ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF COMPUTING

AFAAN OROMO TEXT RETRIEVAL SYSTEM USING PROBABILISTC APPROACH

BY: ISAYAS WAKGARI KELBESSA

ADVISOR: SHIMELIS ASSEFA (PhD)

A Thesis Submitted to Adama Science and Technology University School of Electrical Engineering and Computing in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information System

Adama, Ethiopia

September ,2016

ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF COMPUTING

AFAAN OROMO TEXT RETRIEVAL SYSTEM USING PROBABILISTC APPROACH

Adama, Ethiopia

September ,2016

ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING

DEPARTMENT OF COMPUTING

AFAAN OROMO TEXT RETRIEVAL SYSTEM USING PROBABILISTC APPROACH

BY: ISAYAS WAKGARI KELBESSA

Name and signature of Members of the Examining Board:

Name	**Title**	**Signature**	**Date**
Shimelis Assefa (PhD)	Advisor	_____	_____
_____	Examiner	_____	_____
_____	Examiner	_____	_____

Adama, Ethiopia

September, 2016

APPROVAL SHEET

This Research Thesis entitled "**AFAAN OROMO TEXT RETRIEVAL SYSTEM USING PROBABILISTC APPROACH**" has been read and approved as meeting the preliminary research requirements of the Department of Computing in partial fulfillment for the award of the degree of Master in Information System, Adama Science and Technology University, Adama, Ethiopia.

Shimelis Assefa (PhD)	_____	_____
Advisor	Signature	Date
_____	_____	_____
External Examiner	Signature	Date
_____	_____	_____
External Examiner	Signature	Date
_____	_____	_____
Institute coordinator for Research, P, GS&C	Signature	Date

Acknowledgement

First of all, I would like to thank my creator, 'GOD' for giving me health and patience to complete this work. '***Yaa waaqaayyo, barabaraan galatnikee hin xinnaatin***'. Next to this, I want to express my great gratitude to my advisor Dr. Shimalis Assefa for his continuous advice and guidance, critical and timely comments, providing me some resources and related research papers throughout my thesis work. I also would like to thank my father, Wakgari Kelbessa, who help me in many ways. Lastly, but not the least, I express my special thanks to my brother, Ato Wakena Wakgari; who struggle a lot to take me to this position. Really I never forget you wakena since you are the person who plays a great role for my success next to GOD and who think for me more than yourself. I am deeply grateful to my father, mother, other brothers and sisters for their patience and love.

Table of Contents

List of tables I

List of figures II

List of codes III

List of Acronyms IV

Abstract V

CHAPTER ONE 1

INTRODUCTION 1

1.1. Background of the study 1

1.2. Statement of the problem 4

1.3. Objective of the study 7

1.3.1. General objective 7

1.3.2. Specific objectives 7

1.4. Scope and limitation of the study 8

1.4.1. Scope of the study 8

1.4.2. Limitation of the study 8

1.5. Significance of the study 8

1.6. Organization of the thesis 9

CHAPTER TWO 11

LITERATURE REVIEW 11

2.1. Overview of an IR system 11

2.2. Related Works 12

2.2.1. IR Systems for International Languages 13
2.2.2. IR systems for Local Languages 15
CHAPTER THREE 21
AFAAAN OROMO LANGUAGE 21
3.1. Introduction 21
3.2. A Brief Overview of Afaan Oromo 21
3.2.1. Dialects of Afaan Oromo /Loqoda Afaan Oromoo 22
3.2.2. Oromo Alphabets/ Qubee Afaan Oromo 23
3.2.3. Afaan Oromo Grammar (Seer-Luga Afaan Oromoo) 25
CHAPTER FOUR 32
METHODOLOGY 32
4.1. Sources of data 32
4.1.1. Corpus 32
4.2. Developmental tools 32
4.2.1. Tools used for Text Extraction 34
4.3. Possible Retrieval Models to IR System 34
4.3.1. Boolean model 35
4.3.2. Vector space model (VSM) 36
4.3.3. Probabilistic model 39
4.4. Okapi BM25Similarities for probabilistic model with Lucene 40
4.5. Steps and procedures to integrate BM25Similarities 41
4.5.1. Integration of BM25similarities into Lucene (for Java code) 41

4.5.2. Integration of BM25similarities into Lucene (for xml) 42
4.6. Why probabilistic model? 44
CHAPTER FIVE 46
SYSTEM DESIGN AND DEVELOPMENT 46
5.1. Lexical processing 49
5.2. Evaluation techniques 53
5.3. System architecture for Afaan Oromo 54
CHAPTER SIX 55
EXPERIMENTATION AND DISCUSSION 55
6.1. Query selection 56
6.2. Method of System evaluation 57
6.3. Experimentation 57
6.4. Effects of Synonyms and polysemy in Afaan Oromo 61
6.5. Results after lists of synonymy are added to the document 63
6.6. Answers of research question 64
6.7. Results and findings 67
CHAPTER SEVEN 69
CONCLUSION AND RECOMMENDATION 69
7.1. Conclusion 69
7.2. Recommendation 70
References 71
Appendix I: Websites from which data is collected 76

Appendix II---Afaan Oromo Stop words 77

Appendix III: list of Afaan Oromo synonymy 80

Appendix IV:BM25Similarity integration java code 81

Appendix V: Sample Code Used for Afaan Oromo Stemming 85

List of tables

Table 2.1: Summary of Related Works…….…….16

Table 3.1: Afaan Oromo vowels and sound with example……..…24

Table 3. 2: Afaan Oromo vowels…….....25

Table 3.3:Afaan Oromo Adjectives……27

Table 3.4: Afaan Oromo Adverbs……..28

Table 3.5:Afaan Oromo Prepositions…..……....31

Table 6.1: Relevance judgement ……....…….…….…….56

Table 6.2: Experimentation result before the addition of synonymy …….…….58

Table 6.3: Experimentation result after the addition of synonymy ……63

List of figures

Figure 3.1: Classification of Afaan Oromo Dialects……...23

Figure 4.1: Cosine Similarity…….…….34

Figure 5.1: Solr Pagination.…….…….…….…….…….46

Figure 5.2: Afaan Oromo Word Autosuggestion While Searching for a Document …….47

Figure 5.3: Snapshot for Afaan Oromo Spellchecker Using Solr……47

Figure 5.4: Snapshot of Afaan-Oromo-Core……...……48

Figure 5.5: Search Interface for Afaan Oromo……49

Figure 5.6: Afaan Oromo system architecture ……54

Figure 6.1: Types of Data Used for Afaan Oromo Corpus……...55

Figure 6.2: Architecture Design of Indexing Process for Afaan Oromo…….….66

List of codes

Code 5.1: Afaan Oromo Tokenizers.....50

Code 5.2: Afaan Oromo Stemming.....53

Code 6.1: Synonymy integration sample code.....65

List of Acronyms

IR	Information Retrieval
VSM	Vector Space Model
VOA	Voice of America
OMN	Oromia Media Network
OBS	Oromia Broadcasting service
TV	Television
PRP	Probability Ranking Principle
iOS	iphone Operating system
WWW	World Wide web
LSI	Latent Semantic Indexing
OTS	Open Text Summarizer
MAP	Mean Average Precision
NIST	National Institute of Standard technology
TREC	Text Retrieval Conference
CLIR	Cross Lingual information retrieval
XML	Extendable Markup Language
SVO	Subject Verb Object
ORTO	Oromia Radio and Television organization
GHz	Giga Hertz
GB	Giga Byte
RAM	Random Access Memory
JDK	Java Development Kit
MYSQL	MY Structure Query Language
TF	Term Frequency
IDF	Inverse Document Frequency
BM	Best Match

Abstract

Modern information retrieval (IR) system should be able to take natural language queries and retrieve documents written in natural languages. Since obtaining relevant information from a collection of informational resources in Afaan Oromo is very important for Afaan Oromo speakers, developing a system that help users of Afaan Oromo is mandatory. That is why this study is envisioned to make possible retrieval of Afaan Oromo text documents by applying techniques of modern information retrieval system. In the developed Afaan Oromo prototype, Probabilistic approach was used as an information retrieval models and precision and recall measurement were used as the performance measurement or evaluation technique. In addition to this, F-measure is also used to show the accuracy of the system by avoiding the precision recall trade-off. The similarity measurement used was also BM25Similairty. Apache Solr was also used as an environmental programming language to achieve the evaluation goal. Afaan Oromo text retrieval is evaluated using 158 documents and 13 arbitrarily selected queries that can determine the effectiveness of retrieval using the precision-recall. The average result obtained by our evaluation before the addition of synonymy was 72.91% precision and 86.8% recall respectively. After the addition of synonymy and polysemy the value was changed to 71.39% average precision and 90.5% average recall. The F-measure for the evaluation before synonymy addition was 79.25% and after addition changed to 79.82%. The addition of synonymy improves the system performance by 0.57%. The study therefore, experimentally prove that the addition of the thesaurus system can improve the system performance. Spellchecking, pagination, hit highlighting and autosuggestion are also possible in the developed prototype for Afaan Oromo.

Key words: *Afaan Oromo, Recall, Precision, Information Retrieval, Apache solr, Probabilistic approach.*

CHAPTER ONE

INTRODUCTION

1.1. Background of the study

There is a need for people all over the World to be able to use their own language when using computers or accessing information on the Internet. Since the life of human being is highly connected with information, the method of storing and retrieving the stored information is very important. The task of finding relevant texts within a large amount of unstructured data is called Information Retrieval (IR). According to (James, 2002),afield that concerned with the structure, analysis (a form of literary criticism in which the structure of a piece of writing is analyzed), organization, storage, searching (finding relevant information for the information need of a user), and retrieval of information is called an Information retrieval. To provide access to every resources which can be books, journals, documents and to reduce information overload many universities and public libraries use IR systems. The most visible applications of IR are search engines (George & Hameed, 2013).An IR System is a system that has the ability to store, retrieve and maintenance of information. The general objectives of an Information Retrieval System are to minimize the overhead (the time a user spends to find the information) of a user locating needed information. Information that the user need is retrieved by literally matching terms in documents with those of a query (Yeni, 2003). Because of the contribution of an information retrieval, it is needed to use an IR system in our day to day activities.

Information retrieval (IR) is different to database retrieval. An information retrieval system needs to assess the information needs of its users and rank the answer documents based on likely relevance, whereas, a database retrieval system simply retrieves all documents or objects that satisfy certain criteria (Jelita, 2007). Data base retrieval can retireve documents based on some criteria and also it retrieve documents from structured data. However, information retrieval system is used to retrieve documents or texts from unstructured data without any critera by simply inserting our query.

All people in the world have a great wish if the technology speaks their language instead of they speak the language of technology to get the needed information. That means Human beings are very happy when the technology has the ability to support their language instead of they speak the language of technology which is English most of the time. Because of this Information Retrieval (IR) is one of the current hot research areas for scientists and academic researchers. Many languages, especially on the African continent, are "under-resourced" in that they have very few computational linguistic tools or corpora (such as lexica, tree-banks, part-of-speech taggers, parsers, etc.) which have proven to be a bottleneck for promoting the use of computers and the Internet in their language. The purpose of this thesis is, therefore, to overcome the bottleneck in one of the widely used languages in Ethiopia, i.e. Afaan Oromo.

There is a wide storage of information recorded or stored in natural language that could be accessible via computers which are constantly generated in the form of books, news, business, government reports and scientific papers (Abhimanyu, 2013). The information recorded or stored in natural language which is accessible via computer is needed for human beings, so, that information must be retrieved. The retrieved documents aim at satisfying a user information need usually expressed in natural language.

Searching interesting information is one of the most important tasks in IR. When a user provides a query to a system and that system responds with the information they need it is called information retrieval system process. The system can return both relevant and non-relevant documents. However, the most relevant documents come first and the irrelevant documents come at the end. Search engines presents the retrieved document set as a ranked list of document titles and the documents in the list are ordered by the probability of being relevant to the users request and the highest number of probability of the information need is considered to be more relevant to the query in such like process (Sonia & Reena, 2010).

Information retrieval involves returning a set of documents in response to a user query. An information retrieval process begins when a user enters a query into the system. But, the system has the ability to process text or text operation (example: tokenization, stop word removal, stemming and normalization) to respond to what the user need. Queries are formal statements

of information needs, for example search strings in web search engines. Text retrieval is a branch of information retrieval where the information is stored primarily in the form of text. For this study the text is stored in Afaan Oromo first and the query is also in Afaan Oromo. Users start with information needs, which they translate into query representations. Similarly, there are documents, which are converted into document representations. Based on these two representations, a system tries to determine how good documents satisfy information needs. Therefore, the main aim of this study is to retrieve the information stored in the form of text for Afaan Oromo and checking (measures) the performance of the information needed (query) to the relevant document by using probabilistic approach with apache solr.

According to (Robertson & Sp¨arck, 2006),in probabilistic approach a query typed by a database user to retrieve information is taken as an unstructured list of words or phrase and these terms are matched to documents in the database. Probabilistic approach is used to rank retrieved documents in terms of the degree of match.

Once probabilistic approaches have been computed for the query and for each document in the given collection, e.g., using a probability scheme, the next step is to compute a numeric similarity between the query and each document. To do so, BM25similarity was used. The documents can then be ranked according the probability they have to the query, i.e., the highest probability ranking document is the document most similar to the query. The proposed algorithm or Information retrieval model selected for this study is probabilistic approach. This approach is selected because of: Understanding of user need is uncertain and also whether document has relevant content is uncertain guess. To provide such like uncertain reasoning Probabilistic approach is used.

In addition to this, probabilistic approach has the ability to rank documents by their probability of relevance given a query and also identifies the text as relevant and non-relevant which is called probability ranking principle. In probability ranking principle, if there are collections of documents, user issues a query and set of documents needs to be returned. After that if the probability of relevant document is greater than non-relevant document, that document is relevant to the query. Probabilistic approach is selected because of the following reasons:

- ✓ To figure out best model that works for Afaan Oromo retrieval system from different kinds of IR models.
- ✓ When we use probabilistic approach, if the probability that a given document to the given query is zero then that document is not retrieved, but in VSM when we calculate the similarity we can get negative numbers which shows that the document is irrelevant but it ranked in decreasing order. Probability theory is what we really want an IR system to do: give relevant documents to the user.
- ✓ It would presumably be easier for most users to understand and base their stopping behavior [i.e., when they stop looking at lower ranking documents] upon … a 'probability of relevance' than [a cosine similarity value]
- ✓ The VSM major limitation is that it assumes that the terms are independent (which is deeply described in chapter four).

1.2. Statement of the problem

According to (Xiangji & Robertson, 2000), modern information retrieval (IR) system should be able to take natural language queries and retrieve documents written in natural languages. Users need when the queries provided by them are in their own language (users' own language) and the system (computers) are able to provide the relevant information need to them. Obtaining relevant information from a collection of informational resources in Afaan Oromo is very important for Afaan Oromo speakers. In the current world, the sustainable development of every country is dependent on the technology. The nations in a given country should have the technology implemented as per their culture, economic level, and social background like their language (Gudisa, 2013). What the user conveys to the computer in an attempt to communicate the information need is a query and which a user desire to know is information need.

Afaan Oromo belongs to the Cushitic branch of the Afro-Asiatic languages together with Afar, Somali and Sidama from more than 80 languages of Ethiopia. Afaan Oromo is one of the languages with large number of speakers under Cushitic family (Guutamaa, 2010). If Arabic is counted as African languages, Afaan Oromo is the 4th largest language but the 3rd next to

Kiswahili and Hausa if Arabic is not an African language (Gezehagni, 2012). As there are more users of this language, it needs to be able to support technology; because, Afaan Oromo language is used as the official language for oromia regional state of Ethiopia and also academic language for the oromia state.

The history of Afaan Oromo language can be traced back to the period of Italian occupation. During this time Afaan Oromo language was employed in radio broadcast and in office of administration next to Italian language though the transmission was immediately closed down with the restoration of the Emperor (Nutman, 2014). Afaan Oromo was used in administration and for other Medias next to Italian language when Italy was in Ethiopia, during 1935. However, after Italy left Ethiopia the emperor refused to use this language at that time and the language was banned. That is why this language was left late. But now the language is used in the country, especially in oromia region and the numbers of users are too many. So, modern information retrieval system is needed for the restoration of this language.

There are a number of Medias that use Afaan Oromo as primary language; for instance, Oromia Television and Radio (Web news), Kallacha Oromiyaa, Bariisaa, Yeroo, Voice of America (VOA) (web news), oromia media network (OMN), oromia broadcasting service (OBS), TV Oromia, Geda TV, Sii youtube and different academic and recreational medias to mention a few (Gezehagni, 2012). With the help of this Medias today Afaan Oromo becomes known to some extent and which again need the integration with the technology. There is huge amount of information released with this language through these Medias, since it is the language of education and research, language of administration and political welfares, language of ceremonial activities and social interaction. Therefore, the huge amount of information released with this language must be accessed.

