
**JIMMA UNIVERSITY**

**JIMMA INSTITUTE OF TECHNOLOGY**

**FACULTY OF COMPUTING**

# Amharic Text Summarization for News Items posted on Social Media

**BY ABAYNEW GUADIE**

**This Thesis Submitted to Faculty of Computing to Jimma University for the Partial Fulfillment of the Requirement for the Degree of Master of Science in Information Technology**

**Nov 11, 2017**

**Jimma, Ethiopia**

**JIMMA UNIVERSITY**

**JIMMA INSTITUTE OF TECHNOLOGY**

**FACULTY OF COMPUTING**

# Amharic Text Summarization for News Items posted on Social Media

**BY ABAYNEW GUADIE**

**Main Advisor: Mr. Debela Tesfaye (Assistance Professor)**

**Co-advisor: Mr. Teferi Kebebew (Lecturer)**

**Nov 11, 2017**

**Jimma, Ethiopia**

**Declaration and Certification**

This is to certify that the thesis entitled ‘‘Amharic Text Summarization for News Items posted on Social Media” submitted by Abaynew Guadie, the MSc. Information Technology student of Faculty of Computing, Jimma Institute of Technology (JIT), Jimma University for the award of Master of Science in Information Technology is a record of original work carried out by him under my supervisor and guidance. The thesis has fulfilled all requirements as per the regulations of the University and in my opinion the thesis has reached the standard needed for submission. The results embodied in the thesis have not been submitted to any other University for the award of any degree. I hereby declare that the student has incorporated the comments given during the mock defense to improve the work substantially.

Debela Tesfaye (Assistance Professor)  _____

**Advisor** Signature Date

Teferi Kebebew (Lecturer) _____ _____

**Co-Advisor** Signature Date

Martha Yifiru (PhD) _____ _____

**External Examiner** Signature Date

Getachew Mamo (Assistance Professor) _____ _____

**Internal Examiner** Signature Date

Kibret Zewedu (Lecturer) _____ _____

**Chair Person** Signature Date

_____ _____ _____

**Institutional Research Coordinator for PG.** Signature Date

**Dedication**

This thesis work is dedicated to my father Guadie Nigussie and my mother Alemetu Kinfe, all my brothers and sisters, my wife Banchiamlake Asnake and my son Nahom and who were able to beside me for the fruit of their own.

Abaynew Guadie

Nov 11, 2017

## Declaration

This thesis works to declare that this study is my original work and has not been submitted as partial requirement for a degree in any other university, and that all sources of materials used for the thesis has been appropriately acknowledged.

Abaynew Guadie

Nov 11, 2017

## Table Contents

**List of contents** **Page no.**

## ACKNOWLEDGEMENT

First, I would like to express my heartfelt gratitude to the almighty God and his mother St. Mary for making everything the way it is. All of my efforts would have gone for nothing if it had not been for his importunate help to complete this thesis.

My sincerest thanks to my advisor Mr. Debela Tesfaye (Assistant Professor) and my Co-advisor, Mr. Teferi Kebebew for the helpful cheer before the work was started and constructive comments and direction after the work from beginning for selecting for proposed title for the research study and end of finalize my thesis work.

I owe my deepest heartfelt appreciation to my wife Banchiamlake Asnake for her support and encouragement in conducting this research study. She is always in my heart and my son Nahom.
Special thanks also go to my families, mainly for my father, Guadie Nigussie and my mother Alemetu Kinfe, my uncle Kibret Gela and also my brothers and sisters, who have been behind me in supporting and encouraging me through difficult times.

My deepest thanks go to all my friends, especially Mamaru Dessalegn, Tesfu Makonnen, and Wubetu Board for their strong cooperation by standing with me throughout every step of my work research study. Beside, this I would like to acknowledge all my staff members of Faculty of computing for their help to complete this research corrections during this study.

## Lsist of figures



# List of the algorithm



## List of tables

# List of Acronyms

ASCII----American Standard Code for Information Interchange

BOW----Bag-Of-words

CR---- Compression Ratio

CSV----Comma-Separated Values

DUC----Document understanding Conference

ESTC----Ethiopian Science and Technology Commission

HMM----Hidden Markov Model

NLP----Natural Language Processing

TAC----Text Analysis Conference

TF-IDF----Term Frequency-Inverse Document Frequency

TREC----Text REtrieve Conference

TS----text summarization

TwitIE----Twitter Information Extraction

URL----Uniform Resource Location

UTF----Unicode Text Format

VG ---- Visual Geez

VSM----Vector Space Model

IR---- Information Retrieval

## ABSTRACT

*Nowadays, the information overloads with social media are with the development in natural language application for the local Amharic language speaker posted texts, the amount of data one has to deal with increased rapidly the volumes of posts Amharic text documents that could be posted on Facebook and Twitter. So, the news items on text summarization system user needs for posting that can be summarized from the posted documents that belong to the duration of over a time period of the date, monthly, and yearly posted texts to summarize the tweets from Twitter and Facebook posted Amharic texts. Thus, summarization is dealing with information overload presenting and posted with a text document for the current time representation of the posted documents to summarize [1]. The purpose of this paper is to summarize the news items posted Amharic texts over a time posted documents from social media on Twitter and Facebook; first to find the similarity between posts, and then to cluster of similar posts, and also the groups of clustered documents to summarize with the individual cluster posted documents in the rank sentences that is to identify the higher score of important sentences in the documents to produce for the readers without duplicated posted sentences in the document. The main problem of the social media posted texts are that most people would probably read their posted in Amharic texts with duplicate posted documents. The post texts will most likely contain articles or others news items that are not relevant to the post in question. However, to find the information the user is looking for she or he will have to find summary posted texts and read important portions of posts as Amharic documents to extract desired information on social media. So, duplication information once is posted for the original texts for the Twitter and Facebook that need to minimize the amount of posts to be summarized can be achieved by condensing duplicate posted texts. The main objective of the study is to investigate the development of Amharic text summarization system for social media posted texts (Twitter and Facebook) in Amharic texts. The corpus preparation contains posts news items (protests, droughts, sports and floods) on Twitter and Facebook in total 4951 posted documents in the sentences to prepare for the experimentation and the implementation tool is Java platforms. Our proposed approach has three components: First, calculate the similarity between each posted document within the two pair of sentences. Second, clustering based on the similarity results of the documents to group them by using Kmeans algorithm. Third, summarizing the clustered posted document using TF-IDF algorithms that involve finding statistical ways for the frequent terms to rank the documents. We apply the summarization technique is an extractive summarization approach that is assigned an extract the sentences with highest or list of top ranked sentences in the posted documents to form the summaries and the size of the summary can be identified by the user. The performance of the system is evaluating the results by using both subjective*

*and objective evaluations. In the subjective evaluation is the linguistic qualities of the system summaries are assessed by the human who prepared the manual summaries. Objective evaluation is done for the summaries generated in the experiments by comparing them with an automatic summary of ideal manual summary using f-measure. For the f-measure for Amharic text summarization news items posted on social media that our system indeed performed very well for both single and multiple document summarization tasks. The experiments are prepared for manual and automatic summary using both systems to generate the 30 posted the texts in each clustered posts text files for testing set to extract summary for at 10%, 20% and 30% extraction rate for the posted texts. In the experiment one the highest F-measure score is 87.07% of extraction rate of 30%, in the clustered or group of protests posts. The second experiment the highest F-measure score is 84% for the extraction rate at 30%, in droughts post groups. In the third experiment the highest F-measure score is 91.37% of extraction rate of 30%, in the sports post groups and also the fourth experiments the highest F-measure score is 93.52% of extraction rate of 30% in floods post group to generate the summary post texts. If the system to generate the size of the summary is increased, the extraction rate also increased to posted texts.*

***Keywords:*** *Amharic language, Similarity measure, Text Summarization, Clustering, Tf-IDF algorithm, Social media, Facebook, Twitter, News posted texts*

# CHAPTER ONE

# INTRODUCTION

## 1.1. Background

Amharic language is one of the main African languages and also it is the working language of the Federal Government of Ethiopia and widely spoken throughout the country[2]. It is an Afro-Asiatic language belonging to the Semitic group of its own unique alphabets [3]. It is one of the major languages in Ethiopia and according to the 2007 census, it is the native language of around 22 million people, just under 30% of the population of around 74 million, but over the past 10 years the population of Ethiopia is estimated to have exceeded 100 million and probably over 30 million native speakers now[4]. Amharic language speaker to post Amharic texts on Twitter and Facebook in the social network at this time, which is increasing volumes of posting data are available on the social media, which is observed on the growing online posts, websites, and digital storages in the language. It may not be long before users are faced with huge volumes of Amharic texts in the social networks and other social media. These Amharic text documents are available digitally and the amount is highly increasing every day from a user posting to the social media. In such developments, user will find it very difficult and frustrating to make efficient use of Amharic texts to read the post texts unless they are aided by data processing tools for activities like searching, categorization, clustering, filtering and summarization of the documents. The task of developing for easy retrieval of relevant information is specially challenging on Amharic texts because there are only few recent and uncoordinated efforts of automation and language processing[2]. Currently, with the enhancement in the most people to use and posted and reposted many Amharic text documents in social networks, the amount of data one has to deal with has increased rapidly on Twitter and Facebook for the readers. The number of documents to take into consideration is far more and the time to go through all such documents is very less. If the important information present in these posted documents on social media is made available on Facebook and Twitter[1] to summarize for condensed form the user saves time and the user can focus on important parts of the documents easily.

Text summarization aims to investigate the summarization of user posts over time for the documents from the social media within the time period of each post as a stream posted documents on the social media. Therefore, documents shall be iterated over in news items order for the news items posted on Facebook and also its aims

[1] http://www.twitter.com , Twitter has been credited with providing the most current news about many important user posts and also played to help people who read Twitter posts or tweets texts.

to create and evaluate the news items of posted texts summarization systems for the news post on the social media[5]. Social media refers to interactions among people in which they create, share, exchange, or comment on information or ideas on the Internet or with other virtual communities and networks that precede the Web, such as report post services. It services such as Twitter and Face book generate to rapid access phenomenal volume of content for most real-world posts on a daily source[6]. Social media is a vital source of information on any major post, especially political protests, natural disasters, shooting, bombing and others user posts on the web. However, with the exponential increase in volume of social media posted data, so posts can increase in informal data that does not provide valuable information, and accordingly posts to happen for lessening peoples' ability to find the information that they need in order to organize relief efforts, find help, and potentially save lives to summarize the text document. News posted text summarization can be described as the problem of automatically summarizing information on time, date, month and years. The easiest way to extract Amharic tweets related to a post as a query about the news items user posts in Amharic text. However, for popular posts, this typically results in a significantly large stream of tweets, which makes the task of understanding the aspects of the post and the opinion of the people, a difficult and mostly useless task. It has been observed that, despite the high frequency, the actual information content of the tweet stream is fairly limited. This is due to the fact that several of the Amharic tweets contain redundant information were posted on Facebook and Twitter. Also, many of the tweets that are returned by a search query are not relevant to the post. In this paper, we address summarizing a targeted user posts of interest in a human reader by extracting the most representative posts of the irrelevant Amharic tweet stream for the post news that could be summarizing without duplicate posted Amharic text documents[7].A news items posted text summarization system need user posts that can be summarized from Amharic documents that belong to post on the social media especially on Twitter and Facebook. Thus, summarization is production of information overload to involve presenting to posted on the Twitter with a text documents and time representation of the evolving posts to summarize [1].

The system produces summary by extracting key sentences from the input document and produce a summary by generating posted sentences for the user posts text. Each similar sentence is identified by the combination of a document identifier (which document the sentence come from) and a sentence identifier (the position of the sentence within the document to sequential or randomly selected to merge with similar posts). A summary can consist of a subset of documents that have been considered to be influential and have a lot of situations, it can be a set of sentences extracted from a single or multiple posts. News posted Amharic text documents include summarizing posted political protests, natural disasters, bombing, earthquake[8], shooting, storm and

social media for accident things to happen compiling information from a large set of documents into a single news posted document or generate a summary of user posts from Twitter and Facebook. Summarize for Amharic news items posted documents are to condense the similar posted Amharic text documents on the Twitter, Facebook and also user posts to extract the important Amharic texts, while holding the most important relevance posted document to retrieve the document for Amharic texts produce for the reader.

A summary is a text produced from one or more text documents from different source posted Amharic text to be taken. It contains a significant portion of the information in the original texts and it is no longer than the original texts. Summary posts to consider on shorter than the original input text and contain the important information about the study, as defined by the user posts on the Twitter and Facebook. Summarization of tweet focuses on generating a condensed form of a document that covers the posted document's to get single tweets without repeat tweets. This benefits people get an understanding of the Amharic document content quickly and as a result, people can make a decision faster whether a certain document is relevant to them for the conditions to read for the reader. The purpose of this paper is to summarize the news items posted Amharic texts over a time posted documents from social media on Twitter and Facebook. Hence, first to find the similarity between posts, and then to a cluster of similar posts, and also the groups of clustered documents to summarize with the individual clustered posted documents in the ranked sentences that is to identify the higher score important sentences in the documents to produce for the readers without duplicated posted sentences in the document. So, this research is to design the systems for efficiently monitoring the news items posted in Amharic post texts the information overloaded posts at the time for the social media for Twitter and Facebook posted documents. A good way to get up to date, monthly and yearly information Amharic texts will be to get a stream of sentence length in posted documents about the situation as it develops[9].

The motivation to create for news items posted Amharic texts to summarize is to automate extracting important posts easily, effectively and sufficiently process and generate a more representative important to the summary Amharic text user posts to read in less time and effort. In most cases, Amharic posts text summarization is performed for the purposes of saving users time by condensing the amount (in the size) of posts text documents without repeated posting documents to read on the Twitter and Facebook user posted on social media also the resolve is to produce a short summary for Amharic tweets related to a specific user posts in less time and effort to read the posts. In extracting the Amharic tweets most important advantage of using a summary of Amharic posted text on Twitter and Facebook for extracting out significant information for Amharic texts are its reduced reading time. Summary generates by an automatic extracting Amharic tweet from the input documents that is processed to summarize and has also other advantages:

- The size of the short summary posted news item's user can be controlled on the tweets Amharic text user posts.
- Its content is determinate and to condense the content for the reader, and
- The link between a text user post in the summary and its original text can be easily established.

## 1.2. Statement of the problem

There are several works attempting to summarize texts for several languages, but none is done for Amharic language for news item posted texts on social media particular about Twitter and Facebook. The news items posted on social media which work aims at the once posted and others re-posted in Amharic texts. Accordingly, the information is needed for the users posted Amharic texts on social media. The readers are not obtained the similarity posted text and also clustered document of the news items Amharic text mostly on Twitter and Facebook posted the text of the summary. Readers tend to focus on condensing the information provided to them by taking summarize posted texts and ignoring the details if they are not interested the posted texts. The summarization is not based on text from singing source, it considers texts from different source post texts of the social media user posted text documents. Many text documents for social media pose documents are automatically posted texts on social networks like Facebook, Twitter, and other social network.

For this, due to the development of the technology and change in lifestyle, people have needed less time to spend reading vast amounts of posting information available digitally to satisfy their significantly less information need of the readers. Immediately after a news posted for Amharic texts to summarize, with significant text document on the social media to need the users. Even once, after a few hours, relevant document becomes available and it is often inaccurate or highly redundant documents are posted. At the same time, if users urgently need posted information on summarizing documents to read the Amharic post texts, especially if users are directly affected by the posts like protests, droughts, floods and others news posts. In particular, the news items posted text summarization task focuses on the large posts document on social networks, such as political protests, accidents or natural disasters (earthquakes, droughts, and floods), sports and others posts. It is often difficult to get a simplified overview [10] of the post as a short amount of time and other urgent new posts that can be used of the summary posted texts. The information on social media as user posts that posted on Twitter and Facebook often contains a lot of information that is not relevant and requires a lot of reading to gain an overview of the situation for news posts text. Most people would probably read their posted text summary. However, to find the information the user is looking for she or he will have to find relevant posts texts summary and read important portions of Amharic text documents to extract desired

information on social media. Duplicate post texts on the Twitter and Facebook that need to minimize or compress the amount of data to be summarized can be achieved by removing or ignoring repeat text documents news posts. Many social media platforms the number of people re-posting texts can be interpreted as a measure of its importance, whenever removing duplicates and order by similarity measure and also clustered each posted texts into the summary. The large amount of data on social media on Facebook and twitter posted texts are become strongly time dependent to read a large amount of news posted documents. In this respect, the time dimension has been widely exploited as a highly important relevance measure to improve the effectiveness of document classifying and summarized models. This can relatively summarize texts be combined with an evolving news Amharic posts the challenge is to identify features that provide meaningful information to the summary. Users are now spending time for reading as Amharic documents on social media, which provides lots of text documents creating user information overloads to compress the summary. Thus, the user to read a huge amount of information to have and get the right information on documents a user should not consume all texts in the documents on social media posted texts. To handle this problem all those available posted text and make use of its information contents the text summarization is considered to summarize the large posted text and news items posted on social media especial Facebook and twitter. The research questions for this thesis as follows:

1. How to summarize the clustered documents to extract the summary of each cluster?
2. How to summarize the group of multi-documents in one cluster and rank the sentences for automatically generating the summary of news items posted Amharic texts by the monthly and yearly?

## 1.3. Objective of the study

### 1.3.1. General objective

The general objective of the study is to investigate the development of an Amharic text summarization system for social media posted texts.

### 1.3.2. Specific objectives

It is to achieve the general objective of the study as follows

- To review the literature on the concept of related works on text summarization and news items posted Amharic texts and the available methods of similarity and clustering automatically.
- To build a corpus of Amharic text on social media for the Twitter and Facebook posted texts.
- To develop the design for new proposing system for the research studies.
- To compute similarity between the Amharic texts to determine similarity between posts

- To cluster similar posts
- To extract summary forms each clusters
- To test the system and to measure its performance i.e. how much the system condenses the posted texts

## 1.4. Scope and limitation of the study

### 1.4.1. Scope of the study

The scope of this research focus on text summarization of news items user posts in Amharic texts on social media for Twitter and Facebook. First, calculate the similarity posted documents in a pair of sentences based on word similarity between the two sentences and also cluster based on the similarity results. After clustered posted documents to extract summary and pick the ranked important sentences from users post to select the summarization and finally to summarize each group of clustered documents and also identify the monthly posted documents to the group for the summary automatically to produce for the reader.

### 1.4.2. Limitation of the study

Due to the time limitation to summarize and extract Amharic posted texts from different posts that can be found on Twitter and Facebook to prepare few sample corpus to use for Amharic posts texts. The main drawback while the processing, the study is the unavailability of enough corpus for Amharic tweets and text document categories for natural language processing for the domain on the social media summary tool, particularly on Twitter and Facebook. And also the Amharic language was not its implementing tools for the news items posted text summarization online tools.

## 1.5. Significance of the research study

The result of this research that can be used for the users some of the things that have to be considered when creating the news items posted text summarization system are selecting the right documents for the sentences in posted texts, extracting the interesting information to find in the summary, submitting informs on the right time, avoiding redundant posting user posts and making sure to cover the whole users' Amharic posts without including irrelevant information on the summary sentences. The finding of this thesis as per the consideration of Amharic tweets on social media, it makes to consider the important documents to the summary related to user posts, management of time and attitude towards the core sentences to summarize and value of safe reading Amharic text without duplicate post documents to summarize for the users posted documents with once a month or all posts higher ranked sentences to pick and decide the size of the summary for the reader on social media. Due to the developments of the investigation, people have less time to use and less effort reading huge

volumes of information obtainable to satisfy their significantly less information need for the users as the use of automatic summary.

After finished for this research to use an input to the development of the news items Amharic posts Summarizer and has the importance to initiate further research in the area of Amharic language on social media and also the news items text summarization for others language. Additionally, it can also help to initiate the different Ethiopian languages can be referred to work the news items user posts can be summarized on the different language using on social media user posts and others language out of the country.

## 1.6. Structure of the Thesis

The structure of this thesis is divided as follows. In chapter one, the general introduction, the statement of the problem, objective of the study, scope and significance of the research are presented.

In chapter two, the major concepts literature review or related work for Amharic text summarization on social media, local and global related works for the user posts for Twitter and Facebook are discussed.

In chapter three, to discuss Amharic writing system within emphasis on the representation and processing of electronic format to describe some information to help with social media posted Amharic texts.

In chapter four, to discuss the research methodology and system designs to present the process of data preparation, tools, techniques of the summarization, the system design to explore and developed for news items posted Amharic texts and also, it presents the algorithms.

In chapter five, the tasks and concepts related to the social media on Twitter and Facebook Amharic tweets to implementation algorithm, experimentation and results, evaluation and discussion of Amharic texts for user posts, the news items documents over times are discussed, which includes the description the news items Amharic text summarization on the Twitter and Facebook posted data, the summarization process is described.

Finally, in chapter Six, the conclusion of the findings of the research and the recommendations and future work is presented.

# CHAPTER TWO

# LITERATURE REVIEW AND RELATED WORKS

## 2.1. Introduction

In this chapter to talk the literature review that is relevant to this research is revised and presented a summary news item posted Amharic texts from different sources of user post on the Twitter and Facebook. Also, the relevant concepts that are related to the research are also discussed in this chapter. The chapter presents the description of the summary, user posts as different sources of Amharic tweets, the summarization process of user posts, selection measures for summarization, evaluation methods for text summarization on the social media. There are many research works conducted, the users text summarization for local and global languages. In order to avoid repetition, the global researches done on news items user post text summarization on social media are discussed in this chapter, hence can be referred in section 1.2 in chapter one.

## 2.2. Text Summarization for News Items tasks

The work of the Eidheim[11] to propose the temporal summarization task, which is to inform readers of important novel information about a particular topic. For that the DUC and TAC Update Summarization tasks were designed that as a single pass collection process, processing all the documents at once, while in that year TREC temporal summarization tracks the task designed requires generation of continuous and immediate updates. As with the earlier work, sentences are the unit of selection. The author has been tested according to the requirements of the TREC 2014 temporal summarization way. His results had been evaluated using standard evaluation measures and compared to the results of the 2014 participants or users. The author experimented with several different methods of sentence selections based on language models, TF-IDF and word counting. One of primary issues the authors wanted to address the fact that earlier methods were based on the assumption that all the evaluated documents was relevant. The problems covered by his paper were very similar to the temporal summarization task and it shows that great care necessity be taken when selecting documents to ensure that the results are good. His using algorithm was a search engine and sentence ranking. His paper to show the post models solution achieved a very high score of the gain metric, but a low score of the comprehensiveness metric. A likely explanation for this result was that the strict selection of documents prevented the system from selecting unrelated documents, but also leaves out documents that could be relevant. Because the systems did not score very well in the comprehensiveness metric, it could be a good idea to improve the way documents are selected so that it is not as restrictive. The author had been done only English

news articles was considered when generating updates and the main reason for this is that the chunks and queries are all English, allowing other languages is a potential source of noise, but it is unlikely to contain any interesting information. It is also likely that a user monitoring an update stream would like the stream to be in his or her language, making it a reasonable decision to only look at documents in a specific language. For the sake of keeping the collection of documents small, non-English documents are removed by his the preprocessor of the tweets[12].

