aano'***, ***'-ansho','***, and ***'-imma'***. These morphemes replace the suffixes, ***'-i'*** or ***'-a'*** of verbs to form nouns. See e.g in the table 3.5 below.

Table 3. 5: Derived nouns from verbs

Verbs	**Nouns derived from Verbs**	**Suffix Morpheme**
Bax-a to love	***'bax-ama'***'loved'	***-ama***
'giw-a' 'to hate'	***'giw-ama'*** 'hatred'	***-ama***
'Il-a' 'to born'	***'il-ama'*** **'borned'**	***-ama***
"La'-a" to see	***La'-ama seen***	***-ama***
'bax-a' 'to love'	***'bax-ille'*** 'love'	***-ille***
'geedh-a' 'to become old'	***'geedh-imma'*** 'old age'	***-imma***
'moor-a' 'to steal, to rob'	**'moor-aancho'** 'robber' (singular)	**-aancho**
Ag-o to drink	**'Ag-ancho'** drinker	**-ancho**

Hag-idha to happy	***'Hag-idhancho'*** *'happer'*	***-idhancho***
'huucc-(i)a' 'to pray'	***'huucc-atto'*** 'prayer'	***-atto***

3.3.4. Morphology of Nouns

Like English nouns Sidama nouns are formed by using both inflectional and derivational morphologies. Inflectional nouns are inflected for **number, gender and possession**. Derivational nouns are derived from both **verbs** and **adjectives.**

3.3.4.1. Inflection of Noun for Number

In Sidama language nouns have diverse morphemes for singular and plural nouns. For that reason, to name single animal including person and thing, we will be added different suffixes on the nouns such as, ***'-o'***, ***'-a'***, ***'-shsho'***, ***'-kko'***, ***'-iicho'***, ***'-icho'***, ***'-cho'*** and ***'-aancho'***. On the other side, to name plural noun, we will also be involved various morphemes to the original noun such as ***'-e'***, ***'-go'***, ***'amo'***, ***'-nna'***, ***'-ewo'***, ***'-uuwa'***, ***'-uwa'***, ***'-he'***, ***'-o'***, and ***'-ta'.*** Examples are given in the table 3.6 below.

Table 3.6: Singular and Plural Noun Morphology

Singular	Plural
An-a in English father	***An-uwa in English fathers***
Am-a in English Mother	***-Am-uwa In English mothers***
'fara-shsho' in English 'horse'	***'fara-do' in English*** 'horses'
'yema-kko' in English 'rat'	***'yema-he' in English*** 'rats'

3.3.4.2. Inflection of Noun for Gender

In Sidama language nouns gender distinction between the **masculine** and **feminine** is indicated by using suffix **-ich**a and **-eessa** for masculine and **-ette** and **-eette** for feminine respectively. For example, "**Moot-i-cha**" in English which means Lord for masculine and '**Moot-i-*itte***' in English which means master for feminine. Certain nouns are inflected by removing ***'-imma'*** and replacing it with suffix ***'-eessa '*** for masculine and '***-eette***' for feminine to form new nouns. For example, "Og-imma" in English which means bonesetter turn out to be "Og-e-essa" for masculine and "Og-e-tte" in English which means bonesetter for feminine. The table 3.7 given below presents sample nouns inflected for gender.

Table 3.7: Sample nouns inflected for gender (Senbeto K. , 2019)

Common Noun	**Masculine**	**Femini ne**	**Gender Marker**
'Babba' *in English* 'stutter'	*'babb-icha'*	*'babb-itte'*	*'-icha/-itte'*
'Dunka' *in English* 'passive'	*'dunk icha'*	*'dunk-itte'*	
'Balla' *in English 'blind eye'*	*'ball-icha'*	*'ball-itte'*	
'Cim-imma' *in English* 'arbitrator'	*'cim-eessa'*	*'cim-eette'*	
'Og-imma' *in English* 'professional'	*'og-eessaa'*	*'og-eette'*	*'-eessa/-eette'*
'Dur-'mma' *in English* 'richness'	*'dur-eessa'*	'dur-eette'	

3.3.4.3. Pronoun

Word which used to take the place of noun in the sentence is said to be **Pronouns**. Pronouns can be classified based on their functions and meanings in the sentence. These are personal pronouns, possessive pronouns, demonstrative pronouns, relative pronouns. As like nouns, pronouns have numbers and gender. For example, ***isi/*** *in English* ***he*** *is used for singular masculine* whereas **ise** in English -***'she'*** is used for singular feminine and also **insa**, and **ninke** which mean ***'they'*** *and* ***we*** *are* used for plural both masculine or feminine. When we come to Sidama language it has seven subjective/personal pronouns as like English language. These seven Sidama subjective pronouns are adding different suffixes such as ***'-se'***, ***'-si'***, ***'-ya'***, ***'-nke', '-kki'***, ***'-'ne'***, and ***'-nsa'*** in English "I, we, you, you (P), she, he, they" respectively. See in the table 3.8 below.

Table 3.8: Inflection of nouns for possession

Subjects and their category	***Common Nouns***	**Possession Nouns**
'Ane' 'I' (FPS)	***"mine"*** *'home'*	***'mine-'ya'*** *in English which means*'my home'
'Ati' 'you'(SPS) used for both		***'mine-kki'*** *in English which means* ''your home' (S)
'Ise' 'she' (3PS for F)		***'mine-se'*** *in English which means* ''her home'
'Isi' 'he' (3PS for M)		**'mine-si'** *in English which means*''his home'
'Insa' 'they' (3PP)		**'mine-nsa'** *in English which means* 'their home'

3.3.5. Morphology of verbs

In Sidama language verbs have inflectional morphology for **gender** and **tense**. For example, ***'Hagidh-i-tu'(verb)*** in English which means 'happy'- for feminine whereas ***'Hagidh-i'*** (verb) in English which means 'happy' for masculine. Similarly, ***'Fulitt-tu'(verb)*** in English which means 'go out'-for feminine whereas ***'Ful-i'*** (verb) in English which means 'go out' for masculine.

3.3.5.1. Verb Reduplication

In Sidama language some verbs are formed using repetition to express frequent activities. Usually happening verb root reduplication process looks like C1VC2V. C is the consonant letter and V is the vowel letter. In the Sidama verb repetition process, the first consonant (C1) repeats itself twice in place of the second consonant (C2). In this case the root verbs C1VC2V turn out to be C1VC1C1VC2V as example given in the table 3.9.

Table 3.9: Reduplicated Sidama Verb formation

Root verbs	**Reduplicated Verbs**
*'**Mura'** which means to cut*	***'Murmura'** to cut again and again*
*'**bica'** which means* 'to scratch'	***'bibbica'** which means*'to scratch *again and again*
*'**Faca'** which means* 'to *separate*'	***'Faffaca'** which means* 'to separate *again and again*'
*'**Gana'** which means* 'to hit	***'Gangana'** means* 'to hit *again and again*'
*'**Dara'** which means* 'to split	*'**Daddara'** means* 'to split *again and again.*

3.3.5.2. Reduplicated Verb formation for the Sidama language

In Sidama language inflectional verbs also formed from verb and noun by using combined suffix ***'-ama'***. In Sidama language the last vowel of the original verb replaced by combined suffix to form new verb. For example, ***'Hara'*** (verb) in English which means 'to go' – ***'Har-ama'*** (verb) in English which means 'to go again and again, ***'La'a'*** (verb) in English which means 'to see' – ***'La'a-ama'*** (verb) in English which means 'to see each other', ***'Gana*** (verb) in English which means 'to hit' – ***'Gan-ama'*** (verb) in English which means 'to hit each other', ***'Babbada'*** (verb) in English which means 'to separet' – ***'Babbad-ama'*** (verb) in English which means'separeted from each other'. In Sidama language **verbs** can be also formed from derived noun by removing the last vowel of the noun and by adding suffix ***'-ira'***. For example, ***'Wix-a'*** (noun) in English which means

'seed'- ***'Wix-ira'*** (verb) in English which means 'seeding, ***'Dik-o'*** (noun) in English which means 'market'- ***'Dik-ira'*** (verb) in English which means 'marketing', ***'Sandid-o'*** (noun) in English which means 'tar'- ***'Sandid-ira'*** (verb) in English which means 'taring'. Similarly, **verbs** are derived from **adjectives** replacing the last vowel of the adjective by suffix ***'-ira'***. These verbs are mostly derived from adjectives that show color. The table 2.10 below shows how verbs are derived from such adjectives.

Table 3.10: Verbs derived from adjectives

Adjectives	**Verbs**
'haanja' in English means 'green'	***'haanj-ira' in English means*** 'to become green'
'kolishsho' in English means 'black'	***'kolishsh-ira' in English means*** 'to become black'
'waajjo' in English means 'white'	***'waajj-ira' in English means*** 'to become white'
'bulla' in English means 'gray'	***'bull-ira' in English means*** 'to become gray'

In Sidama language some verbs are derived from adjectives simply by eliminating '-do' from the adjective. For E.g, ***'shaala-do'*** (Adjective) in English 'thiner'- ***'shaala'*** (verb) in English 'to become thin', ***'kaaja-do'*** (Adjective) in English 'hard'- ***'kaaja'*** (verb) in English 'to become hard', ***'ruka-do'*** (Adjective) in English 'narrow'- ***'ruuka'*** (verb) in English 'to become narrow.

3.3.6. Morphology of Adjectives

Adjectives are words which define the **qualities** of nouns. Adjective can exist in three forms like: absolute, comparative and superlative. Sidama language adjectives have morphosyntactic properties different from those of other classes, but share some morphosyntactic properties with nouns and with some verbs. Some adjectives have all or most of the characteristic of nouns (Desta L. , 2020). Both nouns and adjectives can be inflected for **gender, number, and cases**. Sometimes nouns can also serve as an adjective for other nouns. There are restricted adjective in Sidama language for gender formation: like /-eessa/ for male'M' .and /-eette/ for female 'F': E.g., /dur-eessa/, in English which

means rich for male, /dur-eette/ in English which means rich for female. There are two types of adjective classes. These are **free adjective** and **derived adjectives**. **free/true** adjective frequently expresses only some meaningfully important as well as it always indicates **dimension, color, age,** and **value**: for example; /Lowoha/ in English which means 'big', /shiima/ in English which means 'small', /waajjo/ in English which means 'white', /haaro/ in English which means 'new', /dancha/ in English which means 'good'. In general Adjectives in Sidama language are inflected for **gender, and number,** and **agree** with the nouns which they modify (Desta L. , 2020). In Sidama language adjectives are marking suffixes for both Masculine and Feminine genders. Masculine is indicated by suffixes such as /-o/, /-cha/, /-leessa/. e.g., /atoot-aam-o/ 'M', in English which means 'sinful', /ball-i-ča/ 'M', in English which means 'blind' /dur-eessa/ 'M', in English which means 'rich', /fayya-leessa/ 'M', in English which means 'healthy while feminine adjectives are marked by /-e/, /-tte/, and /-leette/ for example /atoot-aam-e/ 'F' in English which means 'sinful' /ball-i-tte/ 'F' in English which means 'blind'/dur-eette/ 'F' in English which means 'rich' /fayya-leette/ 'F' in English which means 'healthy.

Table 3.11: Adjectives inflected for Masculine and Feminine

Root Adjective	**Adjective inflected for Masculine**	**Adjective inflected for Feminine**
'busule' in English 'smart'	***'busule-ho'*** 'smart'(M)	***'busule-te'*** 'smart'(M)
'worb' in English 'clever'	***'worba-ho'*** 'clever'(M)	***'worba-te'*** 'clever'(F)
'haranch' in English 'short'	***'haranch-u'*** 'short'(M)	***'haranch-o'*** 'short'(F)
'umaam' in English 'long-haired'	***'umaa-mu'*** 'long-haired'(M)	***'umaa-me'*** *'long haired'* (F)
'godowaam' in English 'gluttony'	***'godowaa-mu'*** 'gluttony'(M)	***'godowaa-me'*** 'gluttony'(F)

Adjectives are also inflected from number by replacing the last vowel of the root adjective with suffixes such as ***'-oota'***, and ***'-uulle'***. For example, ***'busule'*** in English which means 'smart' (Singular)- ***'busul-oota'*** in English which means 'smarts'(plural), ***'worba'*** in English which means 'clever'(Singular)- ***'worb-uulle'*** in English which means 'clever'(plural), as well as adjectives derived from nouns are formed by replacing the last vowel of the noun with two suffixes ***'-aamo'*** for Masculine and ***'-aame'*** for feminine. The

following table 2.12 express how adjectives derive from nouns for both sexes. The Sidama language adjectives can also be resulting from nouns by using suffixes such as ***'-aancho'***, ***'aano'***, ***'-aamo'***, ***'-ichu'***. For example, ***'macc-a'*** (noun) in English which means 'ear'-***'macc-aamo'*** (adjective) in English which means 'someone that have larger ear', ***'bux-a'*** (noun) in English which means 'poor'-***'but-ichu'*** (adjective) in English which means 'poorly', ***'hamashsh-o'*** (noun) in English which means 'snake'-***'hamashsh-aamo'*** (adjective) in English which means 'treacherous, deceitful person', ***'bax-ille'*** (noun) in English which means 'love'-***'bax-aancho'*** (adjective) in English which means 'lovely', ***'keer-e'*** (noun) in English which means 'peace'***'bax-aano'*** (adjective) in English which means 'peaceful'.

Table 3.12: Adjectives derived from noun

Noun	**Adjective for Masculine**	**Adjective for Feminine**
'ille' 'eye'	***'ill-aamo'*** 'eyed'	***'ill-aame'*** 'eyed'
'wodana' 'heart'	***'wodan-aamo'*** 'heartily'	***'wodan-aame'*** 'heartily'
'guma' 'seed'	***'gum-aamo'*** 'seeded'	***'gum-aame'*** 'seeded'
'hinko' 'tooth'	***'hink-aamo'*** 'toothy'	***'hink-aame'*** 'toothy'

Some of adjectives derived from noun to show negative form of the noun. For example, 'seed' is (noun) - 'seedless' is (adjective) in English. These adjectives in Sidama language are formed by adding suffix ***'-iweelo'*** to the noun by removing the last vowel. See the table 2.13 given below.