Afaan Oromo has 26 characters without apostrophe (‘) which is called ‘Hudhaa’, double consonant also called ‘qubee Dachaa’ in Afaan Oromo (CH, DH, NY, PH, SH, TS). Double consonants are the collection of two alphabets (‘Qubee’). For instance, ‘C’ and ‘H’ becomes CH. There are some names which use such like words. Examples are: ‘Bulchaa’, ‘Nyaata’, ‘Dhadhaa’ and others. ‘Qubee’ is a Latin based script which stands for or represents English

word ‘alphabet’ which was developed by Sheikh Bakri Saphalo. In addition to the above problems, studying how these ‘hudhaa’ and ‘qubee dachaa’ works with technology is very important. In general, Afaan Oromo has 33 characters: general characters (26) + double consonant (6) + apostrophe (1) = 33.

As more and more people use technology to find and use information, developing an IR system for documents and texts in Afaan Oromo help millions of citizen using this language. This study is focused on enabling development of Afaan Oromo text retrieval to grow with current information technology support by using probabilistic approach of IR models. IR is not being optional technology, but it is very important and mandatory to use. Information is highly needed than anything else in this Information Age. But finding this important information needs system support.

There is an attempt in Ethiopia, to develop information retrieval system for the working language of Ethiopia which is Amharic. The main important problems that gave rise to conduct this study is that many Oromo people in Ethiopia and also in others country such as: Kenya, Djibouti Somalia, Egypt and South Africa use Afaan Oromo language. Users who look for Afaan Oromo document with Afaan Oromo query may not find suitable environment to find information for their need. The less knowledgeable ones still cannot understand the English language fully. So, why are those people who do not understand English not helped? Why the Oromo people cannot make their language part of the technology’s language? Generally, the statements of the problem for the study are the following shortly:

- Many Afaan Oromo speakers nowadays search for their information needs on Google; which is very broad to get what they need. So, the best way is visiting specific sites to get real information they need.
- There is also another attempt for Afaan Oromo but most of the attempt is on dictionary based approach instead of corpus based approach.
- The rich literacy works in Afaan Oromo language are not accessible in digital form

- Absence of text retrieval system for Afaan Oromo which is completely applicable and Lack of the development of Afaan Oromo to grow with current information technology support.
- To help users of Afaan Oromo to find information they need without any difficulties
- To satisfy users who look for Afaan Oromo document with Afaan Oromo query.

Hence the aim of this study is to develop a prototype for Afaan Oromo text retrieval system that organize document corpus as per users query based on Probabilistic approach.

So, this study answer the following research questions.

- To what extent probabilistic approach enables to design effective Afaan Oromo text retrieval?
- How can the problems related with polysemy and synonymy be handled and what is the simplest form of indexing to index Afaan O romo text?
- What text operations are needed in Afaan Oromo to apply for document corpus?

1.3. Objective of the study

1.3.1. General objective

The general objective of this study is to design an Information Retrieval system that can enable to search for relevant Afaan Oromo text corpus using probabilistic approach of the retrieved text using apache solr.

1.3.2. Specific objectives

The specific objectives which are used to achieve or accomplish the above general objectives are listed below.

- Review the features of Afaan Oromo writing system pertinent to the purpose of this research.
- To check the accuracy of the system by using probabilistic approach

- To evaluate the performance of the proposed prototype by using the recall and precision measures and recommend future research direction.
- To make Afaan Oromo one of the technology languages, and hence help Afaan Oromo speakers use their language to find and use information.

1.4. Scope and limitation of the study

1.4.1. Scope of the study

The scope of this study is designing Afaan Oromo text retrieval system that effectively searches with in Afaan Oromo text corpus by using probabilistic approach of IR model. The study involves both indexing and searching part of text for Afaan Oromo textual documents. Any other data types, such as image, video, and audio are out of focus of the study.

1.4.2. Limitation of the study

In the study, limited corpus was used for evaluating the performance of the IR system developed as a result of time factor, because, to prepare relevance judgment for corpus with many documents, it takes more time.

1.5. Significance of the study

The speakers of the specific language can benefit from the development of technology in many ways when the language is able to grow with technology. Based on this, localizing works already done in developed language such as; English is very important to solve the problem of the society. Therefore, the speakers' of Afaan Oromo can obtain relevant information from a collection of informational resources in Afaan Oromo which can be considered as the significant of the study. The study has advantage for:

a) **For the speakers of Afaan Oromo**: - This research helps the speakers of the language as they use their own language to search the information they need, i.e. the query is in Afaan Oromo.

b) **For the Development of the language**: The study also helps the language as it becomes the language of technology which is very important for the development of this language. Identify the best information retrieval model for the language, measures the performance of the recall and precision of the language, the way to preprocess in Afaan Oromo and identifies their difficulties for the next researcher.

c) **For the Researcher**: - Since the study is master thesis it has learning out comes for the researcher. It can help the researcher to investigate problems and solving it in scientific manner. It adds the experience of the researcher.

Generally, the main significances of this study are the followings:

- Enabling retrieving document of Afaan Oromo efficiently and effectively for the speakers of Afaan Oromo.
- Helps speakers of Afaan Oromo who cannot use any other language.
- Improving the development of Afaan Oromo Language.
- Understandable to Afaan Oromo users without any training, since it is in their own language
- It can be used as stepping stone for further work in this specific area and Opens way for others researchers to focus on the area and work on it.

1.6. Organization of the thesis

This thesis is organized into six main chapters. The first chapter is about the introductory part which includes: the background on the specific area about information retrieval and status of Afaan Oromo, what initiate the study--Problem statement, objectives of the study, significance of the study, scope of the study, limitation of the study.

Chapter two is about literature review. In this chapter an overview of an IR system and related works (related works for international languages and for local languages) are described.

Chapter three presents the basic concepts and overview of Afaan Oromo language. It is also about Grammars (Seer-Luga), alphabets (Qubee), consonant (dubbifamaa), vowels (dubbachiiftuu) and any other rules of Afaan Oromo to make it clear for those people who cannot speak Afaan Oromo.

The fourth is about major techniques and methods used in this thesis. Methodology—data collection methodology (sources of data, corpus preparation), evaluation technique and developmental tools are discussed in this chapter. The most important techniques for indexing and searching for Afaan Oromo text is used in this section. In addition to this, information retrieval model (Probabilistic approach), probability ranking principle (PRP) and advantages of probabilistic approach discussed in this section.

Dataset selections and preparations, implementations of the proposed work, experimentations, findings of the study, results of the study and issues in implementations are discussed in detail in chapter five. So, chapter five is about work experimentation and result of the study. The answer for the research question is also given in this chapter.

The last chapter but not the least chapter is chapter six which describes about major findings and faced challenges supported by conclusion (short description about what have been done and not done,) and recommendation in which works identified as future work and needs to get the attention of the other researchers is listed in.

CHAPTER TWO

LITERATURE REVIEW

2.1. Overview of an IR system

(RIJSBERGEN, 1979) has described about information retrieval in his research as, the problem of information storage and retrieval has attracted increasing attention since 1940s and it is simply stated that: we have vast amounts of information to which accurate and speedy access is becoming ever more difficult. One effect of this is that relevant information gets ignored since it is never uncovered, which in turn leads to much duplication of work and effort. The idea behind his description is: the absence of an information retrieval duplicates works and effort. So, in the following, overview about information retrieval (meaning) and the advantages of presence of information retrieval is described. IR (information retrieval) is the task of finding relevant texts (texts that match some specific criteria) with in a large amount of unstructured data.

Information retrieval is a broad concept, and a search engine is one of the most widely used implementations of information retrieval systems. Not only search engines there are also other areas of its application. For instance, in recommendation systems, document classification and clustering, filtering, and question-answering systems. The information to be retrieved can be available in any source, format, and volume and can be applied to almost any domain such as healthcare, clinical research, music, e-commerce, and web search (Dikshant, 2015).

In the current era of Android and iOS (iPhone Operating System) smartphones and their even smarter applications, information needs have increased drastically, and with the Internet of Things, the need for information is bigger and even more critical. So, Information retrieval systems are becoming the dominant form of information access with the beginning of web search engines (Dikshant, 2015).The information retrieval is the very hot research area of study, with the main aim of searching for relevant documents from large corpus that satisfies information need of users. A document is relevant if it addresses the stated information need,

not because it just happens to contain all the words in the query and the information need is not overt (Observable; not secret or hidden). IR research involves language dependent process (i.e. system that developed for English doesn't necessarily work for Afaan Oromo).

Searching for relevant documents from document corpus and World Wide Web (WWW) is very important task of an IR since IR searches both structured (organized) and unstructured (unorganized) information. The task of an IR system is not only searching but also indexing documents. Therefore, the whole IR system includes two main subsystems which are called: searching and indexing. Even though all terms existing in the corpus are not useful for the document representation, the search engine would scan every word in the collections if there is no index. So, indexing is very important.

Indexing is the process of preparing index terms which are either content bearing or free text extracted from documents corpus. Searching is the process of matching user's information via query to documents in collection via index terms. The process of searching and indexing is guided by model, i.e. Information Retrieval model, since models can serve as a blueprint to implement actual retrieval system. Mathematical models are needed to understand and reason out some behavior or phenomena in the real world (Dikshant, 2015). The detail information about Information Retrieval model is described in chapter four.

2.2. Related Works

Many researches have been done on text retrieval in the world and African countries but few in Ethiopia. However, some work done is found on the specific area of this study from the reviewed literatures, specifically on the bilingual information retrieval (Afaan Oromo-English cross lingual information retrieval). But, one scholars also made attempt to develop IR system that works for Afaan Oromo by using vector space model of IR model and python as developmental tool with the corpus size of 100 text documents written in Afaan Oromo language. In the following we have reviewed some works that have been done so far on text retrieval for international language and for local language. The gap identified from the previous research and what makes difference this study from other researches which are done previously by other researchers are also explained at the end of this chapter.

2.2.1. IR Systems for International Languages

Author	Title	Models used	Place	Result
Fazli,C.et al.(2008)	Information Retrieval on Turkish Texts	Vector space model	Turkey, Bilkent University, Computer Engineering Department	Improved precision =14.4% Improved recall=13.5%
Kareem,D. (2013)	Arabic information retrieval	VSM	Qatar Computing Research Institute	Precision =61%
Albujasim, Z. (2014)	Search Queries in an Information Retrieval System for Arabic-Language Texts	Vector space model	University of Kentucky	Precision =71.1% Recall = 50%
Xiangji, H. and Stephen, R.(2000)	Probability-Based Chinese Text Processing and Retrieval	Probabilistic model	United Kingdom, London, department of information science	Average precision = 0.37424 (37.42%)
S.Saraswathi, A. et al. (2010)	Bilingual Information Retrieval System for English and Tamil	Naïve Algorithm	India	Precision=61.82 % Recall=86.36%
Yu-Chun,W.(2009)	Korean-Chinese Cross-Language Information Retrieval Based on Extension of Dictionaries and Transliteration	VSM, Apache solr Lucerne	Taiwan,Yuan Ze University, department of Computer	MAP=0.1392

			Science and Engineering	
Yilu, Z. J. et al. (2006)	Experiments on Chinese-English Cross-language Retrieval	Vector space model	University of Arizona	Average precision =92.35%
Savita,C. M.and S. S. Pawar (2015)	Marathi-English CLIR using detailed user query and unsupervised corpus-based Word Sense Disambiguation	VSM, Lucene, apache solr	India, university of SavitribaiPhule Pune	Average precision =0.142(14.2%) Average recall = 0.8647(86.47%)
Javid, D.A. and Heshaam,F. (2014)	Probabilistic Translation Method for Dictionary-based Cross-lingual Information Retrieval in Agglutinative Languages	Probabilistic approach(Okapi)	Iran, university of Tehran	MAP=0.4259(42.59%)
Emad,E.et al. (2015)	Semantic Boolean Arabic Information Retrieval	Semantic Boolean model	Egypt, Menoufia University, Faculty of Computers and Information	Precision =100%
Monther,T. and Emad ,A.(2015)	A Hybrid Approach for Indexing and Searching the Holy Quran	syntactic-and semantic-based approach	Jordan, Yarmouk University, department of Computer Information Systems	Average recall = 95.54%

Tiruveedula ,G. et al. (2014)	Future Model for Information Retrieval Systems Based on Arabic Language	Hierarchal Multilevel Classification technique	Libya, Sirt University	Average precision =75% Average recall =55%

2.2.2. IR systems for Local Languages

Tewodros, H.G. 2003)	Amharic text retrieval: LSI	Latent Semantic Indexing (LSI	Addis Ababa university, Ethiopia	Precision=71 .57% Recall = 69.13%
Hailay,B.B. (2013)	Design and Development of Tigrigna Search Engine	VSM, Lucene, Apache tomcat	Ethiopia, Addis Ababa University	Precision =71% Recall =100%
Abey, B. and Tulu, T.(2015)	Bi-gram based Query Expansion Technique for Amharic Information Retrieval System	Bi-gram query based technique	ArbaMinchi university, Ethiopia	Average precision =0.57 (57%) Average recall= 0.67(67%)
Daniel,B.A. (2011)	Afaan Oromo-English Cross-Lingual Information Retrieval (CLIR): A	VSM	Ethiopia, Addis Ababa university	maximum average precision value of 0.421 and 0.304 are obtained for the Afaan Oromo and English

	Corpus Based Approach			documents respectively
Girma,D.(2012)	Afaan Oromo news text summarizer	*OTS (Open Text Summarizer) as a tools*	Addis Ababa university, Ethiopia	Precision =75% Recall =74.90%
Gezehagn, G.(2012)	Afaan Oromo Text Retrieval System:	Vector space model(VSM)	Addis Ababa university, Ethiopia	Precision=57 .5% Recall= 62.64%

Table 2.1: summary of related works

Note: VSM-vector space model **OTS**-Open Text Summarizer,

LSI-Latent Semantic Index **MAP**-Mean Average Precision

As it is described in Table 2.1, many researches have been done on the international languages as well as few researches on local languages. That is why the researcher of this study identify them as an IR system for international language and IR system for local languages to know which reviewed literatures are similar to the study. Title of the research, year of the research, IR models of the research, the place where that research was done and result obtained have been clearly stated in the table. In the following paragraphs the summary of the reviewed literatures, similarity and difference of the reviewed literatures with this study are explained.

(Fazli,et al., 2008) have been studied on Turkish language, using a large-scale test collection that contains 408,305 documents and 72 ad hoc queries. Their concern was on stemming algorithm of Turkish language to retrieve texts and they hypothesize that within the context of Turkish IR: the use of a stop word list in indexing, the use of language specific stemming algorithms (since more accurate stems would reflect better document content and this would be more noticeable in larger collections), longer queries (since they provide better description

of user needs),longer documents (since the document contents may become more precise as we increase document sizes) can improve the system performance in terms of higher retrieval effectiveness. Even if the language and the model is different, the researcher of this study belief that having language specific stemming algorithms, longer documents as well as queries have the ability to improve system performance.

The other researcher made attempt on international language was: (Albujasim, 2014) on Arabic language and the problems that motivating the researcher to do his research was normalization of Arabic language and a complex morphology of Arabic language that has a significant effect on the performance of an IR system. (Albujasim, 2014) also agree to the idea of (Fazli,et al., 2008) which says 'the use of language specific stemming algorithms' improves the performance of the system. The stemming algorithm of one language cannot work for other language; if we do so, the performance of the system is low, which is also true for Afaan Oromo. (Darwish, 2013) has also done on Arabic text information retrieval. However, the result and the objectives are not clearly explained. Most researches have been done on Arabic language since it has large number of speakers. In addition to (Albujasim, 2014), (Darwish, 2013) another research is done by using semantic Boolean model by (Emad,et al, 2015) on Arabic language. However, even if they have done on the same language (i.e. Arabic) with different model the result is different. The only best result is obtained by (Emad,et al, 2015) which is found in the above table.

Even if Chinese language is the largest language of the world by its native speakers, the development of Chinese language is far below of other language like English. This is because of there is no separator between Chinese words, so, Chinese words cannot be used easily to index or search texts as is possible in English (Xiangji & Robertson, 2000).However, (Xiangji & Robertson, 2000) used the collection from the National Institute of Standard and Technology(NIST) for participating in the Text Retrieval Conference(TREC) which contain 164,768 documents. The IR model used is completely similar to this study. They have used probabilistic model of an IR model and BM25Similarity measure which works based on OKAPI algorithm. But there is no a large collection of Afaan Oromo documents, which make difference.

Bi lingual information retrieval system also has been done on many languages. For example, (Saraswathi, et al., 2010) for English and Tamil language; (Yu-Chung,et al., 2009) for Korean-Chinese cross language; (Yilu,et al., 2006) for Chinese-English Cross-language; (Mayanale & Pawar, 2015) for Marathi-English Cross Lingual Information Retrieval (CLIR). Although most of them used similar information retrieval model, they obtained different results.

(Javid & Dadashkarimi, 2014) used probabilistic model to check Cross-lingual Information Retrieval in Agglutinative Languages to solve the ambiguity problem. The model used is similar to the model that is used in this study. To measure the similarity of the query to the document Okapi was used which is also true for this study. On the other hand, (Tarawneh & Al-Shawakfa, 2015) have done research for Searching the Holy Quran which is used to introduce and build an approach that can help in processing and understanding the Quran text. The concept of XML is used to represent the different chapters of the Holy Quran. Similarly, this study use XML file format to post data to the server.