From the work of those authors Tan Xu, Paul McNamee and Douglas W. Oard **[3],** those papers, they focused on the sequential update summarization task that could be concerned to show the algorithm. This task involves simulating processing a temporally ordered stream of over 1 billion documents to identify sentences that are relevant to a specific breaking posted documents which contain new and important content. This paper they tried to measure the work by static unigram, cosine similarity between the sentence's unigram BOW (Bag-of-Words) term vector and the topic's initial, static unigram BOW term vector. Their analysis of those results to date to suggest several areas for future work, including optimizing both document and sentence selection thresholds; finding better patterns of similar (historical) posts on Wikipedia (e.g., by exploiting the Wikipedia category system). Another study, the work of Sayyadi[7] had proposed a variant of Hidden Markov Models to obtain an intermediate representation to a sequence of tweets relevant to a posted texts. Their approach does not use the continuous time stamps present in tweets and does not address the problem of obtaining the minimal set of tweets relevant to an event. His summarized opinions about entities in Twitter by mining hashtags' to infer the presence of entities and gathering sentiments from tweets. However, not all tweets contain hashtags, which make it difficult to gain sufficient coverage for an event. From the authors J. Yang and S. Counts[13] had proposed the Phrase Reinforcement Algorithm to find the best tweet that matches a given phrase, such as trending keywords. The algorithm first finds the most common phrase on one side of the search phrase, selects those posts that contain this phrase, and then finds posts as the most common phrase on the other side as well. They produce one tweet as a summary of one phrase while they propose to provide a set of tweets to summarize posts.

The work of the Yang and Ruan **[13]** had also proposed a framework of summarizing a stream of tweets. Their main focus is on creating a scalable approach by compressing the tweet stream to fit in limited memory, followed by the use of Nonnegative Matrix Factorization to find topics in the tweet stream. Since they do not filter the tweets for a specific post of interest, the topics discovered using their framework will only contain globally major posts. Their proposed framework finds a summary of a targeted post of interest. However, from the authors D. Metzler, C. Cai, E. Hovy, A. Way, and M. Rey[14] proposed a structured retrieval

approach from obtaining a set of tweets that are the most relevant to a posted texts. It uses a query expansion technique and also exploits the temporal correlation between related post words. The added benefit of their topic model approach is that using the time-variation of each topic for each post, they can measure how fast each aspect of the post decays.

According these authors Imran, C. Castillo, F. Diaz[15],to describe the social media platforms that provide active communication channels during mass convergence and emergency user posts such as natural disasters caused by natural hazards. As a result of that, first responders, decision makers, and the public can use that tweet information to gain insight into the situation as it unfolds. In particular, many social media tweet messages communicated with emergencies take timely, actionable information. However, the author involves solving multiple challenges, including informal messages, parsing brief, handling information overload, and prioritizing different types of information on Twitter found in social media that can be posted messages. Those challenges can be mapped to classical information processing operations such as filtering, ranking, aggregating, classifying, summarizing and extracting the tweet texts. And the author methodically to observe a series of key sub-problems ranging from the detection of posts as the creation of actionable and useful summaries. Previous research has shown that information which contributes to situational awareness is reported on Twitter (and other social media platforms) during mass emergencies. Now, those tasked with formal response efforts of local fire departments to international aid agencies are working to incorporate information broadcast on social media platforms into their processes and procedures. Many emergency responders and humanitarian officials recognize the value of the information posted on social media platforms by members of the public and others, is interested in finding ways of quickly and easily locate and organize that information that is of most use to them.

## 2.3. Temporal patterns text documents

The work of those Ks.R. Premlatha and T.V Geetha [16], for extracting the temporal patterns that is from texts requires to the expression for handling the types of temporal term categories were explicitly, implicitly and vaguely conveyed to the temporal information for the given document. The authors performed by using finite state automata (FSA) in the natural language expression are converted into a calender based timeline. The authors give the temporal expressions the first the temporal explicitly are examples like on the 18th of March 2009, in Nov 2010, 12 Jul 2011 etc. We applied to use this temporal in the news items of posting term, text summarization explicit expression to identify the posted documents by using date, month and year format to group by monthly for the summary posted document in social media posts. The second some temporal implicitly expressions that contain varying degree of the indexical attribute examples like last Sunday, three

weeks ago, on Monday, on Tuesday etc. those cases there should be known the reference time and also the verb tenses. Finally the vague temporal expressions[17] is like July, after several weeks before etc. In the work of this article, to peak event that should be analyzed through the time stamp of the start and end time of the particular event for the topic based clustering for the calender model which go to enable the temporal intervals posted documents was carried out their work for the papers. Those authors to preprocess the used of the clustering such as stop word removal, stemming were performed the document represented using tf-idf representation which is used for clustering for the similarity. The authors describe the clusters were formed into using the cosine similarity between all vectors and comparing the intervals of the two documents within all pairwise the intervals were compared the similarity was calculated based on the size of the intervals and the overlap between them as given the following formula.

$$\text{Similarity} = 2 * \text{overlap} / \text{size of the intervals}$$

Thus the authors to find the peak event and also in the measuring similarity with events, peak the event analysis based on the time stamp of start, end event duration and those the similarity have been calculated using Allan's approach. Their shown as the documents event matrix assigning the news items of temporal relations[17].

| *Category* | *terms* |
|---|---|
| baseterm | day, week, weekday, month, monthname, quarter, season, year, decade |
| indexical | yesterday, today, tomorrow |
| internal | beginning, end, early, late, middle |
| determiner | this, last, next, previous, the |
| temporal | in, on, by, during, after, until, since, before, later |
| post modifier | of, to |
| numeral | one, two . . .. |
| ordinal | first, second . . . |
| adverb | ago |
| meta | throughout |
| vague | some, few, several |
| recurrence | every, per |
| source | from |

***Figure 1: The temporal terms categories*** [16]

## 2.4. Text summarization for Amharic texts

In this author of Meles[18],had described and proposed to his papers for Amharic texts the two generic texts summarization approaches. For his the first technique, to put the topics Latent Semantic Analysis (LSA) and the second technique mixes the graph based ranking among the Latent Semantic Analysis algorithms to identify the topics of a document were used to select the semantically the main sentences for the summary generation. The author used the algorithm was Latent Semantic Analysis and graph based ranking algorithms to explore his work. His work evaluates to propose the performance of the summarization approaches and his prototype Amharic news text summarization system. For this author, to evaluated the summaries systems with manual summaries were generated by six independent human evaluators to be taken the evaluator for the experiments. He prepared the dataset corpus used for evaluating the summarization system was 50 Amharic news's from Ethiopian news reporter the items were in the range of 17-44 sentences. His results to evaluate by comparing the system summaries of their corresponding manual summaries of the results.

The work of Addis[19],had proposed an open sources of customizing by Amharic texts automatic summarization using an open text summarizer tool of two ways of execution the two experiments, the first one experiment is done without changing the code of the tool and the second is done for the changing the Porter stemmed tool for the Amharic stemmed algorithm. His work uses the frequency of terms to determine the relative importance of a sentence in a text. His papers evaluation of the experiments was producing 90 news articles and to test its performance for the summaries for each rate at 10%, 20% and 30% extraction rates for the results. This author to evaluate the system was evaluated using the subjective and the objective evaluation.

According to the work of Eyob [20],had proposed to topic-based Amharic text summarization to investigate the six algorithms to explore as the use of terms by concept matrix to implement for his thesis. His algorithms to take two common steps, the first step, to identified the keywords for the documents were used to select the term of use of the concept matrix to find the document. The second step, for the sentences to find the best keywords contains that were selected for presence in the summary. For this author to take the experiment with news articles in Amharic texts to explore the algorithms for selected the first sentence of the document for inclusion in the summary of the Amharic news texts. His evaluated in the paper by using for the precision/recall for summaries of 20%, 25% and 30% extraction rates to evaluate for the news articles. The author of comparing his system with the previous methods of developed for other languages based on topic modeling approaches summarization that had been used his Amharic data set of the papers.

Since the work of these authors of Kifle and Martha Y[21] had investigated for Amharic texts to an automatic single document summarization tasks for the graph based automatic Amharic text summarizer were proposed. They were to work the generic and domain independent graph based model could successfully to make extracts from Amharic texts for their papers. These authors uses the two graphs based ranking algorithms were introduced, their thesis used for PageRank and HITS that we're using the two sentence centrality measures and sum the relation between the sentences in a text for a graph that show for the results from their experiments. Their developed for the domain and independent extractive summarization techniques was focused on the extraction based summarization to extract the sentences, paragraphs, phrases and words from the original source of the documents. They worked to prepare the data sets for the experiments shown 30 news articles used on economics, politics, society, and sports were conducted for Amharic news articles from collected for Ethiopian reporter news web sites and Addis Admas to test its performance for the summaries for Amharic news articles[21].

In this section for text summarization for Amharic texts, we review the works done in single or multi-document summarization that relied on many different researchers focus to do for Amharic news texts to investigate the model and different authors to work for Amharic and other languages in Ethiopia and outside the country that had been done for developing the research for Amharic text summarization systems. But our research is not being considered the news articles in Amharic texts, we focused on posting documents on social media users posted Amharic texts on Facebook and Twitter user posted news items Amharic texts over a time that could be duplicated post documents to identified to calculate the similarity, cluster and finally summarize the clustered posted documents.

## 2.5. Single and Multi-document text summarization for posting texts

As described the Amharic posted texts to extract Amharic text on the social media. Like the political protests, natural disaster, accidental posts and others to be tokenized and summarize without duplicate Amharic tweets documents, considering to cluster the similar posted texts are measured and others included for pre-processing (like sentences segmentations, tokenization, stop word removal, punctuation mark, stemming method) was done in the extract Amharic tweets and summarized the news items user posts for the Twitter and Facebook. So, summarizing a collection of thematically related documents poses for Twitter and Facebook several additional challenges for filtering Amharic tweets. In order to avoid repetitions Amharic text posts, one has to identify Amharic tweets and locate thematic overlaps. One also has to decide what to include of the remainder, to deal with potential inconsistencies between posted documents, and when necessary, to arrange

posts as various sources of a single timeline. Multi-document[2] summarization is the process of producing a single summary of a collection of related documents and also it is useful in combining information on multiple sources. In[22],multi-document summarization system is developed for the web context. Information may have to be extracted from many different articles and pieced together to form a comprehensive and coherent summary. One major difference between single document summarization and multi document summarization is the potential redundancy that comes from using many source texts. The solution presented in [22]is based on clustering the important sentences picked out from the various source texts and using only a representative sentence from each cluster. A multi-document summary is a short-term representation of the essential contents of a set of related documents. The relation between the documents can various types, for example documents can be related because they are about the same entity, or they discuss the same topic, or they are about the same post or the same post type. Fundamental problems when dealing with multi-source input in summarization are the detection and reduction of redundancy as well as the identification of differing information. In the multi-document summarization problem is studied in the context of multi-source information extraction in specific domains. Here templates instantiated from various documents are merged using specific operators aiming at detecting identical information. In an information retrieval context, where multi-document summaries are required for a set of documents retrieved from a search engine in response to the query [7] can be applied. The method scores text passages (e.g., paragraphs) iteratively taking into consideration the relevance of each passage to a user query and the redundancy of the passage with respect to summary content already selected. In the case of generic summarization, computing similarity between sentences and the centroid of the documents to summarize has resulted in competitive summarization solutions[22].

Sentence ordering is also an issue for multi-document summarization. In single document summarization it is assumed that presenting the information in order, this information appears in the input document would generally produce an acceptable summary. By contrast, in multi-document summarization particular attention has to be paid to how sentences extracted from multiple sources are going to be presented. Various techniques exist on dealing with sentence ordering, for example, if sentences are time stamped by publication date, then they could be presented in sequential order. However, this is not always possible because recognizing the date of a reported post is not unimportant and not all document to contain a periodical date.

[2]Multi-document summarization is useful in combining information from multiple sources and also the number of source articles may be very large. Information may have to be extracted from many articles and pieced together to form a comprehensive and coherent summary.

Based on the number of the document summarization used there are two types of document summarization. These are a single document summarization and multi document summarization. A single document summarization, is as the name implies, takes a single document and presents the most important content of a condensed manner for the needs of further task. A summary is a single document summary, if a summary is prepared for one document only. Whereas multi document summarization extracts information on many text documents written about the same topic. The complexity involved in the integration of different documents about a topic to extract a single summary, makes multi document summarization more complex and wider concept than the single document summarization. Multi-document summary when a summary of one topic is prepared for many different documents. For these reasons, multi-document summarization is much less developed than its single-document and various methods have been proposed to identify cross-document overlaps from different researchers. Summons(which is a paper issued by informing a person that a complaint has been filed against it),or order, a system that covers most aspects of multi-document tweet summarization[23], takes an information retrieval approach. Assuming that all input documents is parsed into templates (whose standardization makes comparison easier), summons clusters the templates according to their contents, and then applies rules to extract tweets items of major imports. In contrast, the problem of organizing information on multi document summarization so that the generated summary is coherent has received relatively little attention[24].

While sentence ordering for single document summarization can be determined from the ordering of sentences in the input texts. This is not the case of multi document summarization where summary sentences may be drawn from different input texts and also parse each sentence into a syntactic dependency structure (a simple parse tree) using a robust parser and then match documents, using to paraphrase rules that alter the trees as needed. To determine what additional material should be included, first identify the tweets the most relevant to the user's query and then estimate the marginal relevance of all remaining tweets using a measure as Maximum Marginal Relevance (MMR). Summons deals with cross-document overlaps and inconsistencies using a series of rules to order templates as the story unfolds, identify information retrieval, identify model inconsistencies (decreasing passing tweets), and finally produce appropriate phrases or data structures for the language generator. Multi-document summarization[25]poses interesting challenges to single post documents and also the information overloads faced by today's society poses great challenges to researchers that want to find a relevant piece of information. Automatic summarization is a field of computational linguistics which can help humans to deal with this information overload by automatically extracting the idea of documents.

Summarization techniques are categorized into two major categories extractive or abstractive summarize. Extractive summarization is assigned an extract the sentences with the highest matching measure to form the summaries. Extraction method is more practicable in many applications of text summarization. In we apply the extraction approach, there are different methods of measuring the importance of a sentence to be used in the summary. Like most researchers in this field, the extractive summarization framework of used in this approach. In this summarization task, the automatic system extracts posted documents from the entire collection, without modifying the documents themselves. Examples of this include key sentence extraction, where the area is to select individual sentences with a document that hold the algorithm to check for the input corpus to find the similarity and to cluster, and document summarization, where the area is to select whole sentences, to create a short paragraph summary to produce for the output results. Extraction techniques merely copy the information considered the most important by the system to the summary (for example, key sections, top score sentences). Whereas abstractive summarization, on the other hand, uses a certain degree of understanding of the content expressed in the original documents and creates the summaries of information fusion. Abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what a human might generate. In general, abstraction can condense a text more strongly than extraction, but the programs that can do this are harder to develop as they require the use of natural language generation technology, which itself is a growing field. While some work has been done in abstractive summarization (creating an abstract summary like that of a human), the majority of summarization systems are extractive (selecting a subset of sentences to place in summary)[24].

An abstract is a brief summary of a research article, thesis, review, conference proceeding or any in-depth analysis of a particular subject and is often used to help the reader quickly determine the paper's purpose. Abstract always appears at the beginning of a document acting as the point-of-entry for any given academic paper. In our research, we couldn't use for abstraction summarization for the posted news items Amharic documents for the user posts. Such a summary might contain sentences, words used to present in the original document to summarize, and also, we considered the documents for summary posted documents on the social media for Twitter and Facebook user posts. The planned strategy that is classified or clustering the posts to be important and not important, each Amharic text gives a score that determines whether this tweet may candidate in the summary or not. Sentence ordering can also be conceived as representing the different topics to be addressed to the summary. For example, a clustering algorithm can be used to identify topics in the set of input documents and discover in what orders the topics are presented in the input documents **[3]**, this in

turns could be used to present sentences in an order similar to that observed in the input set. A probabilistic approach to sentence checking seeks to estimate the likelihood of a sequence of sentences. It tries to find a locally optimal order by learning to order constraints on pairs of sentences **[27]**. An important the study show that for the newspaper article type, even some very simple procedures provide essentially perfect results. For example, taking the first two or three paragraphs of the most recent text of a series of texts about an event provides a summary equally coherent and complete as that produced by human abstractors. Obviously, this cannot be true of more complex types of summary, such as biographies of people or descriptions of objects. Further research is required on all aspects of multi-document summarization before it can become a practical reality[26].

According to this author Kalita[27],to help people that read Twitter posts or tweets, Twitter provides two interesting features of his paper an API that allows users to search for posts on the twitter that contain a topic phrase and a short list of popular topics on the social media. A user can perform a search for a topic and retrieve a list of the most recent posts that contain the topic phrase. The author to define a Twitter post or tweet is at most 140 characters long and in this study they only consider English posts. Because a post is informal, it often has informal syntax, non-standard orthography or non-standard spelling, and it frequently lacks any punctuation. The difficulty in interpreting the results is that the returned posts as the Twitter are only sorted by most recent, not relevancy. Therefore, the user is forced to manually read through the posts in order to understand what users are primarily saying about a particular topic. His paper to discuss ongoing effort to create automatic summaries of Twitter trending topics and also algorithms that can be used to pick the single post that is representative of or is the summary of a number of Twitter posts. Since the posts returned by the Twitter API for a specified topic likely represent several sub-topics or themes, it may be more appropriate to produce summaries that encompass the multiple themes rather than just having one post describe the whole topic and his paper extends the work significantly to create summaries that contain multiple posts on the Twitter. The author compared his multiple post summaries of the ones produced by the leading traditional summarizers, algorithms that produce summaries by selecting several posts from a given set.
According to the work of these author Inouyes, and J. K. Kalita[28] their paper described that an algorithm for summarizing microblog documents on Twitter. Firstly, they presented algorithms that produce single-document summaries, but later extend them to produce summaries containing multiple documents and also they evaluated the generated summaries by comparing them to both manually produced summaries and, for the multiple post summaries, to the summary results of some of the leading traditional summarization systems. The author discussed to overflowing with information on twitter, they taught that just being able to search for

a text tweets and receive user tweet the most recent posts whose text match the keywords are not enough. To get a picture of what is being written about an area in terms of a large number of posts, it is necessary to obtain a summary of those posts. If he used the traditional term document to refer to each post or tweet, for his purposes, obtaining the summary of a large set of documents on certain topic specified by keywords means selecting the most significant one or more documents from the site. It is also possible to construct new documents tweets by piecing together parts of original documents or posts. Data reduction or summarization is essential for understanding and exploring any data on the Twitter. Even if data summarization is used only in the initial phases of data analysis and exploration, it is critical to providing awareness about what questions would be interesting to ask and what other data analysis methods could be useful. In the case of his defined the microblogs, summarization is necessary before any other data analysis can be done because it would simply be impossible to read through millions of tweets. In addition, summarization can be a useful exploratory or investigative tool by itself. Summarization can represent the tweet document with a short piece of text covering the main topics for the user posts, and help users select through the Internet, the most relevant document, and filter out redundant information on the social media to find the relevant tweets[29]. So, the author to describe the document summarization has become one of the most important research topics in the natural language processing and information retrieval communities.

## 2.6. Interpretation of language for posting text documents

During the interpretation of Amharic texts of the social media tweets, text documents, the documents to identify as important, Unicode and UTF-8 encoding are fused (which is joined the language), represented in user posts in Amharic tweet terms, and expressed using a new formulation, using concepts or words not found in the original text. It must interpret the input in terms of something important to the text. But acquiring enough previous domain knowledge is so difficult that summarized to date, texts have only attempted the tweets it in a small way. At first look, the original representations used in information retrieval or other interpretative structures in terms of which to represent user post stories for summarization matching tweets to take one similar post to add one on the Twitter and other to remove the repeated Amharic tweet, hold some talent. But the difficulty of building such structures and filling them makes large-scale summarization impractical at present[30].

The corpus contains a set of documents where each sentence and sentences fragments or splits have been annotated as either essential or nonessential; this kind of dataset could be of help for developing sentences selection and sentence to reduce processing. For this reason, corpora are normally used to assess extracts the important Amharic posts only from the corpus to extract relevant Amharic text tweets. There are three

techniques used to build annotated corpora: manual, automatic and semi-automatic. Manually produced corpora require human power, Dong et al. [26] exploited Twitter to detect and rank fresh URLs that have possibly not been indexed by Web search engines yet. Lately, Chen et al conducted a study on recommending URLs posted in Twitter messages and compare strategies for selecting and ranking URLs by exploiting the social network of a user as well as the general popularity of the URLs in Twitter and also judges to read each text from the corpus and identify the important units. The way judges to decide that a unit is important to depend very much on the category of text and the annotation guidelines provided for them[31].

Automatically produced corpora trust with the idea that very often humans produce summaries of cut-and-paste operations, and therefore it must be possible to identify a set of sentences from the document which cover the information on the human produced summary. This author Marcu**,** said to the employed an acquisition procedure which eliminates sentences from the whole document that did not reduce the similarity between the summary and the remaining sentences. When it is not possible to further to reduce the texts in this way, the linguistic structure of the reduced document are used to eliminate more sentences[26].

Semi-automatically produced try to deal a compromise between the time consuming process of manual annotation, and the errors introduced by the automatically annotated corpora presents an annotation tool which allows the annotator to run several automatic methods which identify the important sentences and then post-edit their results[32].This makes the system to generate the summary according to the user predefined compression rate and also it is evaluated and the results are compared with that obtained by the well-known single and multi-document summarization.

## 2.7. For the summarize the user posts on the Twitter and Facebook

We evaluate the quality of a summary was the growing body of literature on this interesting to suggest that summaries are tasked and category specific and users that no single measurement covers all cases. In section 2.7.1 we describe a few evaluation studies for Twitter and in section 2.7.2 we develop some theoretical background for the tweet texts.

### 2.7.1. Evaluation techniques for previous studies for the summarization

The most existing evaluations of summarization systems are fundamentally the typical, the evaluators create a set of summaries, one for each test text, and then compare the Summarizer's output, measuring the content overlap that often by sentence recall and precision. To simplify evaluating extracts**[33],** independently developed an automated method to create extracts corresponding to abstracts for the Twitter. The two evaluation methods of the text summarization such as intrinsic and extrinsic evaluation methods. An intrinsic

method is the evaluation of the system for summaries according to have the evaluators to take the ratio of some scale of interval the extraction rate (readability; informativeness; facility; coverage and others). It was prepared by creating the human or ideal summaries of the given input text document for comparing the summary of the summarizing system and the human summary of the evaluators. The evaluator used for measuring the average scores found from each evaluation criterion for the given document. Thus, none of them to measure is completely acceptable, since there is no only one correct ideal summary of any given document. The others evaluation method is extrinsic evaluation for measuring the acceptability and also the efficiencies of the automatic summaries of documents to achieve the tasks and easily to motivate, this evaluation method the main difficulty to ensure that are applied to correlate or links with the task of the performance efficiency for the summary documents. Some of the intrinsic evaluation methods to evaluate the summaries as follows.