Table 3. 13: Negative adjectives derived from noun

Noun	**Negative Adjective**
'Anga' which means 'hand'	***'ang-iweelo'*** means 'handless'
'Leka' which means 'leg'	***'lek-iweelo'*** means 'legless'
'Goowa' which meanss 'neck'	***'goow-iweelo'*** means 'neckless'
'hinko' means 'tooth'	***'hink-iweelo'*** means 'toothless'

As discussed above, adjectives can be also derived from verbs. During adjective derived from verb it used suffixes such as ***'-aancho'***, ***'-aano'***, ***'-aamo'***, ***'-icho'***, and ***'-ado'***. These suffixes replace the last vowel in the given verb. For example ***'kaa'l-a'*** (verb) in English which means 'to help'-***'kaa'l-aancho'*** (adjective) in English which means 'someone who

helps', ***'badd-a'*** (verb) in English which means 'to get bald'***'badd-aamo'*** (adjective) in English which means 'someone who has bald', ***'huuccir-a'*** (verb) in English which means 'to beg-***'huucciraancho'*** (adjective) in English which means 'beggarly', ***'he'm-a'*** (verb) in English which means 'to gossip'-***'hem-aanho'*** (adjective) in English which means 'someone who gossip',***'bux-i(a)'*** (verb) in English which means 'to get poor'-***'bux-icho'*** (adjective) in English which means 'poorly', ***'kaajj-a'*** (verb) in English which means 'to become strong'-***'kaajj-ado'*** (adjective) in English which means 'strong'.

CHAPTER FOUR

4. SYSTEM DESIGN AND ARCHITECTURE

4.1. Introduction

An IR system has information analysis, structure, organization, storage, searching, and retrieval potential. The most function of any information retrieval system is to process a user request and retrieve content bearing resource to satisfy the user need. According to this system architecture in Figure 4.1 below, text pre-processing technique performed in both document and user query parts as well as information retrieval system takes both documents and queries as input and produces a set of tokens as an output.

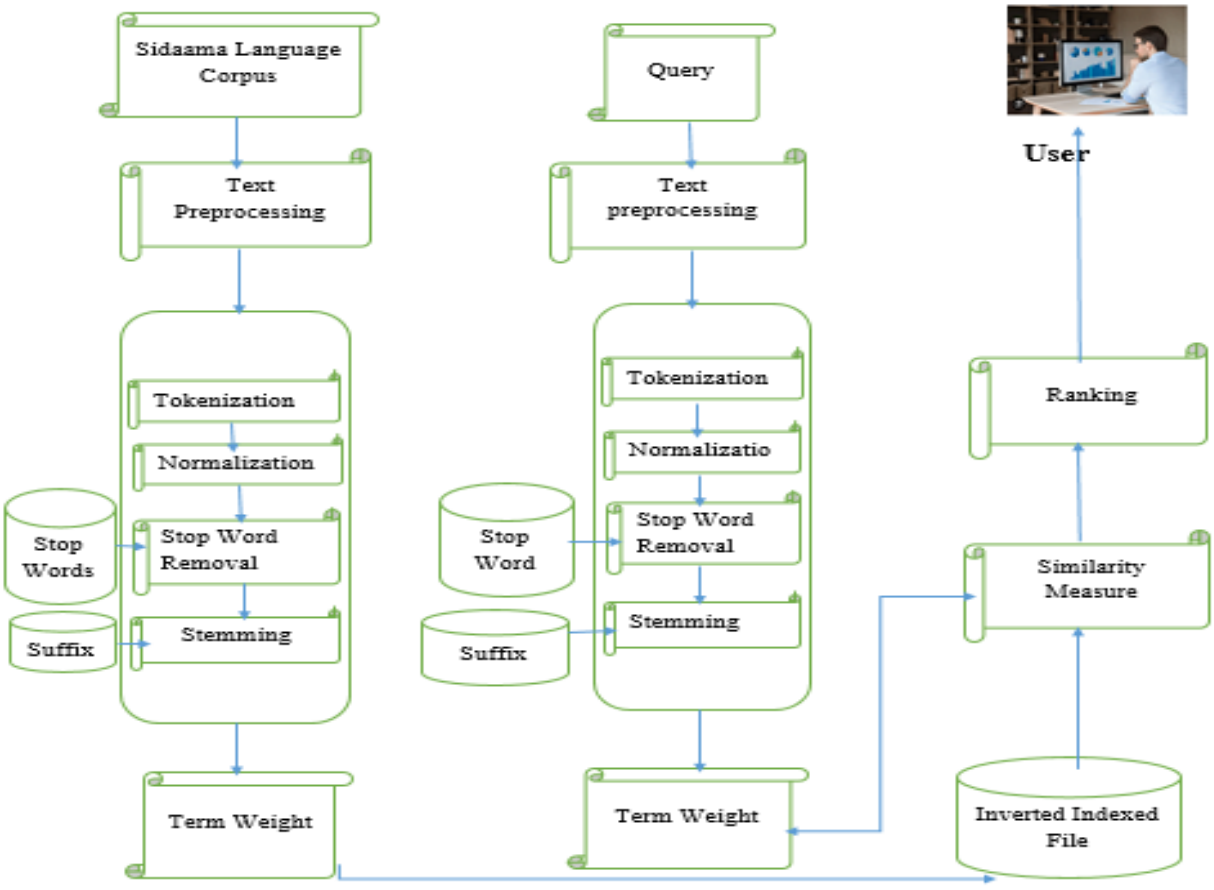

Figure 4.1: The basic architecture of information retrieval systems

Information retrieval system designed involves three main components. These are: **indexing**, ***query expansion*** and **searching**. Information retrieval system organize the given Sidama language text corpus, using index file to improve searching. Indexing is the language dependent process for that reason it requires text pre-processing such as

tokenization, normalization, stop word removal and stemming. Tokenization is the first step which used to identify stream of tokens. Normalization is the next operation according to this study arrangement which remove all punctuation mark except apostrophe mark and change all cases in to the same small case. Then the next stemmed operation follow after normalization operation check is there stop word or not. Then content bearing terms (nonstop word) are stemmed. For all stemmed tokens its appreciated weight computed and then inverted index file is built. Then similarity measuring techniques (cosine similarity) is used to retrieve and rank relevant documents. Information retrieval systems retrieve documents based on some similarity measurement technique. In addition, an information retrieval system measures their performance level in terms of **recall** and **precision**. Thus, the aim of this research is, to design a good system which satisfies the desirable retrieval limitations. This chapter presents the architecture of proposed system together with methods and techniques used to design and develop the Sidama IR system in addition to three basic information retrieval system components (indexing, query expansion and matching). Sidama corpus and query pre-processing and indexing methods as well as their algorithms are also discussed. And also, in this chapter, the VSM, TF*IDF term weighting, and cosine similarity measurement techniques and performance evaluation techniques are discussed

4.2. System Architecture

In order to evaluate the performance of the system information retrieval system required the corpus collection. Sidama language document collected from different resources passes through different steps in order to index and use the retrieval system. In information retrieval searching is efficient when the database is small. But, for the larger database indexing is the key solution. Because searching without using indexing structure to organize the document will take much more time and space for large databases. Therefore, constructing and maintaining indexing on large database is necessary in order to solve the problem mentioned above. **Indexing** is the primary component which produces content bearing index terms as we as it performs text preprocessing functions. Text preprocessing functions are applied to both documents in the corpus and user query. The **searching** function is the second component which will compare query terms with terms in documents then ranked the result of the relevant document. We applied vector space model and cosine similarity method to measure the similarity of documents with user

query. **Query expansion** is the third component which accepts stemmed query terms from text pre-processing function and expands the user query by adding synonymous terms. For each query terms the expansion function adds one or more related terms from the user in order to delivering extra information for query expansion. This supports to maximize the probability of gaining the relevant documents that can be wasted due to word mismatch

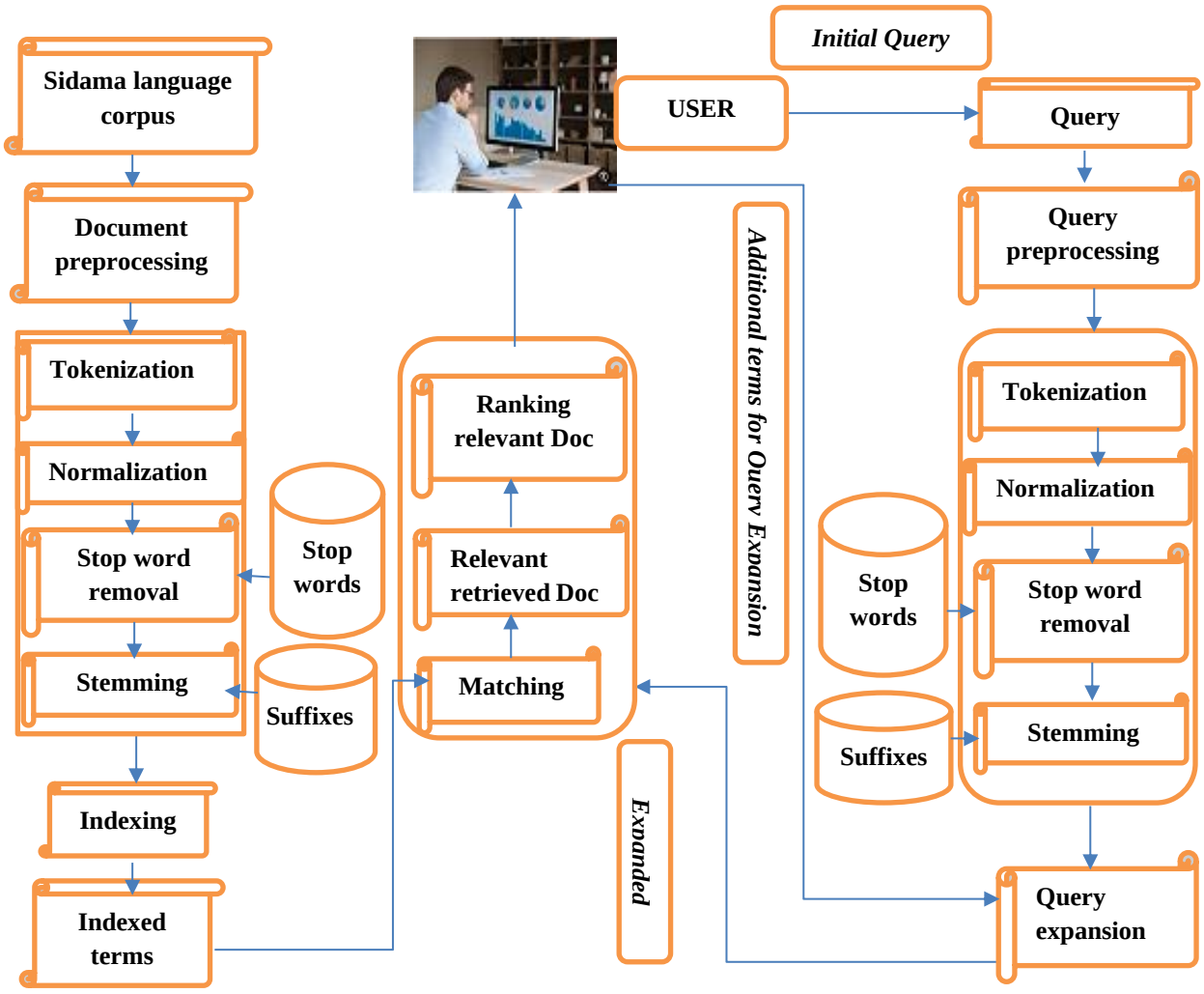

Figure 4. 2: Proposed IR architecture with query expansion

4.3. Text Pre-processing

Modern information retrieval system should be planned in a method to retrieve information of any nature. The required system architecture should be prepared information into boundary demarcation with frequent existences in the data. Information retrieval is essentially a substance of deciding which documents in a collection should be retrieved to satisfy a user's need. The user's need is represented by a query or profile, and contains one or more search terms, plus some additional information such as weight of the

words. Hence, the retrieval decision is made by comparing the query terms with the index terms (vital words) appearing in the document

4.3.1. Tokenization

Tokenization is the process of breaking down text into smaller fragments is said to be tokens. The corpus can be tokenized into paragraphs, sentences, phrases, words, sub-words, or characters. In the tokenization process the most important point is creating boundaries demarcation between tokens. We did not get any literature to review for a sentence delimiter (separate text strings such as commas (,), semicolon (;), quotes (“, ‘), braces ({}), pipes (|), or slashes (/ \)) for Sidama language texts. Afan Oromo a language uses the Latin script and has the same spoken family as Sidama language, it uses a space to show the end of one word, as well as an exclamation mark (!), a period (.), and a question mark (?) to show a word and sentence delimiter (Mekonen M. ,., 2022). So, we tokenized the corpus into sentences using sentence delimiters like an exclamation mark (!), a period (.), and a question mark (?). In order to know the boundary of end sentences, we have added” BOS” to indicate the beginning of a sentence and” EOS” to indicate the end of the sentences.

```
Read documents from the corpus
 for document in corpus
      if white space in line
         split and put as token
             end if
```

Figure 4. 3: Figure Algorithm for tokenization

4.3.2. Normalization (Conflation)

Sidama language normalizer component has two responsibilities the **first responsibility** is to check if the Sidama language words written in short form or abbreviation. To handle this characteristic, check a word written in either by **“.”** (Period) or **“/”** (slash) at the middle of word, if the normalizer gets such kind of word, replace the word with appropriate long forms. For example, M.D in Sidama language is the word which written in short form or abbreviation when it writes in long form it become **marote dirro** in English which means Ethiopia calendar. The **second responsibility** of normalizer is to

handle the existence of redundancy of some characters which written in different causes (upper, lower and mixed cause), to compile them in to one common direction. For example, "***INTO*****",** and **"*into*"**, **''Into''** are three different representation of the same word in English it has the some meaning" to mean ''**let us eat**"". This characteristic of natural language has negative effect on precision of information retrieval. The Sidama writing system has lower case, upper case and mixed case letters in word formation. So, the normalization function converts all uppercase letters of words into consistent lowercases and removes all the punctuation marks, special characters as well as numbers which have no discrimination power in a retrieval system, except single quotation mark (') which is considered as the character used for word formation. Normalization is the process of creating equivalence among or between words which taking a different form but carrying the same message. In general normalization process reduces word variants into a common form. Some Sidama terms that need normalization are presented in the table 4.1 below.

Table 4. 1: Sidama terms that need normalization

Equivalent Sidama Terms	**Normalized to**	**English Equivalent**
w.k.l	Woluno kone lawano	e.t.c
ADO, Ado, ado	ado	Milk

```
Read documents from the dataset
Read punctuation mark list
for the text document
    If text is written in upper case
    convert text into lower case
     else keep text as it is
        end else
          end if
            for print texts in lower cases
              while end of the document
                 for the punctuation mark in document
                     if punctuation mark! ()- []{};:'"\,<>./?@#$%^&*_~|=+ appear
                            remove the punctuation mark
                   end if
                     end for
            end while
```

Figure 4. 4: Algorithm for normalization and punctuation elimination

4.3.3. Stop Word Removal

Stop words removal process is the text preprocessing process which eliminate the stop words that doesn't have significant meaning when stand alone. Stop words are the words which used frequently in the document collection but it no provides serve for information retrieval. The removal of the stop words requires for different purposes, for example: to save disk space and speed up the searching process as well as to reduce the size of the query this also help from degradation of search result. In this study, stop words are collected manually by referring language experts, and researches. Stop words includes prepositions, articles and connectives etc., which are not good predicators of the content in the document. There are different stop word removal techniques. Among them two commonly used techniques are **IDF** and **dictionary lookup**. Inverse Document Frequency is a first frequency analyzing method that counts the frequently occurred words in one or more documents and assumes the more frequent words as stop words. The problem in IDF technique is that it will consider and remove all more frequent word as stop word even if they are not. The dictionary lookup method is the second technique which uses the predefined list of stop words in certain language identified by language experts. For this study, the dictionary lookup technique is used. We have identified common stop word in Sidama Language and sample stop words together with their English equivalents are presented in table 4.2 as follows. We will be present more Sidama language stop word lists in appendix.