(Tiruveedula,et al., 2014) on Arabic language. The idea was:

- to investigate the effects of the Arabic text classification on Information Retrieval for Arabic language and
- provided excellent solutions for Arabic-IR system by means of emerging both Arabic Linguistics and Information Retrieval System.

As it is shown in the above table, from more than 15 reviewed literatures, the only research in which the performance of the system score 100% precision value is the research that is done by (Tiruveedula,et al., 2014). However, there is no any reason which is properly explained for this result in their research.

There are also few researches which are done on local languages in Ethiopia. Some of them are: (Bruck & Tilahun, 2015) and (Tewodros, 2003)on Amharic language, (Hailay, 2013) on Tigrigna language and on Afaan Oromo there are some reviewed literatures which was done by (Daniel, 2011), (Girma, 2012) and (Gezehagn, 2012). The aim of (Bruck & Tilahun, 2015) research is to increase precision of an Amharic information retrieval system while preserving

the original recall. In order to do that query expansion is necessary and for that purpose bi-gram technique has been used. The idea of using query expansion is to provide relevant documents as per users' query that can satisfy their information need. Because users are not fully knowledgeable about the information domain area, they mostly formulate weak queries to retrieve documents. According to their study when they have used bi-gram technique, there is improvement by 8% from the previous work. Since Amharic is morphologically rich language that has lexical variation and has problems related with polysemy and synonymy (Tewodros, 2003) use LSI to improve the result achieved by using VSM (i.e. 0.6913) by using previous study and by LSI the result is 0.7157 which shows that LSI improves the VSM result by 2.4%.

Query reformulation (query expansion) improves system performance by managing polysemy and synonymy words. While developing an IR system, query expansion must be the best idea that must be concerned. However, the technique and model used may have effect on query expansion as it is explained by (Tewodros, 2003). For example, (Gezehagn, 2012), used python programming language and VSM. But, in this study probabilistic model as an IR model, Okapi (which is the most wonderful similarity measurement method in probabilistic approach) and apache solr search engine (which is used for: query expansion, spell checking, word auto completion or word prediction and best for indexing) are used. In many researches that have been done previously by different scholars since python programming language is used in most of the study, it needs a long time investigation to write a program for spell checking, word auto completion or word prediction, query expansion or reformulation and full text search local languages such as Afaan Oromo. Generally, this study is different from any other research in that:

- It uses probabilistic model as an IR model for Afaan Oromo and BM25 similarity measurement to measure the similarity of the document retrieved to query.
- It uses apache solr where we can do spell checking, word auto completion or word prediction, query expansion or reformulation and searching of Afaan Oromo word from the corpus easily.

- Other researches measure only the effectiveness of the system; however, in this study effectiveness of the system is measured and efficiency (response time) of the system is also available.
- No one is coming up with the solution for the problem of polysemy and synonyms in Afaan Oromo, even if it is for small number of words. But this study solves the problem of synonymy.

CHAPTER THREE

AFAAAN OROMO LANGUAGE

3.1. Introduction

Afaan Oromo need an IR system, which is the main aim of this study and to examine the potential of probabilistic approach technique in the retrieval of Afaan Oromo text documents. In this chapter, some general background information about the Afaan Oromo language (overview of Afaan Oromo), alphabets of Afaan Oromo, dialects of Afaan Oromo, rule and regulation in Afaan Oromo (grammars, or Seer-Luga) are described, because, before we are going to solve problems of Afaan Oromo we have to know about it.

3.2. A Brief Overview of Afaan Oromo

There is no doubt that every human language can serve its people in both verbal and written communication since language is a means of communication and a symbol of national identity (Rosenberg & Sillince, 2000). As a means of communication a language serves as a bridge to bring two sides together (verbal and written communication). Written communication requires symbols representing the words while Verbal communication involves using spoken words. Unfortunately, not all languages in the world are written languages; there are also some languages which used only for verbal communication because of some reason. Afaan Oromo language was also in such like situation until the alphabet era.

As it is described in the problem of statement (section 1.2), Afaan Oromo is Cushitic language which is family of Afro Asiatic languages. This language is spoken by more than 33.8 million people according to the 2007 census (Milkessa, 2014). Most of native speakers are people living in Ethiopia, Kenya, Somalia and Egypt. If Arabic is counted as African language, Afaan Oromo is fourth (4^{th}) largest language, but if not, third (3^{th}) largest language in Africa following Kiswahili and Hausa (Gezehagni, 2012).

Afaan Oromo, as a language of wider communication in Ethiopia, could not get the opportunity to be used as a written language until Abba Gammachis (Onesimos Nasib), the prominent

Oromo, translated the Bible into Afaan Oromo in 1880's. Moreover, Abba Gammachis made efforts to use Afaan Oromo as a medium of instruction. However, this was short lived because the Ethiopian ruling regime of that time banned the use of the language in schools.

Before colonization, the Oromo had used their own language in social, religious, educational, political, and legal activities. The Oromo language has been neglected and suppressed by Ethiopian authorities; however, in 1991, after the overthrow of the Derg, an alphabet using Latin characters known as *qubee* was officially adopted for Afaan Oromo. Soon after, Afaan Oromo was allowed to be used as the medium of instruction in elementary schools throughout Oromia and in print and broadcast media. The adoption of a single writing system allowed a certain amount of standardization of the language, and more texts were written in Oromo between 1991 and 1997 than had been in the previous 100 years (Guutamaa, 2010).

It is not enough for the language that it has more number of speakers. This means, having more number of speakers cannot be a slogan (cannot be an indicator of development) for the language, rather, when the language support technology and vice versa, when the users of that language are able to search the needed resources from the computer system without any problem it is possible to say that the language is developed. On this world some languages have more and more number of speakers or followers, but they are less used in technology. For this Chinese language can be an example. (Xiangji & Robertson, 2000) have been described about how much Chinese language is used in technology even if it has large number of speakers.

3.2.1. Dialects of Afaan Oromo /Loqoda Afaan Oromoo

According to the most recent work on the classification of Afaan Oromo dialect (Tilahun, 2015), the Oromo language have been divided into six dialects such as west, central, northern, southern, southeast and eastern dialects as indicated in the following tree-diagram which is done by Feda.

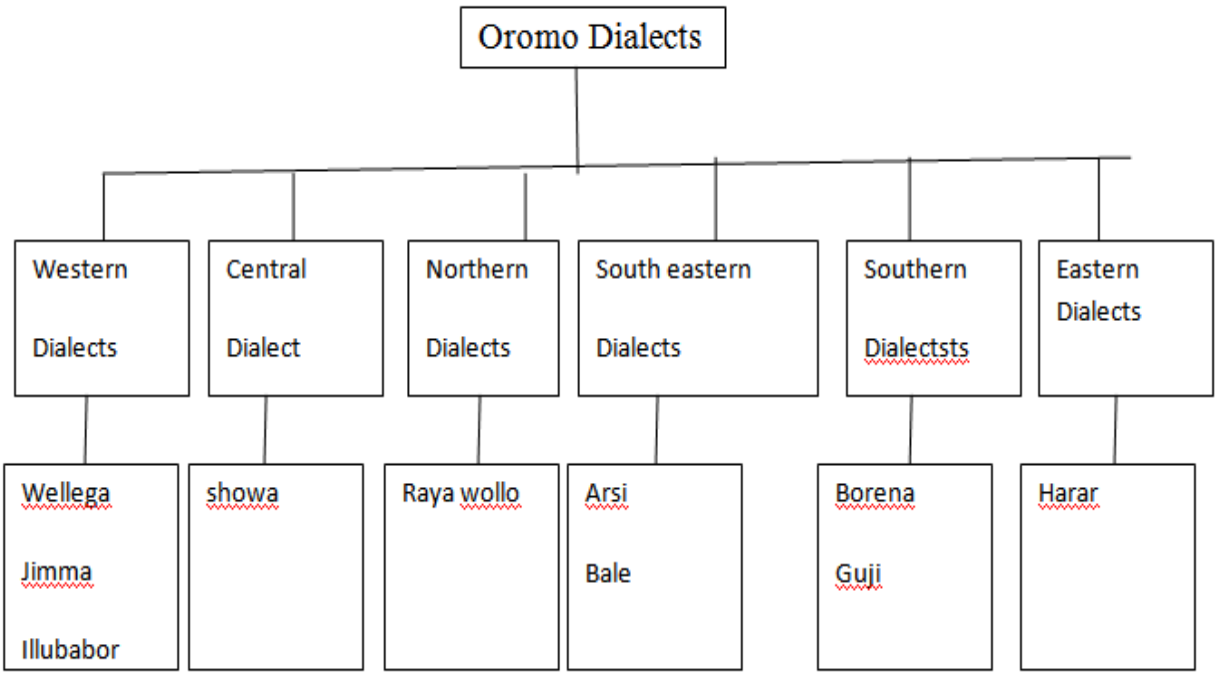

Figure 3.1: Classification of Afaan Oromo dialects. Feda (2015)

The major differences in dialects are in the form of pronouns, certain verb conjugations, and informal lexicon.

3.2.2. Oromo Alphabets/ Qubee Afaan Oromo

Afaan Oromo alphabets can be divided in to two main categories. These are: consonant (dubbifamtoota) and vowels (dubbachiiftota). Afaan Oromo vowels (Dubbachiiftuuwwan/dubbachiiftota) are represented by the five letters, a, e, o, u and i. Or long vowels; aa, ee, oo, uu,ii and consonants (Dubbifamtoota)are represented by 21 letters: b, c, d, f, g, h, j, k, l, m, n, p, q, r, s,t, v, w, x, y, z excluding apostrophe. Double consonant (qubee dachaa) are the combination of 2 consonant characters. They are six in numbers. For instance, CH, DH, NY, PH, TS, SH. Since there are some words which need double consonants in Afaan Oromo, it is must to use them where they are needed (Guutamaa, 2010). Example: 'Dhadhaa' to mean 'butter', 'nyaata' to mean 'food', 'chaappaa' to mean 'stamp'.

Afaan Oromo is phonetic language that is spoken in the way it is written. In contrast to English in which the same letter may sound differently in different words, Afaan Oromo characters

sound the same in every word. For instance, when we sound the word 'qara' we first sound 'q' letter and also when we write this word we start with 'q'.

A. Vowels/ dubbachiiftota

Even though there is a difference in sound or pronunciation in English language and Afaan Oromo, there are five vowels in Afaan Oromo similar to English language. Sound in Afaan Oromo can be: long sound (sagalee dheeraa), short sound (sagalee gabaabaa) and stressed sound (sagalee jabaa). When two consonants are written consecutively without a vowel; it is assumed as though there is a vowel (IrraButaa). Some examples of such like Afaan Oromo sounds are: Farda(horse), morma(neck), Arba (elephant), milkii (chance or the name of the person), galma(auditorium).

Afaan Oromo vowels (Dubbachiiftuu) are represented by the five letters, a, e, o, u and i. Or long vowels; aa, ee, oo, uu and ii (Guutamaa, 2010). If these vowels are fully absent from a given word in Afaan Oromo, it has no meaning. But they can have meaning themselves without any consonant. That is why they called 'dubbachiiftota' in Afaan Oromo.

Example, uuuuuuuuu..! = crying out for help, Ee…/eeyyee =correct, ok

Example of Afaan Oromo sound is listed in the following table,

Types	**Example**	**Meaning**
Stressed sound (sagaleejabaa):	Dammee, damma	The name of girl, refer to a person as honey, honey
Short sound(sagaleegabaabaa)	Bona, lola, mana	Summer, fight, house
Long sound (sagaleedheeraa)	Baala, balaa,	Leaf, accident

Table 3.1: Afaan Oromo vowels and sound with example

Vowels (Dubbachiiftonni) make as the consonant (Dubbifamtoonni) characters have meaning. Generally, the following table shows the five Afaan Oromo vowels.

	Uppercase/ DubbachiiftuuGurguddaa				
Uppercase Vowels (Dubbachiiftuu)/ short uppercase	A	E	I	O	U
Long Uppercase vowels	AA	EE	II	OO	UU
	Lower case/Dubbachiiftuuxixiqqoo				
Lowercase vowels/ short lowercase	a	e	i	o	u
Long lowercase vowels	aa	ee	ii	oo	uu

Table 3.2: Afaan Oromo vowels

B. Consonant/dubbifamtoota

In Afaan Oromo, consonants do not have their own sound which means that the sound of the consonants depends on the vowels. More than two consonants of the same type cannot be written consecutively. In AO When a consonant is stressed, it brings about change in meaning. For instance, Sodaa = Fear; Soddaa = son-in-law. The full description about the Afaan Oromo consonant is explained in section 3.2.2.

3.2.3. Afaan Oromo Grammar (Seer-Luga Afaan Oromoo)

Afaan Oromo has its own grammar which is called 'Seer-Luga', as every language has their own grammar. In Afaan Oromo consonants do not have their own sound which means that the sound of the consonants depends on the vowels. If this is made, it is grammatically in correct. More than two consonants of the same type cannot be written consecutively.

Dammee = refer to a person as honey or it could also be a girl's name; but Dammmmee= is meaningless and grammatically wrong. We can only use two vowels consequently, unless separated by apostrophes (hudhaa).

Taa'i = Have a sit, Walga'ii = Meeting

The logical flow of Afaan Oromo grammar is SOV (Subject Object Verb) as opposed to English which is SVO. For example: when we say 'he came yesterday' (SVO) for English; in Afaan Oromo we have to say 'Inni Kaleessa dhufe' (SOV). He (Inni) is subject which is common for both but came (dhufe) and kalessa (yeterday) exchange their place.

Gender in Afaan Oromo

In Afaan Oromo gender can be *feminine or masculine*. Oromo feminine refers to female qualities attributed specifically to women,'dubartoota' and girls,'durboota' or things considered feminine while nouns such as abbaa 'father', ilma 'son', and sangaa 'ox' are masculine (Gezehagn, 2012).Frequent gender markers in Afaan Oromo include -eessa/-eettii, -a/-ttii or –aa/tuu.

Afaan Oromo	**Construction**	**Gender, English**
Obboleessa	obbol + eessa	male, brother
obboleettii	obbol + eettii	female, sister
beekaa	beek + aa	male, knowledgeable
beektuu	beek + tuu	female, knowledgeable

Negation in Afaan Oromo/ diddaa

In Afaan Oromo, negation is the process that turns an affirmative statement (I am happy) into its opposite denial (I am not happy) which is called in Afaan Oromo 'diddaa'. For example,

Statement	**negation of the given statement**
Nan dubbadha/ I speak	Hindubbadhu/ I don't speak
Nan jaaalladha / I like	Hinjaalladhu/ I don't like

The prefix 'Hin' is used as negation in Afaan Oromo language.

Adjectives (Ibsa-maqaa)

According to (Dikshant, 2015), adjectives define the attributes of a noun (such as red or intelligent) and he called such important things in the language as Part-of-Speech Tagging (Common parts of speech include noun, verb, and adjective). In Afaan Oromo Adjectives are words that describe or modify another person or thing in the sentence. Adjectives can explain: Colors, 'bifa, bifoota', Shapes, 'boca'and Sizes, 'hamma'. Oromo adjectives have the ability to explain noun. When we say '*the old red house'* (manicha diimaa moofaa), the words 'diimaa and moofaa' are used to explain the noun manicha, to identify what kind of house is it. Following lists of adjectives are listed in Table 3 for more information about Afaan Oromo adjectives.

Oromo Adjectives	**English Adjectives**
bifoota	Colors
gurraacha	Black
baluu	Blue
bifabunaa	Brown
daalacha	Gray
magariisa	Green
diimaa	Red
adii	White
hammaguddina	Sizes
guddaa	Big
gadifagoo	Deep
lafarradheerata	Long
dhiphaa	Narrow
gabaabaa	Short
xinnaa	Small
dheeraa	Tall
yabbuu	Thick

qallaa	Thin
ball'aa	Wide
Boca	Shapes
geengoo	Circular
sirrii	Straight
golarfee	Square
golsadee	Triangular
dhandhama	Tastes
hadhaawaa	Bitter
asheeta	Fresh
sogidaawaa	Salty
oganaawaa	Sour
Kan urgoo baayyatu	Spicy

Table 3.3: Lists of Afaan Oromo Adjectives.

Source: https://en.wikibooks.org/wiki/Afaan_Oromo/Chapter_06

Adverbs (Ibsa-Xumuraa)

In every day conversation there are adverbs. They are words that modify any part of language other than a noun. They can modify verbs, adjectives, clauses, sentences and other adverbs. In Afaan Oromo, adverbs can be categorized as: adverbs of place, adverbs of time, adverbs of frequency and adverbs of manner. The followings are some samples of Afaan Oromo adverbs.