**1. Coherence and structured for Summary**

This is one of the evaluator to measure the extraction rate based methods to use the flow of the information to be structured by using the cut and paste processes of the documents on phrases, sentences or paragraphs that produces the result in a summary extracted for the documents which results in coherence to get for the documents. For this measured by using the human summary of rank sentences for structure and coherence to compare the ranks sentences with the scores for reference summary or original sentences in the given document.

**2. Informativeness for Summary**

For the informative summary of comparing the system summary generated the documents within the input texts for summarized the key ideas about manual summary that are included in the automatically summaries and the summary information are presented in the input texts. The informativeness of the summary to measure the summaries are compared to a reference summary information measuring is presented in the system summary producing results of the summary.

**3. Sentence Precision and Recall**

It is a standard measures for information retrieval in terms of sentences in the given document. So, precision of measures by using how many of the sentences in the system summary generated and also in the reference human summary are producing the results. An others hand, recall for the sentences to measure for how many of the sentences in the reference ideal summary are extracted from the system summary of the documents[34]. For this summary information gathering in large input text corpus the time and effort required to post to edit the system summary generated for the specific tasks in the system summarization, which is measurable tasks

for the documents. Still, all the systems performed extracts only, there by simplifying much of the scoring process of IR-like recalls and precision measures against human extracts, the wealth of material and the variations of analysis contained in**[22]** underscore how little is still understood about summarization evaluation. And also, the evaluation resources consist of metrics for measuring the content of automatic summaries of reference summaries. Twitter observatory that allows observing, searching, analyzing and presenting social media is introduced as a part of the research and illustrative examples of using his proposed pipeline show how the Twitter Observatory implementing the pipeline can support the user in interaction with the social media data[35].

According to this author Sharifi et al. [27] described that his algorithms process collections of short posts as specific topics on social media the well-known site called Twitter and create short summaries of those posts as the Twitter. The goal of his research is to produce summaries. His paper evaluated the summaries produced by the summarizing algorithms, compare them with human produced summaries and obtain excellent results. Since this authors P. Meladianos[37] to deal with the task of sub-posts detection in evolving twitter posts using posts collected from the Twitter stream and also by representing a sequence of successive tweets in a short time interval as a weighted graph of words, they were able to identify the key moments sub-posts that combine a post using the concept of graph degeneracy on the twitter. They were selected a tweet to best describe each sub-post using a simple yet effective heuristic (which is used for experimental). They evaluated for their paper and its approach using human generated summaries containing the actual important sub-posts within each post and compare it to two baseline approaches using several performances metrics such as curves and precision and also recall performance.

### 2.7.2. Basic measures for summarization user posted on the tweets

The much of the complexity of summarization user posted evaluation arises from the fact that it is difficult to specify what one really needs to measure, and why, without a clear formulation of what precisely the summary is trying to capture. We general some considerations here, to be a summary, user posts for the Twitter Amharic tweets, the summary must conform two important requirements for tweets[19]:

- It must be shorter than the original input text.
- It must contain the important information on the original (where importance is defined by the user), and no other, totally new, information.

So, text summarization is one of the natural language processing (NLP) application that propose to extract the most important information about a source to produce a condensed version of a particular user post of the twitter task. In order to generate a summary of a document, they have to identify key pieces of information

existing on a document, ignoring the redundant information posted and reducing the complexity of texts details. During the evaluation of the summarization systems and human summaries that must be qualitatively and quantitatively measured in order to subjectively and objectively respectively. One of these measures is the compression ratio (CR) or compression rate which analyses how much the shorter the summary is than the original input text. CR is calculated as the ratio of the summary of length of full text. The other measure is the retention ratio (RR) which analyses how much the information is retained in the summary. RR is also referred as omission ratio and is calculated as the ratio of information on the summary divided by the information about the full input text. A good summary has a small CR closer to zero and large RR values closer to one. One can define two measures to capture the extent to which a summary S follows to these requirements with regard to a text T:

(1) Compression Ratio: CR = (length S) / (length T)

(2) Retention Ratio: RR = (info in S) / (info in T)

However, this choice to measure the length and the information, content, that could say that a good summary which is the compression ratio is small (tending to zero) while the retention ratio is large (tending to unity). It could be evaluated its characterize summarization systems by plotting the ratios of the summaries produced under varying conditions that were disused the above author.

### 2.7.2.1. The co-selection Summary Evaluations

The co-selection method is that to discuss here are the simplest of all evaluation measures co-selection sentences and this simplicity creates at a value score to check the intersection with the system and human summary of sentences[20]. For that the main problem is the difficulty in accounting for variations in what humans consider ideal summary sentences in the documents. The summary evaluation for using the co-selection measure were to take from information retrieval evaluations techniques and for describing the formula for recall (R), precision (P), and F-measure given to calculate the system and the human selected the fraction of sentences that the system has chosen from the total of sentences found in the ideal summary as follows to formula.

$$R = \frac{|system\ and\ human\ choice\ overlap|}{|sentences\ chosen\ by\ human|} \qquad (2.1)$$

Precision (P) measures the fraction of system summaries that are correctly chosen.

$$P = \frac{|system\ and\ human\ choice\ overlap|}{|sentences\ chosen\ by\ system|} \quad (2.2)$$

F-score (F) is the harmonic mean of recall and precision.

$$F = \frac{2 * P * R}{P + R} \quad (2.3)$$

There were many works being done on the area of summarization[38] on social media by Twitter in western world language for news items summarization. But most of the work being done for major technology languages like, English, Chinese ,German and French[22].

Due to those authors [39] that described the shortness of tweets and also TwitIE makes the assumption that each tweet is written in only one language. The choice of languages used for categorization is specified through the arrangement file, complete as an initialization parameter. The authors take three tweets one English, one German, and one French. TwitIE TextCat was used to allocate automatically the language feature of the tweet text (denoted by the Tweet annotation). Those give a collection of tweets in a new language, it is possible to train TwitIE TextCat to support that new language as well and also that is done by using the fingerprint generation (which is patterned to generate) included in the language identification plugin. It builds a corpus of documents and reliable tweet language identification allows them to only process those tweets written in English with the TwitIE English and named entity recognizer. That is achieved by making the execution of these components conditional on the respective tweet being in English, by using a Corpus and also provides the category and named entity recognition in French and German, it is possible to extend TwitIE towards these languages with some training and adaptation effort. The authors to evaluate their paper by using which trusts with n-gram frequency model to discriminate between languages. These days information summarize systems on social media are highly connected with human daily activities. There are many search engines for searching text documents, video, audio, and pictures could be posted the user posts for social media. Additionally, there are special purpose search engines like Facebook, Twitter, which specifically works for online social media of user posts were posted.

## 2.8. The Use of Social Media in the Summarization posted documents

Social media was defined by [6]as a group of Internet-based applications that build on the ideological and technological foundations of Web 2.0, and that allow the creation and exchange of user posted generate content. The use of social media has exploded to recent years and the high availability of such information is then used by many researchers. One of the most popular social media platforms is Twitter and Facebook,

Twitter is currently the most popular microbiology service in the world. Apart from using Twitter to connect with friends and family, people also use Twitter and Facebook daily to share news and knowledge and discover the latest information and updates about various topics of interest. According to the Twitter Statistics measured on December 19th, 2015, the number of active Twitter users reached 554,750,000 and there are 58 billion of tweets posted every day by Twitter Statistics," 2015. As this statistic of Twitter is an online social networking service that enables users to send short 140-character messages called tweets.

According to recent social media industry figures, Twitter currently ranks as one among the leading social networks worldwide based on active users. Registered users can read and post tweets as well as follow other users via update feed. As of January 2016, Katy Perry was the most followed star on Twitter with more than 80 million followers. Recent social media data also prove that Twitter usage is becoming increasingly prominent during posts. Live-tweeting cultural happenings such as sporting posts have become a popular way for consumers to engage online with others while sharing their thoughts on current involvements [28].

The systems that were previously described show evidence of the possible on Twitter and Facebook for user post finding in the social media. However, given a scheduled post, it was possible to identify its key moments using Twitter and Facebook posted user posts. The finding of the important posted that has not yet received much attention by the research community[5]. This author of developed one of the first sub-post detection systems. The authors used posts as Twitter in order to generate summaries of American football games and also the author used the tweet rate to identify sub-posts in American football games, using a sliding window that if the fraction of the tweeting rates of the sub-windows exceeds a beginning, a possible sub-event is identified. Subsequently, a lexicon-based recognition method is employed to label the sub-post as game-related or filter it out in case it is a random sub-post. The author algorithm learns the underlying structure and vocabulary of a football game using a modified HMM. However, his algorithm is valid only to recurring posts as the HMM must be trained for similar posts to reach high performance standards and it is not effective against previously unseen types of posts.

This author[40], to defined a tweet is limited to 140 characters and most references to other web pages are abbreviated via URL shortening services (e.g., http://www.tiny.cc/ and http://bit.ly/ ) so that readers could not guess where the references point at. This is an interesting feature of users and other users add as many trending areas as possible to appear in the top results from any search for Twitter. The drop in the visitor count of tinyUrl.com shows that Twitter's impact on the third party applications is not negligible. Twitter is becoming the next-big-thing on the web, and we can compare it to early giants like Google, YouTube and Facebook.

In this author to work[41] , the author of described his goals is to design novel features "identifying valuable information on Twitter during natural disasters" that can be used as input to machine learning classifiers in order to automatically and accurately identifying informational tweets from the rest in a timely fashion. His approaches used for machine learning algorithms that would be discussed for URL extraction are used extensively in tweets to link to external sources that could not ordinarily acceptable to the long-restricted structure of a tweet. However, URLs found in tweets, are shortened in order to accommodate for the length restriction. Inherently, a few features can be extracted from the URL itself, considering that each shortened URL has a base domain and a randomly generated code appended to that (i.e. https: "t.co/ [code]"), which, when clicked on, will redirect to an actual web page. The author developed a method to bypass this system through having his features extraction method request for the shortened URL to the legitimate web page, returning an analyzable URL. From this URL, He developed a method of analysis to distinguish between informational and conversational tweets. The URLs were analyzed for hyphenated or underscore separated article names in the path, indications of dates, and the presence of reliable news sources that was extracted from the dataset. He's used 10-fold cross-validation for evaluation, using Naïve Bayes classifiers as implemented in the wake toolkit and also the results of the designed feature set were compared with the outcomes of the "bag of word", along with the results of a combined result set of "bag of words" and the designed feature set.

## 2.9. Sentence Extraction by tf-idf and Position Weighting

In this article[44] to describe the sentence extraction via using tf-idf, they would be discussed the posted Japanese Newspaper in order to create a summary and his system is implemented with the sentence extraction approach and weighting strategy to mine from a number of documents. They created an experimental system for the Japanese Summarization to compute the importance value of each sentence based on Japanese newspaper terms. He's used the important sentences whose sum of characters exceed the restricted character amount are eliminated and the remaining sentences are then sorted as they appeared in the original document. The author of summarization to used single and multi-document summarization to implement the different evaluation between a long summary and its short summary is more remarkable. In order to improve his results some semantic information on summarization may be required to reduce the redundancy and to make a constructive summary. Text summarization is an important activity in the analysis of a high volume text documents and also it has a number of applications, recently number of applications use text summarization for the betterment of the text analysis and knowledge representation. In this article a frequent term texts summarization algorithm is designed and implemented in Java and to designed algorithm is implemented

using open source technologies like Java, Porter stemmer etc. and verified over the standard text mining corpus. Japanese summarization to compute the importance values for each sentence based on Japanese newspaper term[45].

### 2.9.1. Hybrid TF-IDF documents

TF-IDF stands for term frequency-inverse document frequency, is a numerical statistic which reflects how important a word is to a document in a collection or corpus is occurring, it is the most common weighting method used to describe documents in the Vector Space Model (VSM), particularly on IR problems. In a hybrid TF-IDF algorithm development and the idea of the algorithm is to assign each word, sentences within a document a weight that reflects the words, sentences the most important to the document. The sentences are ordered by their weight from which the top sentences with the most weight are chosen as the summary. In order to avoid redundancy or duplicate posted words the algorithm selects sentences and tokenize, the next sentences or terms and checks it to make sure that it does not have a similarity with a given threshold with any of the other previously selected because the top most weighted tweets may be very similar[46]. We can calculate the term frequency of a word as the ratio of the number of times the term or word occurs to the document to the total number of words with the document. The tf*idf of term t in document d is calculated as:

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t$$

TF-IDF Algorithm (tf-idf) is basically a mathematical statistic that is meant to show how important a word is to a document in a collection of documents. In information retrieval systems, it is used as a weighting factor. At the number of times a word appears in the document increases, the tf-IDF value increases proportionally. But this tf-idf value is decreased by the frequency of the word with the collection. This helps to take into account the fact that some words appeared more frequently in general. For the value of term frequency tf(t,d), the most easy way to go is to use the frequency of a term of a document, i.e. the number of times that term t repeats in document d. If ft, d, denotes the raw frequency of t then the simple tf scheme can be given as tf (t, d) = ft, d.

The inverse document frequency basically measures the amount of information provided by a word, that is, whether the term is common or rare across all documents in the input files. The IDF is a measure either the term is common or rare (infrequently) for all documents in the corpus. It is a logarithmic obtained value and it is obtained by dividing the total number of documents by the number of documents containing the term,

and then taking the logarithm of that quotient and the log of this term is calculated to a value obtained is the IDF.

$$\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Then tf–idf is calculated as:

$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \cdot \mathrm{idf}(t, D)$$

The inverse document frequency (IDF) is a measure of how much information on the documents provides, that is, whether the term is common or rare (some words such as common stop words are frequent that words do not help discriminate between one document over another) across all documents. It is the logarithm (the logarithm is taken to balance the effect of the IDF component in the formula) scaled inverse fraction of the documents that contain the words obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient.

$$\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

With N: total number of documents in the corpus $N = |D|$

$|\{d \in D : t \in d\}|$: Number of documents where the term t appears (i.e. $(t, d) \neq 0$). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to. $1 + |\{d \in D : t \in d\}|$ Term Frequency -Inverse Document Frequency (TF-IDF) is a statistical weighting technique that has been applied to many types of information retrieval problems. For example, it has been used for automatic indexing[47], query matching of documents and automated summarization[6].

Generally TF-IDF is for a direct automated summarization, the application of TF-IDF is fairly basic and the idea is to assign each posts sentence within a document a weight that reflects the sentence's important within the document. Once each sentence has been weighted, the sentences are ordered or condense to one single post by similarity between post sentences k sentences with the most weight are chosen as the summary.

In this technique, the weight of a term is determined by the following formula:

$$TF\_IDF = tf_{ij} * \log_2 \frac{N}{df_j}, \qquad (2)$$

where tfij is the frequency of the term Tj within the document Di , N is the total number of documents, and dfj is the number of documents within the set that contain the term Tj. To handle this situation **[2]**, we redefine TF-IDF in terms of a hybrid document. One other important contribution to the Hybrid TF-IDF equation is its normalization of words of a document occur, which allows it to carefully control the overall target summary length. As described by Singhall et al.[48], had established TF-IDF is very sensitive to English text document length and often over weights terms from longer documents for capital and small characters. In our application of TF-IDF, we observed not the same effect in Amharic language like English words. Without a normalization word, established TF-IDF, given the most weight to the longer documents since the weight of a document is the simple sum of the weights of the composing words. Therefore, initially our implementation of TF-IDF resulted in simply the longest documents always presented the most weight. The given below is the research study of the related literature about these approaches and systems for multi- document summarization.

A. Similarity measure: The cosine similarity measure is one of the common techniques used to measure similarity between a pair of sentences vectors. Here sentences are represented as a weighted vector between the sentences in the posted documents on social media from Twitter and Facebook.

B. Cluster based method: Basically, clustering is to group similar sentences score values of their classes. For clustering of multi documents, these documents refer to the sentences and the cluster that a sentence belongs to be represented by classes of the group.

C. Word frequency: The basic idea of using word frequency is that important words are found many times in the document. Tf and idf is one of the most common measure used to calculate the word frequency. After this we calculated the combine the tf and idf to score the important of the word to find in the documents to the summary.

D. Feature based method: The extractive summarization types include for identifying the relevant sentences from the text and put them together to create an accurate summary. Some of the features that are considered for the selection of sentences are the significant word location of sentence, length of sentence from the input documents etc.

# CHAPTER THREE

# AMHARIC CHARACTER REPRESENTATION AND WRITING SYSTEM

## 3.1. Introduction

This chapter to discuss for a brief description of the Amharic writing system by focusing mainly on the electronic data representation of Amharic characters and the script for Amharic language on social media (Twitter and Facebook) that have to require digital storages in the language for user posts**.** The Ethiopia with some around 70 million populations is the most populated African country and harbors more than 80 different languages. Three of these are dominant: Oromo, a Cushitic language spoken in the South and Central parts of the country and written using the Latin alphabet; Tigrinya, spoken in the North and in neighboring Eritrea; and Amharic, spoken in most parts of the country, but mainly in the Eastern, Western, and Central regions[49]. The origins of the Amharic language are found back to the 1st millennium B.C. It is supposed that they are the descendants of King Solomon and the Queen of Sabea[49]. From this immigrant from southwestern Arabia crossed the Red Sea into present-day Eritrea and mixed with the Cushitic population. New languages formed as a result of this union, e.g., Ge'ez. Ge'ez was the classical language of the Axum Empire of Northern Ethiopia. It existed on the 1st Century A.D. and the 6th Century A.D. When the power base of Ethiopia shifted from Axum to Amhara between the 10th Century A.D. and the 12th Century A.D, the use of the Amharic language spread its influence, hence becoming the national language. Amharic is a Semitic language that uses a script which originated from the Ge'ez alphabet. It has 33 basic (33x7) characters of each having 7 forms of each consonant-vowel combination. Unlike Arabic, Hebrew or Syrian, the language is written from left to right. Amharic is also one of the most widely studied languages in Ethiopia. It is a field of study at the B.A. and M.A. level and in the school curriculum it is taught as a subject in most elementary and secondary levels of education. In the capital of Ethiopia, Addis Ababa, and in most major towns, it is the medium of instruction in primary level education. Because of its past and present role, Amharic has served as a medium of study of Ethiopian culture and society. Knowledge of the Amharic language is essential to understanding Ethiopian culture. Amharic is very useful for scholars in anthropology, history, and archaeology as well as in linguistics, since Ethiopia is a land of great history and treasures. The discovery of Lucy, the three-million-year-old skeleton, or "dink'nesh" in Amharic ("You are wonderful"), in what is now the Afar area in eastern Ethiopia, was a momentous as the history of Ethiopia, and world archaeology. Amharic has been the language of the court in federal government and the dominant population of Ethiopia spoken Amharic and also currently it is the official working language of the Federal Democratic Republic of Ethiopia and thus has official status

national and the working language of several of the states within the federal system, including Amhara and the multi-ethnic Southern Nations, Nationalities and Peoples the official language of Ethiopia today. As the national language, Amharic is spoken in every province, including the Amhara region. Amharic, like other languages that use the Ethiopic script (Gurage, Harari, Tigre, and Tigrinya), use characters derived mainly come from Geez. The Ethiopic script were first displayed on a computer around 1986.Who was the pioneer in this attempt is controversial but one of them was the then Ethiopian Science and Technology commission (ESTC)[50].

At the time the challenge to the computer representation of the script was developing a software package that can handle character design, keyboard layout and printer set-up. The pioneering work by ESTC started a motivated current to develop Ethiopic software by different IT companies and teams of individuals which led to the problem of lack of standardization. At the present there are at some Ethiopic software products available, each with its own character set, encoding system, font names and keyboard layout. The recent development of the introduction to the Ethiopic range with the Unicode standard could help in standardizing the different incompatible software products. Amharic currently, different Mass Medias like radio, television broadcasts and the press are also using it for disseminating information to the public. Outside Ethiopia, Amharic is the language of millions of emigrants (particularly in Egypt, US, Israel, and Sweden), and is spoken in Eritrea. It is written using a writing system by Fidel((ፊደል)), adapted from the one used by Ge'ez language[50].

## 3.2. Amharic Language Character Representation and Writing System

The characteristics of the Amharic writing system considered in this section are common to posted texts. Geez, has been a language of literature in Ethiopia up to recent time and is now used for the mass of the Ethiopian Orthodox Church. Written Geez can be copied back to at least the 4$^{th}$ century A.D and also the first versions of the Geez script included only consonants while the characters of the later versions represent consonant-vowel (CV) phoneme pairs[48].Amharic has borrowed most of its characters of Geez and thus the Amharic writing uses characters created by a consonant-vowel (CV) fusion. Seven vowels are used in Amharic each of which comes in seven different forms (orders) reflecting the seven vowel sounds (ኧ ኡ ኢ ኣ ኤ እ/e and ኦ/o). The Ethiopic writing system (EWS) in the traditional non-Unicode environment the first order has 33 base forms of origin Amharic characters of their 7 forms, which gives all totals of 231 characters[50]. It also has 7 special symbols, 44 for labialization (Example ሏ, ሯ … ቈ, ኈ, ፏ) symbols, 8 punctuation marks and 20 numerals that raise the total 310 all Amharic characters include all above. Amharic also has its own numbers and its own punctuation system where the symbol ። (hulet neteb) is used to separate words, the symbol ፣

(netela serez) is used as a comma, the symbol ፤ (dereb serez) is used as a semicolon, and the symbol ። (arat neteb) is used as a full stop. The question (?) and an exclamation marks (!) have recently was included in Amharic writing system[48]. As we see the result of its wide presentation for Amharic language currently, large Amharic documents are compiled in electronic forms. Due to this, the amount of electronic Amharic information is increasing from time to time from social media, thus, it is mandatory to perform a task of NLP and utilize the knowledge that confined to natural languages[50]. For the purpose of this research since the news items Amharic texts are considered on the Twitter and Facebook posted texts user posts, it is important to investigate the posted sentences and documents of the social media on the Twitter and finally to summarize the lager text documents into short summary.