Table 4. 2: Stop word Sample in Sidama language texts

Sno.	Sidama language stop words	Equivalent meaning in English
1	Wolootu	Another
2	Mittimma	Together
4	Mereero	Between
5	Gedeno	After
6	Calla	Only
7	Gobba	Out
9	Giddo	In
10	Ayi	Who
11	Maayira	Why
12	Maa	What

13	Ani	Me
14	Annu	Father
15	Duuchu	Many
16	Albaanni	Before
17	Ku'uri	Those
18	Daafira	Therefore
19	Mama	Where
20	Ati	You
21	Baalu	All
22	Ko'o	There
23	Mamoote	When
24	Lamunku	Both
25	Miteege	Once

Input: word
Output: stop word removed

1. Receive query word
2. While (Not end of the query) Read the word

If the word is found in the **stop word** list
Remove it
Else Continue
End if
End while

Figure 4. 5: Stop Word removal algorithm

4.3.4. Word Stemming

A stemming is the text preprocessing process which used to reduce the variant form of the words to a common stem form in order to improving the performance of information retrieval system. Stemming can be classified as: Linguistic, automatic or mixed. Linguistic stemming uses a Linguistic knowledge of the construction of the language, by providing manually compiled list of affixes, allotropy rule and so on. In such case stemming is the process of reducing inflected or sometimes derived words to their stem, base, root form. However, automatic stemmer is built using algorithms and uses a

statistical process such as frequency count, n-gram technique and some combination of both (Demewoz, Automatic Thesaurus Construction From Wolayita Text, 2013). For example, the words 'runn', 'runns', 'running', and 'runned' are various forms of the word 'runn' and these words should be reduced to its normal form, 'runn. Sidama words have rich suffixes and infixes. For example, Sidama word 'Ita' in English to mean 'to eat' has many suffixes such as "Itaancho" in English to mean (eater), "Ito" in English to mean (eat), "Iteemmo" in English to mean (I will eat), "Itoommo" in English to mean (I eat), "Itino" in English to mean (she eat), "Itannino" in English to mean (he is eating), "Intannikkiho" in English to mean (uneatable), "Intanniho" in English to mean (eatable), "Intannireeti" in English to mean (eating things), "Intanniricho" in English to mean (something that is eatable). Regarding to infixes the root form "Mura" in English to mean "cut" will have infix as "Mumura" in English to mean (cut again and again), whereas "Xiwa" in English to mean (to push) will have "Xixiwa" in English to mean (to push again and again). For this study we will used identified suffix list and applied algorithm developed by Amanuel (2012) but this study excludes infixes because it requires further study.

```
Step 1: Open documents and Suffix list
Step 2: Read and split Suffix list
Step 3: for word in document
            If the word ends with Suffix list
             Replace Suffix list with a space
               Else return word
                End else
                 End if
                   End for
Step 4: return stemmed word
```

Figure 4. 6: Suffix removal algorithm

4.3.5. Word Indexing

Indexing is the primary step for information retrieval system. During the indexing process, texts are collected, analyzed and stored in order to facilitate fast and accurate text retrieval. After text retrieve, it ranked and presented to the user according to their relevance with user query. All index terms are not equally important because the degree of the terms in

representing document is different one from other. For indexing and searching goal information Retrieval systems use inverted indexes. An inverted index comprises a group of terms and their corresponding existences in documents. Though, a single word might happen in numerous morphological different in a query and document (Tsadu, 2017). Therefore, upsurge the scope of the index and reducing retrieval performance. The requirement of indexing for reducing such types of issue. Before indexing terms with their documents, all terms pass through the text preprocessing process such as tokenization, normalization, stop word removal and stemming. After preprocessing the filtered documents pass to morphological analysis to get the appropriate root form. Finally, query terms will be indexed using Sidama language index engine.

4.4. Matching

In the proposed system, document processing involves text preprocessing, morphological analysis, stop word removal and indexing (Tilahun, 2020). On the query side, we apply text preprocessing, morphological analysis and stop word removal processes except indexing. As an outcome of this process, we obtain indexed documents. After the query and document are processed and indexed, the next step is similarity measurement and matching. Similarity measurement includes comparing the level of similarity and relatedness of user query and documents in the source. Similarity measurement calculates the number of occurrences of indexed-term existences between query and documents. The outcome of similarity measurement is relevant documents that are matched with the user's query. These documents are retrieved for the users in ranked order (Senbeto K. , 2019).
The information retrieval models are the key for every information retrieval system. The information retrieval systems are established by using information retrieval models. IR models provide procedures to calculate and estimate similarity values between user query and all documents in the source. Both query and document which involved in certain process was denoted by information retrieval models. Most of the time non-stop word(s) which happening in both user query and document determine the relevance of document to the user query. Documents and query are defined and represented in vector space model. The vector space model is applied in three basic phases (Jitendra et al., 2013). In the **first phase** stop words are removed, and only terms that hold concept are selected. In the **second phase** weight is calculated for each indexing term in relation to terms in user query. The value of weight shows the level of the relevance of a document to the user

query. The document weight values permanently fall between 0 and 1 (Haward & W.Bruce, 1992). When the document-query weight value become 0, then document is not relevant to the user query. The value of weight greater 0 shows the level of retrieved document similar to the user query. The term weight determines the significance of the term in the document to the user query. Term weight is dependent on the term frequency (TF), which occurs in the document Di. The number of documents in which the term Ti occurs is said to be document frequency (Df_i). The relation between query terms and documents is computed based on the frequency ($F_{i,}$ j) of query term (T_i) occurs in the document (D_j) and described in matrix as follows:

Documents (D_j) / Query Terms (T_i)	D_1	D_2	D_3	…	D_i
T_1	$F_{1,1}$	$F_{1,2}$	$F_{1,3}$	…	$F_{1,j}$
T_2	$F_{2,1}$	$F_{2,2}$	$F_{2,3}$	…	$F_{2,j}$
T_3	$F_{3,1}$	$F_{3,2}$	$F_{3,3}$	…	$F_{3,j}$
…	…	…	…	…	…
T_j	$F_{i,1}$	$F_{i,2}$	$F_{i,3}$	…	$F_{i,j}$

Figure 4. 7: Query term-document representation in vector space

Where N is the total number of documents in the corpus and $F_{i,\,j}$ is the frequency of query term (Ti) in the document (Di). The total frequency of query term (FT_i) in the documents in the corpus is the sum of frequencies of the term in all the documents it occurs. This is given as: -

$$FTi = \sum_{i=1}^{N} Fij \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots .3.$$

In order to decide the importance of index terms not only in a particular document, but also in documents collection, term weight should be computed and assigned to each term independently[1], in relation to the user query. Most common and most effective technique for term weighting is TF*IDF (Beaza-Yates et al., 2011); (Jitendra & Sanjay, 2013). The TF*IDF term weighting technique employs the values of term frequency (TF) and inverse

document frequency (IDF) to calculate the weight value of terms. The inverse document frequency of term i (IDF_i) is calculated as:

$$IDFi = \log\frac{N}{DFi} \quad \ldots\ldots (3.2)$$

Where N is the total number of documents in the corpus, and DF_i is the document frequency containing query term $_i$. The value of IDF is higher for rare query terms, but lower for frequent query terms occurring in the corpus. The IDF is very important to balance the two extremes (most frequent term and rare terms) and also helps to discriminate the documents within the collection. Having term frequency TF and inverse document frequency IDF, the term weight for query term i in document j using TF*IDF is the product of $TF_{i,j}$ and $IDF_{i,j}$ as follows:

$$TF*IDFij=TFij*IDFij \quad \ldots\ldots 3.3$$

$$TF*IDFij= TFij*\log_{10}\frac{N}{DFi} \quad \ldots\ldots 3.4$$

The computational value of TF*IDF is directly proportional to TF, but inversely proportional to document frequency (DF). That means, if the query term appears more frequently in a few documents, the value of TF*IDF will increase, in contrary, its value decreases if the term occurs too often in many documents in the corpus (Christopher D. et al., 2009). The **third phase** includes calculating the ***similarity*** (matching) between user query and indexed terms in documents and ***ranking*** them depending on the value of similarity. In VSM, the query vector $\vec{Q}$ and documents vector $\vec{D}j$ are denoted in vector space as:

$$\vec{Q}=(W1i,W2i,W3I,Wni) \quad \ldots\ldots 3.5$$

$$\overrightarrow{DJ}=(W1j,W2j,W3j,Wnj) \quad \ldots\ldots 3.6$$

The similarity of document D_j to user query Q is defined by the correlation between vectors $\vec{Q}$ and $\overrightarrow{DJ}$; which is the cosine measure between the two vectors. Cosine measure is the most common similarity measuring method used in most of IR systems. The similarity measurement is cosine angle θ between document Dj and query Q vectors as shown in figure 3.6 below.

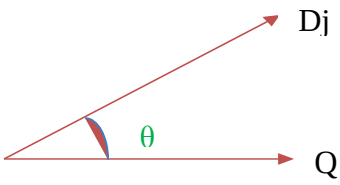

Figure 4. 8: Cosine similarity between document Dj and query Q

For given the query vector Q and document vector Dj, the cosine similarity measure is calculated as the internal product of the two vectors divided by the norms of them as given below.

$$(Dj,Q) = \frac{\vec{Dj}\vec{Q}}{|\vec{Dj}|\,|\vec{Q}|} \quad \text{.....3.7}$$

$$s(Dj,Q) = \frac{\sum_{i=1}^{n} wi,j \; x \; wi,q}{\sqrt{\sum_{i=1}^{n} wi,j} \;\; x \sqrt{\sum_{j=1}^{n} wi,q}} \quad \text{.....3.8}$$

Where $|\vec{Q}|$ and $|\vec{Dj}|$ are normalization of documents. The factor of $|\vec{Q}|$ does not affect the ranking because its value is equal for every document in the corpus (Beaza-Yates e. R., 2011). The value of Sim (Dj, Q) always falls between 0 and 1, we can limit the number of retrieved documents by assigning threshold value, so that documents having a similarity value greater than or equal to threshold value will be retrieved and returned to the user.

4.5. Ranking

Ranking the given document is the most important process in IR systems. The purpose of ranking is to calculates the similarity level for each document in the source with respects to user query. After we have ranked, then we have been sorting the retrieved documents in their similarity level. That sorted documents are ordered in their decreasing similarity mark, so that the maximum ranked documents are most likely similar and relevant to the user query (Beaza-Yates & Ribeiro-Neto, 2011). IR model defines document ranking by using either set theory, or vector or probability distribution theories. For the query term (T_i) and document (D_j), the ranking function $R(T_i, D_j)$ will assign the rank value for each retrieved document (D_j) with relation with user query term (T_i). In VSM, document ranking is based on the value of similarity measure calculated using cosine similarity. A document having the highest similarity value ranked as the first, the next as second and so on in the retrieval list. In that way, the retrieved documents are arranged in decreasing order based on their cosine similarity measure (Christopher D. Manning P. R., 2009).

CHAPTER FIVE

5. EXPERIMENT

5.1. INTRODUCTION

This research intention is designing an IR system for Sidama language to retrieve relevant documents for the user's query. This chapter reflects the implementation and evaluation of the designed Information System Prototype in previous chapter. The designed system has implemented in two subsections. The first subsection is indexing which prepare both document corpus and query preprocessing. The second subsection is document searching and ranking. It computes the similarity of documents in the corpus with the user query. We implemented the proposed system in the Python 3.1 programming language.

In order to evaluate the proposed system, 500 documents as corpus and 20 queries are prepared. All documents in the corpus read and categorized by the researcher and the language experts based on the identified user queries. The query-document relevance decision has done to compare it with the result of proposed system. See the Apendix-1 for Query-Document relevance judgement by language expert. The result of proposed system has presented and evaluation of the system measured by precision, recall and F-measures of 20 queries. The result analysis for each query outcome also made and discussed.

Finally, outcomes and confronted challenge during experimentation are reflected.

5.2. Document Preparation

5.2.1. Corpus Preparation

For corpus preparation we will be collect different data from different data sources Such as The Sidama language Holly Bible and Bakkalcho newspaper, Sidama Medium Network, Dayye FM program agency, Sidama Regional state Culture, Tourism and Sport bureau, Hawassa University Sidaamigna Department and Sidama Regional state Education Bureau (SRSEB). These articles include diverse topics like politics, education, culture, religion, history, social, health, economy and other events.

Table 5.1: Corpus Composition

Document Sources	Document Category	No of Documents
Sidaamu Afoo Holly Bible	Spiritual documents	50
Bakkalcho newspaper	Mixed documents	50
Sidama Medium Network	Mixed documents	62
Dayye FM program agency	Mixed documents	56
Sidama Regional state Culture and Tourism bureau	culture related documents	40
Ehiopia press agency/Sidama Afoo	Mixed documents	62
Hawassa University Sidaamigna Department	Educational Textbooks grade 1-8	60
Sidama Regional state Education Bureau	Education related documents	70
Sidama Regional state prosperity bureau	Politics related documents	50
Total		**500**

5.2.2. Query Preparation

Searching for information began by providing search terms which is denoted as user query. In addition to documents in a corpus, user queries should be prepared to test the performance and efficiency of the proposed system. In our case, we have prepared 20 user defined queries. The vector space model considers query as vector just like documents. All user queries pass through text pre-processing operations such as tokenization, normalization, stop word removal and stemming as like documents in the dataset. Terms in query are deliberately selected from the documents in the corpus by reading each of the documents carefully. For each of the query, relevant documents are identified from the corpus by language experts. This helps to compare the results obtained by experts with the implemented IR system.