	English	**Afaan Oromo**
Adverbs of place	here	as
	there	achi
	everywhere	iddoo hunda
	home	mana
	out	ala
Adverbs of time	yesterday	kaleessa

	today	har'a
	tomorrow	bor
	now	amma
	later	Booda, eger
Adverbs of frequency	always	yeroo hunda
	sometimes	gaaffii gaaf/ Yeroo tokko tokko
	occasionally	gaaffii gaaf
	seldom	darbee darbee
	rarely	darbee darbee
Adverbs of manner	hard	cimaa
	quickly	dafee
	slowly	suuta
	carefully	qalbiidhan
	together	walii wajjin

Table 3.4: Afaan Oromo Adverbs

Pronouns in Afaan Oromo (Bamaqoota Afaan Oromo)

'Bamaqoota' comes from two Afaan Oromo words which is 'bakka' i.e. place and 'maqaa' i.e. noun which gives 'bakka maqaa' i.e. in place of noun. The followings are lists of Afaan Oromo pronouns.

English pronouns	Afaan Oromo Pronouns/ Ba-maqaa (bakka maqaa)
I, You	ani/ ati
He, She	inni/ isheen
We, They	nuhi/ isaan
Me, You	ana/ si
Him, Her	isa/ ishee
Us, Them	nuu/isaan
My, Your	koo/ kee
His, Her	isaa/ ishee
Our	keenya teenya
Their, Mine	isaanii/ kooti
Yours, His	keeti/ kanisaati
Hers	kanisheeti
Ours	kankeenya
Their	kan isaaniiti

Prepositions

Afaan Oromo prepositions can be divided in to two categories. These are: true prepositions and post prepositions. The main tasks of prepositions in Afaan Oromo is to link nouns to an action or to another noun (Tesfaye, 2010). For instance, if we take this statement: 'the book on the table' (kitaaba teessoo irra jiru) and 'go from there' (achii adeemi), the preposition in the first statement link noun to noun, but the second one is used to link noun to action. The only difference of Afaan Oromo preposition is that the true preposition comes before the noun and the post preposition comes after the noun. Usually with place names, used to be mean "in", no preposition or postposition is used. Therefore, we can simply say "**Finfinnee jiratta**" which stands for "you live in Finfinnee [Addis Ababa]", or "**mana kitaabaan ture"** for "I was in the library", no preposition is there. The followings are some of Afaan Oromo prepositions.

Postpositions	**Prepositions**
ala — out, outside	**gara** — towards
bira — besides, with, around	**eega, erga** — since, from, after
booda — after	**haga, hanga** — until
cinaa — besides, near, next to	**hamma** — up to, as much as
dur, dura — before	**akka** — like, as
duuba — behind, back of	**waa'ee** — about, in regard to
irra — on	
irraa — from	
itti — to, at, in	
jala — under, beneath	
jidduu — middle, between	
malee — without, except	
wajjin — with, together	
gubbaa — on, above	
fuuldura — in front of	
gad(i) — down, below	

Table 3.5: Afaan Oromo Prepositions

Source: https://en.wikibooks.org/wiki/Afaan_Oromo/Chapter_10

CHAPTER FOUR

METHODOLOGY

4.1. Sources of data

Related documents, secondary data, which are specifically related to the study's objectives, have been reviewed from different sources (books, journals, internet, and other thesis) etc. to understand Afaan Oromo text retrieval using probabilistic approach and measure the performance.

4.1.1. Corpus

The corpora have been prepared from different websites such as: the official website of oromia national regional state, website of voice of America Afaan Oromo language, oromia media network site, TV oromia, international bible society official website, oromia radio and television organization (ORTO) and any other Internet based sources. Resources from holy bible books, news articles and the Internet are used as data set. The corpus size is 158 documents. The type of data used is clearly described in chapter six. This data set covers different aspects of life like: educational, religious, political, socio-economic, culture, sports and so on. This is because there is no any standard corpus developed yet for Afaan Oromo. Information retrieval needs standard corpus for testing and making experimentation.

Most of the websites from which these data are collected are attached as appendix (Appendix I) at the end of this paper.

4.2. Developmental tools

In this study a single personal computer processor of intel core i3 with 2.4GHz speed, Microsoft windows 7 professional operating system, 500GB Hard Disk capacity and 4GB of RAM is used. For the development of the search engine, we have utilized the following tools.

JDK 1.8.0.91: Since all components of our apache solr is developed using java programming language, JDK (java development kit) version 1.8.0.91 is utilized and installed on windows environment for implementing the various components of the solr**.** It is difficult to use solr, if our machine does not have JDK; that is why we use it.

NetBeans 8.0.1: we have used NetBeans to implement BM25 similarities in apache solr. It used for writing, running and compiling any java codes that we need to connect to Lucene.

Apache solr (solr-6.0.0): Apache Solr was used as windows environment and programming language for the purpose of this research. Apache Solr is an open source search platform built upon a Java library called Lucene (Dikshant, 2015). Solr is a popular search platform for Web sites because it can index and search multiple sites and return recommendations for related content based on the search query's taxonomy. Apache Solr is a fast open-source Java search server. Solr enables to easily create search engines which searches websites, databases and files. It is also highly reliable, scalable and fault tolerant, providing distributed indexing, replication and load-balanced querying, automated failover and recovery. Solr is NOT a relational database like MySQL or Oracle. Rather, it can work in tandem with databases to power web applications.

Apache solr support both ***query effectiveness*** and ***query efficiency*** (Vu, 2015) which motivate the researcher of this study to use this tools. *Query efficiency* is a performance metric to measure how fast a search engine can retrieve data. Retrieval time is defined as the period after the query is accepted into the search engine until the result is returned. *Query effectiveness* is a performance metric to measure how relevant is the search result to the query terms.

Generally, in addition to the above most important uses of solr, the followings are also the main reasons why apache solr is selected as developmental tools for this study:

- ✓ Solr enables us to easily create search engines.
- ✓ Fast enough relative to comparable other general-purpose search libraries.
- ✓ Fast to improve overall indexing performance.
- ✓ Contains memory improvements that leverage some more-advanced data structures.

- ✓ It is also highly reliable, scalable and fault tolerant, providing distributed indexing, replication and load-balanced querying.

4.2.1. Tools used for Text Extraction

Pages on the web can contain text, audio, video, image, etc. The page can also be in different formats, such as text, Microsoft word, excel, HTML, PDF and XML. This types of formats also may contain image, audio, video. So, it needs the means to index them. Because we are using Lucene solr for indexing the documents which contain the above data types (image, audio, video), it needs helping tools that extract from different document formats (Hailay, 2013). Since Lucene does not have its own method that does this task, it needs third-party tools. Therefore, we have reviewed open source text extraction tools for the above mentioned document formats only. For this purpose, PDFBOX and WORD EXTRACT are used.

PDFBOX: is free open source library that extracts text from PDF format web documents. PDFBOX version 2.0.2 is used for this purpose because it is Java based library that includes classes that work with Lucene particularly.

WORDEXTRACT: This is simple and optimized for extracting and used to extract Microsoft word documents which are plain-text and difficult for the lucene to index.

4.3. Possible Retrieval Models to IR System

To search relevant documents, we need a retrieval model. Because, it is very important to reason out the reason why one document is ranked above the other. So, a retrieval model is a framework for defining the retrieval process by using mathematical concepts. It provides a blueprint for determining documents relevant to a user need, along with reasoning of why a document ranks above another. Each model uses a different approach for determining the similarity between query and documents (Dikshant, 2015). There are many information retrieval models. From them some are:

4.3.1. Boolean model

It is possible to understand from its name that; Boolean model classifies documents as either the documents match or don't match. From the existing information retrieval models, Boolean model is a simple model which works based on set theory and Boolean algebra (Hiemstra, 2009). Additionally, in Boolean model, the terms are extracted from documents while indexing and the index is created. Hence, documents are as a set of terms. While searching, the user query is parsed to determine the terms and form a Boolean expression. The user query can contain the operators AND, OR, or NOT, which have special meanings in a Boolean expression—to identify conjunction, union, or disjunction, respectively. In Solr it uses + and - symbols to specify terms as "must occur" or "must not occur." Therefore, all the documents that satisfy the expression are deemed as matches, and everything else is discarded. For instance, the query **social** AND **economic** will produce the set of documents that are indexed both with the term social and the term economic, i.e. the intersection of both sets. Combining terms with the OR operator will define a document set that is bigger than or equal to the document sets of any of the single terms. So, the query **social** OR **political** will produce the set of documents that are indexed with either the term social or the term political, or both, i.e. the union of both sets.

An advantage of the Boolean model is that it gives (expert) users a sense of control over the system. This means the user know that which documents must retrieved. It is immediately clear why a document has been retrieved given a query. If the resulting document set is either too small or too big, it is directly clear which operators will produce respectively a bigger or smaller set. For untrained users, the model has a number of clear disadvantages. Its main disadvantage is that it does not provide a ranking of retrieved documents. The model either retrieves a document or not, which might lead to the system make rather frustrating decisions (Hiemstra, 2009). There is no the concept of partial match in Boolean model.

Generally speaking, Boolean model has the following drawbacks which discourage the researcher of this study to use this model:

- ✓ Retrieval based on binary decision criteria with no notion of partial matching
- ✓ Absence of grading scale or no ranking of the documents
- ✓ The Boolean model frequently returns either too few or too many documents in response to a user query
- ✓ Information need has to be translated into Boolean expressions which most users find awkward.

4.3.2. Vector space model (VSM)

Vector Space Model is a classic model of document retrieval based on representing documents and queries as vectors of index terms (Inkpen, 2005).Gerard Salton introduce VSM in which partial matching is possible in 1960s to oppose to Boolean model where no partial matching method is used. So, the problem in the Boolean model (problem of no partial matching) is solved by Gerard Salton.

In VSM, similarity measurements allow decisions to be made which documents are similar to each other and to keyword queries. According to (Salton & Buckley, 1988), VSM generates weighted term vectors for each document in the collection, and for the user query. Then the retrieval is based on the similarity between the query vector and document vectors. The output documents are ranked according to this similarity. The similarity is based on the occurrence frequencies of the keywords in the query and in the documents. The angle between query and document vectors are measured by using the cosine similarity measurement since both document and a user 's query is represented as vectors in vector space model. Since relevance is modeled via similarity between a document and a query in vector space model(VSM), VSM assume that if document vector V1 is closer to the query vector than another document vector V2, then the document represented by V1 is more relevant than the one represented by V2.

In VSM, to decide the similarity of the document to the given query, *term weighting* technique (tf *idf) is used. It is impossible to talk about term weight, if we do not have the concept of term frequency(tf) and inverse document frequency(idf). So, the following section describe about them.

Term frequency(tf):is the count of the term i in document j (the number of times a given term appears in that document). The term-frequency of a term is normalized by the maximum term frequency of any term in the document to bring the range of the term-frequency 0 to 1 (Datta, 2010).Terms appearing in larger documents would always have larger term frequency, so, this count is usually normalized to prevent a bias towards longer documents. Mathematically, the term frequency of a term i in document j is given by:

$$Tf_{i,j} = \frac{freq\ i,j}{Max\ (freq\ i,j)} \quad \text{..... Equation 4.1}$$

Where freq i,j is the count of how many times the term i appear in document j.

Inverse Document Frequency(IDF): idf is used to measure whether the terms are common or rare across all documents. Higher idf value is obtained for rare terms whereas lower value for common terms

$$Idf = \log_2 \left(\frac{\text{Total number of documents(N)}}{\text{Number of documents containing the given term(df)}} \right)$$

Document frequency (*df*): number of documents containing the given term.

$$Idf,i = \log_2(N/df) \quad \text{.....Equation 4.2}$$

where df is the number of documents in which the term i occurs,

N is the total number of documents and

idf is inverse document frequency.

Then the product of equation 4.1 and equation 4.2 is called term weighting.

Term weighting (tf*idf) =Tf i,j $*$ $\log_2(N/ df)$……Equation 4.3

After calculating the term-weights, document and query vectors in k dimension (where k is number of index terms in vocabulary), we need to find the similarity between them. The widely used measure of similarity in vector space model is called the cosine similarity. The cosine similarity between two vectors $\vec{d}_j$ (the document vector) and $\vec{q}$ (query vector) is given by:

……. Equation 4.4

$$similarity(\vec{d_j}, \vec{q}) = \cos\theta = \frac{\vec{d_j} \cdot \vec{q}}{|\vec{d_j}|\,|\vec{q}|} = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

Where θ is the angle between the two vectors,

wi,j is the term-weight for i-th term of j-th document

w1, q is the term weight assigned to i-th term of the query

Since wi,j ≥ 0 and wi,q ≥ 0 for all i,j and q, the similarity between d,j and 8q varies from 0 to 1.

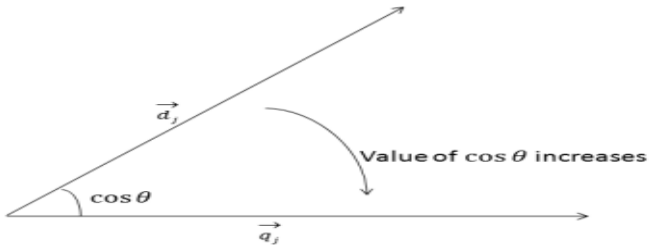

Figure 4.1: Cosine Similarity

Source: Ranking in Information Retrieval (Datta, 2010)

The main advantages of cosine similarity are:

- Simple model when compared to others

- The cosine similarity measure returns value in the range 0 to 1. Hence, partial matching is possible.
- Ranking of the retrieved results according to the cosine similarity score is possible.

Although vector space model has the above advantage, it has also some disadvantages. For instance: *assumption of term independence*, text VSM can't deal with lexical ambiguity and variability. e.g.: "he's affected by AIDS" and "HIV is a virus" don't have any words in common. So, in the Text VSM the similarity is 0: these vectors are orthogonal even though the concepts are related. on the other hand, similarity between "the laptop has a virus" and "HIV is a virus" is not 0: due to the ambiguity of the word "virus".

VSM also *un able to convey any relationship including the Boolean relationship, existing between the terms*. It theoretically assumes that terms are statistically independent

There are *no predictions* about the techniques for effective and *long documents are poorly represented* because they have poor similarity values. In addition to this, in VSM the order in which the *terms appear in the document is lost* in the vector space representation. And weighting is intuitive but not very formal.

4.3.3. Probabilistic model

Probabilistic approach which was proposed by Maron and Kuhans in 1960 (Albujasim, 2014), computes the probability that the document is relevant to the query and ranks the documents according to their probability of being relevant to the query (Inkpen, 2005). Several versions of probabilistic models are available today. BM1 is the first version of the model which was introduced by Robertson-Sparck Jones in 1976. The refinement of the first version (i.e. BM1) is replaced by BM5 which is again proposed by Robertson and others in 1998. The concept of BM, Best Match, is to mean the document that has best similarity to the given query within the collections. The most widely used ranking algorithm among the alternative implementations provided by Lucene for probabilistic approach is BM25Similarity. This similarity is based on the Okapi BM25 algorithm (Dikshant, 2015). By default, Lucene uses vector space and Boolean model to rank the documents, however, it can also use BM25similarity which is one

of the most powerful ranking algorithm of OKAPI which is probabilistic model. Therefore, we can also use probabilistic approach in Lucene by changing the 'Default Similarity' to the 'BM25Similarity'.

4.4. Okapi BM25Similarities for probabilistic model with Lucene

According to (Vu & Huy, 2015), BM25 (where *BM* stands for *Best Match*) is a ranking function that sorts documents according to their relevance to a specific query, which is implemented with the Okapi information retrieval system first. It uses two free parameters k and b that distinguish it from other ranking methods. In probabilistic model the ranking method of okapi BM25 is calculated as follows.

Term frequency (TF): The number of times a given term appears in the documents.

$$TF(q_i, D) = 0.5 + 0.5\frac{f_{q_i,D}}{\max(f_D)} \quad \text{.....equation (4.5)}$$

Inversed document frequency (IDF): Since the document frequency measures how many documents a term appears in, Inverse document frequency (1/df) then measures how special the term is. Is the term is very rare (occurs in just one document)? Or relatively common (occurs in nearly all the document)?

$$IDF(q_i) = log\frac{N - |d(q_i)| + 0.5}{|d(q_i)| + 0.5} \quad \text{..... .....equation (4.6)}$$

Where:

N ➔Total number of documents and Df(qi)➔ number of documents containing term q

Weight:

$$weight_{TF.IDF} = TF \times IDF$$ ..... equation **(4.7)**

Similarity:

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{TF(q_i,D) \cdot (k+1)}{TF(q_i,D) + k(1 - b + \frac{b \cdot |D|}{avgDl})}$$ .....equation **(4.8)**

where:

Q ➔contains q1, q2, q3, ...qn

D ➔ interested documents (d1, d2,d3,.....dn)

Avgdl➔average length of all documents

K and b are the two free parameters that are used to distinguish this method from other ranking methods.

Note: b has value from 0 to 1; most of the time 0.75 is optimized

K is often ranged from [1.2,2.0], we can use 1.2

4.5. Steps and procedures to integrate BM25Similarities

The following sections shows the steps and procedures followed to integrate BM25Similarities into Lucene.