## 3.3. Processing Amharic Text Documents

This research uses the news items posted Amharic text summarize on social media using Twitter, Facebook that can be extracted Amharic tweets to extract Amharic text and summarize for Amharic posted documents. The first step in an Amharic tweet extract from the input corpus documents and extract the Amharic tweet and also the selection of Amharic tweet texts that represent on the input documents. As there can be many of the posted sentences or tweet documents in text datasets storage and processing time costs require document processing for efficient and reliable automatic extract the Amharic tweets and summarized the tweets. Document processing is therefore an important task to get features that sufficiently represent a document without being redundant/repeat and irrelevant tweet from the Facebook user posts. In the process of this research have been investigated to relevant posted text documents and irrelevant posted text documents that could have been taken as a document of the Amharic writing system. The pre-processing requirement depends on the purpose of the news items posted Amharic text user posts problem to identify easily. This research the special nature of the Amharic language and its writing system, the lack of a standard Amharic corpus, and the unavailability of processing tools for Amharic software presented unique pre-processing challenges. This research is working full Amharic morphological analyzer and Amharic stemmer to remove the affixes easily. Therefore an algorithm for simple extract Amharic tweets repeated posting to calculate the pair of sentences level for similar posted to be written user posts that was adapted to the tool and enhanced to cluster, to summarize the clustered Amharic tweet documents for using the tf - IDF algorithm.

## 3.4. Computerizing the Amharic Script (Amharic Alphabets)

The visual Geez software has to represent the over 280 Amharic characters using the English language keyboard designed to recognize only 26 letters. This means the Amharic software have to use key combinations of the characters of the Amharic FIDEL (i.e. the software has to consider the most Amharic characters as a combination of two characters)[51]. As an example, the symbolic representations of the seven forms of the Amharic characters ሀ (ha), ለ (le), መ (me) are to write below.

| 1st order | 2nd order | 3rd order | 4th order | 5th order | 6th order | 7th order |
|---|---|---|---|---|---|---|
| ሀ (Hä) | ሁ (Hu) | ሂ (Hi) | ሃ (Ha) | ሄ (He) | ህ (H) | ሆ (Ho) |
| ለ (Lä) | ሉ (Lu ) | ሊ (Li ) | ላ (La ) | ሌ (Le ) | ል (L) | ሎ (Lo) |
| መ (Mä ) | ሙ (Mu) | ሚ (Mi ) | ማ (Ma ) | ሜ (Me) | ም (M ) | ሞ (Mo) |

***Table 3.1.The Amharic Character Representation***

## 3.5. The Amharic Grammar

As we discussed in [49] a sentence in Amharic can be a statement which is used to declare, explain an issue and the combination of phrases to create another phrase that can express a full idea about something is a sentence. When Amharic sentence is viewed from the grammatical structure point of view, it is a combination of noun phrase and verb phrase. The noun phrase comes first and then the verb phrase. Based on the number of phrases they contain sentences in Amharic are categorized under two basic categories simple sentences and complex sentence. Simple sentence only contains a single verb while complex sentence is constructed by combining more than one noun phrases and verb phrases. Since Amharic sentence formation follows its own structure, the syntax of the language also exhibited a unique structure.

The grammar Amharic is written from left to right with its own writing system structure is generally S+O+V (Subject + Object + Verb) or S+N+V (አበበ ነበዝ ነዉ), or S+V (Subject + Verb e.g. አበበ መጣ). The modifiers in such structure generally precede the word or the phrases they modify. This order is different from English the (Subject + Verb+ Object) order for the English language (He is a student). Amharic has inflectional morphological structures, which requires a complex morpheme analytical for morpheme generation and word formation as well.

## 3.6. Amharic Software

Fundamentally, computers just deal with numbers. They store letters and other characters by assigning a number for each one[50]. Amharic alphabets do not have a representation of the ASCII (American Standard Code for Information Interchange) code table. Apparently, different Amharic word processing software makes use of the ASCII code for writing, Amharic by associating the English keyboard button with Amharic symbols. Since the number of Amharic character together with punctuation marks is much greater than English, two and three keys are used to represent a single Amharic symbol[50]. Technically speaking, the software converted the default code table where each key is associated with English symbols by Amharic code table, so that users can use the same keyboard to write and edit Amharic letters. In other words, these programs, associate the keyboard buttons with Amharic symbols. In fact, this is done at the screen level. That is, the symbols stored inside files are the associated ASCII symbols of the default code table, not the Amharic symbols. The software converts these symbols for associated Amharic symbols when they are read from files into memory. As a result, users see Amharic symbols of their screen and when printing. Ever since 1987, there have been different software developed to assist users to write and edit Amharic text inside the computer. Most of the software is written to work only with Microsoft Word. However, there are few which can work in other programs, one of them is Visual Geez, which currently becomes very popular. The software is developed by Custor Computing Pvt. Co. The software has two versions, VG2 and VG2000 developed for the various versions of Microsoft Office products. Ethiopian news agency (ENA) is using the first version, which is VG2. All Amharic softwares had succeeded in helping users to enter and edit Amharic text inside the computer (especially for word processing purpose).

## 3.7. Amharic Punctuation

The analysis of Amharic texts exposes that different Amharic Punctuation marks are used for different purposes. There are many punctuation marks of which only a few of them are commonly used and have representations to Amharic software. For example the sentence separator for Amharic text writing is four dots arranged in a square sequence as (።) and are referred as "አራት ነጥብ/arat neteb". The comma equivalence punctuation in Amharic is "ነጠላ ሰረዝ/ netela serez" which is symbolized as (፣) used to separate lists. The equivalence between the semicolon, which is symbolized as (,) which is used to separate phrasal lists. Compound sentences are referred as "ድርብ ሰረዝ/ dereb serez" which is denoted by the symbol (፤). Punctuation marks like the question mark (?) and the exclamation marks (!) are borrowed from the English language and used in Amharic language for the same purpose as they are used in other foreign languages[50].

## 3.8. Amharic numbers

The number system in Amharic writing has 20 single characters which represent numbers from the ones (1/፩ up to 9/፱), tenths (ten/፲ to ninety/፺), and hundred (፻) and ten thousand (፼). These characters are derived from Greek letters are modified to look like the Amharic characters by adding a horizontal line of top and bottom of each character. The Amharic number system, however, does not have any representing symbol for zero value and it does not use any decimal points and commas. As a result arithmetic computation is very complicated using the Amharic number system. It is mainly used in calendar dates to show the dates in the Ethiopian calendar[50].

# CHAPTER FOUR

## 4. Research Methodology and System Design

### 4.1. Literature Review

This thesis to have conceptual understanding the posted text documents and identify the gap between the social media on the Twitter, Facebook posted documents in Amharic texts that is not covered by previous studies different materials, including journal articles, conference papers, newspapers that have been reviewed. In this study the review is mainly concerned works that had direct relation of the topic for news items posted Amharic texts, text summarization and the objective of the research study. These include previous work done on Amharic language text summarization system, giving more attention on the news items posted aspects, Amharic posted text summarization for the social media. Generally, this chapter description of the data preparation, tools, summarize techniques, the system architecture or design of news items posted Amharic text summarization on social media posts (Facebook and Twitter) working in this thesis and in order to explore and develop the framework for the systems involves doing the steps for the pre-process system designs.

### 4.2. Data preparation

The corpus consists of a set of the news items posted Amharic text documents from Twitter and Facebook covering the time period intervals from October1, 2015 through October, 2016 end, from a diversity of user posts within one year data collected on social media in this interval and also users should get access to the input text corpus. We used to collect training dataset on the Twitter and Facebook posted documents to Amharic text input corpus 4951 posted sentences in totals (protests (3943), droughts (667), floods (101) and sports (240)) in different news items. For those in the collected posted documents from Facebook and Twitter to obtain the sources in protests in totals 120,862 posts, in droughts posted is 43,774 posts, in sports posted is 10,299 posts and in floods is 1,209 are the training sets posts are identified by the format of date, month and years. We selected the testing set 30 posted texts for each news item in a clustered post documents. A social media would be used to get the posted text document that would be important information on summarized, and the related users post about this news item posted Amharic texts are obtained all posts as the dataset that pointed to Amharic texts in the document. We add the duplicate tweets term of the word, phrases, sentences by assuming that two tweets written by the same sentences at the same timestamp or dates is similar posted texts to extract the redundancy of sentences, and to find the unique tweets (single posted) words news items Amharic documents for the final summary. We add stop words using the default Amharic stop word list; removed punctuation and we applied TF-IDF algorithm, we obtained Amharic text for the news items user

posts of the social media that can be summarized. We made to prepare the data set of Amharic posting text documents that posted on the Twitter and Facebook on the social media. We used to extract the significant documents and sentences from the input corpus of the Amharic tweets that define each important sentence within users posted. In order to assess the Amharic posts of a system to trust with annotated corpus for Amharic texts and also each sentence from the corpus is included in the selected sentences in the summary.

### 4.2.1. Data sets

We have the following training data sets and we could be tested the data sets are #protests, #droughts, #sports and #floods data on social media posted texts that could be found in the news items posted Amharic text documents over a time in social media. The data set to consist 4 news items posted texts to collect data onto the Facebook and Twitter posted text documents and selected to extract from summary the important sentences in the input to the summary.

| **Datasets items** | **Size (in sentences)** | **Similarity posts (in sentences)** | **Clusters(in documents), K=3 groups, sentences** | | **Amharic, posted texts (dd-mm-yyyy) format** |
|---|---|---|---|---|---|
| Protests | 3,943 | 534,994 | C1(doc1) | 201,486 | 43,236 |
| | | | C2(doc2) | 166,797 | 39,805 |
| | | | C3(doc3) | 166,711 | 37,821 |
| | | | **Total** | **534994** | **120,862** |
| Droughts | 667 | 224,971 | C1(doc1) | 84,380 | 16850 |
| | | | C2(doc2) | 70,429 | 12979 |
| | | | C3(doc3) | 70,162 | 13945 |
| | | | **Total** | **224971** | **43,774** |
| Sports | 240 | 28,824 | C1(doc1) | 10,804 | 3470 |
| | | | C2(doc2) | 9010 | 3631 |
| | | | C3(doc3) | 9010 | 3198 |
| | | | **Total** | **28,824** | **10,299** |
| Floods | 101 | 4960 | C1(doc1) | 1833 | 459 |
| | | | C2(doc2) | 1554 | 419 |
| | | | C3(doc3) | 1573 | 331 |
| **Totals** | **4,951** | **793,748** | **Total** | **4,960** | **1,209** |

***Table 4.2. Preparation of data sets pre-processes***

In the above table 4.2, we made for training and testing the posted Amharic texts for social media on Facebook and Twitter posted in Amharic texts to take and analysis the samples to process the similarity and cluster for the similar posted sentences automatically. Based on these data to find the similarity with each posted Amharic texts with a pair of sentences on social media and identify the group of the similar clusters, the numbers of input k are three (created on distance values or nearest distance value to group the similar items of sentences to cluster and after cluster we had been found three clustered documents for each posted text documents for sentences to summarize individually clustered.

### 4.2.2. Manual Summary Preparation Posted Texts

For the purpose of the evaluations an ideal or human summary is prepared for manually by linguistic experts for the Amharic texts that is based on a guideline to prepare for ranking the sentences of a given Amharic post text on social media for Facebook and Twitter post texts. We have three linguistic experts to need a manual summary of the posted documents. The three experts to evaluate the summary posted texts are involved in this process, three experts from Jimma University two of them masters of art (M.A) lecturers and one bachelor of art (B. A) in Amharic department. The guidelines on the Summarizer preparations are combined with the (Melese, 2009), (Eyob, 2011) and (Addis, 2013). The Amharic summary is to make on the different compression rates of 10%, 20% and 30% to use for each groups cluster posted texts. Due to those used the evaluation processes to measure the performance of the customized system summary by comparing the automatic summary generated with the corresponding human summary (manual). The same as the ideal summary is used in all experiments to evaluate the summaries generated by the system in the given posted text documents.

## 4.3. Approach for the research study

### 4.3.1. Extractive summarization

Extractive methods work by selecting a subset of existing words, phrases, sentences in the original text to form the summary. Extraction is a summary consisting of a number of sentences selected from the input documents for Amharic texts to produce for the reader. We used to the summarization techniques for our research by extractive summarization for the input documents to summarize the important posted sentences to produce for the users of the social media summarize document by using the hybrid tf-IDF algorithm.

## 4.4. Development tools for summarizing techniques

The research system to investigate and develop a program would be developed the code using Java (NetBeans 8.0.1) programming language to implement the news items posted Amharic text summarization on social media for Twitter and Facebook posted Amharic text user posting documents, which is run on the prepared input corpus.

### 4.5. System design for the research study

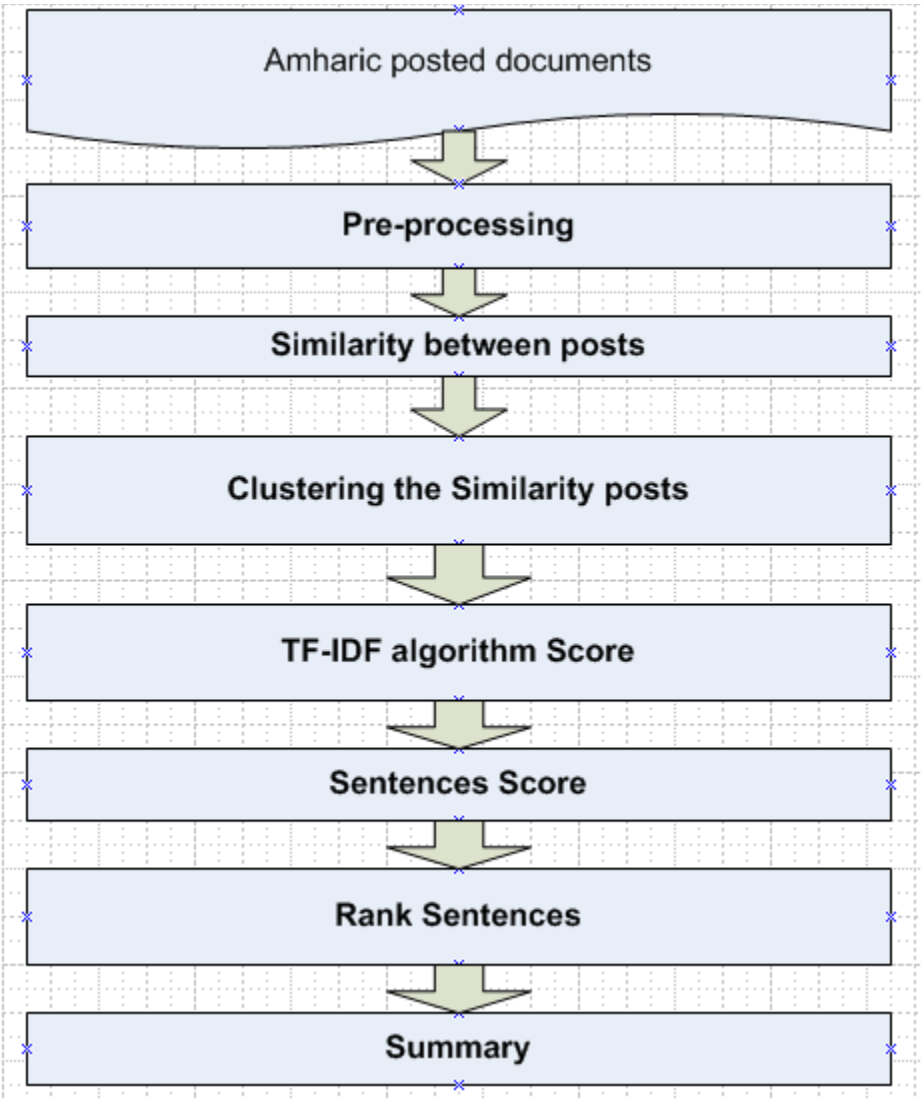


***Figure 2. The Architecture of the system design***

As the above figure 2, to describe the posted current news items posted Amharic texts on social media that could be used to organize and preprocess Amharic texts in the sentences using the similarity between posts to calculate for each sentence Amharic tweet text only to find the similarity posts and clustered for each similarity

value of sentences posted documents. After clustered posted Amharic tweets to summarizing each clustered document individually using a TF - IDF algorithm to enhance the output documents for the users to select the highest sentences into the summary. For extracting text summarization for news items posted Amharic text is selecting the most significant sentences in text documents to be summarized. This method consists of seven steps to process the systems above figure, they are:

1. Pre-processing posted documents, 2. Similarity measures to calculate for sentences in the documents
3. Cluster the similarity posted similar items, 4. Tf-IDF calculates for each term of the document, 5. Sentence scoring for each sentence, 6. Sentence ranking and finally summarization.
The following is a list of the functional components of the research implement.

Step 1. **Pre-processing of input post document:** This will work on the Amharic posted documents for processing by the rest of the system. The preprocessing involves

**Sentence segmentation in the document:** It is the process of breaking down or segment the given posted document into sentences. For this the system the posted document is segmented by identifying sentence which ends with a full stop symbol or Arat neteb (።),question mark (?), exclamatory mark (!), and sometimes complex sentences separated by derb serez (፤) ,the total number of sentences present in the posted document are identified. Sentence separator is through the document and separates the sentences based on some rules (like a sentence ending is determined by Arat neteb (።), question mark (?), exclamatory mark (!) and derb serez (፤), a space etc.). Any other appropriate criteria might also be added to separate the sentences in the given posted documents. Word separator separates the words based on some criteria like a space denote the end of a word. **Tokenization of the segmented sentences:** the tokenization is very important to this research study. It is the process of breaking for the posted documents for the sentences into words or terms. The corpus, which is a set of sentences first tokenized into words. Tokenization is done by identifying with the white spaces, comma (,) and special symbols between the words. In this process frequency of each word is calculated and stored for further processing of the input documents.

**Stop word removal of the list of words:** Stop words are the words that is carried as the important meaning of the keywords. These stop words are identified by supplying a list of words with less importance of the system. The system compares these stop words with the tokenized words found from the previous phase. For the stop words eliminator: this eliminates the regular Amharic words like ‘ስለ, ወይም, ከላይ, ላይ, በኋላ, ከዚያ, ብቻ etc. for further processing. A list of valid stop-words in Amharic is available for a softcopy and these words are known as stop words. For this research stop word could not be scored in a sentence if the algorithm to

check the stop word if it is stopping word to contain the key to the HashMap, to continue that could not be added in once sentence score; because of stop word is not important to the sentences scores. But it could be balancing the meaning of the sentences. So, we couldn't remove it. But we could be the stop word values to scale down its importance's for the sentences score of the posted documents.

The word-frequency calculator: this calculates the number of times a word appears in the document (stop-words have not been eliminated earlier itself and will not figure in this calculation) and also the number of sentences that word appears in the document.

**Stemming words**: This step is the process of conflating tokens to their roots form of the word (eg. ኢትዮጵያውያን → ኣትዩጵዩ). All Amharic stem words are changed into 6th order (ሳድስ or Sades Fidel) character sets. This the word of the posted document can be found in the different forms of the same document. These words have to be converted to its root form of the simplicity on posting documents for social media the process is known as stemming. We use the porter stemming algorithm, porter stemming is a process of removing the common morphological and inflectional words starting and endings of words of the affixes. This algorithm is used to transform the words of its stem forms. This system for the processing, the stemmer method is used to return a word of its root form using a predefined prefixes and suffixes list. Finally, the frequency of each word of the stem is calculated a retained for the next phase. The process of removing prefixes and suffixes may be more or less efficient, depending on the quality of the dictionary and to identify the score of words occurrence in the posted document. The Amharic words the affix removals are done for removing prefixes first the starting word with the left side and then the suffixes word of the right side of the word using the set of rules to provide for the lists of the words of the stemmer under the "pre- and -post" lists of rules of the dictionary files for the Amharic words. The base of this the stem word used for calculating the word frequency of the terms of the document to select the key word in most representing term to rank among the sentences of the posted text in social media and to extract the summary.

We could import the Unicode characters of Amharic filed ሀ - ፐ into Java tools to import base of this we could be removed the affixes in the posted Amharic words to find the stem of words (see the details in the appendix I and VIII). If we would be worked the rules of the stemming for Amharic words to remove the prefixes, words and suffixes Amharic words of posted texts as the following algorithm.

**Prefix remove algorithm**

1. Read a word from the list that holds the representative unique words of the terms after preprocessing, normalization and stop words removal processes are completed.
2. If the size of the word is less than three, then return the word as a word without prefix and pass to the next word in the list.
3. If the word has W number of characters (where W is greater than two), strip the left most W-2 characters from the word.
4. Check if the striped group of characters exists in the prefix list. If it matches with a prefix, then do step 7 and return the word without prefix and pass to the next word in the list.
5. Else, if the W-1 group of characters does not match with the prefixes in the prefix list, then strip the left most W-2 characters from the word and repeat step 4 above.
6. Else, if the W-2 characters are not found in the prefix list, step 4 and 5 are repeated iteratively by decreasing the number of characters in the left striped from the word by one in each loop until one character (W-(W-1)) is striped and checked in the prefix list.
7. If the prefix removed has only one character, then return the word after the prefix removal.
8. Else, if all the characters are checked and none of the striped group of characters have match in the prefix list, then return the word with no prefix.

### *Algorithm 2: Stemming rules for prefix removal algorithm*

In the same manner to develop the rules for suffixes in Amharic words in post fixes (suffixes) words as follows

**Suffixes remove algorithm**

1. Read a word from the list that holds the list of words after all prefixes are removed.
2. If the size of the word is less than three, then return the word as a word without suffix and pass to the next word in the list.
3. If the word has E number of characters (where E is greater than two),
   strip the right most E-1 characters from the word.
4. Check if the striped group of characters exists in the suffix list. If it matches with a suffix,
   Then do step 7 and return the word without suffix and pass to the next word in the list.
5. Else, if the E-1 group of characters does not match with the suffixes in the suffix list, then strip the right most E-2 characters from the word and repeat step 4 above.
6. Else, if the E-2 characters are not found in the suffix list, step 4 and 5 are repeated iteratively by decreasing the number of characters in the right striped from the word by one in each loop until one character (E-(E-1)) is striped and checked in the suffix list.
7. If the suffix removed has one character, then return the word after the suffix removal.
8. Else, if all the characters are checked and none of the striped characters have match in the suffix list, then return the word with no suffix.

### *Algorithm 3: Stemming rules for suffix removal algorithm*

As can see the above rules we could be made the most common prefixes in Amharic words are some examples like "እነ-","አል-","ከነ-", "ስለ-","የ-","ከ-","ለ-" which is used in both nouns and verbs in post texts in the documents to remove the prefixes in the beginning words. In Amharic prefixes can be formed by a combination of other prefixes of words more than one prefixes. For example the prefix "እንደየ-", "እንደማ-","እስኪ-" is a combination of the individual prefix "እስክት-","የማያስ-"and "እንደም-" etc. based on this we could be developed the rules one, two, three, four and five prefixes and suffixes words to remove or eliminate the affixes. And the other words, variant forming affixes is suffixed and the most common suffixes in Amharic nouns are the possession marker "-ን", the emphasis marker "-ም", the object marker "-ና" and the plural marker "-ዎችናም", "-ውያንን","-ዎችንም","-አዊ", "-ውያን" etc. we could be developing the rule to read as Java codes for Nyala font the words like ኢትዮጵያውያን (እትይጵይ. See the details of the stemmer affixes code the appendix VIII.