Table 5.2: Selected User Queries

Roll No.	Selected Initial User Query	Query Identification Number (QID)
1.	Maganu qaalenna Amaale agadha	QID_1
2.	Sidaamu afii jirtena qoonqote qinaawo	QID_2
3.	Qalote Borro	QID_3
4.	Maganu seejjo agadha hegere heeshsho ragira	QID_4
5.	Sidaamu afii coyi fooliishsho	QID_5
6.	Rosu udiinne qixxeessa, dooranna woyyeessa	QID_6
7.	Sidaamu buninna wole giwirinnu latishshanna laalcho woyyeessa	QID_7
8.	Bushshu shaqqille woyyeessa	QID_8
9.	Sidaamu Qoqqowi Jireenyu Paarte	QID_9
10.	Kaphu ammano lubbo direyitanno yite rosiissannohu kaphu rosichooti	QID_10
11.	Saada	QID_11
12.	gorsu looso	QID_12
13.	Afuu lophonna borreessamme	QID_13
14.	Qoqqowu mootimma amaalete mine	QID_14
15.	Maganu Beetti Yesuusi	QID_15
16.	Kaphu Ammano Laaltanno Guma	QID_16
17.	Bushshunna wayi agarooshshe	QID_17
18.	Weesete Loosi Mixo	QID_18
19.	waasa, shaafeeta, buurisame	QID_19
20.	Imaananna budu uduunne	QID_20

To identify each query uniquely, we have assigned unique Query Identification Number to each of the queries as given in the table 5.2 above. As documents pass through text pre-processing operations, the queries all pass through a series of text processing functions.

5.2.3. Query expansion term selection

We identified expansion terms for each of twenty queries, together with available relevant documents in the corpus. Query expansion terms are synonymous terms derived from available documents in the corpus. Relevant documents for each expanded query identified by language experts by deep reading of all documents in the corpus, and are given in the table 5.3 below.

Table 5.3: Query Expansion Terms

Roll No.	Initial User Query	Additional terms for query expansion
QID_1	Maganu qaalenna Amaale agadha	Qullaawa Maxaafa, Yihowa seejjonna ayiirrisa
QID_2	Sidaamu afii jirtena qoonqote qinaawo	Borrotenna Borreessate amanyoote
QID_3	Qalote Borro	Xiinxallotenna dhaggete looso
QID_4	Maganu seejjo agadha hegere heeshsho ragira	Yesuusi Kiristoosi hajajonna qaale ayiirrisa
QID_5	Sidaamu afii coyi fooliishsho	Sidaamu Qaali Borrote amanyoote
QID_6	Rosu udiinne qixxeessa, dooranna woyyeessa	Maxaafa, rosu mine halashsha
QID_7	Sidaamu buninna wole giwirinnu latishshanna laalcho woyyeessa	Laalchonna baattote shiilo lossa
QID_8	Bushshu shaqqille woyyeessa	Baatto harishshanna loosa
QID_9	Sidaamu Qoqqowi Jireenyu Paarte	Sidaamu Gashooti Qoqqowu mootimmara lophote Paarte
QID_10	Kaphu ammano lubbo direyitanno yite rosiissannohu kaphu rosichooti	Wole Magano faarsiranna hexxo assira
QID_11	Saada	lalo meichonna gerecho cea
QID_12	gorsu looso	wayi jiro garunni horoonsiranna irshu latishsha halashsha
QID_13	Afuu lophonna borreessamme	Qaalu Rosicho lossa
QID_14	Qoqqowu mootimma amaalete mine	Qoqqowu gashshooti borrotenna hasaawu mine
QID_15	Maganu Beetti Yesuusi	Kiristoosi Qullaawu Kaaliiqi
QID_16	Kaphu Ammano Laaltanno Guma	Sheexaanu, kaphu duduwi HEGERE HEESHSHO hunanno
QID_17	Bushshunna wayi agarooshshe	Giwirinnu loosira baattote shiilonna laalchimma hattono xeenu wayi keeraanchimma
QID_18	Weesete Loosi Mixo	weesete Kaashonna Baatto harishsha
QID_19	waasa, shaafeeta, buurisame	Sidamaho Budu Sagale
QID_20	Imaananna budu uduunne	seemma, qolonna, gonfa

5.3. Text Processing

Aiming to increase retrieval efficiency in terms of storage space and retrieval time, text pre-processing both document corpus and query undertaken. Text pre-processing involves creation of index terms for increased retrieval performance.

All document and query pass through text pre-processing operations, which include: tokenization, normalization, stop-word elimination, stemming and index-term selection. These operations are language specific which require a text pre-processing algorithm designed for particular language morphology.

5.3.1. Tokenization (lexical Analysis) and Normalization

Tokenization (lexical Analysis) break down both document and query statements given into individual sequence of words termed as tokens. Tokenization converts provided Sidama texts (documents and queries) into a range of tokens which can be statistically computed and processed using white space between words. As we shown in figure 5.1 below the python code segment for tokenization, a python inbuilt function split() break down given text into individual tokens, which are ready for further text preprocessing operations.

```
def tokenize(filename):
    file=open("text.txt",encoding='utf-8')
    token = file.read().split()
return token
```

Figure 5.1: Code segment for tokenization

The normalization operation switches case folding of words. The Sidama writing system has-lower case and upper-case alphabets in its morphology. The text normalization function converts all uppercase letter words into common lowercases letters. The text normalization module also eliminates all the punctuation marks, special characters as well as numbers which have no discrimination power in a retrieval system, except single quotation mark (‘) which is considered as an alphabet for Sidama word formation. The

following figure 5.2 depicts code segment for normalization and punctuation mark exclusion.

```
def PunctuationMarkRemover(text):
  PM_List=['’’', '—', '~', '•', '≤', '□', '§', '¦', '"', '″', '·', '[', ']', '{', '}', '(', ')', '.', ',', '?', '; ', '!', '#',
            '$', '%', '^', '&', '*', '…', '‹‹', '››', '-', ',', '«', '»', '–', '÷', '›', '‹', '']
  for punc in text:
     if punc in PM_List:
          text=text. replace(punc, ' '). lower ()
          text= re.sub('[\d+]','', text)
 return text
```

Figure 5.2: Code Segment for Text Normalization and Punctuation Elimination

The function in Python code segment “PunctuationMarkRemover(text)” reads document as well as query and if any of the punctuation marks listed in PM_List appeared in text, the function replaces it with blank space. The resulting text after punctuation mark removal given to Python inbuilt function “lower()” which converts all uppercase letters into lowercase letters that controls word mismatch due to case folding.

5.3.2. Stop Word Removal

In IR point of view, there are a group of words that appear most frequently in the documents or queries. Sidama words such as ‘aana’, ‘ledo’, ‘calla’, ‘woy’, ‘dana’, ‘ikko’, etc are considered as stop words. Such words have no discriminating value for retrieval purposes so that they should be removed from documents and queries. The removal of stop words helps to boost the processing speed of the retrieval process reducing time taken for indexing thereby saving memory space. We applied the dictionary lookup method that employs predefined list of stop words in certain language categorized by language expertise. In our case, we have identified stop word lists in Sidama language and put in one file named “StopWords.txt”. The complete list of stop word we identified for our evaluation purpose is presented in appendix 2. The stop word removing function defined as ‘stopWordRemoval(text)’ in indexing subsection open and read stop word lists from StopWord.txt file and tokenize into individual words. The function checks whether stop words are there in the documents (text). The function removes all the stop words if appeared in the documents, and produces a non-stop word list which bears concepts. The

non-stop words go for stemming function. The figure 5.3 depicted below displays the code segment for stop word removal.

```
def  Remove_Stop_Words (word):
Stem_Word_list = [ ]
StopWord=open ("StopWord.txt", encoding='utf-8')
Read_StopWord = StopWord. read (). split () //Tokenization
for i in range (0, len(word)):
      if word[i] not in Read_StopWord:
          Stem_Word_list.append(word[i])
   return Stem_Word_lists
```

Figure 5.3: Code Segment for stop word Removal

5.3.3. Stemming

Stemming is a text preprocessing technique which used in natural language processing to reduce words to their base form. The stemming operation convert different morphological variations of the same base word to common stem. After the stop words have removed from both documents and query, then process goes to stemming function where affixes are reduced to root word form. Sidama language word formation uses infixes and suffixes only, only one prefixe "di-" found in Sidama Language, which is used for negative thought. For example, 'di-iteemmo' meaning 'I don't eat'. We applied suffix removal methods by selecting Sidama Language suffix list as shown in appendix-3. The following python code segment remove suffixes.

```
    def suffix(word):
          suffix_list=open("Suffix.txt",'r',encoding='utf-8')
          suffix= suffix_list.read().split ()
          for n in range (0, len(word)):
             stem_word =' '
             stem_word = stem_word + word[n]
             for suffix in range (0, len(suffix)-1):
               if(len(stem_word)>2):
                    if (stem_word. ends with(suffix[suffix])):
                       stem_word = stem_word.Replace(suffix[suffix], ' ')
```

```
                    suffix=len(suffix)
                    word[n] = stem_word
            suffix_list. Close ()
        return word
```

Figure 5.4: Code fragment for stemming word

5.4. Query Expansion

We applied manual query expansion that add one or more additional searching terms from user. The query expansion module asks user for additional information. The added query terms given to query expansion module, which append initial user query terms with the newly added terms forming expanded query. This will address the problem of word discrepancy and omission of documents due to short and inadequate query formation. The following code segment receive additional query terms from user and forms expanded query joining with the initial query.

```
expanded_query_list = [ ]
  additional_user_query=input("write additional query terms here for better searching: ")
  while len(additional_user_query) == 0:
    additional_user_query=input ("You did not write the additional query terms to search.
Please write additional query terms for better search!>>")
  additional_user_query=additional_user_query.lower()   #Query Normalization
  u_query = puc_remover(additional_user_query).split()  #Query Tokenization
  u_query= stop_word_removal(u_query)                   #Stop Word Removal
  for i in u_query:
    query = puc_remover(i)                      #Punctuation Removal
    query=query.split()
    query = sufix_prefix(query)                  #Query Stemming
    expanded_query=expanded_query_list.append(query)
    expanded_query=query_list+expanded_query_list
  print("Indexed Terms of Initial Query  :",query_list)
  print("Indexed Terms of Expanded Query are:",expanded_query)
```

Figure 5.5: Code segment for query expansion

5.5. Document Searching and Ranking

We applied cosine similarity measure to compare the expanded user query with the documents in the corpus. The similarity score value obtained ranked in descending order resulting documents with highest similarity value first, second, third, etc. the following Python code segment represents the cosine similarity measurement and ranking.

```
def cosine_similarity(query_text,doc_id):
    cos_similarity = 0
    normalized_query=query_TFIDF_Norm_fun(query_text)
    for norm_term in normalized_query:
        if norm_term in normalized_terms[doc_id]:
  cos_similarity+=normalized_query[norm_term]*normalized_terms[doc_id][norm_term]
    return cos_similarity


query_TFIDF_norm=query_TFIDF_Norm_fun(expanded_query)
    scores = {}
    for doc_id in docs:
        scores[doc_id] = cosine_similarity(expanded_query,doc_id)
        scores_list =[(doc_qu_sim,doc_id) for doc_id,doc_qu_sim in scores.items()]
    scores_list.sort() # it sort in increasing order or from smallest to largest
    scores_list.reverse() # it reverse the sorted result so that the highest result in first order
```

Figure 5.6: Document searching and ranking

5.6. Experimentations and System Testing

The proposed information retrieval system uses the vector space model with query expansion and without query expansion. The implementation of the proposed information retrieval system broken down in to two subsections: Indexing and searching with and without query expansion. The indexing subsection includes all the text preprocessing operations and produces an index term which are ready for similarity measurement. Last of all, the indexing part produces four central files such as: term frequency (tf), document frequency (df), posting and vocabulary files. The number of documents which contain terms(t) in corpus is said to be document frequency. File which stores each term with its

occurrence is said to be vocabulary file. The vocabulary file contains each single non-stop-word tokens in the documents with its document frequency and collection frequency. Both the document frequency and collection frequency are cross referenced to posting file. Posting file contains terms with their term occurrence, locations in the documents and each document name containing the term. Users 'information requirement is denoted as query. Consequently, component similarity measurement and ranking are done using TF*IDF measure in relation to the user query. The output of indexing part given to the searching part for matching with user query terms. The searching part receives middle file from indexing part and query statement from the user as inputs.

5.6.1. Experimentation 1: Retrieval Evaluation without Query Expansion

In the first experiment, we have evaluated the performance of the system before query expansion by using performance evaluations metrics such precision, recall and F-measure. Performance evaluations metrics are used to measure the degree of objectives stated to achieve by the system. In our research we have used 20 queries to test the Sidama language information retrieval system. We have evaluated the systems to check if the system meets the specified target or not. The objective of this research is to improve the performance of Sidama language information retrieval system using manual query expansion.

The searching result for the first query (QID_1) "**Maganu qaalenna Amaale agadha**" was in the figure 5.7 below. Among 500 documents in the corpus 25 documents were retrieved for the given query (QID_1). The cosine similarity measurement score between user query and document corpus falls always between 0 and 1. To limit the number of retrieved documents we have been using threshold value of cosine similarity score 0.5 as specified by (Xiaodan, Xiaohua, & Xiaohua, 2008). Retrieved documents with cosine similarity score greater than or equal to 0.5 (50%) are counted as relevant documents to user query. The retrieved documents having cosine similarity score less than 0.5 considered as less relevant to user query, so that they are not used in efficiency evaluation. According to the result of QID_1 as given in the figure 5.7 above total of 25 documents were retrieved of which 18 documents are relevant and 7 are non-relevant to the query. This manifests that a smaller number of documents are retrieved from 30 relevant documents available in the corpus resulting less recall value of 60.00%. That means, 12 relevant documents available in the corpus didn't retrieve due to search term mismatch. In this case the obtained precision was 72.00% and F-measure of 65.45%.