4.5.1. Integration of BM25similarities into Lucene (for Java code)

The default sorting mechanism in Solr is known as relevance ranking. Solr calculates the relevance score of each document in the result set and ranks the documents so that the highest scoring documents move to the top during a search, Scoring is the process of determining how relevant a given document is with respect to the input query. The default scoring mechanism in solr is the mix of both *Boolean* and *vector space model*, where *Boolean* is used to show

documents that match the query set, and the *vector space model* is used to calculate the score of each and every document in the result set (Kumar, 2015). In addition to the VSM, the Lucene scoring mechanism supports a number of pluggable models, such as probabilistic models and language models. In this study we have plugged-in okapi BM25similarity into solr to measure the relevance ranking.

For modern full-text search collections, a model should pay attention to term frequency and document length. Okapi BM25 is a probabilistic model that incorporates term frequency and length normalization. BestMatch25 (BM25 or Okapi) is sensitive to these quantities. BM25 is one of the most widely used and robust retrieval models (Sojka & Schütze, 2015).

The sample of java code used for the integration of BM25similarity to plug-in into solr for the purpose of this study is attached at the end of this paper as appendix (appendix IV). However, since we take only the .**JAR file** of this code, it is not enough; so, we have to follow the step described in the section 4.5.2.

4.5.2. Integration of BM25similarities into Lucene (for xml)

While we are using apache solr, we have to create a core or collections first. The core created contains Afaan Oromo data which is called *Afaan-Oromo-Core*. So, in the core the following four (4) main important files created before starting to import our data.

- ✓ *Schema.xml*: -defines the fields to be indexed and the type for the field and similarity.
- ✓ *Solrconfig.xml*: - Defines indexing options, highlighting, spellchecking autosuggestions and various other configurations.
- ✓ *Solr.xml*: - specifies configuration options for our Solr server instance
- ✓ *Core. Properties*: - a simple Java Properties file where each line is just a key=value pair, e.g., name=Afaan-Oromo-Core

Therefore, schema.xml file is where we have to define our *BM25Similarity* to use probabilistic approach with apache solr. The following is the steps to do this:

Step 1:

We have to register the *BM25SimilarityFactory* in *schema.xml* by using the similarity element. Globally, only one similarity class should be defined. If no similarity definition is available in schema.xml, by default, *Default Similarity* is used.

<schema>

…… ➔Schema. xml codes are defined here

……

<similarity class="solr.BM25SimilarityFactory" />

</schema>

Step 2:

Specify the configuration parameters supported by BM25SimilarityFactory in the configuration, solrconfig.xml; to do so,

Assume that K, b and *discount Overlaps* are parameters and their data type is float, float and Boolean respectively. The value of k ranges from 0 to 1 and b from 1.2 to 2.0. However, the default value of K =1.2, b= 0.75 and *discount Overlaps* = *true*. Let us see from the following:

Parameter	Type	Description
k1	Float	This optional parameter controls the saturation level of the term frequency. The default value is 1.2. A higher value delays the saturation, while a lower value leads to early saturation.
b	Float	This optional parameter controls the degree of effect of the document length in normalizing the term frequency. The default value is 0.75. A higher value increases the effect of normalization, while a lower value reduces its effect.
discountOverlaps	Boolean	This optional parameter determines whether the overlapping tokens (the tokens with 0 position increments), should be ignored while computing the norm based on the document length. The default value is true, which ignores the overlapping tokens.

From: Dikshant Shahi book's (i.e. Apache Solr-A Practical Approach to Enterprise Search), (2015)

Then; we use the following in the solrconfig.xml

<? xml version="1.0" encoding="UTF-8"?>

<config>

…… ➔there are a lot of solrconfig.xml file here

……

<similarity calss="BM25similarities">

<float name="k1">1.2</float>

<float name="b">0.75</float>

</similarity></config>

Step 3:

Re index the documents.

4.6. Why probabilistic model?

The first and most important reason that motivate this study to use probabilistic approach or model is to examine the potential of probabilistic approach technique in the retrieval of Afaan Oromo text. In addition to this, according to (Chu, 2003) probabilistic approach also has the following advantages.

- ❖ Includes term dependencies and relationships (i.e. the occurrence of one event affect the other) in its operation

- More user friendly than Boolean model because, probabilistic model does not employ the Boolean logic facility that many users find hard to apply.

- The query-document similarity measurement is determined by the model itself instead of some arbitrary decision as in VSM.
- The model is able to take advantage of feedback information to develop well founded methods; self-improving features adds another positive element to the probabilistic model.

CHAPTER FIVE

SYSTEM DESIGN AND DEVELOPMENT

According to (Grainger & Potter, 2014), Solr provides a number of important features that help us deliver a search solution that is easy to use, intuitive, and powerful. Solr is selected as developmental tool for this study, so that it is not so much difficult to do the following features.

Pagination and Sorting

Rather than returning all matching documents, which is difficult, for example, when we have many documents, Solr is optimized to serve paginated requests, in which only the top N documents are returned on the first page. If users don't find what they're looking for on the first page, they can request subsequent pages using simple next request parameters.

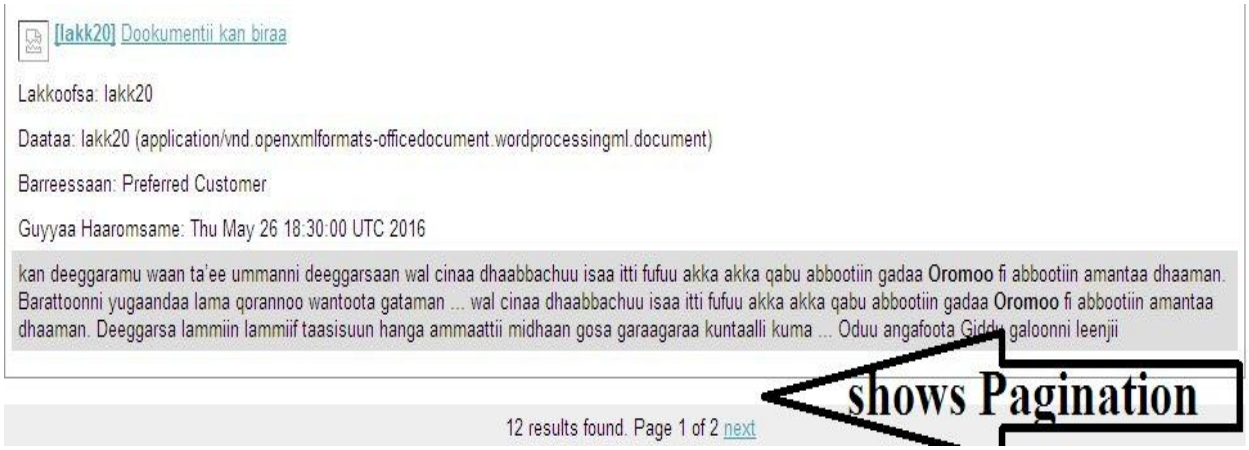

Figure 5.1: solr pagination

Autosuggest

Autosuggest allows users to see a list of suggested terms and phrases based on documents in the index. Autosuggest features allow a user to start typing a few characters and receive a list of suggested queries with each keystroke. This reduces the number of incorrect queries. It also gives users examples of terms and phrases available in the index. Example, as a user types

'Bil…….', autosuggestion feature can return suggestions like "Bilisummaa", "bilbila", "Billaacha", 'Bilchaataa' or 'Bilcheessi' and so on.

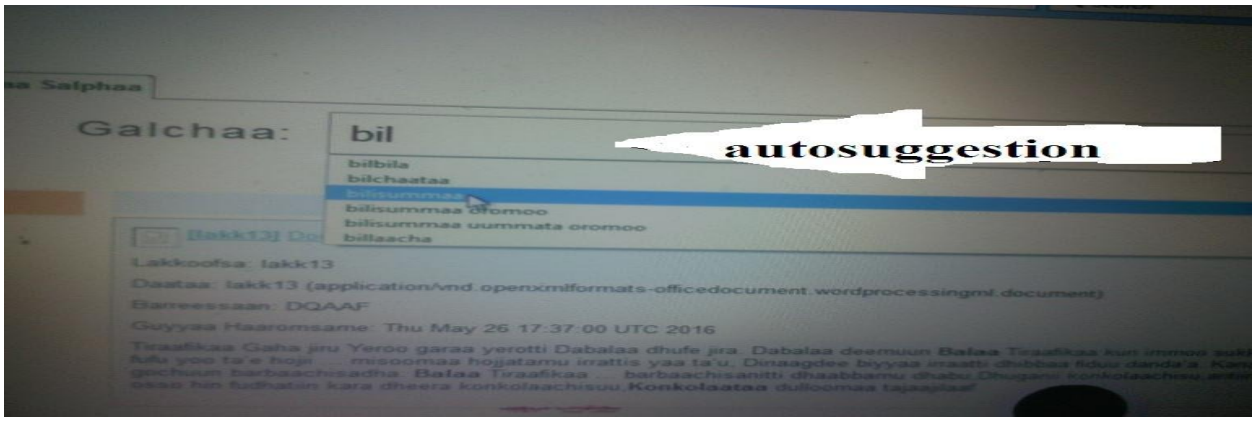

Figure 5.2: Afaan Oromo word Autosuggestion while searching for a document

Spell-Checker

In the age of digital devices (mobile devices), spelling correction support is essential. Solr's spell checker supports two basic modes:

■ **Autocorrect** —Solr can make the spell correction automatically, based on whether the misspelled term exists in the index.

■ **Did you mean (**in Afaan Oromo used as: **'kan jechuu barbaaddan kanaadhaa?')** —Solr can return a suggested query that might produce better results so that we can display a hint to our users, such as "Did you mean Bilisummaa?" if the user typed' blsummaa'.

Figure 5.3: snapshot for Afaan Oromo spellchecker using solr

Hit Highlighting

When searching documents that have a significant amount of text, we can display specific sections of each document using Solr's hit-highlighting feature. Most useful for longer format documents, hit highlighting helps users find relevant documents by highlighting sections of search results that match the user's query. Sections are generated dynamically based on their similarity to the query. The following is the home page for the designed system.

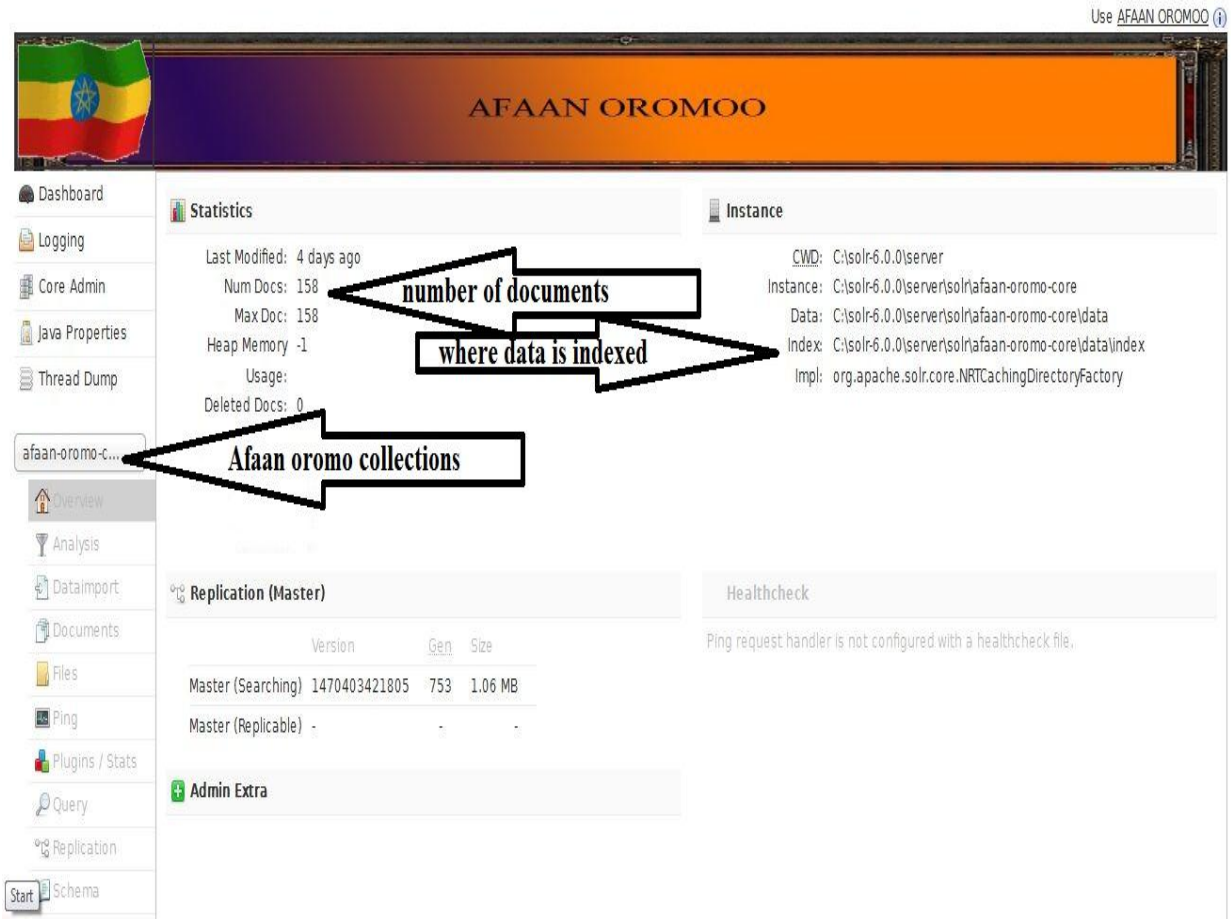

Figure 5.4: snapshot of Afaan-Oromo-Core

When we browse the above home page by using ***localhost:8983/solr/afaan-oromo-core/browse***, the search box which is used to insert Afaan Oromo query is displayed as follows. For example, if we search for the query *'bilisummaa'*, which is freedom in English, it looks like the following.

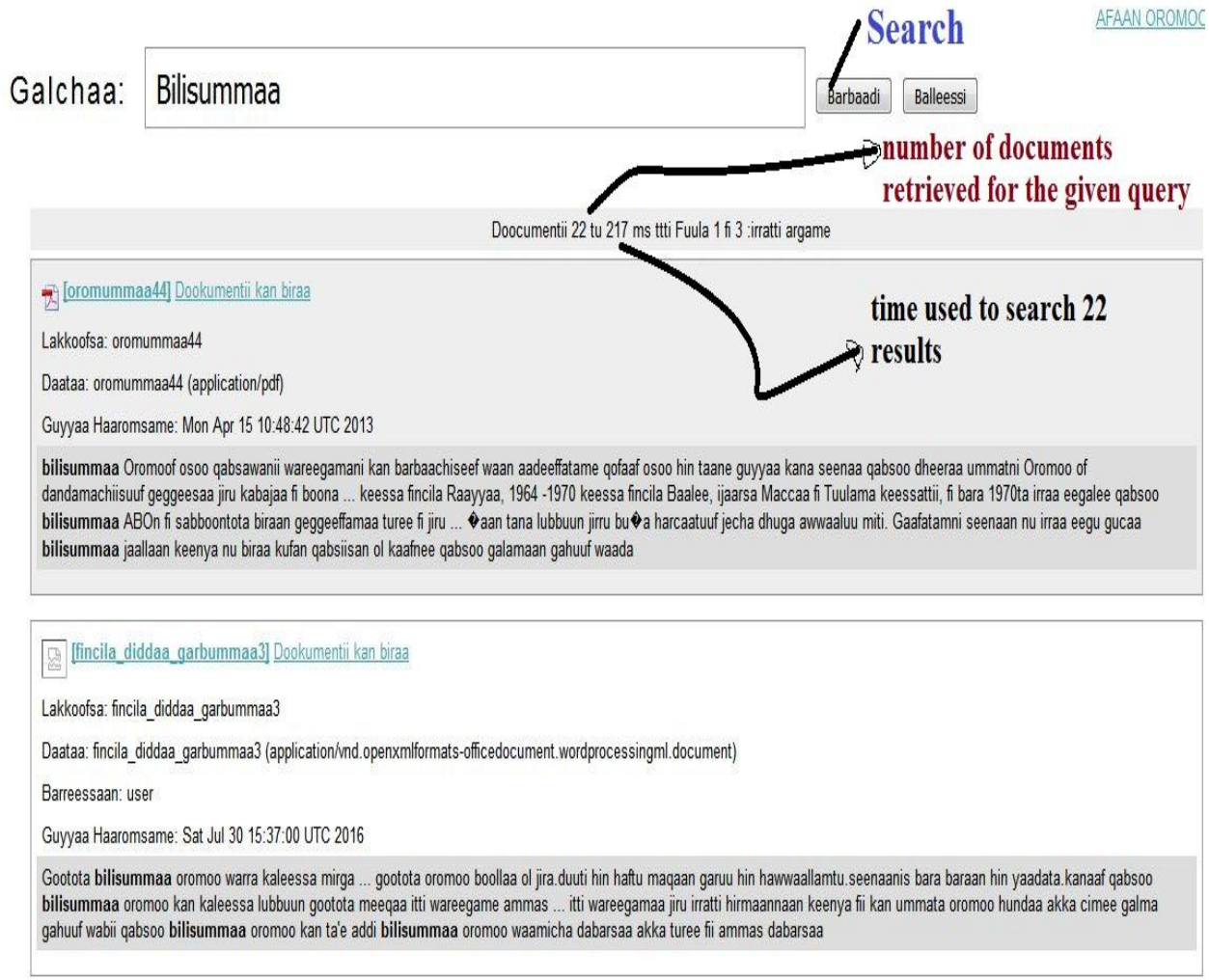

Figure 5.5: Search interface for Afaan Oromo

5.1. Lexical processing

Tokenization: Tokenization is the process (task) of chopping it up into pieces, which is called tokens, and also at the same time it is the process of throwing away some characters,

illuminating punctuation and special characters in a given character sequence and in a document unit (Albujasim, 2014) and (Gezehagn, 2012). Tokenization of Afaan Oromo is the same with that of English except apostrophe (‘) which is considered as part of word in Afaan Oromo. If we have the original document *‘Oromiyaan qabeenya uumamaa bareedaa qabdi*’, then the tokens are: *‘Oromiyaan’,’qabeenya’,’uumamaa’, ‘bareedaa’, ‘qabdi’*. However, there are challenges related to tokenization in every language. The first challenge is differentiating single word from compound word. For example, if we take the word ‘*harka-qalleessa*’ to mean ‘*poor*’; if it is tokenized as ‘*harka*’ and ‘*qalleessa*’, the meaning of the word will be changed. Because we cannot separate such like words from one another in Afaan Oromo. Such like challenge is not only in Afaan Oromo, but also in other languages such as English. Example, words like*: San Francisco, Hewlett-Packard Versus Hewlett and Packard, Addis Ababa Versus Addis Ababa* are common in English. The following is the tokenizer that is used for the purpose of this study.

```
<! -- ===============Afaan Oromo tokenizer ================== -->

<fieldType name="text_ao" class="solr.TextField" positionIncrementGap="100">

<analyzer>

<tokenizer class="solr.StandardTokenizerFactory"/>

<filter class="solr.ApostropheFilterFactory"/>

<filter class="solr.LowerCaseFilterFactory"/>

<filter class="solr.StopFilterFactory" ignoreCase="true" words="lang/stopwords_ao.txt" />

<filter class="solr.SynonymFilterFactory" synonyms="punctuation-whitelist.txt"
ignoreCase="true" expand="false"/>

</analyzer>

</fieldType>
```