Step 2. **Similarity measure**: Sentence similarity is computed as a linear combination of sentence similarity and word similarity. The cosine similarity measure is one of the common techniques used to measure similarity between a pair of sentences to be calculated the similarity between each sentence and after similarity to group the similar clusters to form. Sentence similarity is calculated as the similarity between the sentences vectors for the two sentences intersection, and to build the sentences vectors, the union of words in the two sentences is treated as the characters of the file length. We measure the similarity by using the intersection of the two pair of sentences with word similarity with summation for each word and the union of the both sets of sentences.

Step 3. **Clustering**: For clustering of multi documents, these items refer to the posted documents for sentences and the cluster that a similar sentence belongs to represented by classes of the group in a shorter or nearest distance value calculated by the centroids using the formula of Euclulidean distance function. Once we used with post text data, k-means clustering can provide a great way to organize the thousands-to-millions of words, phrases or sentences for posted documents. In the social media posted documents that could be used by the user to clusters repeat the task and inform steps until the cluster does not change, or equivalently, until the center of the mean does not change. After that the calculated the similarity values of two or more than posted document for the sentences to find the similarity sentence to check for each sentence. Since, the sentences for randomly for each document set for the input similarity results, posted documents in the sentences and to give clusters based on similarity values of news items posted Amharic texts on social media and each set clusters the similar posted documents for the user posts on the social media[36].

The similarity document clustering is to discover the natural grouping of a set of patterns, documents for posting the sentences in the input similarity results calculation. User posts that are in the same cluster for similar among themselves and dissimilar to the posts belonging to other clusters. The purpose of document clustering is to meet human interests to read posted Amharic text information to group similar items and also to read the posted Amharic text documents for the reader and understanding for the user that happen to get the summary posts. When we summarize the user posts by clustered document for each post of clusters by similar distance values Amharic text documents for the social media on the Twitter and Facebook posted data. Day to day user posts to arrange with the same cluster like political #Protests with #Protests similar contents, #droughts with #droughts, #floods with #floods, #sports with # sports, #health with #health the same posted documents or sentences and others user posts to arrange or order the tweets for the algorithm and summarize for each user posts. In our research the data onto social media could not be identified automatically protests

at droughts, protests at sports posted or other data sets. So, we could have a different corpus for data sets to train and test the performance for Amharic text documents posted on social media. Each collected protests corpus is not the same subject matter, depend on the similarity algorithm to find the similar subject matter to get the contents and search for the documents in sentence level to check the sentences by sentences sequence method to found the same sentence or some word similarity to find its return to some value, if it is not getting the similar sentences that display to return zero value. The characteristics of k-means algorithms have been designed by J. Macqueen[52] was probably the most popular and simplest clustering technique. It can be characterized briefly as follows:

- Requires a number of clusters (k) as part of the input
- Sensitive to initial conditions and outliers: because of random initialization
- Euclidean distance being its most natural similarity measure, allowing spherical zones only
- Partitions are mutually disjoint (exclusive – no overlap allowed)
- Guaranteed to terminate.

Step 4. **TF-IDF algorithm Score:** The TF-IDF value is composed of two primary parts. The term frequency (TF) and inverse document frequency (IDF). TF component assigns more weight to words that occur frequently to a document. Because important words are often repeated in one document and a single document that encompasses all the documents together[5].

In this case, the TF component's description is straightforward since we can compute the frequencies of the terms of all the posts as one document. The weight of a sentence is the summation of the individual term weights within the sentence. In the information retrieval, the most standard algorithms make use of term frequencies and inverse document frequencies.

The goal of such an algorithm TF-IDF is to rank documents according to their relevance to a query about the document on statically ways of weights of the terms of the documents. TF measures how relevant is a word with a specified in the document. Tf to calculate the frequency of term occurrences of each of the stemmed words of the posted document. IDF measures how relevant is a word according the full corpus of documents. For example, a word that appears in most of the documents should not have a big impact on the relevance and a word that appears in very few documents make them very relevant when it appears in the query. Why implementing it? We find that implementing it is interesting for many reasons. First of all, the algorithms that we are going to implement the next experiment are very likely to use TF and IDF too. Furthermore, the information retrieval algorithm might be proven useful when testing and checking the results of other algorithms.

Therefore, TF-IDF gives the most weight towards that occurs most frequently within a small number of documents and the least weight to terms that occur infrequently within the majority of the documents.

$$\mathrm{Tf}(t,d) = \frac{\text{term counts for the stem}}{\text{total terms in the document}}$$

$$\mathrm{IDF}\ (t,\mathrm{Docs}) = \log 10\ (\ \frac{\mathrm{Docs}}{\text{t appeared in all Docs}}\ )$$

$$\mathrm{Tfidf}\ (t,d,\mathrm{Docs}) = \mathrm{tf}\ (t,d) * \mathrm{idf}\ (t,\mathrm{Docs})$$

Where, t is term or word, d is document, Docs = all documents in the corpus.

This above equation defines the weight of a term of the content of a document. But we have a set of posts that are related to a topic.

Step 5. **Sentence scoring**: this step after the tfidf stem word score, each term of obtaining the score its values if this sentence determines the score of each sentence and several possibilities exist. The score can also be made to the number of sentences in which the words in the sentence appear in the document. Scoring each sentence is to rank each sentence we need to score each sentence using the tf-idf values calculated before. Rather than simply taking the sum of all the values of a given word with one sentence sequentially. These include only summing tf-IDF value score of the term where the word is a noun, verb and others, but stop words could not be added for the sentences that could be scaled down the stop words in a sentence.

Step 6. **Sentence Ranking**: after the sentences scores for the list of stem words, the sentences will be ranked according to the sentence scores values and any other measures like the position of a sentence in the document can be used to control the ranking. After applying the sentences scored we can finally sort our sentences in descending (top value to low value) orderly the sentences score values to sort. For example, even though the scores are high, we would be putting first rank sentences and compare to each score value of the sentences higher score values to order the sentences. After each sentence is scored they are arranged in descending order of their score value, i.e. the sentence whose score value is the highest is in top position and the sentence whose score value is the lowest is in the bottom position of the sentences in the Amharic text document.

Step 7. **Summarizing Extraction**: After ranked the sentences list, if the user to select the input on the size of the summary, the sentences will be picked from the ranked lists orderly. The news items text summarization is the process of automatically creating a compressed version of a given text that provides useful information about the user in the social media. The length of a summary depends on the user's needs. In this thesis, we focus on single or multi-document extracting text summarization, where the goal is to produce a summary of

single documents. Extractive summarization produces summaries by choosing a subset of the sentences in the original document to find the significant sentences in the summary. This is to increase the structure and coherence of the summary. Now a summary of news items posted Amharic text documents are the most important part of our application. For summarization of documents we are implementing by using TF-IDF algorithm score. The algorithm is to isolate the frequently words' weight of news items Amharic relevant post texts, documents from other irrelevant posted documents that can occur immediately to extract the relevant Amharic tweets to summary without repeating the news items document user posts of the summary posted texts. For each of these unique Amharic posted documents, the algorithm adds them to the summary tweets for texts. As to duplicate users post text, when a summary of Amharic tweets from Twitter and Facebook existing posts are changed to summary posts. Each user will need to process those streams in time and spending of time to read the posts on the social media, clean out irrelevant posts as the Twitter and Facebook post texts. Then select Amharic tweets from those documents to return to the user each post as time for monthly grouped, users to select for the summarize sentences in rank order list and summarize the sentences for Amharic texts in a once-a-month match posted text documents for ranking sentences. So, users select the numbers of the top ranked sentences in text documents and the system to ask enter the summary date time for the month and year to require the summary of one month and year after that to produce summary posted documents to the users by ranking order within the select month and year only. Based on the literature review discussed in the previous section (2.5) the process of offered single and multi-document summarization using news items for temporal summarization for Amharic posted the text to summarize. The length of the sentence looks like the importance of the sentence into the summarization. Generally, sentences that are very long and very short are not suitable for summary. Sentence that is very long will have unnecessary information which is not useful for summarization of documents. Whereas, sentences that are too short, do not give much of information about the document. The proposed system takes as inputting a set of Amharic texts from the stream and outputs a summary of the posted text.

### 4.5.1. The implementation of the flow of the algorithm

The implementation of the algorithm as the following diagram to summarize the news items Amharic posted documents based on the following steps of coding the full system from input document until final summarizing each clustered document individually for the target summary document.

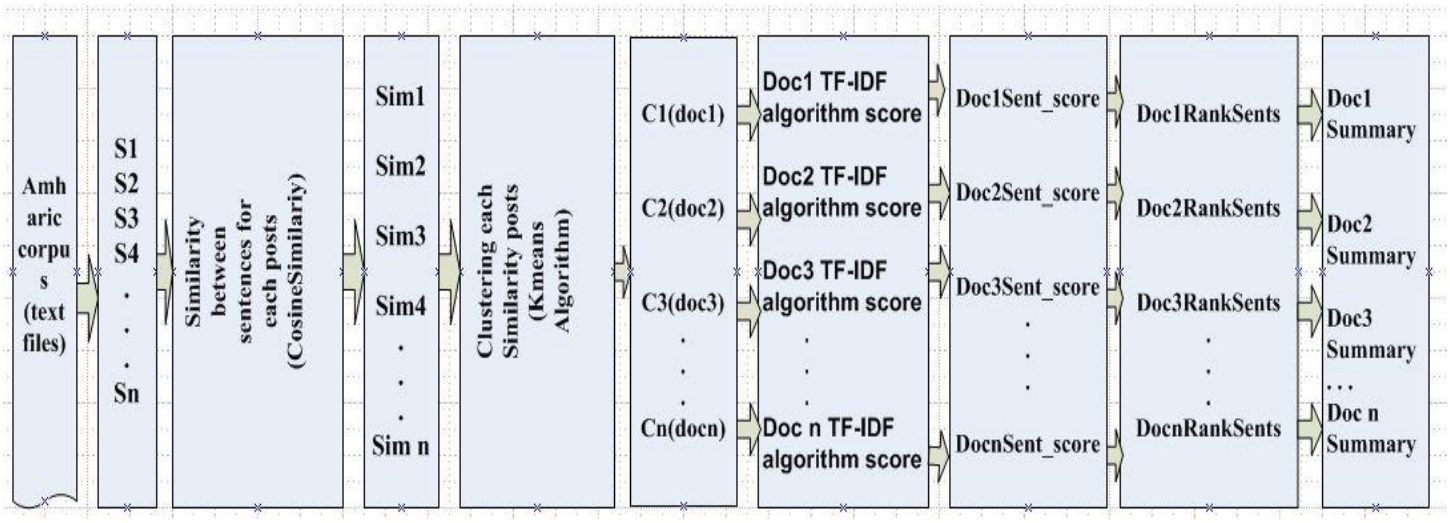


***Figure 3. The flow of the algorithm that could be summarized the news items posted Amharic texts steps***

As indicated the above figure 3 to show that if the general diagram of the steps for the implementation in Java code from the input Amharic text file documents until to each clustered document summarize. If each sentence is to find the similarity between others sentences to compare and checked the word similarity in a pair of two sentences is similar or dissimilar to check by sequence method, if the sentence one (S1) to iterate until the end of the of the sentences in the document to find to similarity (Sim1), again the second sentence (S2) to sequentially to find the similarity (Sim2) for the third, fourth until end of the sentences to while loop its etc. After similarity results, we should be grouped the similar items of the sentences in one cluster (C1), is not similar to another cluster documents (C2), based on clustered documents to analysis by using TF-IDF algorithm to normalize the frequently terms in the text document. Subsequently, to decide whether the term is important to the sentences scores, sentences rank of the Amharic texts and finally to summarize each clustered document individually. Based on this implementation step we could be generated automatically to summarize the text document, if user to decide the size of the summary of the sentences to take the news items, texts summarization for Amharic text documents on social media and also enter as the user to need the monthly grouped and enter the years to get the summarize Amharic text documents in selected the user month and year only to find the summarize for selected sentences.

### 4.6. Evaluations techniques

Once of the customizations of the news items posted Amharic text Summarizer on social media posted text to evaluate the finalize results, and also the system summary is generated for the Amharic post text documents in social media, evaluation of the outputs of the automaton for Amharic text summaries are completed in two

ways of evaluating. The first one is subjective evaluations which are involved in checking whether the automatic summary of the system make to the informativeness, non-redundancy, the grammatical correctness of the terms, linguistic quality, coherence and structure, referential clarity and all over the observation for the summary. For this evaluation the guideline is prepared for the Summarizer experts to follow this for evaluating the given post texts of summaries of each clustered post. The system evaluations are prepared by using three experts to use for the manual summary of the posted documents.

The second objective evaluation is made by the comparing the automatically summaries generated with the manually prepared human summaries post texts and calculating for the average effectiveness measured of evaluation, F-measures for each posted clustered documents. The evaluation tool for designing using Java tool used to evaluate the Amharic post text summaries objectively F-measure a performance is measured. The score of the evaluation is given in percentages of hundred for each summary posted texts. In this research evaluation is to measure the compression ratio, precision, recall and f-measure for the news items text summarization from user selected the higher rank sentences automatically, we have taken three users to evaluate each document to evaluate the summary manually, how much the user selected relevance documents in sentences automatically by the system for posting text documents. One of the simplest ways to evaluate a summary is by considering it independently from the source of news items posted Amharic text documents for compression ratio automatically generate summary[53]. The evaluation of news items Amharic tweets/posts can be done on both systems and human summaries, but the guidelines have to be adjusted depending on the user of summaries to be assessed. When the information contained in the two posted sentences are compared, humans can be required to identify important sentences from the top ranked score sentence and see how they are reflected in the summary. One of the main disadvantages of manual evaluation (user) is that human judges have to be involved in the process, which leads to high cost and also makes the evaluation disposed to human errors. In addition, user evaluation has several other shortcomings. Firstly, it takes time to apply this evaluation method, so a researcher developing a summarization system cannot assess immediately whether a change in the system leads to an improvement or descent in the system's performance. Secondly, the related of the judges have to be taken into consideration, because it could influence their assessment. For example, if judges are required to evaluate summaries produced from scientific documents, but they do not have the necessary background knowledge to understand them properly, the results of the evaluation will be questionable. Another criticism is that judges are not evaluating summaries of a realistic situation (i.e. in a real world task), instead relying on their perception to decide the summary's usefulness[54].

CHAPTER FIVE

# 5. Implementation, Experimentation and Evaluation Results from the Research Study

## 5.1. Introduction

In this chapter of used to understand the concept behind, customize the system, test corpus, experiments that had been conducted implementation code of the system, to summarize the news items posted Amharic text user posted documents on social media and evaluate the output of the systems is discussed. To achieve the objectives of the research study the following major tasks had been used in each main responsibility involved in doing the research study. Like similarity measure, clustering, tf-idf algorithm terms score and finally summarization for each clustered documents. The selected posted texts in social media are 30 posted texts as testing sets are saved separately from a ".txt" files for each clustered post as Amharic texts after clustered post documents being cleaned from any noise, other non-text content like video, images, audio and others unnecessary information to the research study. In order to group the clustered post documents into four groups based on the similarity post Amharic texts to group the similarity posted news items of cluster of each item of posts. In these categories are protests post texts containing their posts and sentences, droughts post texts containing their posts and sentences, sports post containing their posts and sentences and floods post texts containing its posts and sentences in social media on Facebook and Twitter post texts. From each of the groups clustered posted Amharic texts consist of its different posts texts which are saved for a text files name for each clustered post as protests post, droughts post, sports post and floods post in order to sequentially save the post texts like "prot1.txt, prot2.txt" and so on. Finally to evaluate the results obtained, evaluations to test the performance of our approach, and discussion of the findings.

## 5.2. Implementation and Experimental Results for posting texts on social media

### 5.2.1. Similarity measures between Sentences

Sentences are made up of words, so it is reasonable to represent a sentence using the words with the sentence. The most relevant research area, to our task is the Amharic text summarization news items posted on social media. The cosine similarity measure is used to measure similarity between a pair of sentences and also after similarity to group the similar clusters to form. Here sentences are represented as a weighted vector and established information retrieval methods use a set of a predetermined index term (words or collocations) that are used to represent a document in the form of document term vector. Vector similarity is then used to identify documents that are most related to the query about the terms. Because the index terms are pre-determined and

in large numbers. Recent research achievements in similarity analysis are also improved to accomplish an efficient similarity vector for a sentence to calculate the similarity in words. Given two pairs of sentences:

S1= {w11, w12 ... w1m1} S2= {w21, w22 ... w2m2}

Where wij is the jth word of Si (i=1, 2), mi is the number of words in Si.

A joint-word set S = S1 U S2 is then formed into distinct words with S1 and S2.

S =S1 U S2 = {w1, w2... Wm} that has m distinct words. The joint word sets S contains all distinct words of S1 and S2. And also we could be intersected the similarity in words with the two sentences word similarity.

$$S = \begin{pmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,N} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ s_{\rho,1} & s_{\rho,2} & \cdots & s_{\rho,N} \end{pmatrix}; \; s_{\mu,j} = \begin{cases} \omega_{\mu,j} & \text{when the word } j \in \vec{s}_{\mu} \\ 0 & \text{otherwise} \end{cases}$$

Where each $\mu$ row contains the $\omega_{\mu,j}$ weighting of the word j in $\vec{s}_{\mu}$ sentence

Example.

S1= 9-Feb-2016 ኢትዮጵያውያን ሰላማዊ ሰልፍ በዋሽንግተን ዲሲ #OromoProtests #Ethiopia Protests #Oromo #Ethiopia "በኦሮሚያ፥ በጎንደርና ቢጋምቤላ በዜጎች ላይ እየደረሰ ያለው በደል ይቁም!"

S2= 2-Feb-2016 ኢትዮጵያውያን ዛሬ በዋሽንግተን ዲሲ ሰላማዊ ሰልፍ አደረጉ "በኦሮሚያ፥ በጎንደርና ቢጋምቤላ በዜጎች ላይ እየደረሰ ያለው በደል ይቁም" "አምባገነኖች የልማት ሃጋር አይደሉም" የዛሬው የዋሽንግተን ሰልፈኞች። etc.

From the above examples, if the two sentences compare are all words similar occur both in the sentence, its value for the results is 1. If it doesn't, the similarity in all the words is computed against all the other words, of the sentence and if it happens to be above a threshold value ∅ (empty set), then the value of the two sentences is ∅, else it values of the results is 0. This value is further reduced by the information, data onto the words as found in the corpus. This similarity is computed by dividing the sum of similarity values of all matching candidates of both sentences X and Y by the total number of set tokens. An important point is that it is based on each of the individual similarity values, so that the overall similarity always reflects the influence of them. In equations for the vectors for the two sentences: Two sentences similarity is calculated as formula:

S1= sentence1 S2 = sentence2 $\cap$ = intersection $\cup$ = union

Sim = similarity between two sentences

$$\mathrm{Sim}(S1,S2) = \frac{2 * match\ (s1\ \cap\ s2)}{\mathrm{len}(S1\ \cup s2\ )}$$

The union of the two sentences is the set of elements which are in either set. The intersection of the two sentences is the set of elements which are in both sets. The word similarity attempts to correct for a fact that sentences with the same words can have calculated the similarity values and also cross check between the two sentences that could be found the similarity sentences, words, phrase to be calculated the posted Amharic news items texts. This is done by computing the word order vectors for each sentence and computing a normalized the similarity measure between them. Similarity calculated to word similarity between two sentences in posted sentences to use the intersection, this paper also lists some sentences computed the similarity with their algorithm.

In the work of Simpson[42], his presented the method of the syntactic similarity with the two strings of capturing the similarity between words that was concerned to work. The work of the author to compute the similarity between two sentences and he basically to capture for the semantic similarity between the two word senses for the strings of the path length similarity. In the WordNet, according to the author each part of the speech of the words (nouns/verbs...) was organized to the taxonomies and each node was the set of synonyms that could be represented in one sense. The WordNet defined for the relations between for the synsets and relations between word senses that is a semantic relation between word senses is a lexical relation. The work of the author was defined the semantic relations were the hypernym, hyponym, homonym etc. and lexical relations were antonym relation, the derived form of the relation. From this the measured of the semantic similarity between two synsets for using the undirected graph and measure for the distance between them through WordNet to treat by the taxonomy. The measuring similarity (MS2) the formula was proposed by Wu & Palmer, for the measure both path length and depth of the least common sub-summer:

$$\mathbf{Sim(s,\ t) = 2\ *\ depth\ (LCS)/\ [depth(s) + depth\ (t)\ ]}$$

Where S and t: denote the source and target words being compared. Depth (s): is the shortest distance from root node to a node S on the taxonomy. Where the synset of S lies. LCS: denotes the least common sub-submerge of s and t.