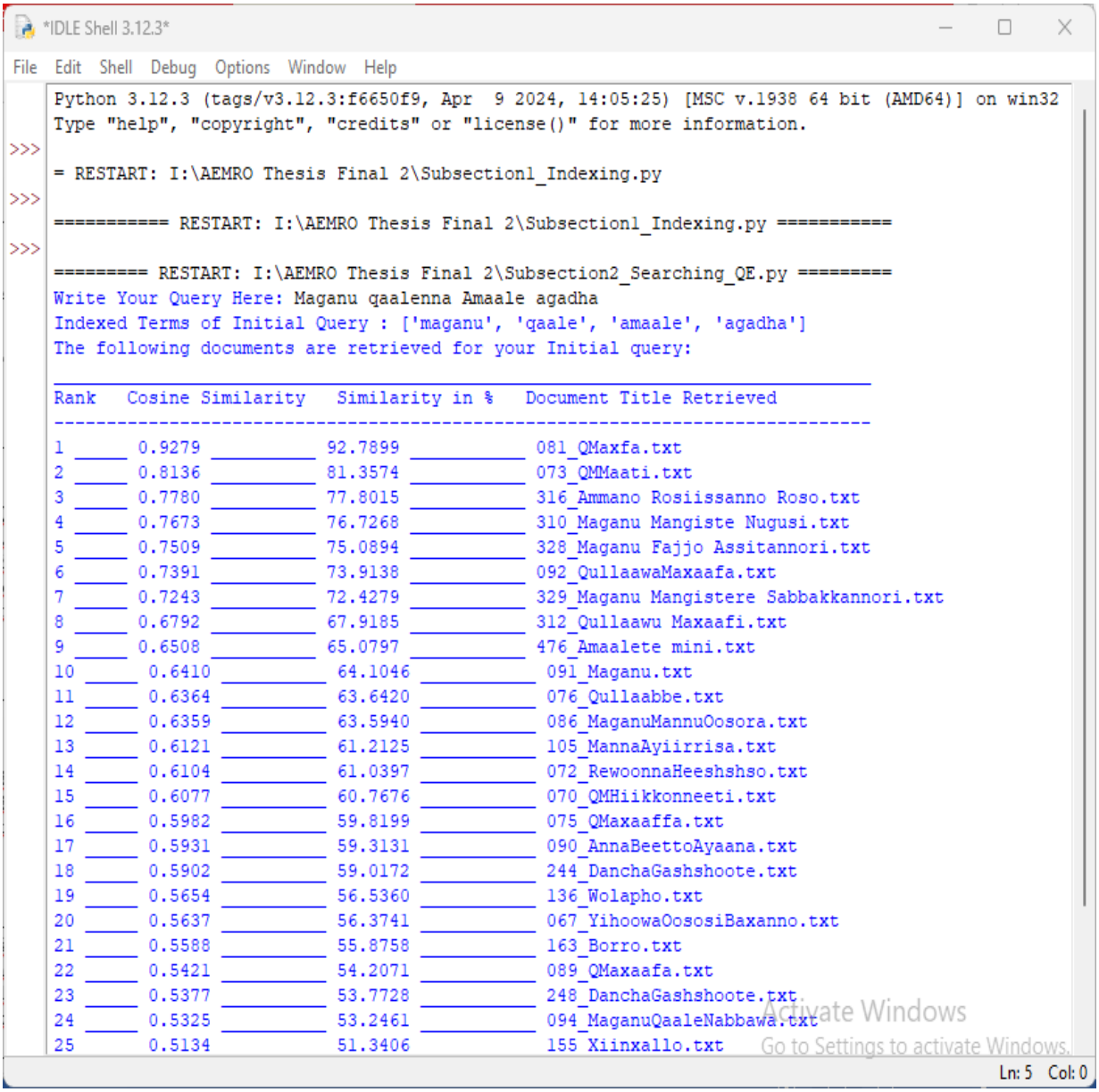

Figure 5.7: Search result for query QID_1 in first experimentation

Similarly for the second query (QID_2), total of 17 documents were retrieved from which 15 documents are relevant whereas 2 documents are non-relevant. Total of 29 relevant documents are there in the corpus according to experts' judgement. This shows that searching term mismatch with the terms in the documents become a serious problem. For this reason, 12 relevant documents didn't retrieve. In this case the resulting performance evaluation matrix's values were 88.24% precision, 51.72% recall and 65.22% F-measure.

The summarized precision, recall and F-measure values for each of the 20 Sidama *initial* queries are given in the table 5.4 below.

Table 5.4: Precision, recall and F-measure results of first experimentation

User Initial Query	Query Code	Relevant Documents in corpus	No. of Documents	No. of Relevant Documents Retrieved	Precision	Recall	F-Measure
Maganu qaalenna Amaale agadha	QID_1	30	25	18	72.00	60.00	65.45
Sidaamu afii jirtena qoonqote qinaawo	QID_2	29	17	15	88.24	51.72	65.22
Qalote Borro	QID_3	13	21	9	42.86	69.23	52.94
Maganu seejjo agadha hegere heeshsho ragira	QID_4	9	21	7	33.33	77.78	46.67
Sidaamu afii coyi fooliishsho	QID_5	29	32	23	71.88	79.31	75.41
Rosu udiinne qixxeessa, dooranna woyyeessa	QID_6	20	19	14	73.68	70.00	71.79
Sidaamu buninna wole giwirinnu latishshanna laalcho woyyeessa	QID_7	24	28	19	67.86	79.17	73.08
Bushshu shaqqille woyyeessa	QID_8	12	13	8	61.54	66.67	64.00
Sidaamu Qoqqowi Jireenyu Paarte	QID_9	18	16	14	87.50	77.78	82.35
Kaphu ammano lubbo direyitanno yite rosiissannohu kaphu rosichooti	QID_10	8	7	5	71.43	62.50	66.67
Saada	QID_11	8	7	1	14.29	12.50	13.33
gorsu looso	QID_12	25	30	21	70.00	84.00	76.36
Afuu lophonna borreessamme	QID_13	24	31	21	67.74	87.50	76.36
Qoqqowu mootimma amaalete mine	QID_14	25	32	22	68.75	88.00	77.19
Maganu Beetti Yesuusi	QID_15	32	**32**	23	71.88	71.88	71.88
Kaphu Ammano Laaltanno Guma	QID_16	15	10	8	80.00	53.33	64.00
Bushshunna wayi agarooshshe	QID_17	19	14	12	85.71	63.16	72.73
Weesete Loosi Mixo	QID_18	16	14	12	85.71	75.00	80.00
waasa, shaafeeta, buurisame	QID_19	11	7	7	100.00	63.64	77.78
Imaananna budu uduunne	QID_20	8	7	3	42.86	37.50	40.00
				Average	**67.86**	**66.53**	**65.66**

The performance evaluation of the proposed system implementation without query expansion was made in terms of precision, recall and F-measure. For each of the query, the search results are recorded as shown in the table above. The average precision value resulted was 67.86%. Likewise, the average recall value obtained was 66.53%. The overall mean value in F-measure was 65.66%. When we compare precision value with the recall value, decreased recall value was recorded, which shows that non-relevant documents were retrieved to user queries whereas multiple relevant documents remain unretrieved. In order to retrieve all the relevant documents to the query, additional query terms should be incorporated using query expansion technique. We applied query expansion in our second experimentation using manual query expansion technique as discussed below.

5.6.2. Experimentation 2: Retrieval Evaluation with Query Expansion

The second experimentation was done using manual query expansion technique. After producing the first retrieval results for the initial query, the system asks the user to feed additional searching terms for query expansion. The user once again provides with new words related with the information need of the user. The query expansion module of the system appends the newly added searching terms with the initial query terms and produce new sets of indexed terms called expanded query. The searching module of the system once again compare the expanded query with all the documents in the corpus and produce the ranked results of documents in cosine similarity score. The documents that have 50% and more similarity measure value are retrieved as relevant to the query. For QID_1, query expansion terms such as "**Qullaawa Maxaafa, Yihowa seejjonna ayiirrisa**" are added to the initial query terms such as "**Maganu qaalenna Amaale agadha**" producing new expanded query terms which are tokenized, normalized, stemmed and indexed like " **'maganu', 'qaale', 'amaale', 'agadha', 'qullaa', 'maxaafa', 'yiho', 'seejjo', 'ayiirri'** ". The searching result for QID_1 was given in the figure 5.8 below.

Figure 5.8: Search result for query QID_1 in second experimentation

As we can see in figure 5.8, in the second experimentation using query expansion for the query QID_1, total of 35 documents have been retrieved from which 30 documents are relevant and 5 documents are non-relevant. In this experimentation, increased precision, recall, and F-measure values were recorded like 85.71%, 100.00 and 92.31% respectively. We can consider that the query expansion-based searching for QID_1 changed in Precision value from 72.00% to 85.71%, recall value from 60.00% to 100.00% and F-

measure value from 65.45% to 92.31%. Better result was recorded for all performance evaluation matrixs in QID_1.

In the same way, for query QID_2, an increased performance values were recorded. The precision value increased from 88.24% to 90.00%, the recall value increased from 51.72% to 93.10% and similarly the F-measure value increased from 65.22% to 91.53%. The precision, recall and F-measure values for each of the expanded query has given in the table 5.5 below.

Table 5.5: Precision, recall and F-measure results of second experimentation

User Initial Query	**Additional terms for query expansion**	**Query Code**	**Relevant Documents in corpus**	**No. of Documents Retrieved**	**No. of Relevant Documents Retrieved**	**Precision (RR/Total Retrived)**	**Recall (RR/Total Relevant in Corpus)**	**F-Measure**
Maganu qaalenna Amaale agadha	Qullaawa Maxaafa, Yihowa seejjonna ayiirrisa	QID_1	30	35	30	85.71	100.00	92.31
Sidaamu afii jirtena qoonqote qinaawo	Borrotenna Borreessate amanyoote	QID_2	29	30	27	90.00	93.10	91.53
Qalote Borro	Xiinxallotenna dhaggete looso	QID_3	13	24	12	50.00	92.31	64.86
Maganu seejjo agadha hegere heeshsho ragira	Yesuusi Kiristoosi hajajonna qaale ayiirrisa	QID_4	9	28	9	32.14	100.00	48.65
Sidaamu afii coyi fooliishsho	Sidaamu Qaali Borrote amanyoote	QID_5	29	35	27	77.14	93.10	84.38
Rosu udiinne qixxeessa, dooranna woyyeessa	Maxaafa, rosu mine halashsha	QID_6	20	28	19	67.86	95.00	79.17
Sidaamu buninna wole giwirinnu latishshanna laalcho woyyeessa	Laalchonna baattote shiilo lossa	QID_7	24	34	24	70.59	100.00	82.76
Bushshu shaqqille woyyeessa	Baatto harishshanna loosa	QID_8	12	14	12	85.71	100.00	92.31
Sidaamu Qoqqowi Jireenyu Paarte	Sidaamu Gashooti Qoqqowu mootimmara lophote Paarte	QID_9	18	22	17	77.27	94.44	85.00

Kaphu ammano lubbo direyitanno yite rosiissannohu kaphu rosichooti	Wole Magano faarsiranna hexxo assira	QID_10	8	9	7	77.78	87.50	82.35
Saada	lalo meichonna gerecho cea	QID_11	8	11	7	63.64	87.50	73.68
gorsu looso	wayi jiro garunni horoonsiranna irshu latishsha halashsha	QID_12	25	36	25	69.44	100.00	81.97
Afuu lophonna borreessamme	Qaalu Rosicho lossa	QID_13	24	36	24	66.67	100.00	80.00
Qoqqowu mootimma amaalete mine	Qoqqowu gashshooti borrotenna hasaawu mine	QID_14	25	37	25	67.57	100.00	80.65
Maganu Beetti Yesuusi	Kiristoosi Qullaawu Kaaliiqi	QID_15	32	40	31	77.50	96.88	86.11
Kaphu Ammano Laaltanno Guma	Sheexaanu, kaphu duduwi HEGERE HEESHSHO hunanno	QID_16	15	16	14	87.50	93.33	90.32
Bushshunna wayi agarooshshe	Giwirinnu loosira baattote shiilonna laalchimma hattono xeenu wayi keeraanchimma	QID_17	19	22	18	81.82	94.74	87.80
Weesete Loosi Mixo	weesete Kaashonna Baatto harishsha	QID_18	16	17	15	88.24	93.75	90.91
waasa, shaafeeta, buurisame	Sidamaho Budu Sagale	QID_19	11	11	11	100.00	100.00	100.00
Imaananna budu uduunne	seemma, qolonna, gonfa	QID_20	8	8	8	100.00	100.00	100.00
					Average	**75.73**	**96.08**	**83.68**

As we see in the table 5.5 above, better precision, recall and F-measure values were recorded in almost all user expanded queries. The average precision value obtained was 75.73%, whereas the recall value was 96.08%. The average F-measure value was 83.68%. We can then realize that, the overall system performance after query expansion outweighed the efficiency of the system without query expansion.

5.7. Result Discussion and Challenges

In this study, we strive to design and develop manual query expansion-based IR for Sidama language. After we designed and implemented the proposed system, the effectiveness measurement and result discussion became essential.

In Information Retrieval Systems, four categories of documents can be found. They are Relevant Retrieved (RR), Relevant Not Retrieved (RNR), Non-Relevant Retrieved (NRR) and Non-Relevant Not Retrieved (NRNR). Considering these document sets; recall, precision and F-measure are considered as basic measures for retrieval effectiveness (Manish & Rahul, 2013). In this study we have applied these techniques to evaluate the performance effectiveness of our proposed system.

Precision measures the ratio of the number of relevant and retrieved documents. It calculates the number of relevant documents retrieved (RR) that are essential to the user and match his information need divided by the number of ***total retrieved documents*** by the system.

Recall measures the ratio of the number of retrieved and relevant (RR) documents to the number of total relevant documents in the corpus. The precision is inversely proportional to relative recall, meaning that, when precision increases recall decreases and vice versa. F-measure calculates weighed mean for both precision and recall giving a balanced measure for harmonic mean from both the values.

The performance evaluation of the proposed system was measured in two experimentations. First experimentation was done without query expansion and the second experimentation was done using manual query expansion. The precision, recall and F-measure values for each of user query are evaluated and presented above in table 5.4 and table 5.5 without query expansion and with query expansion respectively. The comparison of the two experimentations have given in the table 5.6 below.

In recall value measurements, increased values were obtained in query expansion-based searching.