Lists of Afaan Oromo stop words

Lists of punctuations to be tokenized

Code 5.1: Afaan Oromo Tokenizers

Stop words removal: Stop words are Some extremely common words that would appear to be of little value in helping select documents matching a user need and their removal (hopefully) does not change the meaning of a documents (Rababah, 2011). Examples of some Afaan Oromo stop words are: Aanee, agarsiisoo, akka, akkam, akkasumas, akkuma, ala, alatti , amma , ammo, ammoo , ishiitti , isii ,isiin ,isin , isini ,isinii ,isiniif ,isiniin ,isinirraa ,isinitti ,ittaanee ,itti, yammuu, yemmuu , yeroo , yoomii , yommuu ,yoo , yookaan , yookiin , yookiinimmoo , yoom.

Afaan Oromo stop word lists which are taken from Gezahagn Gutema's paper and others are attached as appendix at the end of this paper. (i.e. Appendix II). Lists of Afaan Oromo stop words (*stopwords_ao.txt*) is inserted in the managed_schema.xml (simply, schema.xml). This help apache solr to index the documents by removing these stop words.

Stemming: stemming is reducing a word, inflectional forms and sometimes derivationally related forms of a word to a common base form into its root or stem. Stemming means transforming a word to its morphological root (stem) and indexing that instead. It is often removing inflectional and derivational morphology. For example, the words "*compute*", "*computer*", "*computation*", "*computers*", "*computed*" and "*computing*" might all be indexed as "*compute*" (Ajit & Biswas, 2011).In Afaan Oromo, for example, the word 'barattootarratti','barattootaaf', 'barattootaan', 'barattootni' is indexed to 'barattoot'. Stemming can help us by handling problems related to inflectional and derivational morphology which makes words with similar root to retrieve. Even if stemming has advantages, it has also some disadvantages. For example, some terms might be over stemmed, and this can change the meaning of terms in the document.

Although it is not hundred percent perfect, the code that we have used for the stemming purpose is attached as appendix (appendix V) at the end of this research. In that code *"C:\\solr-6.0.0\\server\\solr\\afaan-oromo-core\\data\\index"* is the file path where the data to be stemmed is found and the list of suffixes which must be removed to get the root word is found at: *"C:\\solr-6.0.0\\bin\\afaan oromo stemming\\stemming.txt"*.

afaanOromoStemming(word) {

RemoveCase(word);

return;}

RemoveCase(word) {

if (word ends with “-tii”) then remove “-tii” return;

if (word ends with “-ttii”) then remove “-ttii” return;

if (word ends with “-ittii”) then remove “-ittii” return;

if (word ends with “-f”) then remove “-f” return;

if (word ends with “-if”) then remove “-if” return;

if (word ends with “-iif”) then remove “-iif” return;

if (word ends with “-ota”) then remove “-ota” return;

if (word ends with “-oota”) then remove “-oota” return;

if (word ends with “-an”) then remove “-an” return;

if (word ends with “-e”) then remove “-e” return;

if (word ends with “-uu”) then remove “-uu” return;

if (word ends with “-ete”) then remove “-ete” return;

if (word ends with “-ete”) then remove “-ete” return;

if (word ends with “-eti”) then remove “-eti” return;

if (word ends with “-eti”) then remove “-eti” return;

if (word ends with "-wwan") then remove "-wwan" return;

if (word ends with "-lee") then remove "-lee" return;

if (word ends with "-[aeiou]") then remove "-[aeiou]" return;

return;}

Code 5.2: Afaan Oromo Stemming

5.2. Evaluation techniques

This study was involved in designing Afaan Oromo text retrieval system and evaluating its performance. In this study, the researcher has prepared corpus, constructed queries and relevance judgment is made for evaluating effectiveness of the work to measure the precision's and recall's effectiveness. The most commonly used techniques to measure relevance of an IR systems are recall and precision. Therefore, a good IR system should retrieve as many relevant documents as possible (i.e. have a high recall), and it should retrieve very few non-relevant documents (i.e. have high precision).

Recall is the fraction of relevant document that are retrieved; precision is the fraction of retrieved documents that are relevant. In this study, probabilistic approach (*BM25similarity*) is used as similarity measurement method.

5.3. System architecture for Afaan Oromo

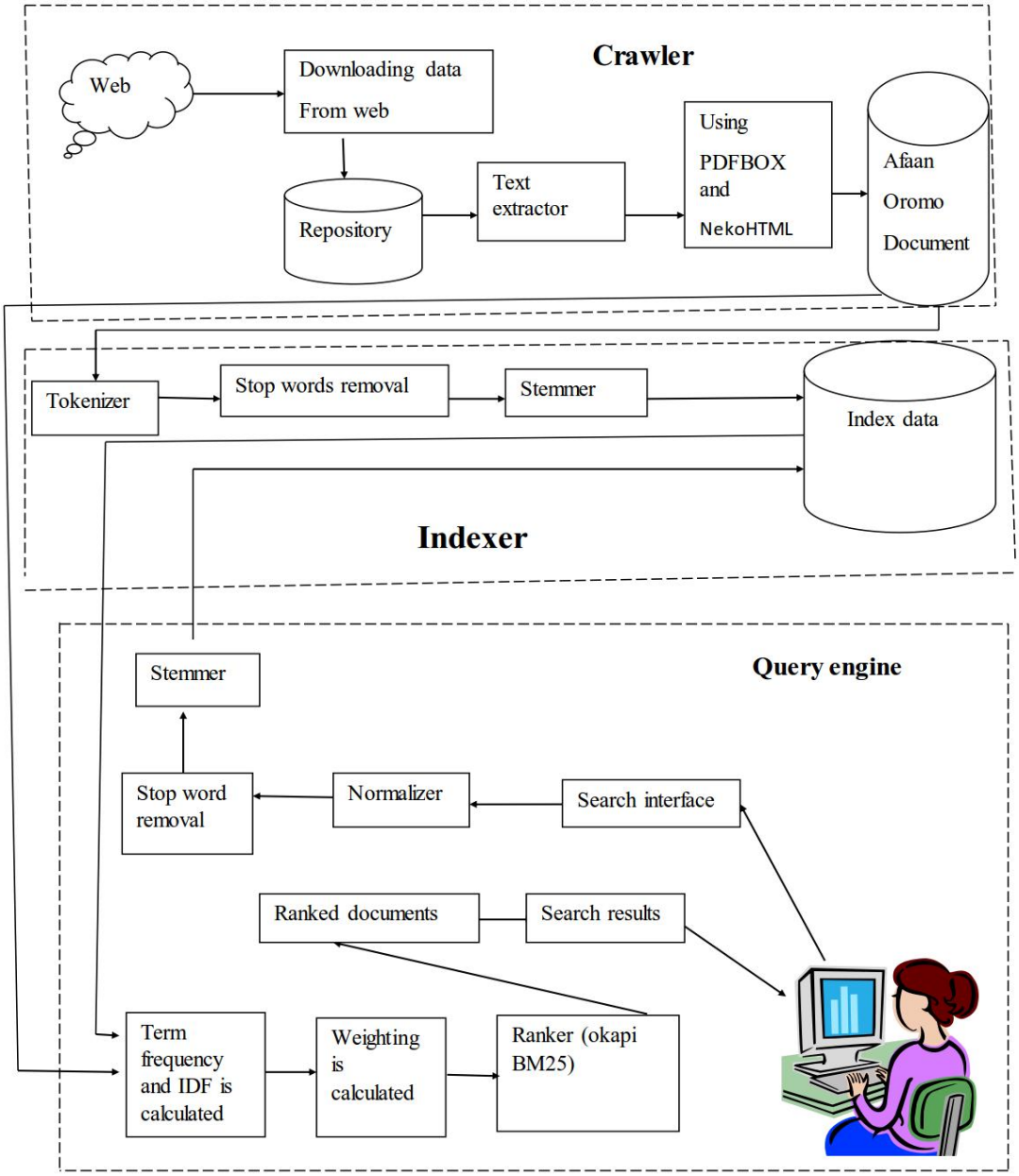

Figure 4.6: Afaan Oromo system architecture

CHAPTER SIX

EXPERIMENTATION AND DISCUSSION

To test this study experimentally, the performance of the IR system using corpus which is called in our case *Afaan-Oromo-core* is used. We have collected data from different websites that contain Afaan Oromo words and statements. These websites are attached as appendix (appendix I) at the end of this study next to references. The corpus contains different subjects like politics, education, culture, religion, history, social, health, economy and other events as we have discussed in chapter four. This corpus also contains different formats of data which seems like the following.

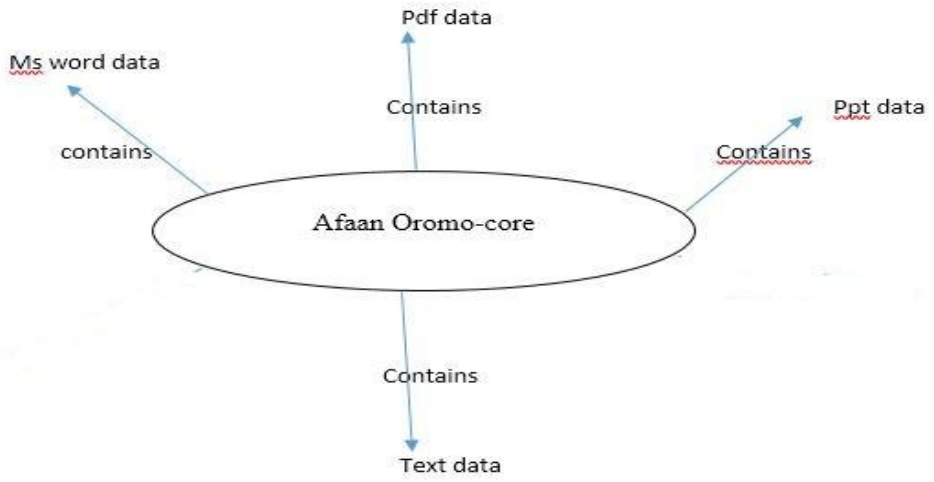

Figure 6.1: Types of data used for Afaan Oromo Corpus

The data which is in (Microsoft word, power point, portable digital format, extensible markup language,) and any other has been converted to *text* format since encrypted data is not supported by solr. But if it is not encrypted solr can index it. The total size of the collections used in this study was thirty-four point three megabytes (34.3 MB).

6.1. Query selection

To show the performance of the system with the given corpus, query selection is very important. In this study 13 queries are randomly selected to check the performance of the system. The average value of 13 queries are taken to judge the precision and recall of the system. The total number of documents we have been used were only 158, since it needs time and effort to collect more documents.

Num.of queries	Queries	Number of documents which are:	
		Relevant	Irrelevant
Query 1	Rakkoo Bulchiinsa gaarii	28	130
Query 2	Taphoota aadaa uummata oromoo	3	155
Query 3	Ayyaana irreechaa	5	153
Query 4	Macaafa qulqulluu	5	153
Query 5	Aadaa uummata oromoo	11	147
Query 6	Dorgommii guutuu Itoophiyaa	7	151
Query 7	Sirna gadaa oromoo	10	148
Query 8	Atileetota itoophiyaa	7	151
Query 9	Geerarsa oromoo	4	154
Query 10	Magaalaa guddoo Oromiyaa	5	153
Query 11	Fayyaan faaya	5	153
Query 12	Bilisummaa oromoo	10	148
Query 13	Sochii diddaa gabrummaa	7	151

Table 6.1: relevance judgement

6.2. Method of System evaluation

It is known that measuring or evaluating the performance and accuracy of the system is very important after IR system is designed. According to (Singhal, 2001), there are two main things to measure in IR system; these are: effectiveness of the system and its efficiency. In this study the effectiveness is measured by precision and recall. It is known that precision and recall are the main important ways of evaluating an IR system (Kumar, 2015). In addition to precision and recall, F-measure which is good for test of accuracy have been used.

$$Precision = \frac{number\ of\ retrived\ that\ are\ relevent}{tota\ l\ number\ of\ retrived\ documents} \ldots\ldots (6.1)$$

$$Recall = \frac{number\ of\ retrived\ that\ are\ relevent}{total\ number\ relevent\ in\ the\ collection} \ldots\ldots (6.2)$$

6.3. Experimentation

In the evaluation of this experiment, free text query (query with two words or more) is used. Because, One-word query makes the output ambiguous for morphologically complex language; so, we used free text query. Free text query also makes it easier for the system to determine the user 's information needs. To calculate the average precision and average recall for the manually selected 13 queries, first we have identified the total number of documents in the collection, the total number of retrieved documents by the system and the retrieved relevant documents. After that for every query the precision and recall is calculated as shown in table 6.2. At the end, the average precision and average recall is calculated as follows and the value of precision and recall is in percent.

Average Precision = The summation of individual precision

The total number of queries

Average Recall = the summation of individual recall

The total number of queries

No Query	**Queries**	**Total-Relevant In the collections**	**Total Retrieved**	**Relevant-Retrieved**	**Precision In %**	**Recall In %**
Q 1	Rakkoo Bulchiinsa gaarii	28	33	28	84.84%	100%
Q 2	Taphoota aadaa uummata oromoo	3	3	3	100%	100%
Q 3	Ayyaana irreechaa	5	6	4	66.7%	80%
Q 4	Macaafa qulqulluu	5	7	5	71.4%	100%
Q 5	Aadaa uummata oromoo	11	16	11	68.75%	100%
Q 6	Dorgommii guutuu Itoophiyaa	7	7	6	85.7%	85.7%
Q 7	Sirna gadaa oromoo	10	17	8	47.05%	80%
Q 8	Atileetota itoophiyaa	7	5	5	100%	71.4%
Q 9	Geerarsa oromoo	4	8	4	50%	100%
Q 10	Magaalaa guddoo Oromiyaa	5	7	4	57.14%	80%
Q 11	Fayyaan faaya	5	5	4	80%	80%
Q 12	Bilisummaa oromoo	10	15	8	53.3%	80%
Q 13	Sochii diddaa gabrummaa	7	6	5	83%	71.4%
Average					**72.91%**	**86.8%**

Table 6.2: Experimentation result before the synonym is added

Where,

Total-Relevant ➔total number of documents that are relevant to the query in the collection

Retrieved➔ total number of documents retrieved by the system

Relevant-Retrieved➔total number of retrieved documents that are relevant to the query

As it is shown in Table 6.2, the average recall and precision obtained from this study is 86.8% and 72.91% respectively. In the study, 100% (i.e. 1 when we change to integer) precision and recall is obtained for some queries. The reason for such event is explained as follows:

Precision =1 (perfect precision) means every result returned by the search was relevant, but there may be other relevant documents that were not part of the research result. In other statement, perfect precision means every result retrieved by the search was relevant (but says nothing about whether all relevant documents were retrieved).