The author the major steps could be followed: tokenization and the word stemming for his formulate for capturing the semantic similarity between sentences were to identify the problem computing the maximum the total matching weight of the bipartite graph to design for semantic similarity, where X and Y were the two sets of disjoint nodes for the similarity. The match of the results from the previous step was mixed with the

single similarity values for the two sentences. The author presented the two formulas for calculating semantic similarity between the two word senses. And the formula he applied the proper strategy to compute the overall score:

$$\frac{2 \times Match(X, Y)}{|X| + |Y|}$$

The matching average for this, where match (X, Y) is the matching word tokens between X and Y. The above formula, the similarity is computed through dividing the sum of similarity values of all matching candidates of both sentences X and Y by the total number of set tokens for each sentence. Since the important point is that each of the individual similarity values for the sentences, so the overall the similarity always reflects the impact of them. We apply this strategy with the measure the similarity formula to consider

$$\frac{2 \times |X \cap Y|}{|X| + |Y|}$$

For example the above formula: the given two sentences are X and Y, the X and Y have lengths of 3 and 2, respectively. The bipartite matcher returns that are the X [1] has matched within the Y [1] with a score of 0.8, the X [2] has matched within the Y [2] with a score of 0.7. Based on this using the matching average of the overall score is calculated: 2* (0.8 + 0.7) / (3 + 2) = 0.6. The using Dice with a threshold is 0.5, since both the matching pairs have scores greater than the threshold, so we have a total of 2 matching pairs. The overall score is: 2* (1 + 1) / (3+2) = 0.8. Sentences delimiter shown below the table.

| Punctuations | Description |
|---|---|
| ? | Question mark |
| ። | አራት ነጥብ/ Arat native (four point or full stop) |
| ፤ | Double point or derb serez / ድርብ ሰረዝ |
| ! | Exclamation mark |

**Table 5.3. Sentences delimiter**

From above table 5.3, if the given documents to separate each sentence in a document the algorithm to use a delimiter to separate (break the sentences for each) each sentence by sentences in Amharic sentences.

```
Read input corpus
Read list of sentences within the pairs of two sentences cross check the words similarity for the intersection and
  Union of the set
String =each sentence to compare and get the similar values in the input corpus
Similarity = each sentence in the list of similar posted documents for sentences to group similar
For String
   For Similarity
     If string == sent
          Normalize string to calculate similarity posted documents for a sentence for each list of sentences
       Else continue
     End if
   End for
End for
```

**Algorithm 1. Sentence Similarity measure normalizes algorithm**

Example of some sample of sentences to take the protests training sets for posting texts on Facebook and Twitter collected to identify the similarity of the pairs of the sentences between the Amharic post texts given below table 5.4

S1→13-Oct-2016 በአማራና ኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ።

S2→በኦሮሚያና በአማራ ክልሎች ሲካሄዱ ከነበሩ ህዝባዊ ተቃውሞዎች ጋር በተገናኘ ከ500 በላይ ሰዎች መሞታቸውን መንግስት አመነ።

S3→ሂውማን ራይትስዎችና ሌሎች አለም አቀፍ የሰብዓዊ መብት ተቋማት ለወራት በአማራና ኦሮሚያ በዘለቀው በዚሁ ተቃውሞ ከ700 የሚበልጡ ሰዎች በመንግስት የጸጥታ ሃይሎች መገደላቸውን ሲገልፅ ቆይተዋል።

S4→ጠቅላይ ሚኒስቴር አቶ ሃይለማሪያም ደሳለኝ በኦሮሚያ ክልል ብቻ 170 ሰዎች፣በአማራ ክልል ደግሞ 120 አካባቢ ሰዎች መሞታቸውን በመግለፅ በአጠቃላይ ከተቃውሞ ጋር በተገናኘ ከ500 በላይ ሰዎች መሞታቸውን ለጋዜጠኞች አረጋግጠዋል።

S5→ድርጊቱ ሰዎች በአለም አቀፍ ገለልተኛ አካል ምርመራ እንዲካሄድበት ጥያቄን እያቀረቡ ያሉ አካላት ትክክለኛ የሟቾች ቁጥር ሊታወቅ የሚችለው ምርመራው ሲካሄድ ብቻ መሆኑን በመግለጽ ላይ ናቸው።

| Similarity calculation | Sentence1 | Sentence2 |
|---|---|---|
| 0.41386 | 13-Oct-2016 በአማራና ኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ | በኦሮሚያና በአማራ ክልሎች ሲካሄዱ ከነበሩ ህዝባዊ ተቃውሞዎች ጋር በተገናኘ ከ500 በላይ ሰዎች መሞታቸውን መንግስት አመነ |
| 0.24446 | 13-Oct-2016 በአማራና ኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ | ሂውማን ራይትስዎችና ሌሎች አለም አቀፍ የሰብዓዊ መብት ተቋማት ለወራት በአማራና ኦሮሚያ በዘለቀው በዚሁ ተቃውሞ ከ700 የሚበልጡ ሰዎች በመንግስት የጸጥታ ሃይሎች መገደላቸውን ሲገልፅ ቆይተዋል |
| 0.181882 | 13-Oct-2016 አማራና በኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ | ጠቅላይ ሚኒስቴር አቶ ሃይለማሪያም ደሳለኝ በኦሮሚያ ክልል ብቻ 170 ሰዎች፣በአማራ ክልል ደግሞ 120 አካባቢ ሰዎች መሞታቸውን በመግለፅ በአጠቃላይ ከተቃውሞ ጋር በተገናኘ ከ500 በላይ ሰዎች መሞታቸውን ለጋዜጠኞች አረጋግጠዋል |
| 0.139523 | 13-Oct-2016 በአማራና ኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ | ድርጊቱ ሰዎች በአለም አቀፍ ገለልተኛ አካል ምርመራ እንዲካሄድበት ጥያቄን እያቀረቡ ያሉ አካላት ትክክለኛ የሚኾች ቁጥር ሊታወቅ የሚችለው ምርመራው ሲካሄድ ብቻ መሆኑን በመግለጽ ላይ ናቸው |
| 0.12113 | በኦሮሚያና በአማራ ክልሎች ሲካሄዱ ከነበሩ ህዝባዊ ተቃውሞዎች ጋር በተገናኘ ከ500 በላይ ሰዎች መሞታቸውን መንግስት አመነ። | ሂውማን ራይትስዎችና ሌሎች አለም አቀፍ የሰብዓዊ መብት ተቋማት ለወራት በአማራና ኦሮሚያ በዘለቀው በዚሁ ተቃውሞ ከ700 የሚበልጡ ሰዎች በመንግስት የጸጥታ ሃይሎች መገደላቸውን ሲገልፅ ቆይተዋል። |

**Table 5.4.Sentences similarity calculation results**

In the above table 5.4 shown that if the two sentences are found to the similarity how much is similar results from the similarity between the two sentences to display the outcomes to run some example of our corpus protests.

## 5.2.2. Clustering Based Algorithms using Kmeans

Clustering is to group similar posted sentences into their classes. It is a process of creating groups of similar items or posted documents. Clustering to find clusters of data posts that are similar in some nearest distance values of one another. The members of a cluster are more like each other than they are like members of other clusters. The main aims at clustering algorithms are similar to one another within the same cluster and dissimilar to the objects in other clusters. The selected similarity values of two posted documents for sentences

in Amharic texts on the Twitter and Face book data randomly to check for each sentence to find the similarity sentences, after calculated the similarity sentences for randomly for each document set for the input similarity results, posted documents in the sentences and to give clusters based on similarity values of news items posted Amharic texts and each set clusters the similar posted documents for the user posts on the social media. For clustering of multi documents, these posted objects refer to sentences and the cluster that a similar sentence belongs to be represented by classes of a folder grouped in a nearest distance value calculated by the centroids using the formula for Euclulidean distance function. Among various clustering based algorithm, we have selected K-means algorithm and also the implementation of K-means algorithm was carried out in java platform. The similarity measures the Euclidean distance between two objects A, and B in N-dimensional space, defined as is one of the most commonly used distance measures, where each object, i.e. time series subsequence of length N, is considered as a point in N-dimensional space.

$$D_E = (\Sigma_{i=1}^{N} (x_A - x_B)^2)^{1/2}$$

Advantages of Euclidean distance measure to include: simplicity and nature, intuitive sense.The disadvantages are: high sensitivity to noise and outliers (especially for sparse data), covering only spherical domain space, demand for extensive data preprocessing if to be applied as time series similarity measure. The Kmeans data set to access and Microsoft Excel have been used for initial preprocessing and storing the data, and in particular to analysis the data sets.

- Indexing subsequences and points within subsequences
- Applying different weights to different dimensions (consecutive data points within each subsequence)
- Normalizing and recycling data (necessary to use text mining)
- To generate the graphs of the clusters

K-means clustering are one of many unsupervised learning techniques that can be used to understand the underlying structure of a dataset that can be grouped the similar posted sentences or documents like the political protests at protest, natural disasters within disasters and others user posts on the social media were posted. Clustering requirement is achieved with the help of K-Means algorithm to group the similar posts Amharic documents with the common phrases to condense one single post and also is the part of the proposed technique. Basic k-means algorithm is relatively simple and first select k initial centers where k is a user specified parameter that is the desired number of clusters. Assign each point of the nearest center, points who is assigned to the same center is a cluster to group with one. Then inform the center of each cluster according to assigned group of cluster posted user posts. On the best clusters formed, document summarization[43] is

executed the documents on sentences weight to focus on important point of the whole document that can be summarizing each group of clusters of summary by using TF-IDF algorithm, which makes it easier about people to determine the information they want and thus read only those post documents which are relevant to their point of view.

**K-means Algorithm**

Input: input the number of cluster k with centroid (mean)

Process: Step 1: partition the data into k- cluster or k- non empty subsets

Step 2: compute the mean for each partition

Step 3: assign each objects of the cluster of its nearest centroid (mean)

Step 4: Step 2 and Step 3 continuous until no change in the mean values and also sum of sequence of distance is minimized within the clusters

Output: number of cluster of partitioned data objects

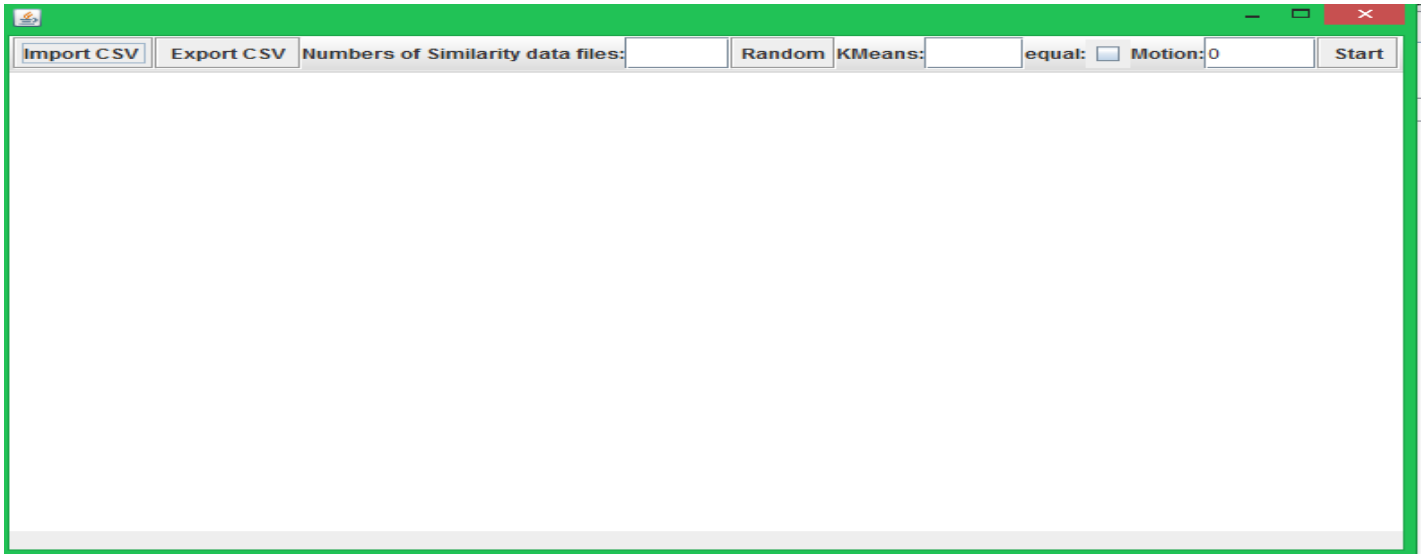


**Figure 3. User interfaces with display clusters**

The above figure 3, shown the user interface of the cluster data onto the text files to change by CSV format to import and clusters to group the similar items of the sentences during the experimentation of the data sets automatically. Example. Clustering the Sports data sets into a K cluster inputs number Kmeans are 3 to display the results of groups of cluster as follows the map.

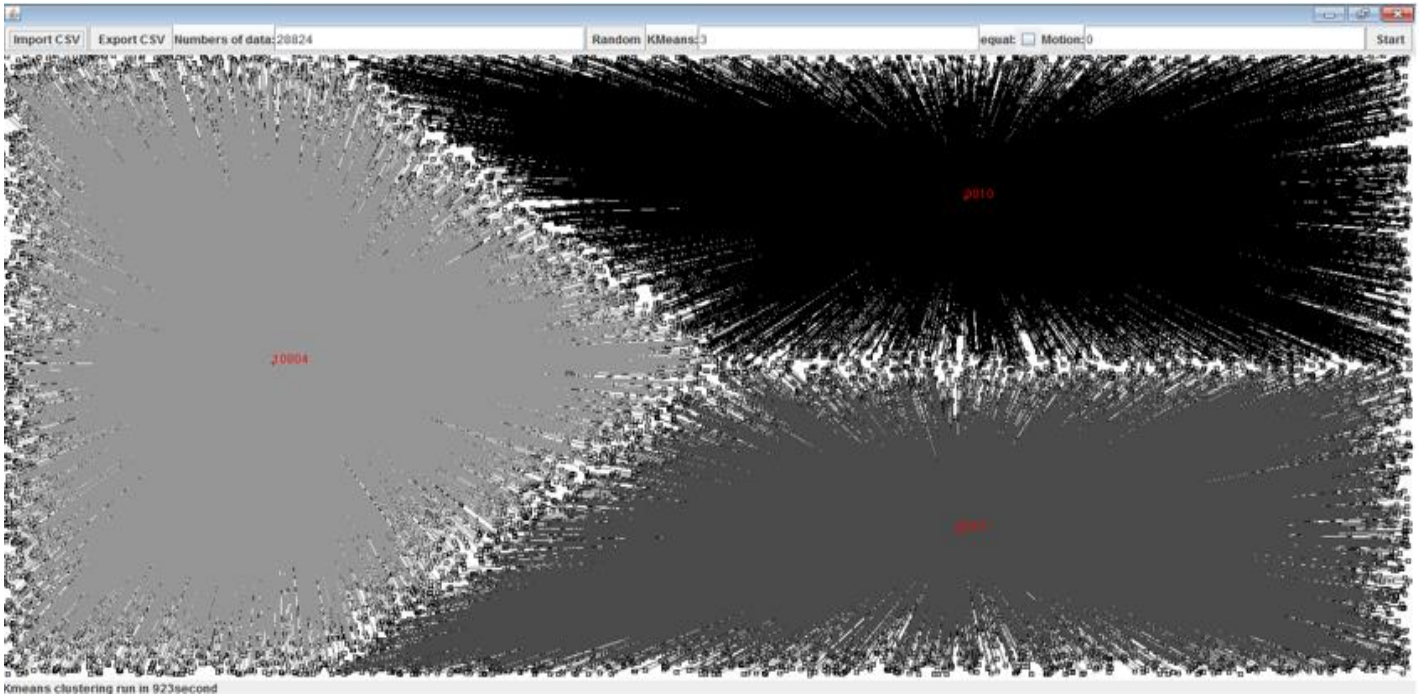


***Figure 4. Example K means clustering for Sports to run the interface results***

As shown to above figure 4, we give the Kmeans of the k number to the cluster the similar posted documents based on similar values. The initialization is well known that results of k-means depend strongly on initial positions of the cluster centroids and their number (the latter being user-defined and thus part of the input).

## 5.2.3. Hybrid TF-IDF algorithm calculates

The TF-IDF is one of the technique that is to pick the most frequently occurring terms (words with high term frequency or tf). However, the most frequent word is a less useful since some words like the stop word occur very frequently to all documents. Hence, we also want a measure of how unique a word is i.e. how infrequently the word occurs to all documents (inverse document frequency or IDF). Hence, the product of tf x IDF of a word gives a product of how frequently, this word is in the document multiplied by how unique the word is written the total corpus of documents. Words in the document with a high tfidf score to occur frequently to the document and provide the most information about that specific document. For instance, to account for bias against longer documents for term frequency or to account for IDF being undefined from a division by zero when the word is not present in the corpus. TF-IDF is not a single method, but a class of techniques where the similarity between queries and documents is measured via the sum of term frequency-like numbers (TFs) multiplied by terms' importance. The term importance is frequently expressed via the IDF (the inverse document frequency, IDF, actually it is the logarithm of the IDF that is used in practice). The using word

frequency is that important words are found many times in one document. Tf and IDF is one of the most common measure used to calculate the word frequency. We were taken to a sample of examples, each document for clustered protests that have been the term of words and its stem found in each post texts in the document.

Document1 Document2 Document3

| Terms words | Stem | Term count | Term | stem | Term count | Term | stem | Term count |
|---|---|---|---|---|---|---|---|---|
| በጃዋር | ጃዋር | 1060 | ጎንደርን | ጎንደር | 4105 | በጃዋርና | ጃዋር | 1857 |
| የኢትዮጵያን | ኢትዮጵያ | 46176 | በመንግሥት | መንግሥት | 19197 | የማይጨርሰው | ጨርሰ | 54 |
| ተደብድቦ | ተደብድቦ | 22 | ተቃዋሚዎች | ተቃዋሚ | 6474 | የማንነት | ማን | 2696 |
| እየተጣሉ | ተጣሉ | 22 | የተማሪዎች | ተማሪ | 16836 | ጥያቄዎቻቸውን | ጥያቄ | 9271 |
| ከተራው | ተራ | 897 | የታሠሩ | ታሠሩ | 2394 | ለነዋሪዎቹ | ነዋሪ | 9516 |
| ስለአማራነት | አማራ | 17919 | ለኢሕአዴግ | ኢሕአዴግ | 3095 | ስለሰልፉና | ሰልፉ | 1495 |
| የወጣው | ወጣ | 8100 | ኃላፊነት | ኃላፊ | 6948 | ታጣቂዎችና | ታጣቂ | 919 |
| | | | ኮሚቴና | ኮሚቴ | 4630 | በወያኔዎች | ወያኔ | 15202 |
| ከመንጋዎቿ | መንጋ | 22 | በኦሮሚያ | ኦሮሚያ | 33662 | ለጦርነት | ጦር | 2036 |
| እንደተነገርው | ተነገር | 161 | በፊትም | ፊት | 3418 | በአሸባሪዎች | አሸባሪ | 594 |
| ኮሚቴና | ኮሚቴ | 6633 | በሽፍትነት | ሽፍት | 9 | እንደትግሬነታችን | ትግሬ | 1891 |
| ጥያቄውንም | ጥያቄ | 15183 | በተደጋጋሚ | ተደጋሚ | 1075 | እርምጃው | ርምጃ | 5201 |
| በመቶዎች | መቶ | 2729 | የምታስጨርሱ | ታስጨርሱ | 6 | የዘገባዎቹን | ዘገባ | 4481 |

***Table 5.5. Terms counts in the documents for posting protests***

As can be seen from above table 5.5 the term count in the document, how many times the term occur to one document to get the important word or term of the summary.

**Term frequency (*ft*)**

TF, which measures how frequently a term occurs of a document. Since every document is different to length, it is possible that a term would appear much more time for the longer document than shorter ones. A Term Frequency is a count of how many times a word occurs to a given document (synonymous with bag of words use multi set). Thus, the term frequency is often divided by the document length (i.e. the total number of terms of the document) as a way of normalization: TF (t, d) = (Number of times term t appears in a document) /

(Total number of terms in the document). Term frequency, we evaluate in only one document, but inverse document frequency (IDF) calculates the term of the word in all documents that found all documents to check the importance of the term in the documents.

**Algorithm for Term frequency**

Read the list of term and frequently key values in HashMap

Iterate itr = tf for the Keyset to iterate for each

While (itr to hash next)

String term= itr next to string

*Find tf* → to calculate the number of times term t appears in a document count over the total number of terms in the document

tfr to put in term and double values

Display the output terms and tfr values results

End while

End

***Algorithm 3. Algorithm for term frequency***

Based on the above algorithm we know the terms or the words how many times to appear in the documents frequently occur to one document. We made to get the results of the documents step by steps as follows.

#### *5.2.3.1. Protests posted texts*

Cluster 1 or document1 term frequency calculates some words to take for the display output within the total of document1 words occur in protests data set is 4,802,014. Below the table doc1 term counts.

| Stem words | Term counts | Term frequency ($ft$) for protests |
|---|---|---|
| ጃዋር | 1060 | 2.2074071420866327E-4 |
| ኢትዮጵያ | 46176 | 0.00962450138119102 |
| ተደብድቦ | 22 | 4.5814100955536374E-6 |

| | | |
|---|---|---|
| ተጣሉ | 22 | 4.5814100955536374E-6 |
| ተራ | 897 | 1.867965843505278 5E-4 |
| አማራ | 17919 | 0.003832433123448611 |
| ወጣ | 8100 | 0.0016867918988174755 |
| መንጋ | 22 | 4.580514408424482E-6 |
| ተነገር | 161 | 3.3521037261651894E-5 |
| ኮሚቴ | 6633 | 0.0013810250941399813 |
| ጥያቄ | 15183 | 0.0031878298230630566 |
| መቶ | 2729 | 5.681919918450187E-4 |

***Table 5.6.Term frequency calculates for protests post texts***

From the above table 5.6, the term frequency calculates tf (t, d) = terms count/ total of the terms of document1 words, Example the term ("በጃዋር") to the word occur in document one is 1060 and total document contain words 4,802,014, to calculate tiff as follows. So, tf ("በጃዋር", d) = 1060/4,802,014=2.2074071420866327E-4

Document2 or cluster2 term frequency some stem words to take for the display output total of document2 words occur in protest post texts in document2 protests data set is 3,941,798. Below table doc2 term counts

| Stem words | Term counts | Term frequency (*ft*) |
|---|---|---|
| ጎንደር | 4105 | 0.0010414029333821774 |
| መንግሥት | 19197 | 0.00487011257299334 |
| ተቃዋሚ | 6474 | 0.0016423977078480429 |
| ተማሪ | 16836 | 0.004271147329213724 |
| ታሠሩ | 2394 | 6.073370578603977E-4 |
| ኢሕአዴግ | 3095 | 7.851746842430789E-4 |
| ታላፊ | 6948 | 0.0017626474010083723 |
| ኮሚቴ | 4630 | 0.0011745908846673523 |
| ኦሮሚያ | 33662 | 0.008539757745069636 |
| ፊት | 3418 | 8.67116985700434 2E-4 |
| ሽፍት | 9 | 2.2832220220315703E-6 |
| ተደጋሚ | 1075 | 2.7271818596488203E-4 |
| ታስጨርሱ | 6 | 1.5221480146877136E-6 |

The above table already the same to calculate the term frequency table 5.6 to term frequency calculate, tf (t, d) = terms counts for the stem/ total of the terms of document2 words.

| Stem words | Term count | Term frequency (*ft*) for protests |
|---|---|---|
| ጀዋር | 1857 | 4.655919524048188E-4 |
| ጨርሰ | 54 | 1.3539022848605393E-5 |
| ማን | 2696 | 6.759482518488915E-4 |
| ጥያቄ | 9271 | 0.0023244496449892705 |
| ነዋሪ | 9516 | 0.002385876693098684 |
| ሰልፉ | 1495 | 3.748303547900937E-4 |
| ታጣቂ | 919 | 2.3041411107163623E-4 |
| ወያኔ | 15202 | 0.0038114856545277626 |
| ጦር | 2036 | 5.104713059214922E-4 |
| አሸባሪ | 594 | 1.489292513346593E-4 |
| ትግሬ | 1891 | 4.741165223465333E-4 |
| ርምጃ | 5201 | 0.0013040084784369749 |
| ዘገባ | 4481 | 0.0011234881737889032 |

Document3 term frequency calculates some stem words to take for the display output within the total of the document3 words occur in the protest cluster data set is 3,988,471. See below the table doc3 term counts.

The above table is the calculation as table 5.6 of term frequency calculate tf (t, d) = terms counts/ total of the terms of document3 words.

**Inverse document frequency (IDF)**

The inverse document Frequency is the number of times a word occurs to a corpus of documents. IDF, which measures how important a term is. While computing TF, all terms are considered equally important. However, it is known that certain terms, such as "ነበር", "ናቸው", "እና", "ይህ", "ያ", may appear a lot of times but have little importance. Thus we need to weight down the frequent terms while scale up the rare ones, by computing the following: IDF (t, Docs) = $\log_{10}$ (Total number of documents of the corpus/ Number of documents with term t in it appears on all documents).