Table 5.6: Comparison between the two experimentations

Query Code	Experimentation-1 (IR without QE)			Experimentation-2 (IR with QE)			Difference in Experimentation 1 and 2		
	Precision	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**
QID_1	72.00	60.00	**65.45**	85.71	100.00	**92.31**	13.71	40.00	26.85
QID_2	88.24	51.72	**65.22**	90.00	93.10	**91.53**	1.76	41.38	26.31
QID_3	42.86	69.23	**52.94**	50.00	92.31	**64.86**	7.14	23.08	11.92
QID_4	33.33	77.78	**46.67**	32.14	100.00	**48.65**	-1.19	22.22	1.98
QID_5	71.88	79.31	**75.41**	77.14	93.10	**84.38**	5.27	13.79	8.97
QID_6	73.68	70.00	**71.79**	67.86	95.00	**79.17**	-5.83	25.00	7.37
QID_7	67.86	79.17	**73.08**	70.59	100.00	**82.76**	2.73	20.83	9.68
QID_8	61.54	66.67	**64.00**	85.71	100.00	**92.31**	24.18	33.33	28.31
QID_9	87.50	77.78	**82.35**	77.27	94.44	**85.00**	-10.23	16.67	2.65
QID_10	71.43	62.50	**66.67**	77.78	87.50	**82.35**	6.35	25.00	15.69
QID_11	14.29	12.50	**13.33**	63.64	87.50	**73.68**	49.35	75.00	60.35
QID_12	70.00	84.00	**76.36**	69.44	100.00	**81.97**	-0.56	16.00	5.60
QID_13	67.74	87.50	**76.36**	66.67	100.00	**80.00**	-1.08	12.50	3.64
QID_14	68.75	88.00	**77.19**	67.57	100.00	**80.65**	-1.18	12.00	3.45
QID_15	71.88	71.88	**71.88**	75.61	96.88	**84.93**	3.73	25.00	13.06
QID_16	80.00	53.33	**64.00**	87.50	93.33	**90.32**	7.50	40.00	26.32
QID_17	85.71	63.16	**72.73**	81.82	94.74	**87.80**	-3.90	31.58	15.08
QID_18	85.71	75.00	**80.00**	88.24	93.75	**90.91**	2.52	18.75	10.91
QID_19	100.00	63.64	**77.78**	100.00	100.00	**100.00**	0.00	36.36	22.22
QID_20	42.86	37.50	**40.00**	100.00	100.00	**100.00**	57.14	62.50	60.00
Average	**67.86**	**66.53**	**65.66**	**75.73**	**96.08**	**83.68**	**7.87**	**29.55**	**18.02**

In precision difference (in 8th column in table 5.6) seven queries show negative values. This indicates that the precision and recall values are inversely proportional to each other. While the value of precision decrease, the value of recall increases (Peerzada, Abdul, & Bilal, 2016). Under "Difference in Experimentation 1 and 2" recall column, all recall values are positive and highly increased in the second experimentation. This indicates that, an expanded query retrieved more documents than non-expanded query. For query QID_4, the precision value has been decreased from 33.33% in first experiment to 32.14% in second experiment by 1.19%, contrary the recall value has been increased from 77.78% in first experiment to 100.00% in the second experiment by 22.22%.

On another side, highly increased recall result was obtained in QID_1 in triple measurements (precision, recall and F-measure). The searching with query expansion in QID_1 produced increased results by 85.71%, 100.00% and 92.31% respectively. The reason was that different terms that are highly distributed in the relevant documents have been given during query expansion than in the initial query formulation. Due to this reason 12 unretrieved relevant documents during searching without query expansion have been retrieved during searching with query expansion.

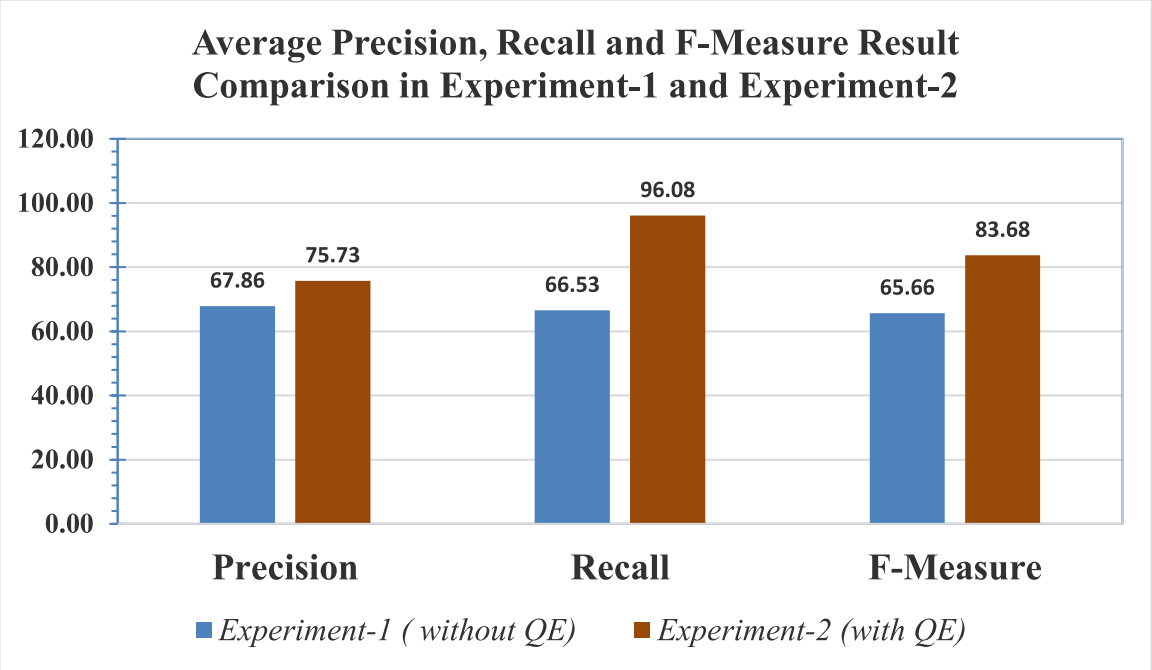

Figure 5. 9: Bar-chart comparison of the two experiments

In general speaking, as we can see in the comparison table 5.6 above, the efficiency of the manual query expansion-based searching outdoes the efficiency of searching without query expansion. It shows the average improvements in the three of measurements. F-measure improved by 18.02%, precision improved by 7.87% whereas recall improved by 29.55%. The greater improvement has been recorded in recall values which indicates that more relevant documents have been retrieved with less number non-relevant documents.

The major problem that affected the overall system result was wrong stemmed terms. For instance, the term “afii” was wrongly stemmed which was taken as it was in the indexed term lists even after stemming. We have used suffix trimming stemmer preparing list of suffixes in Sidama language. The correct stem form of inflectional word “afii” was “afoo”. The rule-based stemming resolves such kind of problems, which needs further researches. The absence of standardized data set and queries are also another challenges.

The thesaurus-based query expansion work done by (Senbeto K. D., 2019) in Sidama language. The researcher has done experimentations in two ways. These are baseline vector space model without the query expansion and using vector space model with thesaurus-based query expansion. The researcher has recorded Precision, recall and F-measure average values for each of the predefined 10 queries and 250 documents in the corpus. In the first experiment the researcher evaluated average performance results of the system without query expansion and obtained precision value of 55.13%, recall value of 77.91% and F-measure value of 63.19%. In the second experiment the researcher obtained the average performance results of the system with the thesaurus-based query expansion in terms of precision 58.18%, in terms of recall 90.99% and in terms of F-measure 69.67%. The experimentation results show that, the IR systems with thesaurus-based query expansion outperforms the standard VSM model by 6.48% F-measure.

Even though, the number of queries and documents varies in thesaurus-based query expansion searching of Senbeto's work with our study, improved results were recorded in precision, recall and F-measurements in manual query expansion-based searching. Significant improvements were observed in all precision, recall and F-measure values as shown in the table 5.7 below

Table 5. 7: Comparison of IR Researches done on Sidama Language

Performance Evaluation Metrix	**Average value of IR using Thesaurus QE**	**Average value of IR using Manual QE**	**Improvement difference**
Precision	58.18%	75.73%	17.55%
Recall	90.99%	96.08%	5.09%
F-measure	69.67%	83.68%	14.01%

We can generalize that the query expansion-based information retrieval produces better results for information needs of the users. Using long expanded and compound queries improve the precision and the recall values (Peerzada, Abdul, & Bilal, 2016).

CHAPTER SIX

6. CONCLUSION AND RECOMMENDATION

6.1. Conclusion

Information retrieval system becoming cutting edge research area, where information seekers now a days using in all aspects of their day to day lives. Information retrieval systems have different challenges such as, short user query expression, ambiguity nature of the natural language and the vocabulary mismatch between query terms and relevant documents. But good information retrieval system to some extent eases the above challenges to retrieve best results.

The main objective of this study is designing, implementing and improving the performance of Sidama language information retrieval system using manual query expansion techniques. In order to achieve our above-mentioned objective, we have collected and reviewed recent IR based literatures, studied Sidama language morphology, and language structures. In chapter 4, we have designed the proposed manual query expansion-based IR system using Vector Space Model (VSM) for Sidama language and in chapter 5 we have done the implementation of the designed system prototype.

We have implemented the designed prototype system in two subsections. These are indexing subsection, and query expansion, searching and ranking subsection. Indexing subsection involves the text preprocessing operation such as tokenization, normalization, stop word removal and stemming for both query and documents in corpus. The query expansion, searching and ranking subsection accepts user's information need (initial query), produce the first comparison result and once again accept additional terms from user for query expansion. Then after, the searching and ranking operation produces the ranked list of documents using expanded query.

In order to evaluate the performance of an implemented IR system, we have collected and organized 500 documents from different domains and categorized them into predefined 20 queries by language experts. The Query-Document relevance judgement by language expert is presented in appendix-1.

We have done two experimentations. First experimentation was done searching using initial user query (without query expansion). The second experiment has been done searching with expanded query (with query expansion). The performance measurements have been calculated using common efficiency measurement units such as precision, recall, and F-measure. For the first experimentation, we recorded the results for each of 20 initial queries. The outcome results were average values of 67.86% precision, 66.53% recall, and 65.66% F-measure. The second experimentation has been performed after manual query expansion and obtained results were 75.73% average precision, 96.08% average recall and 83.68% average F-measure values.

As we can see the average results of the two experimentations, a significant improvement has been seen in the second experimentation (manual query expansion-based). An average precision value increased by 7.87%, whereas an average recall value increased by 29.55%. Similarly, an average F-measure value increased by 18.02% in the second experimentation than the searching results in the first experimentation. Greater improvement has been seen in recall value, which indicates that almost all relevant documents in the corpus have been successfully retrieved during query expansion.

Finally, we can conclude that, the proposed manual query expansion-based searching results in greater improvement than searching without query expansion. The absence of rule-based stemmer was basic problem we faced during this study.

6.2. Recommendations

IR systems are hot research areas these days. In Sidama language Information Retrieval System and NLP are in infant stages. Researchers need to take steps in one more of the following research directions we have identified base on our study results. IR needs standard corpus for testing and making experimentation. But there is no standard corpus developed yet for Sidama language. This needs great focus as future work.

We have applied suffix removing stemming algorithm in our study, since Sidama language didn't have prefix except "di-". But this language does have complex infixes and suffixes. Our suffix removing algorithm produced wrong stems in some words, which degrade the system performance. Interested researcher can do further study in Sidama language rule-based stemmer that handles both infixes and suffixes.

For good IR systems development, standardized thesaurus construction, standard corpus preparation and domain specific queries formulations are crucial, which are still open for further study.

One can undertake additional study in Information retrieval system development using probabilistic model in Sidama language as well.

The development of automatic query expansion and interactive query expansion-based IR systems for Sidama language are also open for future studies.

Only little attempts have been done in Natural Language Processing (NLP) related works in Sidama language. For natural language development by technology supports, NLP based studies such as Sentiment Analysis, Text Classification, Machine Translation e.t.c. require further study.

REFERENCES

Abdelmgeid, A. -A. (2008). Using a query Expansion technique to improve document retrieval. *Vol.2*, 1-6.

Adamu, A. (2002). *Students' Attitude Towards Mother Tongue Instruction as a Correlate of Academic Achievement: the Case of Sidama.* Addis ababa: Addis Ababa University.

Alemayehu, N., & Willett, P. (2003). The effectiveness of stemming for information retrieval in Amharic. *Electronic library anaad Information Systems, 37*(4), 254-259.

Alemu, K. (2004). *Sidaamu Afii Borqaalla.* Hawassa, Sidama Zone, Southern Ethiopia: Unpublished.

Alhroob, A., Khafajeh, H., & Innab, N. (2013, July 7). Evaluation of Different Query Expansion Techniques for Arabic Text Retrieval System. *American Journal of Applied Sciences*, 1018-1024.

Amanuel, H. M. (2012). *Probabilistic Information Retrieval System For Amharic Language.* MSc Thesis, Addis Ababa University, School of Information Science, Addis Ababa.

Anbessa, T. (1987). Ballishsha Womens Speech of Avoidance Among the Sidama. *Journal of Ethiopian Studies, XX*.

Anbessa, T. (1987). Ballishsha: Women's Speech Among the Sidama. *Journal of Ethiopian Studies, 20*, 44-59.

Anbessa, T., & Tafesse, G.-M. (2019). A Linguistic Analysis o fPersonal Names in Sidama. *Journal of Afroasiatic Languages, History, and Culture, 8*(1).

Andargachew, M. (2009). *Automatic Thesaurus Construction for Amharic Text Retrieval.* Addis Ababa University, Department of Information Science. Addis Ababa: Addis Ababa University.

Arjun, B., & Sindhu, L. (2014, December). A Survey of Information Retrieval Models for Malayalam Language Processing. *International Journal of Computer Applications, 107*(14), 19-23.

Ashish, K. (2015). Query Expansion techniques. 1-5.

Asnakew, L. A. (2018). *A Relevance Feedback Approach For Amharic Text Retrieval.* MSc Thesis, Bahir Dar University, School of Research and Post Graduate Studies, Bahir Dar.

Atalay, L. (2014). *A Probabilistic Information Retrieval System For Tigrinya.* MSc Thesis, Addis Ababa University, School of Information Science, Addis Ababa.

Beaza-Yates, R., & Ribeiro-Neto, B. (2011). *Modern Information Retrieval: the concepts and technology behind search* (2nd ed.). England: Pearson Education Limited.

Bilal, A. A.-S. (2018). Applying Vector Space Model (VSM) Techniques in Information Retrieval for Arabic Language. *Computing Research Repository (CoRR)*, 1-54.

Biruduraju, A. (2011). Consolidated study on query expansion. 1-75.

Biruk, D. (2013). *Linguistically Motivated Amharic IR (LM-IR).* MSc Thesis, Addis Ababa University, Department of Information Science, Addis Ababa.

Bissan, A. (2012). Experiments on two Query Expansion Approaches for a Proximity-based Information Retrieval Model. 407-412.

Bruk, A., & Tilahun, T. (2015). Enhancing Amharic Information Retrieval System Based on Statistical Co-Occurrence Technique. *Journal of Computer and Communications*(3), 67-76.

Carpineto, C. a. (2012). A Survey of Automatic Query Expansion in Information Retrieval. 1-50.

Carpineto, C., & Romano, G. (2012, January). A Survey of Automatic Query Expansion in Information Retrieval. *ACM Computing Surveys, 44*, 1-50.

Christopher D. Manning, P. R. (2008). *Introduction to Information Retrieval.* Cambridge: Cambridge University Press.

Christopher D. Manning, P. R. (2009). *Introduction to information retrieva.* England: Cambridge University Press.

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

David, D. L., & Karen, S. J. (1996). Natural language processing for information retrieval. *Communications of the ACM*, 92-101.

Demewoz, B. (2013). *Automatic Thesaurus Construction From Wolayita Text.* Addis Ababa: Addis Ababa University.

Desta, L. L. (2020). *Development of Morphological Analyzer for.* Addis Ababa University. Addis Ababa, Ethiopia: Addis Ababa University.

Dinesh, M. (2014). *Grammar Rule Based Cross Lingual Inforation Retrieval for Telugu.* B.S. Abdur Rahman University, Computer Science. Vandalur, Chennai: B.S. Abdur Rahman Institute of Science and Technology.