Recall =1 (perfect recall) means all relevant documents were retrieved by the search but says nothing about how many irrelevant documents were also retrieved.

Recall measures how well the search system finds relevant documents; precision measures how well the system filters out the irrelevant documents, Precision is the ratio of the number of relevant documents retrieved to the total number retrieved. Recall is the ratio of the number of relevant documents retrieved to the total number of documents in the collection that are believed to be relevant. Recall considers the total number of relevant documents (Borhade & Agarkar, 2014).

In addition to this, (Kumar, 2015) also described that, as precision increases, recall decreases and vice versa. This is called a trade-off between precision and recall as they are inversely related which is also true as the result of this study shows. Depending on this, Query 5(*aadaa uummata oromoo*) registered better recall (100%) but lower precision (68.75%). This means that all relevant documents were retrieved by the system irrespective of the irrelevant document included in the result set to the given query. Additionally, there are many documents which talks about 'uummata Oromoo' (Oromo people) and all those documents are retrieved, but the

relevant retrieved documents were not many. So, when small relevant retrieved documents (11) is divided by total retrieved documents (16), the result become low.

The result registered in Query 10 (*magaalaa guddoo oromiyaa*) in English (The capital city of Oromia) is also because of the word variation of the word 'guddoo' (capital). In Afaan Oromo the query '*magaalaa guddoo Oromiyaa'* can be written as: '*magaalaa beekamaa Oromiyaa*',' *magaalaa gudditii Oromiyaa'*, '*magaalaa teessoo mootummaa naannoo Oromiyaa*' and so on. So, this can be improved when the lists of synonymy are inserted. Synonymy can increase recall by decreasing precision. Because, when lists of synonymy are added more number of documents are retrieved by the system.

F -measure

F-measure was used so that the system increase recall by decreasing precision and sometimes vice-versa; there is a precision-recall tradeoff (for example, recall can be increased by simply retrieving more relevant documents, but the precision will go down). The F-measure combines precision and recall, taking their harmonic mean. The F-measure is high when both precision and recall are high. The formula is:

$$\mathrm{F-measure} = \frac{2PR}{P+R}$$

Where; F-measure➔stands for a measure of a test's accuracy

P -precision,

R-recall

The evaluation of the F-measure for the above average precision and recall is calculated as:

$$F = \frac{2(0.7291)\,(0.868)}{0.7291 + 0.868} = \frac{1.2657}{1.5971} = 0.7925$$

The F-measure assumes values in the interval [0,1]. It is 0 when no relevant documents have been retrieved, and 1 if all retrieved documents are relevant and all relevant documents have been retrieved. The F-measure result obtained for this study is 0.7925 which indicates that the accuracy of the system is 79.25%. Therefore, it is possible to have a search for Afaan Oromo text retrieval system using probabilistic approach.

6.4. Effects of Synonyms and polysemy in Afaan Oromo

In most of natural languages including Afaan Oromo, the problem of synonym is very challenging task. Because, the absence of synonym has effect on the performance of the IR system. In this study we have made attempt to solve this problem for Afaan Oromo. There are two ways to specify synonym mappings in our collections. These are:

- Separate lists of the two words with the symbol "=>" between them. If the token matches any word on the left, then the list on the right is substituted. The original token will not be included unless it is also in the list on the right.

  Example: gaarii=>misha

- A comma-separated list of words. If the token matches any of the words, then all the words in the list are substituted, which include the original token.

  Example: gaarii, misha,bayeessa,dansaa

Synonym mappings can be also used for spelling correction too.

Example: oromiya => Oromiyaa

Oromiya => oromiyaa

ormiya => Oromiyaa

when there is no synonymy in the developed system, for example, when the query 'ormiya' is inserted, the system responds error message; but after we have inserted synonymy, the system

displays all documents which contain the word ‘oromiyaa’. So, it can improve the performance of the system.

Lists of Afaan Oromo synonymy (synonyms.txt) file is located under the folder C:\solr-6.0.0\server\solr\afaan-oromo-core\conf to add the synonym for our data. Some lists of synonymy are attached at the end of this study as (appendix III) but the full lists are in the above folder since it took many pages. The following is where the lists of synonymy(synonymys.txt) were inserted.

```
<fieldType name="text_general" class="solr.TextField" positionIncrementGap="100">
    <analyzer type="index">
      <tokenizer class="solr.StandardTokenizerFactory"/>
      <filter class="solr.StopFilterFactory" ignoreCase="true" words=" stopwords_ao.txt" />
      <filter class="solr.LowerCaseFilterFactory"/>
    </analyzer>
    <analyzer type="query">
      <tokenizer class="solr.StandardTokenizerFactory"/>
      <filter class="solr.StopFilterFactory" ignoreCase="true" words=" stopwords_ao.txt" />
      <filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt"
ignoreCase="true" expand="true"/>
      <filter class="solr.LowerCaseFilterFactory"/>
    </analyzer>
  </fieldType>
```

6.5. Results after lists of synonymy are added to the document

The following table shows the effect of Afaan Oromo synonymy on system performance.

No Query	**Queries**	**Total-Relevant in the collections**	**Total Retrieved**	**Relevant-Retrieved**	**Precision in %**	**Recall in %**
Q 1	*Rakkoo Bulchiinsa gaarii*	*28*	*33*	*28*	*84.84%*	*100%*
Q 2	*Taphoota aadaa uummata oromoo*	*3*	*3*	*3*	*100%*	*100%*
Q 3	Ayyaana irreechaa	5	6	4	66.7%	80%
Q 4	Macaafa qulqulluu	5	7	5	71.43%	100%
Q 5	*Aadaa uummata oromoo*	*11*	*21*	*11*	*52.38%*	*100%*
Q 6	Dorgommii guutuu Itoophiyaa	7	7	6	85.7%	85.7%
Q 7	Sirna gadaa oromoo	10	17	8	47.1%	80%
Q 8	Atileetota itoophiyaa	7	5	5	100%	71.4%
Q 9	Geerarsa oromoo	4	8	4	50%	100%
Q 10	*Magaalaa guddoo Oromiyaa*	*5*	*10*	*5*	*50%*	*100%*
Q 11	Fayyaan faaya	5	5	4	80%	80%
Q 12	Bilisummaa oromoo	10	21	8	40%	80%
Q 13	*Sochii diddaa gabrummaa*	*7*	*7*	*7*	*100%*	*100%*
Average					**71.39%**	**90.5%**

Table 6.3: Experimentation result after the addition of synonymy

As it is shown in ***Table 6.3*** the addition of synonymy has effects on information retrieval system performance. As the result indicates after the synonymy is added the average number of recall is increased from 86.8% to 90.5% (i.e. 3.7% difference) and the average precision is decreased from 72.91% to 71.39% (i.e. 1.52% difference). This is because, when the lists of synonymy are added to the collections many documents can be retrieved. When many documents are retrieved and the relevant retrieved documents remain unchanged, the precision decreased. According to the experimentation made, the F-measure value after the addition of lists of synonymy is 79.82%, which is increased by 0.57 from the first. That means after the addition of synonymy the result is 79.82% and before the addition the result was 79.25%, so, the difference is 0.57. This shows that if standard thesaurus system is available for Afaan Oromo, the system performance is improved.

6.6. Answers of research question

As it is listed in section 1.2, there are three main research questions for this study. As the result of the experiment made, the answer is given to these questions as follows.

Research question 1: To what extent probabilistic approach enables to design effective Afaan Oromo text retrieval?

Answer: As the results of this study shows, the probabilistic approach has the ability to design effective Afaan Oromo text retrieval. 86.8% average recall and 72.91% average precision was obtained by using probabilistic approach and the value of recall is increased to 90.5% after lists of synonymy are added. However, after the addition of lists of synonymy, the value of precision is decreased by 1.52%.

The probability that a relevant document is not retrieved (also called False Negative or Type II error) which is given by: **1- Recall** (Sanderson, 2005) is used to know the error rate of the system. Therefore, to get the probability that the system is unable to retrieve relevant document, we have to subtract the recall obtained by the experiment from one (1). That means; 1-0.905 =0.095, where 0.905 the result of recall in this study. So, for this study: The probability that the developed system can retrieve relevant document is =90.5% and the probability that the developed system cannot retrieve relevant document is =9.5%.

Researchquestion 2: How can the problems related with polysemy and synonymy be handled and what is the simplest form of indexing to index Afaan Oromo text?

Answer: The problems related with polysemy and synonymy can be handled by using apache solr software easily. First preparing list of synonymy and polysemy is very necessary, then, integrating it in the schema.xml. For this study first lists of synonymy are prepared by the researcher and some of them are taken from Girma Debele's research paper, then, this list is saved in *C:\solr-6.0.0\server\solr\afaan-oromo-core\conf* where the schema.xml is found. After that, connecting to the schema file as follows.

```
</analyzer>  <analyzer type="query">

 <tokenizer class="solr.StandardTokenizerFactory"/>

 <filter class="solr.StopFilterFactory" ignoreCase="true" words=" stopwords_ao.txt" />
<filter class="solr.SynonymFilterFactory" synonyms="synonyms.txt" ignoreCase="true"
expand="true"/>

<filter class="solr.LowerCaseFilterFactory"/>  </analyzer>
```

Code 6.1: Synonymy integration sample code

The simplest form of indexing to index Afaan Oromo text

Since inverted file helps an IR system to rapidly control what documents contain a given set of words, how often each word appears in the document and it is an optimized data structure that can be used for information retrieval, inverted index is the simplest form of indexing for Afaan Oromo. It is a data structure for efficiently indexing texts by their words. Inverted index also used by apache solr. A list of words where each word is followed by the identifier of every text that contains the word can be viewed by an inverted file. The number of occurrences of each word in a text is also stored in this structure. The set of terms is called a vocabulary (V). Each term (t) in the inverted index has a pointer p (t) that points to the posting list, which is a list of all the occurrences of a term (t). In this work inverted index is used for sake of indexing.

Therefore, the simplest form of indexing to index Afaan Oromo documents is inverted index. The following figure shows the step to get inverted index for Afaan Oromo.

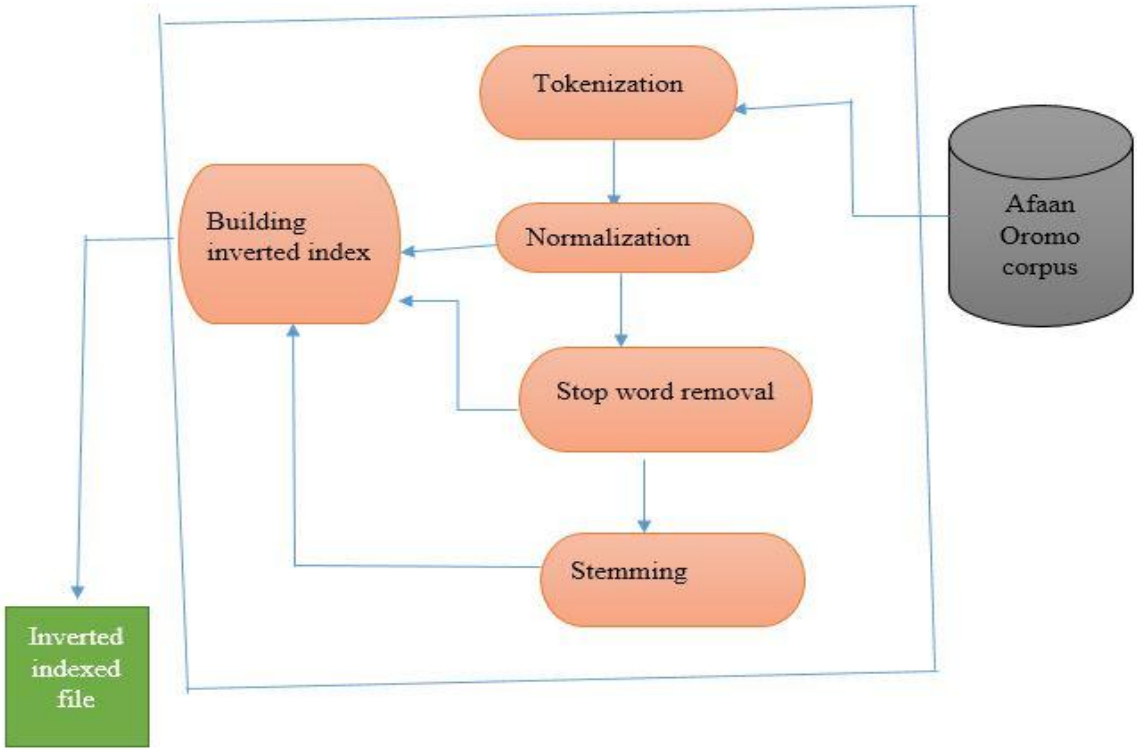

Figure 6.2: Architecture design of indexing process for Afaan Oromo

Research question 3: What text operations are needed in Afaan Oromo to apply for document corpus?

Answer: As we have identified by this study, the text operations needed for Afaan Oromo document corpus are: tokenization, normalization, stop word removal and stemming. In Afaan Oromo when we tokenize the given token, it is very important to careful about apostrophe. This is because apostrophe is considered as a word in Afaan Oromo. It is known that there is *standard solr factory* which is used by apache solr to tokenize *a* token. We have inserted lists of punctuation of Afaan Oromo except apostrophe, to tokenize the token. The other option is also we can replace the apostrophe with the letter 'h' since they have the same pronunciation and also meaning.

6.7. Results and findings

The results obtained by this study is encourage able to develop an Afaan Oromo search engine when compared to other researches that have been done previously by other scholars those have been made attempt on Afaan Oromo. Because, the experimentation made for 158 of Afaan Oromo documents by using 13 queries registered average precision of 72.91% and average recall of 86.8% with 79.25% of F-score before Afaan Oromo synonymy and polysemy are added. However, the study also checked that the existences of synonymy and polysemy can improve the performance of the system, especially, recall. The average precision, average recall and F-measure was 71.39%, 90.5% and 79.82% respectively, after the addition of synonymy and polysemy. It improves the performance of the system by 0.57%.

(Gezehagn, 2012) has attempted on Afaan Oromo by using vector space and the system registered 0.575 precision and 0.6264 recall. This result is very low when compared to our experimentation. The other attempt was (Girma, 2012), who has done on Afaan Oromo and registered 74.4% and 58% precision and recall respectively. This is also very low when compared to this study.

Generally speaking, this study contributes some useful things to Afaan Oromo. The *first one* is that designing Afaan Oromo text retrieval system using probabilistic approach with apache solr is possible, which was checked with experiment. *Second*, this study also checked the effects of synonymy and polysemy on the performance of the system by experiment even if there was no real thesaurus system for Afaan Oromo language. *Third*, the system's spellchecking power while searching some document is also another contribution of this study. This means when someone searches for some documents, (i.e. if he or she enters incorrect statement), for example, 'Blsummaa umta Oromoo' instead of 'Bilisummaa uummata Oromoo', the spellchecker can correct it by showing the following with the document available in the corrected query:

The other contribution of this study in addition to text retrieval is that *hit highlighting* is also possible. This means when we search the document for some query, each and every words in the query are written in bold in the document. Which simplify works for the user to quickly identify which documents contain the information need.

CHAPTER SEVEN

CONCLUSION AND RECOMMENDATION

7.1. Conclusion

In this study we have attempted to develop a prototype of Afaan Oromo text retrieval system using probabilistic approach. The two modules for this development are indexing and searching. The text pre-processing or text operations are done in indexing. This is because to index the documents, the first thing we have to do is pre-processing. The searching component has also pre-processing and similarity measurement. The similarity measurement used was BM25similarity which is used in probabilistic approach and also applicable in apache solr.

According to the experimentation made in this study, the system registered the average precision and recall of 72.91% and 86.8% respectively, before the synonymy and polysemy were added. However, after the addition of some synonymy and polysemy, the result is changed to 71.39% average precision and 90.5% average recall for 13 queries of 158 documents. There were 3.7% improvements on recall and the precision was decreased by 1.52%. The F-measure (F-score) result for the first experimentation (before synonymy is added) was 79.25% and after the addition of synonymy the result was 79.82%, which was improved by 0.57% from the first. Therefore, we can say that the existence of both synonymy and polysemy have effect on the performance of an information retrieval system for Afaan Oromo. The probability that a relevant document is not retrieved (also called False Negative or Type II error) which is given by: **1- Recall** (Sanderson, 2005) is used to know the error rate of the system. Therefore, to get the probability that the system is unable to retrieve relevant document, we have to subtract the recall obtained by the experiment from one (1). That means; 1-0.905 =0.095, where 0.905 the result of recall in this study. So, for this study: The probability that the developed system can retrieve relevant document is =90.5% and the probability that the developed system cannot retrieve relevant document is =9.5%.

7.2. Recommendation

IR technology is an ever-growing area of research, especially with regard to Afaan Oromo text where challenges presented in the language. Since studying about Afaan Oromo text is the beginning level, it needs the collaboration of the researcher. So, we would like to recommend the following recommendation to concerning body.