**Algorithm for Inverse document frequency**

Find the idf string and double values found in the hash map

Iterator ir = idf for keyset to iterator

While (ir hash next)

String term=ir next to string

Double idf=math.log10 (total documents in the corpus divided by the term t appears in all documents)

Idfr1 to put in term and idf results

Display output in the form of term and idf values

End while

End

*Algorithm 4.Algorithm for Inverse document frequency*

| stem words | Term count in all docs | Inverse document frequency (*IDF*) |
|---|---|---|
| ጀዋር | 3.0 | 1.0 |
| ኢትዮጵያ | 3.0 | 1.0 |
| ተደብድቦ | 1.0 | 1.4771212547196624 |
| ተጣሉ | 1.0 | 1.4771212547196624 |
| ተራ | 3.0 | 1.0 |
| አማራ | 3.0 | 1.0 |
| ወጣ | 3.0 | 1.0 |
| መንጋ | 3.0 | 1.0 |
| ተነገር | 3.0 | 1.0 |
| ኮሚቴ | 3.0 | 1.0 |
| ጥያቄ | 3.0 | 1.0 |
| ሙዶ | 3.0 | 1.0 |

*Table 5.7. Inverse document frequency for protests*

As indicated in the above table 5.7, the inverse document frequency calculated the term of the protest documents to appear, IDF (term, docs) = $\log_{10}$ (docs/ term appears in all docs). Example the term "በጃዋር" to calculate its edge of all documents to appear the term with in three documents to find and we had been three documents in the corpus. IDF ("በጃዋር", 3) = 1+ $\log_{10}$ (3/3) =1+0. The IDF value is zero means if the term occurs to all documents in the corpus occurred or frequently appear in all documents in the corpus, so this term is usefulness of the documents the IDF to scale down the term values. If the computed score is close to 0, this means that given word is commonly seen among all the documents, meaning that it can be marked as specifically for a certain document.

**TF-IDF (Term frequency- Inverse document frequency)**

TF-IDF stands for "Term Frequency, Inverse Document Frequency". It is a way to score the importance of words (or "terms") in a document based on how frequently they appear across multiple documents. The Tf - IDF is used to weight words, according to how important they are. Words that are used frequently in many documents will have a lower weighting while infrequent one term will have a higher weighting. Intuitively, if a word appears frequently in a document, it's important. It gives the word a high score. But if a word appears in many documents, it is not a unique identifier. It gives that the word a low score in the documents. Therefore, common words like "እና", "ስለ"," ወይም", which appear in many documents, will be scaled down. Words that appear frequently in a single document will be scaled up weight. There can be ways to improve the TF-IDF algorithm, such as by ignoring stop words that could not be added the sentence score in a document.

**Algorithm for Term frequency Inverse document frequency**

Read the results of the tfr1 from hash map the term frequency

Read the results of the idfr1 from hash map in the inverse document frequency

Iterate itrs = tfr1 key Set to iterator

While (itrs has next)

String term = itrs next to string

Double tf = tfr1 get the term

Double idf = idfr1 get the term

Calculate the tf-idf = tf *idf

tf-idf1 to put the term and its Tf-idf values for each

Display the output the term and tf-idf1 values

End

*Algorithm 6.Algorithm for Term Frequency-Inverse document frequency*

| Stem words | Term frequency (*ft*) | Inverse document frequency (*IDF*) | TF-IDF Scores for protests |
|---|---|---|---|
| ጃዋር | 2.2074071420866327E-4 | 1.0 | 2.2074071420866327E-4 |
| ኢትዮጵያ | 0.00962450138119102 | 1.0 | 0.00962450138119102 |
| ተደብድቦ | 4.5814100955536374E-6 | 1.4771212547196624 | 6.7659470161263775E-6 |
| ተጣሉ | 4.5814100955536374E-6 | 1.4771212547196624 | 6.7659470161263775E-6 |
| ተራ | 1.8679658435052785E-4 | 1.0 | 1.8679658435052785E-4 |
| አማራ | 0.003832433123448611 | 1.0 | 0.003832433123448611 |
| ወጣ | 0.0016867918988174755 | 1.0 | 0.0016867918988174755 |
| መንጋ | 4.580514408424482E-6 | 1.0 | 4.580514408424482E-6 |
| ተነገር | 3.3521037261651894E-5 | 1.0 | 3.3521037261651894E-5 |
| ኮሚቴ | 0.0013810250941399813 | 1.0 | 0.0013810250941399813 |
| ጥያቄ | 0.0031878298230630566 | 1.0 | 0.0031878298230630566 |
| መቶ | 5.681919918450187E-4 | 1.0 | 5.681919918450187E-4 |

*Table 5.8. Term frequency-inverse document frequency of protests posted texts*

As discussed the above table 5.8 we calculated the tf-idf value score, example the term ("በጃዋር") to the word occur in the document the multiplication of tf and idf. Tf-IDF (term, d) = tf (term, d) *idf (term, docs) = 2.2074071420866327E-4*1.0 = 2.2074071420866327E-4. This tf-idf results zero or the same for the tf values means if the term occurs to all documents in the corpus. So, tfidf score is zero or the same of tf values, for the word "በጃዋር" which implies that the word is not very informative as it appears on all documents.

### *5.2.3.2. Droughts posted texts*

The numbers of droughts training sets for document to find the frequency term count in social media on Facebook and Twitter posted texts for Amharic, we were taken to a sample of examples, each clustered document that should be the stem of the terms found in each document.

Doc1

| Terms | stem | Term counts |
|---|---|---|
| ኢኮኖሚ | ኢኮኖሚ | 254 |
| የኢትዮጵያን | ኢትዮጵያ | 38755 |
| ረሃብን | ረሃብ | 2925 |
| እርዳታ | ርዳታ | 14235 |
| ዋሽንግተን | ዋሽንግተ | 4483 |
| ድርቅና | ድርቅ | 25800 |
| የምግብ | ምግብ | 6533 |
| የከፋ | ከፋ | 2391 |
| አሳሳቢው | አሳቢ | 1345 |
| ትብብር | ትብር | 986 |
| የመርዳት | መርዳት | 2639 |

Doc2

| Terms | Stem | Term counts |
|---|---|---|
| የገንዘብ | ገንዘብ | 6270 |
| የዩናይትድ | ዩናይትድ | 5771 |
| የልማት | ልማት | 981 |
| ሀላፊነትም | ሀላፊ | 153 |
| እየተደረገ | ተደረገ | 913 |
| የውሀ | ውሀ | 454 |
| የጉድጓድም | ጉድጓድ | 153 |
| እየተቆፈረ | ተቆፈረ | 153 |
| ዝናብ | ዝናብ | 4391 |
| እጥረትም | ጥረት | 9960 |
| የተከሰተውን | ተከሰተ | 3059 |

Doc3

| Terms | stem | Term counts |
|---|---|---|
| ድርቅን | ድርቅ | 20549 |
| ሰራተኞች | ሰራ | 2423 |
| እርዳታው | ርዳታ | 18913 |
| የአሜሪካ | አሜሪካ | 3101 |
| ድጋፍሀን | ድጋፍ | 2854 |
| አካባቢዎችን | አካባቢ | 7313 |
| ያበረከተችውን | በረከተ | 348 |
| ለተራቡ | ተራቡ | 1296 |
| መረጃዎችን | መረጃ | 2057 |
| ገበሬዎች | ገበሬ | 5230 |
| ለችግር | ችግር | 3593 |

***Table 5.9. Term counts for the documents in droughts***

Cluster 1 or document1 term frequency calculates some stem words to take for the experiments that could be displayed output within total of the terms of document1 words occur to droughts training set is 1,605,592. See below doc1 terms counts.

| Stem words | Terms counts | Term frequency ($ft$) for droughts |
|---|---|---|
| ኢኮኖሚ | 254 | 1.5819710113154525E-4 |
| ኢትዮጵያ | 38755 | 0.024137514387216678 |
| ረሃብ | 2925 | 0.001821757955943976 |
| ርዳታ | 14235 | 0.008865888718927349 |
| ዋሽንግተ | 4483 | 0.0027921165526484937 |
| ድርቅ | 25800 | 0.016068839406275067 |
| ምግብ | 6533 | 0.00406890417989128 |
| ከፋ | 2391 | 0.00148917034962805 |
| አሳቢ | 1345 | 8.376972481178282E-4 |

| | | |
|---|---|---|
| ትብር | 986 | 6.141037075421402E-4 |
| መርዳት | 2639 | 0.001643630511362787 |

The above table already the same to term frequency calculate tf (t, d) = terms count/ total of document1 words.

Cluster 2 or document2 term frequency calculates some stem words to take for the experiments that could be displayed output, total of terms with document2 words occur to drought training sets is 1,282,333.

Doc2 term frequency of droughts

| Stem words | Term count | Term frequency (*ft*) for droughts |
|---|---|---|
| ገንዘብ | 6270 | 0.004889525575650006 |
| ዩናይትድ | 5771 | 0.004500391084063188 |
| ልማት | 981 | 7.650118962859101E-4 |
| ሀላፊ | 153 | 1.1931378198954561E-4 |
| ተደረገ | 913 | 7.119835487350009E-4 |
| ውሀ | 454 | 3.5404220276636413E-4 |
| ጉድጓድ | 153 | 1.1931378198954561E-4 |
| ተቆፈረ | 153 | 1.1931378198954561E-4 |
| ዝናብ | 4391 | 0.003424227560235914 |
| ጥረት | 9960 | 0.007767093258927283 |
| ተከሰተ | 3059 | 0.0023854958111504576 |

***Table 5.10. Term frequency of droughts***

As can see the above table 5.10 already the same to term frequency calculate droughts document, tf (t, d) = terms count/ total of document2 words.

Cluster 3 or document3 term frequency calculates some stem words to take for the experiments that could be displayed output total of document3 words occur to drought training set is 1,325,679. Doc3 term frequency for droughts, below table

| Stem words | Term counts | Term frequency (*ft*) for droughts |
|---|---|---|
| ድርቅ | 20549 | 0.015500735849327024 |
| ሰራ | 2423 | 0.0018277426134079215 |
| ርዳታ | 18913 | 0.01426665127832605 |

| | | |
|---|---|---|
| አሜሪካ | 3101 | 0.0023391786397762957 |
| ድጋፍ | 2854 | 0.002152859025450354 |
| አካባቢ | 7313 | 0.005516418378808143 |
| በረከተ | 348 | 2.625069869855372E-4 |
| ተራቡ | 1296 | 9.776122273944144E-4 |
| መረጃ | 2057 | 0.0015516576788196842 |
| ገበሬ | 5230 | 0.003945148109006781 |
| ችግር | 3593 | 0.0027103092075834345 |

As can see the above table already the same to term frequency calculate droughts document, tf (t, d) = terms count/ total of document3 words. See below the IDF calculate of droughts posted texts

| Stem words | Term count all docs | Inverse document frequency (*IDF*) |
|---|---|---|
| ኢኮኖሚ | 1.0 | 1.4771212547196624 |
| ኢትዮጵያ | 3.0 | 1.0 |
| ረሃብ | 3.0 | 1.0 |
| ርዳታ | 3.0 | 1.0 |
| ዋሽንግተ | 3.0 | 1.0 |
| ድርቅ | 3.0 | 1.0 |
| ምግብ | 3.0 | 1.0 |
| ከፋ | 3.0 | 1.0 |
| አሳቢ | 3.0 | 1.0 |
| ትብር | 3.0 | 1.0 |
| መርዳት | 3.0 | 1.0 |

***Table 5.11. Inverse document frequency of droughts***

As indicate to the above table 5.11, the inverse document frequency calculated the term of the drought documents to appear, IDF (term, docs) =1+ $\log_{10}$ (docs/ term appears of all docs).

TF-IDF calculate of droughts

| Stem words | Term frequency (*ft*) | Inverse document frequency (*IDF*) | TF-IDF Score for droughts |
|---|---|---|---|
| ኢኮኖሚ | 1.5819710113154525E-4 | 1.4771212547196624 | |
| ኢትዮጵያ | 0.024137514387216678 | 1.0 | 0.024137514387216678 |
| ረሃብ | 0.001821757955943976 | 1.0 | 0.001821757955943976 |
| ርዳታ | 0.008865888718927349 | 1.0 | 0.008865888718927349 |
| ዋሽንግተ | 0.0027921165526484937 | 1.0 | 0.0027921165526484937 |
| ድርቅ | 0.016068839406275067 | 1.0 | 0.016068839406275067 |
| ምግብ | 0.00406890417989128 | 1.0 | 0.00406890417989128 |
| ከፋ | 0.00148917034962805 | 1.0 | 0.00148917034962805 |
| አሳቢ | 8.376972481178282E-4 | 1.0 | 8.376972481178282E-4 |
| ትብር | 6.141037075421402E-4 | 1.0 | 6.141037075421402E-4 |
| መርዳት | 0.001643630511362787 | 1.0 | 0.001643630511362787 |

***Table 5.12.Term frequency-inverse document frequency of droughts posted texts***

As discussed the above table 5.12, we calculated the tfidf is the multiplications between the term frequency and inverse document frequency to get the tfidf score of the terms.

### *5.2.3.3. Sports posted texts*

As we check the to find the frequency terms counts in the sports posted texts in the documents input corpus table 5.13 below. Doc1 Doc2 Doc3

| Terms | Stem words | Term counts |
|---|---|---|
| የኦሎምፒክ | ኦሎምፒክ | 2537 |
| ማራቶን | ማራቶ | 969 |
| ተሸላሚው | ተሸላሚ | 58 |
| በሞራልና | ሞራ | 40 |
| ኢትዮጵያውያን | ኢትዮጵያ | 3735 |
| እንደተሰማው | ተሰማ | 31 |
| የስፖርት | ስፖርት | 938 |
| አትሌተው | አትሌት | 1501 |

| Terms | Stem words | Term counts |
|---|---|---|
| በሪዮ | ሪዮ | 2718 |
| ሜዳልያን | ሜዳልያ | 381 |
| በኖርዌይ | ኖርዌይ | 26 |
| የአሸናፊነት | አሸናፊ | 52 |
| ከማሸነፉ | ማሸነፉ | 5 |
| እንደገለጸው | ገለጸ | 121 |
| ከሰራው | ሰራ | 8 |
| የለሊሳ | ለሊሳ | 16 |

| Terms | Stem words | Term counts |
|---|---|---|
| የስፖርትና | ስፖርት | 955 |
| የበላይነቱን | በላይ | 399 |
| አበረታች | አበረታ | 1073 |
| መድሃኒትን | መድሃኒት | 578 |
| በመስጠት | መስጠት | 509 |
| ለሯጯ | ሯጯ | 733 |
| የገንዘቤ | ገንዘቤ | 749 |
| የሕገወጥ | ሕገወጥ | 396 |

| በውጪ | ውጪ | 146 | | የህዝብ | ህዝብ | 299 | | ተቀዳጅታለች | ተቀዳጅ | 217 |
|---|---|---|---|---|---|---|---|---|---|---|
| አጋርነቱን | አጋር | 90 | | በመላው | መላ | 401 | | የዳይመንድ | ዳይመንድ | 217 |

*Table 5.13.Term counts of Sports posted texts*

As can be seen from above table 5.13 the term count in the document, how many times the term occur to one document to get the important word or term of the summary. tf (t, d) = term counts for the stem/ total of the terms in the document contains the stem words.

The total of words of document1 occurred is 173, 941,total of words of document2 occurred 208,433, and also the total of words of document3 occurred 174,569, this is to use the tf calculate the total words of the document to use the sports post Amharic texts input clustered documents respectively. Below table 5.14 to calculate the term.

Document1 Document2 Document3

| Stem word | TF for Sports post texts |
|---|---|
| ኦሎምፒክ | 0.014585405396082579 |
| ማራቶ | 0.005570854485141514 |
| ተሸላሚ | 3.334463984914425E-4 |
| ሞራ | 2.299630334423743E-4 |
| ኢትዮጵያ | 0.021472798247681685 |
| ተሰማ | 1.78221350917895E-4 |
| ስፖርት | 0.005392633134223674 |
| አትሌት | 0.00862936282992509 |
| ውጪ | 8.393650720646656E-4 |
| አጋር | 5.174168252453418E-4 |

| Stem word | Term Frequency |
|---|---|
| ሪዮ | 0.013040161586696923 |
| ሜዳልያ | 0.001827925204310257 |
| ኖርዌይ | 1.247403242288889E-4 |
| አሸናፊ | 2.494806484577778E-4 |
| ማሸነፉ | 2.398852389017094E-5 |
| ገለጸ | 5.805222781421368E-4 |
| ሰራ | 3.838163822427351E-5 |
| ለሊሳ | 7.676327644854702E-5 |
| ህዝብ | 0.001434513728632224 |
| መላ | 0.001923879615917095 |

| Stem word | Term Frequency |
|---|---|
| ስፖርት | 0.005470616205071 |
| በላይ | 0.002285629147323 |
| አበረታ | 0.006146587099385 |
| መድሃኒት | 0.003310169520117 |
| መስጠት | 0.002915710466398 |
| ሯጩ | 0.004198912732049 |
| ገንዘቤ | 0.004290567053715 |
| ሕገወጥ | 0.002268439567417 |
| ተቀዳጅ | 0.001240614520528 |
| ዳይመንድ | 0.001243061480528 |

*Table 5.14. Term frequency of Sports posted texts*

As can see the above table already the same to calculate term frequency calculate sports document, tf (t, d) = terms counts / total of the terms of the document contains words.

| Term | Term frequency (*ft*) | Inverse document frequency (*IDF*) | TF-IDF Score for Sports |
|---|---|---|---|
| ኦሎምፒክ | 0.014585405396082579 | 1.0 | 0.014585405396082579 |
| ማራቶ | 0.005570854485141514 | 1.0 | 0.005570854485141514 |

| | | | |
|---|---|---|---|
| ተሸላሚ | 3.334463984914425E-4 | 1.1760912590556813 | 3.921633946293831E-4 |
| ሞራ | 2.299630334423743E-4 | 1.1760912590556813 | 2.7045751353750553E-4 |
| ኢትዮጵያ | 0.021472798247681685 | 1.0 | 0.021472798247681685 |
| ተሰማ | 1.78221350917895E-4 | 1.1760912590556813 | 2.096045729915668E-4 |
| ስፖርት | 0.005392633134223674 | 1.0 | 0.005392633134223674 |
| አትሌት | 0.00862936282992509 | 1.0 | 0.00862936282992509 |
| ውጪ | 8.393650720646656E-4 | 1.0 | 8.393650720646656E-4 |
| አጋር | 5.174168252453418E-4 | 1.1760912590556813 | 6.085294054593875E-4 |

***Table 5.15. TF-IDf calculation both tf and idf posted sports texts***

As indicate to the above table 5.15, the inverse document frequency calculated the term of the posted sport documents to appear, IDF (term, docs) = $1+\log_{10}$ (docs/ term appears of all docs) and also the tfidf to calculate the multiplications of the term frequency and the inverse document frequency that we found the results of the term.

### *5.2.3.4. Floods posted texts*

The floods training sets to find the frequency terms counts for the given documents to identify the frequent terms and its roots as follows table 5.16

Document1

| Terms | Stem words | Term counts |
|---|---|---|
| በኢትዮጵያ | ኢትዮጵያ | 316 |
| ከመኖሪያ | መኖሪያ | 61 |
| ቀያቸው | ቀያ | 11 |
| በጎርፉ | ጎርፉ | 325 |
| የከተማዋ | ከተማ | 62 |
| የጣለው | ጣለ | 145 |
| ዝናብም | ዝናብ | 454 |
| አደጋዎች | አደጋ | 728 |
| በድሬደዋ | ድሬደዋ | 301 |
| መወሰዱንም | ወሰዱ | 20 |
| በዘገባው | ዘገባ | 41 |

Document2

| Terms | Stem words | Term count |
|---|---|---|
| አካባቢዎች | አካባቢ | 99 |
| በባሌና | ባሌ | 120 |
| በወላይታ | ወላይታ | 143 |
| የጠፉና | ጠፉ | 24 |
| ሞተዋል | ሞተ | 48 |
| በጎርፍ | ጎርፍ | 770 |
| ከ50 | 50 | 72 |
| እንደዘገብንው | ዘገብ | 24 |
| የመሬት | መሬት | 45 |
| በቤተሰብም | ቤተሰብ | 72 |
| ወረዳዎች | ወረዳ | 24 |

Document3

| Terms | Stem words | Term count |
|---|---|---|
| የውሃና | ውሃ | 55 |
| የምግብም | ምግብ | 110 |
| ለነዋሪዎች | ነዋሪ | 165 |
| በህይወት | ህይወት | 53 |
| በመጥለቅለቁ | መጥለቅለቁ | 55 |
| በአሰላ | አሰላ | 4 |
| የሃገሪቱን | ሃገሪቱ | 37 |
| ኤጀንሲስ | ኤጀንሲ | 46 |
| በአፋር | አፋር | 20 |
| በጎርፍ | ጎርፍ | 723 |
| ጠፉ | ጠፉ | 46 |

*Table 5.16.Term counts of floods posted texts*

As shown the above table 5.16, the term counts for the words contains in the document, how many times the term occur in one document.