Djoerd, H. (2009). Information Retrieval Models∗.

Dr Israa Burhanuddin, A. (2019, April 18). Morphology.

Fang, H. (2008, June). A Re-Examination of Query Expansion Using Lexical Resources. *proceedings of Association for Computetional Linguistics(ACL)*, 139-147.

Garald, K. (1999). *Information Retrieval Systems: Theory and Implementation.* (C. W. Bruce, Ed.) London, London, Boston, Dordrecht : Kluwer Academic Publishers.

Gezehagn, G. (2012). *Afaan Oromo Text Retrieval System.* MSc Thesis, Addis Ababa University, Department of Information Science, Addis Ababa, Ethiopia.

Haward, R., & W.Bruce, C. (1992). A Comparison of Text Retrieval Models. *The computer science Joyrnal, 35*(3), 279-290.

Hiemstra, D. (2001). *Using Language Models for Information Retrieval.* City University. The Netherlands: Neslia Paniculata.

Hiteshwar, K. (2019, January 24). Query Expansion Techniques for Information Retrieval: a Survey.

Howard, R. T., & W. Bruce, C. (1992). A Comparison of Text Retrieval Models. *The Computer Science Journal, 35*(3), 279-290.

Indrias, X., Sileshi, W., Yohannis, L., & Desalegn, G. (2007). *Sidama-amharic-English Dictionary.* (S. G. (MA), Ed.) Addis Ababa, Ethiopia: Master Printing Press.

In-Ho, K., & GilChang, K. (2003). Query Type Classification for Web Document Retrieval. *SIGIR, 3*, 64-71.

Jitendra, N. S., & Sanjay, K. D. (2012). Analysis of Vector Space Model in Information Retrieval. *National Conference on Communication Technologies & its impact on Next Generation Computing CTNGC* (pp. 14-18). Lucknow: International Journal of Computer Applications® (IJCA).

Jitendra, N. S., & Sanjay, K. D. (2013). A Comparative Study on Approaches of Vector Space Model in Information Retrieval. *International Journal of Computer Applications*, 37-40.

Johanna, L. d. (1976). *The History of the International Feeration of Library Associations From Its Creation to the Second World War 1927-1940.* Loughborough University of Technology, Department of Library and Information Studies. Leiden, Netherlands: Loughborough University.

jw.org. (2018, April 1). *Yihowa Farci'raasine*. (Jehovah's Witnesses) Retrieved November 19, 2018, from Official Website of Jehovah's Witnesses: https://www.jw.org/sid/

K. Pradeep Reddy1, T. R., & Vardhan4. (2018). Impact of Similarity Measures in Information Retrieval. *08*.

K.Ezhilarasi, G. K. (2018). *Literature Survey: Analysis on Semantic Web Information Retrieval Methodologies* (Vol. 142). Anna , India: Anna University.

Kawachi, K. (2007). *A Grammar of Sidama, A Cushitic Language Of Ethiopia.* University at Buffalo, Department of Linguistics. Buffalo: ProQuest Information and Learning Company.

Kehinde, D. A., Dipo, T. A., & Babajide, A. (2016, January 13). A Full Text Retrieval System in a Digital Library Environment. *Intelligent Information Management*(8), 1-8.

Khadim, D., Fleur, M., & Gayo, D. (2014). Query expansion using external resources for improving information retrieval in the biomedical domain. *3*, 189-194.

Kiran, P. B. (2016). Information Retrieval: search process, techniques and strategies. *International Journal of Next Generation Library and Technologies (IJNGLT), 2*(1), 1-10.

Kotaro, N., Takahiro, H., & Shojiro, N. (2007). A Thesaurus Construction Method from Large Scale Web Dictionaries. *IEEE International Conference on Advanced Information Networking and Application(AINA)*, 1-5.

Kunpeng, Z., Xiaolong, W., & Yuanchao, L. (2007). A new query expansion method based on query logs mining. *International Journal on Asian Language Processing, 1*(19), 1-12.

Liadh, K., Lorraine, G., Hanna, S., Tobias, S., Gondy, L., Danielle, L. M., . . . Joao, P. (2014). Overview of the ShARe/CLEF eHealth Evaluation Lab 2014. *Springer* (pp. 172-191). Switzerland: Springer International Publishing.

Loay, E. G., & Ibrahem, A. H. (2013). Text-Based Information Retrieval System using Non-Linear Matching criteria. *International Journal of Advancements in Research & Technology, 2*(6), 172-176.

Manish, S., & Rahul, P. (2013). A Survey on Information Retrieval Models, Techniques And Applications. *International Journal of Emerging Technology and Advanced Engineering, 3*(11), 542-545.

Manwar, A., Hemant, S. M., Chinchkhede, D. K., & Vinay, C. D. (2012, May). A Vector Space Model For Information Retrieval: A Matlab Approach. *Indian Journal of Computer Science and Engineering (IJCSE), III*, 222-229.

Mekonen, M. ,. (2022). *Context-Based Spell Checker For Sidama Afoo.* Hawassa: Hawassa University.

NALRC. (2018, August 1). *National African Language Resource Center*. (S. Antonia, Producer, & Indian University, SCHOOL OF GLOBAL AND INTERNATIONAL STUDIES) Retrieved December 31, 2018, from NALRC website: https://nalrc.indiana.edu/doc/brochures/Sidamo.pdf

Peerzada, M. I., Abdul, M. B., & Bilal, A. B. (2016). Precision, Recall and F-Measure of Select Search Engines: Retrieval of Research Article in the Field of Library and Information Science. *International Journal of Information Studies & Libraries, 1*(2), 1-10.

Pradeep, R. K., Raghunadha, R. T., Apparao, N. G., & Vishnu, V. B. (2018). *Impact of Similarity Measures in Information Retrieval.*

Ramakrishna, K., B. Padmaja, R., & Swapna, N. (2013, October). Pseudo Relevance Feedback by linking WordNet for Expanding Queries in Information Retrieval Process. *International Journal of Modeling and Optimization, 3*(5), 462-467.

Rivas, A. R., Iglesias, E. L., & Diz, M. L. (2014, March 2). Study of query expansion techniques and their application in the biomedical information retrieval. *The Scientific World Journal*, 1-9.

Roshdiand, A., & Roohparvar, A. (2015). Review: Information Retrieval Techniques and Applications. *International Journal of Computer Networks and Communications Security, 3*(9), 373-377.

Ruban, Swathi, A., & Misrina, M. (2017). A Study on Query Suggestion and Expansion in Information Retrieval. *International Journal of Latest Trends in Engineering and Technology*, 162-164.

Sakthi Murugan, R. P., & Aghila, D. G. (2013, October). Ontology Based Information Retrieval - An Analysis. *International Journal of Advanced Research in Computer Science and Software Engineering, III*(10), 486-493.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic retrieval. *Information Processing and Management*, 513-523.

Samia, Z. (2007). summary and statical report of the 2007 population and housing census.

Samrawit, Z. (2014). *Word Sense Disambiguation using Semantic Similarity for Query Expansion in Amharic Information Retrieval.* MSc Thesis, Addis Ababa University, Department of Information Science, Addis Ababa.

Sara, A., Mohammed, D., & Mahmoud, K. (2016, October). A Novel Information Retrieval Approach using Query Expansion and Spectral-based. *(IJACSA) International Journal of Advanced Computer Science and Applications, 7*(9), 364-373.

Senbeto, K. (2019). *Thesaurus-Based Query Expansion For Designing Sidama informational retrieval.* Bahir Dar, Ethiopia: Bahir Dar, University.

Shweta J.patil, D. K. (2017, April). IMPROVING INFORMATION RETRIEVAL FROM PDF DOCUMENTS.

Shweta, P. J., & Dilip, B. K. (2017). Efficient Information Retrieval Using Indexing. *nternational Journal of Computer Science and Network, 6*(2), 106-109.

Steven, B., Ewan, K., & and Edward, L. (2009). *Natural Language Processing with Python* (1st ed.). United States of America: O'Reilly Media, Inc.

Stokes, N., & Lawrence Cavedon, J. Z. (2008, October 29). Exploring criteria for successful query expansion in the genomic domain. *Springer Inf Retrieval* , 17-50.

Tesfa, K. H. (2013). *Word Sense Disambiguation for Afaan Oromo language.* Addis Ababa University, Computer Science. Addis Ababa: Unpublished.

Tewodiros, A. C. (2014). *Applying Thesaurus Based Semantic Compression for Improving the Performance of Amharic Text Retrieval.* MSc Thesis, University of Gondar, Department of Information Technology, Gondar.

Tewodros, H. (2003). *Amharic Text Retrieval: An Experiment Using Latent Semantic Indexing (LSI) With Singular Value Decomposition (SVD).* MSc Thesis, Addis Ababa University, Information Science, Addis Ababa.

Thompson, P. (2007, October 30). Looking back: On relevance, probabilistic indexing. *ELSEVIER*, 1-8.

Tilahun Yeshambel1, J. M. (2020). Amharic Document Representation for Adhoc Retrieval. 1-11.

Tsadu, Z. (2017). *Query Expansion for Tigrigna Information Retrieval.* Addis Ababa: Addis Ababa University.

Tuan, T. A. (2014, March). A Hybrid Approach of Query Expansion for Vietnamese Query. *International Journal of Computer Science and Mobile Computing, 3*(3), 623-631.

Turid, H. (2003). *Dictionary-Based Cross-Language Information Retrieval; Principles, System Design and Evaluation.* University of Tempere, Department of Information Studies. Finland: Bookshop TAJU.

Vera, H., Jaap, K., Christof, M., & Maarten, D. R. (2004). Monolingual Document Retrieval for European Languages. *Language & Inference Technology Group, ILLC, 7*, 33-52.

Welde, J. (2014). *Ontology Based Query Expansion for Enhancing the Performance of Amharic Information Retrieval: The Case of Tourism Sector.* Addis Ababa University, Department of Information Science. Addis Ababa: Unpublished.

Wesley, J. C. (2009). *Programming, Core Python.* Mexico City: Pearson Education, Inc.

William, H., Susan, P., & Larry, D. (2000). Assessing Thesaurus-Based Query Expansion Using the UMLS Metathesaurus. *Division of Medical Informatics & Outcomes Research*, 344-348.

William, R. L. (2016). Languages of the world (August 2016 draft for Oxford Research Encyclopedia of Linguistics). *Preliminary draft for Oxford Research Encyclopedia of Linguistics*, 1-27.

Wolf, F. (2013). *Linguistically Motivated Ontology-Based Information Retrieval.* University of Augsburg, Department of Computer Science. Germany: Augsburg.

Xiaodan, Z., Xiaohua, H., & Xiaohua, Z. (2008). A Comparative Evaluation of Different Link Types on Enhancing Document Clustering. *Information Search and Retrieval*, 1-9.

Yegosh, G., Ashish, S., & Sexena, A. (2013). A Survey on Important Aspects of Information Retrieval. *Journal of Engineering and Technology*, *4*(2), 49-72.

Zhihan, L. (2009). *Improvement to Chinese Information Retrieval by Incorporating Word Segmentation and Query Expansion.* Queensland University of Technology, Faculty of Science and Technology.

Appendices

Appendix 1: Query-Document relevance judgement by language expert

Query ID	Initial User Query	Codes of Relevant Documents to the query in the corpus	No. of Relevant Documents to the query
QID_1	Maganu qaalenna Amaale agadha	067, 069, 070, 071, 072, 073, 074, 075, 076, 081, 082, 086, 089, 090, 092, 093, 094, 105, 136, 163, 165, 222, 230, 232, 243, 310, 308, 326, 328, 329	30
QID_2	Sidaamu afii jirtena qoonqote qinaawo	003, 007, 034, 037, 040, 041, 048, 049, 050, 055, 127, 131, 151, 163, 164, 165, 166, 167, 171, 173, 175, 182, 184, 188, 194, 199, 208, 209, 284	29
QID_3	Qalote Borro	004, 011, 022, 058, 062, 064, 072, 172, 173, 180, 196, 200, 202	13
QID_4	Maganu seejjo agadha hegere heeshsho ragira	067, 068, 072, 074, 093, 230, 260, 324, 330	9
QID_5	Sidaamu afii coyi fooliishsho	001, 039, 053, 066, 095, 098, 099, 131, 164, 194, 160, 161, 162, 163, 165, 166, 167, 173, 175, 186, 188, 189, 203, 208, 210, 215, 323, 328, 431	29
QID_6	Rosu udiinne qixxeessa, dooranna woyyeessa	494, 487, 464, 289, 192, 158, 288, 488, 007, 008, 183, 493, 030, 081, 018, 304, 464, 442, 148, 097	20
QID_7	Sidaamu buninna wole giwirinnu latishshanna laalcho woyyeessa	133, 153, 220, 252, 281, 282, 333, 338, 343, 345, 346, 370, 372, 373, 374, 380, 386, 406, 407, 408, 409, 410, 416, 456	24
QID_8	Bushshu shaqqille woyyeessa	272, 276, 284, 340, 347, 376, 381, 394, 395, 406, 424, 426	12
QID_9	Sidaamu Qoqqowi Jireenyu Paarte	004, 084, 119, 120, 172, 176, 196, 243, 253, 271, 287, 301, 302, 321, 409, 470, 474, 497	18
QID_10	Kaphu ammano lubbo direyitanno yite rosiissannohu kaphu rosichooti	068, 072, 078, 089, 232, 317, 318, 319	8
QID_11	Saada	003, 023, 029, 424, 425, 436, 446 247	8
QID_12	gorsu looso	254, 262, 264, 272, 275, 276, 277, 281, 283, 294, 295, 298, 322, 333, 334, 339, 345, 350, 354, 358, 361, 381, 382, 389, 451	25
QID_13	Afuu lophonna borreessamme	010, 011, 012, 030, 031, 059, 060, 061, 062, 063, 065, 066, 125 151, 171, 185, 164, 214, 215, 200, 207, 238, 202, 460	24

QID_14	Qoqqowu mootimma amaalete mine	053, 100, 163, 165, 216, 217, 222, 242, 253, 286, 289, 302, 303, 307, 321, 346, 452, 465, 466, 474, 476, 481, 487, 488, 494	25
QID_15	Maganu Beetti Yesuusi	068, 070, 071, 072, 074, 075, 076, 077, 079, 082, 083, 084, 085, 087, 089, 090, 091, 092, 093, 094, 138, 256, 258, 260, 307, 310, 311, 312, 316, 325, 326, 329	32
QID_16	Kaphu Ammano Laaltanno Guma	072, 086, 069, 222, 313, 317, 318, 320, 321, 322, 324, 326, 327, 328, 340	15
QID_17	Bushshunna wayi agarooshshe	110, 298, 333, 334, 347, 380, 386, 388, 390, 394, 395, 405, 406, 407, 410, 413, 451, 452, 453	19
QID_18	Weesete Loosi Mixo	229, 253, 259, 271, 274, 276, 280, 281, 293, 336, 353, 358, 370, 371, 385, 478	16
QID_19	waasa, shaafeeta, buurisame	100, 113, 119, 120, 129, 130, 132, 140, 224, 234, 237	11
QID_20	Imaananna budu uduunne	032, 119, 120, 133, 149, 228, 279, 367	8