- To test and make experiment, IR system needs standard corpus. However, since there is no standard corpus developed for Afaan Oromo till now, it needs more emphasis.
- Although there are several strategies that could be used to evaluate IR system, this study focuses on precision/recall and F-measure method. Other methods such as: mean average precision (MAP) and multi-grade strategies did not used. So, by increasing the number of documents, preparing thesaurus system and correct stemming algorithm someone can measure the performance of the system since all the synonymy and polysemy of Afaan Oromo did not used here.
- Apache solr accept the document and index it, it is also possible to download that document, however, the study only focusses on searching some documents to examine the power of the chosen model (probabilistic model) for Afaan Oromo. So, finding the means of getting the documents back or downloading any document that we need is also another future work by adding all types of data that must be searched.
- Since the existence of the synonymy and polysemy of the language have the ability to improve the performance of the retrieval system, it is also very important if there is thesaurus system for Afaan Oromo.
- Any other document like video, audio, graphics and pictures are not studied in this research. To make this research fully functional, therefore, it needs additional work on these types of data. Because, it needs time and more investigation.

References

Abhimanyu, Z. (2013). Natural Language Processing. *International Journal of Technology Enhancements and Emerging Engineering Research*, 50-62.

Ajit, M. K., & Biswas, S. (2011). Inverted indexes: Types and techniques. *IJCSI International Journal of Computer Science, 8*(4), 384-392.

Albujasim, Z. M. (2014). Search Queries in an Information Retrieval System for Arabic-Language Texts. *Thesis and Desertation--Computer Science*, 23-77.

Borhade, S. B., & Agarkar, P. (2014). Lucene Based Performance Evaluation Of Relational Database. *Internatonal Journal Of Innovative Research In Technology, I*(6), 1981-1987.

Bruck, A., & Tilahun, T. (2015). Bi-gram based Query Expansion Technique for Amharic Information Retrieval System. *I.J. Information Engineering and Electronic Business, vol.6*, 1-7.

Chu, H. (2003). *Information representation and retrieval in Digital age*. Thomas H. Hogan Sr.

Daniel, A. B. (2011). Afaan Oromo-English Cross-Lingual Information Retrieval (CLIR): A Corpus Based Approach. *Master Thesis*, 1-95.

Darwish, K. (2013). Arabic Information Retrieval. *Foundations and Trends in Information Retrieval, Vol.7*(no.4), 239–342.

Datta, J. (2010). *Ranking in Information Retrieval*. Powai, Mumbai: Indian Institute of Technology, Bombay.

Dikshant, S. (2015). *Apache Solr: A Practical Approach to Enterprise Search*. Pune, India: The Digital Group.

Emad,et al. (2015). Semantic Boolean Arabic Information Retrieval. *The International Arab Journal of Information Technology, Vol.1*(No.3), 311-316.

Fazli,et al. (2008). Information Retrieval on Turkish Texts. *Journal of the American Society for Information Science and Technology, Vol. 59*, 407-421.

George, L. E., & Hameed, I. A. (2013). Text-Based Information Retrieval System using Non-Linear Matching criteria. *International Journal of Advancements in Research & Technology*, 1-5.

Gezahagn, E. G., & at.el. (2012). Afaan Oromo Text Retrieval System. *Master Thesis*, 33-40.

Gezehagn, E. G. (2012). Afaan Oromo Text Retrieval System. *Master Thesis*, 12-20.

Gezehagni, E. G. (2012). Afaan Oromo Text Retrieval system. *master's thesis*, 16-20.

Girma, D. D. (2012). Afan Oromo news text summarizer. *Master Thesis*, 1-100.

Grainger, T., & Potter, T. (2014). sample chapter. In T. Grainger, & T. Potter, *Solr in Action* (pp. 1-28). Manning Publications.

Gudisa, T. (2013). Design and Implementation of Predictive Text Entry Method for Afan Oromo on Mobile Phone. *Master Thesis*, 2-3.

Guutamaa, I. (2010). *Afaan Oromo As Second Language*. USA: Gubimans publishing.

Hailay, B. B. (2013). Design and Development of Tigrigna Search Engine. *Master Thesis*, 1-95.

Hiemstra, D. (2009). Information Retrieval Models. *Information Retrieval*, 4-10.

Inkpen, D. (2005). *Information Retrieval on the Internet*. Ottawa: University of Ottawa.

James, A. (2002). Challenges in Information Retrieval and Language Modeling. *of Massachusetts Amherst*.

Javid, & Dadashkarimi. (2014). A Probabilistic Translation Method for Dictionary-based Cross-lingual Information Retrieval in Agglutinative Languages. *Third Computational Linguistic Conference in Sharif University* (pp. 1-13). Tehran: Tehran University.

Jelita, A. (2007). Effective Techniques for Indonesian Text Retrieval. *A thesis submitted for the degree of Doctor of Philosophy*, 42-43.

Kumar, J. (2015). *Apache Solr Search Patterns*. (T. Patil, Ed.) MUMBAI: Packt Publishing.

Manning, C. D., Raghavan, P., & Schütze, H. (2009). *An Introduction to Information Retrieval*. Cambridge University, England: Cambridge University Press.

Mayanale, S. C., & Pawar, M. S. (2015). Marathi-English CLIR using detailed user query and unsupervised corpus-based WSD. *Int. Journal of Engineering Research and Applications, Vol.5*(6), 86-91.

Milkessa, M. (2014). Official Language Choice in Ethiopia: Means of Inclusion or Exclusion? *Open Access Library Journal* , 1-13.

Nutman, A. (2014). Afan Oromo Mass Media: Challenges and Prospects. *ResearchGate* (pp. 4-6). Addis Ababa: Addis Ababa University.

Rababah, S. (2011). Modern Information Retrieval. *information retrieval system*, 6-28.

RIJSBERGEN, C. J. (1979). *INFORMATION RETRIEVAL*. Glasgow: Office of Scientific and Technical Information.

Robertson, S., & Sp¨arck, K. (2006). *Simple, proven approaches to text retrieval*. Cambridge: University of Cambridge.

Rosenberg, D., & Sillince, J. A. (2000). Verbal and Nonverbal communication in computer mediated settings. *International Journal of Artificial Intelligence in Education*, 299-319.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic retrieval. *Information Processing and Management*, 1-6.

Sanderson, M. (2005). *Information Retrieval System Evaluation: Effort,Sensitivity and Reliability* . Sheffield: Sheffield University.

Saraswathi, S.et al. (2010). BiLingual Information Retrieval System for English and Tamil. *Journal of Computing, Vol.2*(4), 85-89.

Singhal, A. (2001). Modern information retrieval: A brief overview. *IEEE Data Engineering Bulletin, vol.24*(3), 5-7.

Sojka, P., & Schütze, H. (2015). *Introduction to Information Retrieval.* Brno: University of Munich.

Sonia, B., & Reena, G. (2010, November 1). A Novel Probabilistic Approach for Efficient Information Retrieval. *International Journal of Computer Applications, 9*(2), 1-5.

Tarawneh, M., & Al-Shawakfa, E. (2015). A Hybrid Approach for Indexing and Searching the Holy Quran. *Jordanian Journal of Computers and Information Technology, vol.1*, 48-50.

Tesfaye, D. (2010). Designing a Stemmer for Afaan Oromo Text: A Hybrid Approach. *Masters thesis*, 50-51.

Tewodros, G. H. (2003). Amharic Text Retrieval: An Experiment Using Latent Semantic Indexing (LSI) with Singular Value Decomposition (SVD). *Master Thesis*, 1-113.

Tilahun, N. (2015, July 1). Homorganic Nasal Assimilation in Arsi-Bale Afan Oromo: A Non-Linear Phonology. (T. Negash, Ed.) *Humanities and Social Sciences, III*, 240-248.

Tiruveedula,G.et al., (2014). Future Model for Information Retrieval Systems Based on Arabic Language. *International Journal of Advanced Trends in Computer Science and Engineering, vol.3*(no.2), 01-03.

Vu, H. T. (2015). Ndex Strategies For Efficient And Effective Entity Search. *Master's Projects*, 396.

Xiangji, H., & Robertson, S. (2000, november 4). Probability-based chinese text processing and retrieval. *computitional intelligence*, 1-3.

Yeni, H. (2003). Information Retrieval System In Bahasa Indonesia Using Latent Semantic Indexing And Semi-Discrete Decomposition. *International Journal of IR*, 2-3.

Yilu, Z. et al. (2006). Experiments on Chinese-English Cross-language Retrieval. *Information Retrieval to Knowledge Management*, 5-6.

Yu-Chung,et al. (2009). Korean-Chinese Cross-Language Information Retrieval Based on Extension of Dictionaries and Transliteration. *Institute of Information Science journal*, 8-9.

Appendix I: Websites from which data is collected

http://www.voaafaanoromoo.com/

http://bilisummaa-oromo.blogspot.com/

http://kiilolee-birraa-malkaa.blogspot.com/

http://www.voaafaanoromoo.com/

http://www.biblestudyproject.org/

http://www.wikimezmur.org/

https://om.wikipedia.org/

http://www2.fanabc.com/

http://kallacha.8m.com/

http://gotoota-oromo.blogspot.com/

http://www.voicefinfinne.org/

http://maddawalaabuupress.blogspot.com/

http://waaqeffannaa.org/

http://waaqeffannaa.org/

http://omicwomo.info

Appendix II---Afaan Oromo Stop words

aanee	gararraa	ishiirraa	koo	siin
agarsiisoo	garas	ishiitti	kun	silaa
akka	garuu	ishiitti	lafa	silaa
akkam	giddu	isii	lama	simmoo
akkasumas	gidduu	isiin	malee	sinitti
akkum	gubbaa	isin	manna	siqee
akkuma	ha	isini	maqaa	sirraa
ala	hamma	isinii	moo	sitti
alatti	hanga	isiniif	na	sun
alla	henna	isiniin	naa	tahullee
amma	hoggaa	isinirraa	naaf	tana
ammo	hogguu	isinitti	naan	tanaaf
ammoo	hoo	ittaanee	naannoo	tanaafi
an	hoo	itti	narraa	tanaafuu
ana	illee	itumallee	natti	ta'ullee
ani	immoo	ituu	nu	ta'uyyu
ati	ini	ituullee	nu'i	ta'uyyuu
bira	innaa	jala	nurraa	tawullee

booda	inni	jara	nuti	teenya
booddee	irra	jechaan	nutti	teessan
dabalatees	irraa	jechoota	nuu	tiyya
dhaan	irraan	jechuu	nuuf	too
dudduuba	isa	jechuun	nuun	tti
dugda	isaa	kan	nuy	utuu
dura	isaaf	kana	odoo	waa'ee
duuba	isaan	kanaa	ofii	waan
eega	isaani	kanaaf	oggaa	waggaa
eegana	isaanii	kanaafi	oo	wajjin
eegasii	isaaniitiin	kanaafi	osoo	warra
ennaa	isaanirraa	kanaafuu	otoo	woo
erga	isaanitti	kanaan	otumallee	yammuu
ergii	isaatiin	kanaatti	otuu	yemmuu
f	isarraa	karaa	otuullee	yeroo
faallaa	isatti	kee	saaniif	yommii
fagaatee	isee	keenna	sadii	yommuu
fi	iseen	keenya	sana	yoo
fullee	ishee	keessa	saniif	yookaan
fuullee	ishii	keessan	si	yookiin
gajjallaa	ishiif	keessatti	sii	yoolinimoo

gama	ishiin	kiyya	siif	yoom
Akka	Yoo	Nuti	Fi	waan
Ituu		tawullee	Immoo	ofii
Odoo	akkasumas	Isee	Moo	akka
Silaa	ofii	Keeti	Illee	Kun
Yeroo	Malee	otuu	Akka	Sun
hanga	erga	utuu	jechuu	an
erga	erga	otuma	jechuun	kan
Osoo	waggaa	ka	jechaan	inni
Oggaa	waan	Saniif	tanaaf	tanaafi
Kanaafi	otumallee	tanaafuu	itumallee	jechuun

Appendix III: list of Afaan Oromo synonymy

uummata, saba

gaarii, misha, bayessa,dansaa

aaduu, albee, haaduu

ajjaa, omborii

aankoo, jaldeessa

aantii, aanaa, fira

aarii, haarii

aayyoo, aayyaa

abaabayyuu, habaabayyuu

ababoo,habaaboo,ilillii,dararaa

abadan,siruma

abashaa,habashaa

abbaagadaa,luba

abbala,hawwa

abboomama,adabamaa

abboomuu,adabuu

abishii,sunqoo

ablee,hablee

alalee,halalee

biyyalafaa,addunyaa

Appendix IV:BM25Similarity integration java code

```
package org.apache.lucene.search.similarity.dfr;

import java.io. IOException;

import org.apache.lucene.index.DocsEnum;

import org.apache.lucene.index.IndexReader;

import org.apache.lucene.index.Stats.DocFieldStats;

import org.apache.lucene.search.DefaultSimilarityProvider;

import org.apache.lucene.search.similarity.AggregatesProvider;

import org.apache.lucene.util.BytesRef;

public class BM25Similarities extends DefaultSimilarity{

private final float k1;

 private final float b;

 private final AggregatesProvider aggs;

 public BM25Similarities (Aggregates Provider aggs, float k1, float b) {

  this.aggs = aggs;

  this.k1 = k1;

  this.b = b;

 }
```

```
 public BM25Similarity (float k1, float b) {
     if (Float.isFinite(k1) == false || k1 < 0) {
      throw new IllegalArgumentException("illegal k1 value: " + k1 + ", must be a non-negative finite value");
     }
     if (Float.isNaN(b) || b < 0 || b > 1) {
       throw new IllegalArgumentException("illegal b value: " + b + ", must be between 0 and 1");
      }
public BM25Similarities(AggregatesProvider aggs) {
  this(aggs, 1.2F, 0.75F);
 }
 private class BM25TermScorer extends BoostBytesFieldDocScorer {
  private final float queryWeight;
  private final DocsEnum docsEnum;
  public BM25TermScorer (IndexReader segment, float queryWeight,
     TermAndPostings termAndPostings, final BM25FieldSimilarity fieldSim) throws IOException {
    super(segment, queryWeight, termAndPostings, fieldSim,
```

```
        new ComputesLengthNorm() {
          protected float lengthNorm(int docID, DocFieldStats stats) {
            // nocommit: incorporate doc boosting into this
            float norm = k1 * ((1 - b) + b * (stats.termCount) / (fieldSim.avgTermLength));
            return norm;
          }
        });
      this.queryWeight = queryWeight * fieldSim.idf(termAndPostings.term);
      this.docsEnum = termAndPostings.docsEnum;
    }
  public float score() {
      final float tf = docsEnum.freq();
      return queryWeight * tf / (tf + getDocBoost(docsEnum.docID()));
    } }
  public class BM25FieldSimilarity extends DefaultFieldSimilarity {
    final IndexReader topReader;
    final float avgTermLength;
  public BM25FieldSimilarity (IndexReader topReader, String field, double avgTermLength)
throws IOException {
```

```
    super(topReader, field);
    this.topReader = topReader;
    this.avgTermLength = (float) avgTermLength;}
public FieldDocScorer getTermScorer(float queryWeight, TermAndPostings
termsAndPostings, IndexReader segment)
     throws IOException {
    return new BM25TermScorer (segment, queryWeight, termsAndPostings, this);  }
public float idf(BytesRef term) throws IOException {
    final float dfj = topReader.docFreq(field, term);
    final float n = topReader.maxDoc();
    return (float) Math.log (1 + ((n - dfj + 0.5F)/(dfj + 0.5F)));
  } }
public FieldSimilarity getField(IndexReader topReader, String field) throws IOException {
  return new BM25FieldSimilarity (topReader, field, aggs.getAvgTermLength(topReader,
field));  }}
```

Appendix V: Sample Code Used for Afaan Oromo Stemming

```
package stemming;

import java.io. *;

import java.util.regex.Pattern;

import java.util.Scanner;

//import org. tartarus. snowball. *

public class Stemming {

   public static void main (String [] args) throws FileNotFoundException, IOException {

      String a,s,line,token;

FileReader file_to_read=new FileReader("C:\\solr-6.0.0\\server\\solr\\afaan-oromo-core\\data\\index"); //our file path.

   Scanner filesc=new Scanner(file_to_read);

   FileWriter fstream = new FileWriter("C:\\solr-6.0.0\\bin\\afaan oromo stemming\\stemming.txt"); // our lists of stemming

   BufferedWriter out = new BufferedWriter(fstream);

 while(filesc.hasNextLine())

   {line=filesc.nextLine();

   Scanner linesc=new Scanner(line);
```

```
        while(linesc.hasNext())
        { token=linesc.next();
 a=stemming.getInstance().process(token);//method  to access the porter stemmer  for afaan oromoo
         out.write(a);
         out.write(" ");}
        out.newLine();
        linesc.close();  }
    filesc.close();}
    private  static  Object getInstance()  {
        throw new UnsupportedOperationException("Not  supported yet.");     }}
```