Appendix 2: Sidama Language stop word Lists

Aada	Addi	Aana	Aananni	Aacha	Aanchine
Aane	Agari	Agarina	Aja	Bire	Birqiiqa
Biishsha	Bitima	Bura	Busule	Buuxa	Buqisa
Calla	Ajisha	Daafira	Daafo	Dana	Dancha
Dino	Duucha	Duuchanka	Duuchuri	Duumba	Akata
Duumbara	Duume	Duke	Dunka	Duumo	Duura
Eenbara	Eega	Eewo	Akkala	Eweli	Ewelo
Fafo	Fakana	Fakkaninno	Fakkanu	Ga’a	Gede
Gedensaanni	Akeeka	Gedena	Geeshsha	Giddo	Giddooni
Gobba	Gowwa	Haaro	Hakkiicho	Akkimale	Hakkiinni
Hakko	Hala’lado	Harancho	Hasiissano	Hatti	Hatto
Hattooha	Hattoonni	Aliba	Aliba	Hattono	Hattoote
Hattori	Hooga	Hoogoro	Hoogo	Ikka	Ikke
Ikkona	Albaanni	Ikkino	Ikkinohura	Ikkolanakayi	Ikkiro
Ikkirono	Ikko	Ikkoomero	Insa	Ise	Ane
Isera	Isi	Jawa	Jooga	Kaajjado	Kae
Kalaa	Kayi	Kayinni	Ani	Kayi	Kayinni
Ani	Kayinnilla	Kayisse	kee’mado	Kolishsho	Konne

Kuni	Kuula	Kuulaamo	Kuulaame	Anje	Lase
Lede	Ledo	Lowo	Ma	Maa	Mamoote
Mayinni	Mayira	Assa	Mereero	Mito	Mule
Mullicho	Nafa	No	Noo	Noohu	Nooti
Asse	Noola	Qacce	Qaccera	Qaccete	Qacceni
Qara	Qaramo	Qarame	Qariweelo	Assi	Qolcha
Qolchama	Qolchancho	Qolchi	Qolchitu	Qole	Qora
Ra'ado	Sada	Ate	Sae	Sayise	Seeda
Seesa	Shaqado	Shaqa	Shiima	Sona	Taaga
Ati	Tene	Tini	Togo	Togoha	Togoni
Tuqa	Tuqisa	Tuqisi	Tuqiu	Baabba	Tuuqa
Urga	Widoonni	Widira	Wo'ma	Wo'manka	Wo'munkuri
Wo'mare	Baca	Wole	Wolewa	Wolere	Wolu
Wona	Woroonni	Wote	Woyi	Woluno	Badhe
Xa	Xaate	Xea	Xeisa	Baala	Baalanka
Baalunkura	Balaxa	Bashsho	Badhenni	Batiyne	Batira
Batiro	Batisa	Bero			

Appendix 3: Sidama language suffix Lists

Aati	Anchi	Ancho	Anka	Anno
Anote	Atto	Ba	Be	Cho
Eelo	Eemmo	Eete	Ete	Ge
Gge	Gge	Gge	gge	Gge
Ha	Hena	Ho	Hu	I
idi	Idhu	Innani	ineemmo	Ini
inoommo	Ino	inooni	i'nummo	Inummo
iri	isi	isiisi	isu	Itini
Itino	itinoonni	itta	itto	Ittokki
itu	kka	kki	mma	Maancho
mme	mmete	mmo	na	Nna
ni	nni	nitu	nke	Nku

nno	no	noommo	nnota	Nnotto
nsa	nte	nummo	omero	Onni
oni	oomma	oommo	oonke	Ootta
ootto	oti	ra	re	Ro
sa	se	sha	ta	Tano
te	tenni	tinanni	tini	Tino
tinoonni	to	tto	u	uba
ullu	Umma	ummo	unku	Wa
We	wiinni	'ya		

Appendix 4. Performance evaluation Metrix

Query ID	Initial User Query	Additional terms for query expansion	Codes of Relevant Documents to the query in the corpus	No. of Relevant Documents to the query	No. of Retrieved Documents without QE		No. of Retrieved Documents with QE	
					Total Retrieved	Relevant Retrieved	Total Retrieved	Relevant Retrieved
QID_1	Maganu qaalenna Amaale agadha	Qullaawa Maxaafa, Yihowa seejjonna ayiirrisa	067, 069, 070, 071, 072, 073, 074, 075, 076, 081, 082, 086, 089, 090, 092, 093, 094, 105, 136, 163, 165, 222, 230, 232, 243, 310, 308, 326, 328, 329	30	25	**18** [067, 070, 072, 073, 074, 075, 076, 081, 086, 089, 090, 091, 092, 094, 136, 310, 328, 329]	35	**30** [067, 069, 070, 071, 072, 073, 074, 075, 076, 081, 082, 086, 089, 090, 092, 093, 094, 105, 136, 163, 165, 222, 230, 232, 243, 310, 308, 326, 328, 329]
QID_2	Sidaamu afii jirtena qoonqote qinaawo	Borrotenna Borreessate amanyoote	003, 007, 034, 037, 040, 041, 048, 049, 050, 055, 127, 131, 151, 163, 164, 165, 166, 167, 171, 173, 175, 182, 184, 188, 194, 199, **208**, 209, 284	29	17	**15** [003, 007, 037, 040, 048, 049, 050, 055, 131, 151, 173, 175, 184, 188, 284]	30	**27** [003, 007, 034, 037, 040, 041, 048, 049, 050, 055, 127, 131, 151, 163, 164, 165, 166, 167, 171, 173, 175, 182, 184, 188, 194, 199, 284]
QID_3	Qalote Borro	Xiinxallotenna dhaggete looso	004, 011, 022, 058, 062, 064, **072**, 172, 173, 180, 196, 200, 202	13	21	**9** [004, 011, 058, 062, 064, 172, 180, 196, 202]	24	**12** [004, 011, 022, 058, 062, 064, 172, 173, 180, 196, 200, 202]

QID_4	Maganu seejjo agadha hegere heeshsho ragira	Yesuusi Kiristoosi hajajonna qaale ayiirrisa	067, 068, 072, 074, 093, 230, 260, 324, 330	9	21	**7** [067, 072, 093, 230, 260, 324, 330,]	28	**9** [067, 068, 072, 074, 093, 230, 260, 324, 330]
QID_5	Sidaamu afii coyi fooliishsho	Sidaamu Qaali Borrote amanyoote	001, 039, 053, **066**, 095, 098, 099, 131, **164**, 194, 160, 161, 162, 163, 165, 166, 167, 173, 175, 186, 188, 189, 203, 208, 210, 215, 323, 328, 431	29	32	**23** [001, 095, 039, 131, 164, 208, 194, 203, 215, 210, 163, 175, 165, 186, 188, 160, 166, 323, 189, 162, 167, 173, 160]	35	**27** [001, 039, 053, 095, 098, 099, 131, 194, 160, 161, 162, 163, 165, 166, 167, 173, 175, 186, 188, 189, 203, 208, 210, 215, 323, 328, 431]
QID_6	Rosu udiinne qixxeessa, dooranna woyyeessa	Maxaafa, rosu mine halashsha	494, 487, 464, 289, 192, 158, 288, **488**, 007, 008, 183, 493, 030, 081, 018, 304, 464, 442, 148, 097	20	19	**14** [494, 487, 464, 289, 158, 008, 183, 493, 030, 007, 018, 464, 148, 097]	28	**19** [494, 487, 464, 289, 192, 158, 288, 304, 007, 008, 183, 493, 030, 081, 018, 464, 442, 148, 097]
QID_7	Sidaamu buninna wole giwirinnu latishshan na laalcho woyyeessa	Laalchonna baattote shiilo lossa	133, 153, 220, 252, 281, 282, 333, 338, 343, 345, 346, 370, 372, 373, 374, 380, 386, 406, 407, 408, 409, 410, 416, 456	24	28	**19** [153, 220, 252, 282, 333, 338, 343, 345, 346, 373, 374, 380, 406, 407, 408, 409, 410, 416, 456]	34	**24** [133, 153, 220, 252, 281, 282, 333, 338, 343, 345, 346, 370, 372, 373, 374, 380, 386, 406, 407, 408, 409, 410, 416, 456]
QID_8	Bushshu shaqqille woyyeessa	Baatto harishshanna loosa	272, 276, 284, 340, 347, 376, 381, 394, 395, 406, 424, 426	12	13	**8** [284, 340, 347, 381, 395, 406, 424, 426]	14	**12** [272, 276, 284, 340, 347, 376, 381, 394, 395, 406, 424, 426]
QID_9	Sidaamu Qoqqowi Jireenyu Paarte	Sidaamu Gashooti Qoqqowu mootimmara lophote Paarte	004, 084, 119, 120, 172, 176, 196, 243, 253, 271, 287, 301, 302, 321, 409, 470, 474, **497**	18	16	**14** [004,084, 120, 172, 176, 196, 253, 271, 287, 301, 302, 321, 409, 474]	22	**17** [004, 084, 119, 120, 172, 176, 196, 243, 253, 271, 287, 301, 302, 321, 409, 470, 474]
QID_10	Kaphu ammano lubbo	Wole Magano faarsiranna hexxo assira	068, 072, 078, 089, 232, **317**, 318, 319	8	7	**5**	9	**7**

	direyitanno yite rosiissannohu kaphu rosichooti					[068,072,232, 318, 319]		[068, 072, 078, 089, 232, 318, 319]
QID_11	Saada	lalo meichonna gerecho cea	003, 023, 029, 424, 425, 436, 446 247	8	7	1 [436]	11	7 [003, 023, 029, 424, 425, 436, 446]
QID_12	gorsu looso	wayi jiro garunni horoonsiranna irshu latishsha halashsha	254, 262, 264, 272, 275, 276, 277, 281, 283, 294, 295, 298, 322, 333, 334, 339, 345, 350, 354, 358, 361, 381, 382, 389, 451	25	30	**21** [254, 262, 264, 272, 275, 276, 277, 281, 283, 294, 295, 298, 322, 333, 334, 339, 345, 350, 354, 358, 361]	36	**25** [254, 262, 264, 272, 275, 276, 277, 281, 283, 294, 295, 298, 322, 333, 334, 339, 345, 350, 354, 358, 361, 381,382, 389, 451]
QID_13	Afuu lophonna borreessamme	Qaalu Rosicho lossa	010, 011, 012, 030, 031, 059, 060, 061, 062, 063, 065, 066, 125 151, 171, 185, 164, 214, 215, 200, 207, 238, 202, 460	24	31	**21** [010, 011, 012, 030, 031, 059, 060, 061, 062, 063, 065, 066, 125 151, 171, 185, 164, 214, 215, 200, 207]	36	**24** [010, 011, 012, 030, 031, 059, 060, 061, 062, 063, 065, 066, 125, 151, 171, 185, 164, 214, 215, 200, 207, 238, 202, 460]
QID_14	Qoqqowu mootimma amaalete mine	Qoqqowu gashshooti borrotenna hasaawu mine	053, 100, 163, 165, 216, 217, 222, 242, 253, 286, 289, 302, 303, 307, 321, 346, 452, 465, 466, 474, 476, 481, 487, 488, 494	25	32	**22** [053, 100, 163, 165, 216, 217, 222, 253, 286, 289, 302, 303, 307, 346, 452, 465, 474, 476, 481, 487, 488, 494]	37	**25** [053, 100, 163, 165, 216, 217, 222, 242, 253, 286, 289, 302, 303, 307, 321, 346, 452, 465, 466, 474, 476, 481, 487, 488, 494]
QID_15	**Maganu Beetti Yesuusi**	Kiristoosi Qullaawu Kaaliiqi	068, 070, 071, 072, 074, 075, 076, 077, 079, 082, 083, 084, 085, 087, 089, 090, 091, 092, 093, 094, 138, 256, 258, 260, **307**, 310, 311,	32	**32**	**23** **[**071, 074, 076, 077, 079, 082, 083, 085, 089, 090, 091, 092, 093, 094, 138, 256,	**41**	**31** **[**068, 070, 071, 072, 074, 075, 076, 077, 079, 082, 083, 084, 085, 087, 089, 090, 091, 092, 093, 094, 138, 256, 258, 260,

			312, 316, 325, 326, 329			258, 260, 310, 311, 312, 326, 329**]**		310, 311, 312, 316, 325, 326, 329**]**
QID_16	Kaphu Ammano Laaltanno Guma	Sheexaanu, kaphu duduwi HEGERE HEESHSHO hunanno	072, 086, 069, 222, 313, 317, 318, 320, 321, 322, 324, 326, 327, 328, 340	15	10	**8** [069, 222, 313, 317, 318, 320, 321, 327]	16	**14** [072, 086, 069, 222, 313, 317, 318, 320, 321, 322, 324, 326, 327, 328]
QID_17	Bushshun na wayi agarooshs he	Giwirinnu loosira baattote shiilonna laalchimma hattono xeenu wayi keeraanchimm a	110, 298, 333, 334, 347, 380, 386, 388, **390**, 394, 395, 405, 406, 407, 410, 413, 451, 452, 453	19	14	**12** [298, 334, 388, 393, 395, 413, 451, 452, 453]	22	**18** [110, 298, 333, 334, 347, 380, 386, 388, 394, 395, 405, 406, 407, 410, 413, 451, 452, 453]
QID_18	Weesete Loosi Mixo	weesete Kaashonna Baatto harishsha	229, **253**, 259, 271, 274, 276, 280, 281, 293, 336, 353, 358, 370, 371, 385, 478	16	14	**12** [229, 259, 271, 276, 281, 293, 275, 336, 353, 370, 371, 478]	17	**15** [229, 259, 271, 274, 276, 280, 281, 293, 336, 353, 358, 370, 371, 385, 478]
QID_19	waasa, shaafeeta, buurisame	Sidamaho Budu Sagale	100, 113, 119, 120, 129, 130, 132, 140, 224, 234, 237	11	7	**7** [100, 119, 120, 129, 132, 140, 234]	11	**11** [100, 113, 119, 120, 129, 130, 132, 140, 224, 234, 237]
QID_20	Imaanann a budu uduunne	seemma, qolonna, gonfa	032, 119, 120, 133, 149, 228, 279, 367	8	7	3 [032, 119, 149,]	8	8 [032, 119, 120, 133, 149, 228, 279, 367]