The total of the terms in the document1 contains words occurred 28,355, the total of the terms in the document2 words occurred 24,620, and also the total of the terms in the document3 words occurred 23,719, to the following table to use the floods respectively in the document for words to calculate the term frequency respectively. See below table 5.17

Document1 Document 2 Document3

| Stem words | TF for floods posted texts |
|---|---|
| ኢትዮጵያ | 0.011144418973725975 |
| መሞሪያ | 0.0021512960677129256 |
| ቀያ | 3.879386351613472E-4 |
| ነርፉ | 0.011461823311585259 |
| ከተማ | 0.002186563216363957 |
| ጣለ | 0.005113736554399576 |
| ዝናብ | 0.01601128548756833 |
| አደጋ | 0.025674484217950978 |
| ድሬደዋ | 0.0106154117439605 |
| ወሰዱ | 7.053429730206313E-4 |
| ዘገባ | 0.0014459530946922942 |

| Stem words | TF |
|---|---|
| አካባቢ | 0.004021121039805037 |
| ባሌ | 0.00487408610885459 |
| ወላይታ | 0.005808285946385053 |
| ጠፉ | 9.748172217709179E-4 |
| ሞተ | 0.0019496344435418359 |
| ነርፍ | 0.03127538586515029 |
| 50 | 0.002924451665312754 |
| ዘጉበ | 9.748172217709179E-4 |
| መሬት | 0.0018277822908204712 |
| ቤተሰብ | 0.002924451665312754 |
| ወረዳ | 9.748172217709179E-4 |

| Stem words | TF |
|---|---|
| ውሃ | 0.002318816138960327 |
| ምግብ | 0.004637632277920654 |
| ነዋሪ | 0.006956448416880982 |
| ህይወት | 0.0022344955520890424 |
| መጥለቅለቁ | 0.002318816138960327 |
| አሰላ | 1.6864117374256924E-4 |
| ሃገሪቱ | 0.0015599308571187655 |
| ኤጀንሲ | 0.0019393734980395463 |
| አፋር | 8.432058687128462E-4 |
| ነርፍ | 0.030481892153969393 |
| ጠፉ | 0.0019393734980395463 |

*Table 5.17.Term frequency of posting floods in Amharic texts*

As can see the above table 5.17, already the same to term frequency calculate floods document, tf (t, d) = terms count/ total of document words. From the above table to calculate the term "ነርፉ" is in the document one term counts is 325 occur. The total document the word occurs document 1 is 28,355. To calculate the term "ድርጅቶች" tf ("ነርፉ", doc1) = term count/total document1 words occur= 325/28,355=0.011461823311585259.

| Stem words | Term frequency | Inverse document frequency(Idf) | TF-IDF score for floods posted |
|---|---|---|---|
| ኢትዮጵያ | 0.011144418973725975 | 1.0 | 0.011144418973725975 |
| መሞሪያ | 0.0021512960677129256 | 1.0 | 0.0021512960677129256 |
| ቀያ | 3.879386351613472E-4 | 1.4771212547196624 | 5.730324035237625E-4 |
| ጎርፉ | 0.011461823311585259 | 1.0 | 0.011461823311585259 |
| ከተማ | 0.002186563216363957 | 1.0 | 0.002186563216363957 |
| ጣለ | 0.005113736554399576 | 1.0 | 0.005113736554399576 |
| ዝናብ | 0.01601128548756833 | 1.0 | 0.01601128548756833 |
| አደጋ | 0.025674484217950978 | 1.0 | 0.025674484217950978 |
| ድሬደዋ | 0.0106154117439605 | 1.0 | 0.0106154117439605 |
| ወሰዱ | 7.053429730206313E-4 | 1.0 | 7.053429730206313E-4 |
| ዘገባ | 0.0014459530946922942 | 1.0 | 0.0014459530946922942 |

***Table 5.18. TF-IDf calculation both tf and idf of floods posted texts***

As discussed the above table 5.18, we calculated the tf-idf is the multiplications between the term frequency and inverse document frequency to get the tfidf scored for the term weight. Tfidf score to identify the most important term to score the sentences in the summary posted texts.

**Sentence Scores calculated**

If the stem words again to build the sentence and the sentences to score each stem word read the value of the tf-IDF score term weights in the hash map stores line by line of the summation of sentences within the word score values. The score sentence can be made to be proportional to the sum of frequencies of the different words in line comprising the sentence. Some example of the corpus to take and we made to summation of each sentence of terms or words to build the sentence final to get the score sentence.The formula for the equations

$$\text{Score}(q,d) = \sum_{t \in q} \text{tf-idf}_{t,d}.$$

| Sentences score | Values |
|---|---|
| 29-Sep-2016 የኢትዮጵያን መንግሥት የሚቃወም የአሜሪካን መንግሥት ድጋፍ የሚጠይቅ ሰልፍ በዋሽንግተን ተካሄደ። | 0.026182066152337157 |
| 18-Dec-2015 "ወጣቱ ይህ መንግሥት በሚያደርገው ነገር ሁሉ ተንገሽግሿል" #OromoProtests #Ethiopia ላለፉት ሦስት ሳምንታት የኦሮሞ ብሔረሰብ አባላት ተማሪዎች የአዲስ አበባ እና የፊንፊኔ ዙሪያ የኦሮሚያ ልዩ ዞን የተቀናጀ ማስተር ፕላን ተግባራዊ ይደረጋል መባሉን ሲቃወሙ ሰንብተዋል። | 0.05766890854273876 |
| 2-Feb-2016 ኢትዮጵያውያን ዛሬ በዋሽንግተን ዲሲ ሰላማዊ ሰልፍ አደረጉ "በኦሮሚያ፥ በጎንደርና በጋምቤላ በዜጎች ላይ እየደረሰ ያለው በደል ይቁም" "አምባገነኖች የልማት ሃጋር አይደሉም" የዛሬው የዋሽንግተን ሰልፈኞች የኦሮሚያን ክልላዊ ምክር ቤት እንበትናለን ባሉት መሰረት በተደረገው ግምገማ የክልሉ ምክር ቤት ፕረዚዳንት እና ኣባላትን ጨምሮ የፈደራል መንግስታቸውን ባለስልጣናት እና የኦሕዴድ ኣመራሮች እንደሚያሰናበቱ እየተጠበቀ ይገኛል። | 0.07195030682152953 |
| ሂውማን ራይትስዎችና ሌሎች አለም አቀፍ የሰብዓዊ መብት ተቋማት ለወራት በዘለቀው በዚሁ ተቃውሞ ከ700 የሚበልጡ ሰዎች በመንግስት የጸጥታ ሃይሎች መገደላቸውን ሲገልፅ ቆይተዋል | 0.03182795429330759 |
| 13-Oct-2016 በአማራና ኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ። | 0.02963092170848662 |

***Table 5.19. Sentences score results***

The above table 5.19 is an example Amharic text for each word to sum of the key values to add and to represent the sentences score without sorting the sentences to show that a simple illustrations of the sentence scoring algorithm.

**Sentences ranking**

Sentence ranking is after scoring the sentences of each sentence, sentences are arranged in descending order of its score by higher value, i.e. the post sentences whose score value is the highest that is in top position and the sentence whose score value is the lowest is in the bottom position. Some example to take for the protests corpus and we made to sentence rank formed higher score into lowest score sentence.

| Top ranked sentences | Values |
|---|---|
| 11-Mar-2016 በኢትዮጵያ ኦሮሚያ ክልል ውስጥ ከተቀሰቀሰው ተቃውሞ ጋር በተያያዘ የኦሮምያ ዳያስፖራ ማኅበረሰብ ሰላማዊ ሰልፍ በዋሽንግተን ዲሲ #Oromo #OromoProtests #Ethiopia የኢትዮጵያ መንግሥት ኦሮምያ ውስጥ የተፈጠረውን ተቃውሞ በሰላማዊ መንገድ እንዲፈታ የሚጠይቁ ሰልፎች ዋሺንግተን ዲሲ ላይና አውሮፓ ውስጥ ደግሞ በርሊን ተካሂደዋልነ። | 0.11110838102282201 |

| | |
|---|---|
| 1-Apr-2016 ከተጀመረ አምስተኛ ወሩን ሊደፍን ጥቂት ቀናት የቀረው የኦሮሚያ ክልል ከተሞች ተቃውሞ በአንዳንድ ከተሞች በሚገኙ ትምሕርት ቤቶች ከትናንት በስቲያ፣ ትናንትና ዛሬ በቢሾፍቱ ከተማና ደባዩ በተባለ አንደኛ ደረጃ ትምሕርት ቤት እንዲሁም በበደሌ ከተማና ለሊሳ ሃሮ ቶሬ በተባለች የገጠር ቀበሌ ተማሪዎች እንደታሠሩና በአስለቃሽ ጭስ ምክንያት ሆስፒታል እንደገቡ ነዋሪዎችና ተማሪዎች ለአሜሪካ ድምጽ ተናግረዋል በበደሌና በቢሾፍቱ ተማሪዎች እየታሠሩ መኾኑን ተማሪዎች ገለጹ። | 0.08698136505807359 |
| 10-Dec-2015 በኦሮሚያ ግድያና እስራት ይቁም" በዋሽንግተን ዲሲ የተካሄደ ሰልፍ #OromoProtests #Ethiopia #Oromo #DC #HornofAfrica ከዋይት ሃውስ ቤተ-መንግሥት እስከ ውጭ ጉዳይ መስርያ ቤት "በኦሮሚያ መብት ይከበር፣ የተማሪዎች ግድያና እስራት ይቁም" በማለት በመቶዎች የሚቆጠሩ ኢትዮጵያውያን ድምጻቸውን አሰምተዋልነ። | 0.07829483699910365 |
| 2-Feb-2016 ኢትዮጵያውያን ዛሬ በዋሽንግተን ዲሲ ሰላማዊ ሰልፍ አደረጉ "በኦሮሚያ፣ በጎንደርና በጋምቤላ በዜጎች ላይ እየደረሰ ያለው በደል ይቁም" "አምባገነኖች የልማት ሃጋር አይደሉም" የዛሬው የዋሽንግተን ሰልፈኞች የኦሮሚያን ክልላዊ ምክር ቤት እንበትናለን ባሉት መሰረት በተደረገው ግምገማ የክልሉ ምክር ቤት ፕረዚዳንት እና ኣባላትን ጨምሮ የፊደራል መንግስታቸውን ባለስልጣናት እና የኡሕዴድ ኣመራሮች እንደሚያሰናበቱ እየተጠበቀ ይገኛልነ። | 0.07195030682152953 |
| 18-Dec-2015 "ወጣቱ ይህ መንግሥት በሚያደርገው ነገር ሁሉ ተንገሽግሿል" #OromoProtests #Ethiopia ላለፉት ሦስት ሳምንታት የኦሮሞ ብሔረሰብ አባላት ተማሪዎች የአዲስ አበባ እና የፊንፊኔ ዙሪያ የኦሮሚያ ልዩ ዞን የተቀናጀ ማስተር ፕላን ተግባራዊ ይደረጋል መባሉን ሲቃወሙ ሰንብተዋልነ። | 0.05766890854273876 |
| 13-Oct-2016 በአማራና ኦሮሚያ ክልሎች ከ500 ሰዎች በላይ መሞታቸውን የኢትዮጵያ መንግስት አመነ። | 0.02963092170848662 |

***Table 5.20.Top ranked sentences results***

As can see the above table 5.20 to show that the higher top ranked sentences as a simple example of the Amharic sentence ranking algorithm.

## 5.2.4. Summarize clustered documents

We could summarize based on clustered documents for each group of clusters document to summarize using the TF-IDF score to produce the summary documents for each clustered document. We have been three clustered post documents. This means cluster1, we can say that document1, cluster2 we can say that document2, cluster3 we can say document3 and also three summary documents, cluster1 to have one summary, cluster2 to one summary and the third cluster to have one summary individual it has been their own summary of each cluster posts text (shown in table 5.5, 5.9, 5.13 and 5.16). A general summary is good if it covers all the important words of the document the training data sets, whereas a user-focused summary should contain

only the topics which are directly related to those selected by a user. In addition to comparing the information on the posted text documents for a sentence, judges can be asked to the selection of the system summary.

The summary is after ranking the Amharic post document in the sentence level on its total score the summary is produced selecting N number of top ranked sentences, where the value of N is provided by the user. For the readers' suitability, the selected relevant document for Amharic sentences in the summary is reordered according to its original positions in the Amharic post texts. The purpose of a summary is to give a reader a condensed and objective justification of the main ideas and features of a text of posting documents. In the summary document, the following are the important reasons to support of manual and automatic document summarization to use the reader for Amharic documents as the following:

- A summary saves reading time and efforts
- It improves Amharic document indexing efficiency
- A summary is simplifying the selection and literature of investigations of all summary documents
- Modified summaries can be useful in question-answering systems where they provide personalized information to be easily recorded.
- Machine generated summary is free from bias or wrong
- The use of automatic or semi-automatic documents summarization of profitable, abstract services can allow them to scale the number of available document they can evaluate
- To sum up information
- To present information concisely or briefly
- To state the main or essential points without any detailed explanations
- To refer to a body of work

### *5.2.4.1. News items posted text summaries*

We implement the news items posted texts in the current time to post texts in social media like protests, droughts, sports, floods and others to post in the social media on Facebook, Twitter and others social media, but we focused on the post texts on Facebook and Twitter post Amharic texts to summary by using date, month and years to summaries the poster texts to give to the readers within a short period of time to summaries the posted documents. News items posted to identified by the time of the intervals of the period posted texts in the social networks/media like to use the date, weeks, months, years, one year's etc. posted texts to summarize the texts. For example, if the user selected for political protests summary to need within the monthly to group for multi documents within an one month and to want to summarize by entering the month

and year into the system to find automatically the system group posted text document based on the user asked the month to group the same month and years rank each posted sentences for a multi documents post and also give to the summarize sentences to the reader. If the compression ratio to calculate the selected posted sentence for monthly in to post text documents divided by the total monthly posted text documents the give the percentage. Example protests posted texts to display each clustered text with console in Java programming (NetBeans).

For clustered document 1

How many sentences do you want select in doc1 to the summarization: 2?

Enter date of summary time Month-Year: Mar-2016

Total numbers of posted text documents in doc1:104

The compression ratio of doc1 is: 98.07692307692308

1)8-Mar-2016 በኦሮሚያ ዘገባ ለማቅረብ የተንቀሳቀሱ ጋዜጠኞች በቁጥጥር ስር ውለው ተለቀቁ #Ethiopia #FreePress #OromoProtests #Journalism: 1.8259625047899546E-5

2)12-Mar-2016 በኢትዮጵያ ኦሮሚያ ክልል ውስጥ ከተቀሰቀሰው ተቃውሞ ጋር በተያያዘ የኦሮምያ ዳያስፖራ ማኅበረሰብ ሰላማዊ ሰልፍ በዋሽንግተን ዲሲ #Diaspora #Oromo #OromoProtests በኦሮሚያም ሆነ በሌሎች የሃገሪቱ ክልልች ለተፈጠረው የሕዝብ ቅሬታ፥ መንግሥታቸውና ፓርቲያቸው ሙሉ ኃላፊነት እንደሚወስዱ ጠቅላይ ሚኒስትር ኃይለማርያም ደሳለኝ አስታወቁ: 1.60538780117E-5

==========

For clustered document 2

How many sentences do you want select in doc2 to the text Summarization: 1?

Enter date of summary time Month-Year: Jun-2016

Total number posted text documents in doc2: 40

The compression ratio of doc2 is: 97.5

1)15-Jun-2016 ቤንች ማጂ ዞን የርእሰ መምሕር ቢሮ በመታሸጉ ትምሕርት መስተጓጎሉን ወላጆች በተቃውሞ ተናገሩ: በደቡብ ክልል ቤንች ማጂ ዞን የሚገኝ የሁለተኛ ደረጃ ትምሕርት ቤት ርእሰ መምሕር ቢሮ በመታሸጉ : 0.102

==========

For clustered document 3

How many sentences do you want select in doc3 to the summarization: 3?

Enter date of summary time Month-Year: Aug-2016

Total numbers of posted text documents in doc3: 90

The compression ratio of doc3 is: 96.666667

1)23-Aug-2016 መንግስት እየወሰደ ያለው የሃይል እርምጃ ተቃውሞን ከማባባስ ውጭ መፍትሄ አያመጣም ተባለ ኢሳት (ነሃሴ 16 ፥ 2008) የኢትዮጵያ መንግስት በኦሮሚያና በአማራ ክልሎች ሰላማዊ ሰልፈኞች ላይ እየወሰደ ያለው የሃይል እርምጃ ህዝባዊ ተቃውሞውን ከማባባስ ውጭ መፍትሄ አያመጣም ሲሉ የአባማ አስተዳደር የዲሞክራሲና የሰብዓዊ ጉዳዮች ከፍተኛ ባለስልጣን ገለጹ የኢትዮጵያ ቀጣዩ ብሄራዊ ስራ የፖለቲካ ምህዳሩን ማስፋት እንደሆነ ገልጸዋል:2.0143475267321004E-4

2) ሃላፊው በቅርቡ በአማራና በኦሮሚያ ክልሎች የተከስቱ ህዝባዊ ተቃውሞዎች ለኢትዮጵያና ለአሜሪካ ከፍተኛ ፈተና እንደሆኑ ቶም ማሊኖውስኪ ባሰራጩት ጹሁፍ ገልጸው፣ በህገመንግስቱ የተገለጹ የአስተዳደርና ችግርና የፖለቲካ ብዝሃነት ማነስ ላይ ተመስርተው የቀረቡ ጥያቄዎች ናቸው ሲሉ በጹሁፋቸው አብራርተዋል:1.704464172197319E-4

3) በራስ ተነሳሽነት ለሰላማዊ ሰልፍ በነቂስ የወጣው የቡሬ ከተማና አጎራባች መንደሮች ህዝብ፣ የብዓዴን የድርጅት አርማም እንዳቃጠለ፣ "ወያኔ ሌባ" በማለት ተቃውሞ ሲያሰሙ እንደዋሉ ለማወቅ ተችላል:1.598495843953218E-4

========== the three documents are completely finished the process==========

**Droughts posted texts to summaries by using the news items posted identified to date, month and year's texts**

If the user selects the posted droughts to need within monthly to the group and to want to be summarized by enter month and year to the system to find automatically the system group posted text document based on the user asked the month to group the same month and years, ranked each posted sentences list for a multi documents post and also give to the summarize sentences for the reader easily. Example droughts to display on console in Java.

For clustered document 1

How many sentences do you want select in doc1 to the summarization: 2?

Enter date of summary time Month-Year: Jun-2016

Total numbers of posted text documents in doc1: 90

The compression ratio of doc1 is: 97.77

1)10-Jun-2016 በትግራይ የቀበሌ አስተዳዳሪዎች ለእርዳታ የተላከውን እህል ለማዳበርያ ክፍያ እየሸጡት ነው ሲሉ ያአከባቢው ነዋሪዎች ይከሳሉ #Tigray #FoodAid #Ethiopia መቐለ በትግራይ ክልል በራያ አላማጣ ወረዳ የገጠር ቀበሌ አስተዳዳሪዎች ለእርዳታ የተላከውን እህል ለማዳበርያ ክፍያ እንዲውል በጨረታ እየሸጡት ነው ሲሉ ያከባቢው ነዋሪዎች ይከሳሉ: 0.0031675525793529893

2)28-Jun-2016 በኢትዮጵያ በኤል-ኒኞ ምክንያት የተከሰተው ድርቅ ቢያበቃም ተፅዕኖው እንደሚቀጥል ተገለጸ #Ethiopia #ElNino #Drought በኤል-ኒኞ ምክንያት የተከሰተው ድርቅ ቢያበቃም አሉታዊ ተፅዕኖው እንደሚቀጥል የብሔራዊ የአደጋ ስጋት ሥራ አመራር ኮሚሽነር አስታወቁ ኮሚሽነሩ አቶ ምትኩ ካሳ ለአሜሪካ ድምፅ እንዳስታወቁት እርዳት የሚቀርብለት ህብረተሰብ ከዚህ ማእቀብ ሊወጣ የሚችለው ምርት ከተሰበሰበ በኃላ ነው: 0.0027313860513231702

==========

For clustered document 2

How many sentences do you want select in doc2 to the text Summarization: 2?

Enter date of summary time Month-Year: Apr-2016

Total number posted text documents in doc2: 162

The compression ratio of doc2 is: 98.76543209876543

1) የመንግስት እና የሰብዓዊ ርዳታ አጋሮች ልዑካን ቡድን ለኢትዮጵያ ድርቅ ምላሽ የሚሆን ድጋፍ ለማሰባሰብ ወደ አውሮፓና አሜሪካ ተንቀሳቅሷል የተመድ የሰብዓዊ ርዳታ ማስተባባሪያ ጽህፈት ቤት እንዳለው ጉዞው አጣዳፊ ምላሽ ለማግኘት የታለመ ነው እስከንድር ፍሬው ዝርዝሩን ልኳል: 0.0022751180743149587

2) 28-Apr-2016 የመንግስት እና የሰብዓዊ ርዳታ አጋሮች ልዑካን ቡድን ለኢትዮጵያ ድርቅ ምላሽ የሚሆን ድጋፍ ለማሰባሰብ ወደ አውሮፓና አሜሪካ ተንቀሳቅሷል #EthiopiaDrought: 0.0017135382964777223

---

For clustered document 3

How many sentences do you want select in doc3 to the summarization: 1?

Enter date of summary time Month-Year: Dec-2015

Total numbers of posted text documents in doc3: 66

The compression ratio of doc3 is: 98.48484848484848

1) ማላዊትዋ ሊሳ ፋቺ ያብራራሉ ‘ከነርፍ መጥለቅለቁ አደጋ በፊት ልጇ ደህና ነበር እህላችንን ካጣን በኋላ ግን ልጇ ታሞ በምግብ እጥረት ተጎዳ ወደ ሆስፒታሉ የመጣሁትም ህክምናና ምግብ ፍለጋ ነው’’ ይላሉ ሊሳ ፋቺ የማላዊ ፕረዚዳንት የብሄራዊ ቀውስ አዋጅ አውጀዋል: 8.757701567205155E-4

==========the three documents are completely finished the process==========

**Sports posted texts to summaries by using the news items posted identified to date, month and years**

If the user selected the posted text in sports to need within the monthly to group only one month and to need for summarizing by entering the month and year into the system to originate automatically the system group posted text document based on the user asked the month to group the same month and years, ranked each posted sentences list for a multi documents post and also give to the summarize sentences for the reader easily. Example Sports to display in the Java platform.

For clustered document 1

How many sentences do you want select in doc1 to the summarization: 3?

Enter date of summary time Month-Year: Dec-2015

Total numbers of posted text documents in doc1: 5

The compression ratio of doc1 is: 40.0

1)በዚህ ስፖርት ሥነ ስርዓት ላይ ክቡር ጠ/ሚ/ር ኃ/ማርያም ደሳለኝ፣ ክቡር አቶ ሬድዋን ሁሴን የኢፌዲሪ ወጣቶችና ስፖርት ሚኒስቴር ሚኒስትር ክቡር አቶ አንበሳው እንየው የኢፌዲሪ ወጣቶችና ስፖርት ሚኒስቴር ሚኒስትር ዴኤታ ክቡር አቶ ብርሃነ ኪ/ማርያም የኢትዮጵያ

አሎምፒክ ኮሚቴ ፕሬዝዳንት፣ ክቡር አቶ አለባቸው ንጉሴ የኢትዮጵያ አትሌቲክስ ፌዴሬሽን ፕሬዝዳንት ጥሪ የተደረገላቸው ከፍተኛ የመንግስት የሥራ ኃላፊዎችና የኢአፌ ሥራ አስፈፃሚ ኮሚቴ አባላት በስፍራው ተገኝተዋል:0.015820155944169686

2) ክቡር ጠ/ሚ/ሩ ባሰሙት ንግግር በአለም ላይ የስፖርቱ በተለይም የአትሌቲክሱ ስጋት እየሆነ የመጣው የአበረታች መድኃኒቶችና እፆች ጉዳይ ወደ ሃገራችን እንዳይመጣ ሁሉም የድርሻውን እንዲወጣና አትሌቶችም ከፍተኛ ኃላፊነት ያለባቸው መሆኑንና የሃገራቸውን ስምና ዝና በዚህ ረገድም ማስጠበቅ እንደሚገባቸው በአፅንኦት አስገንዝበዋል:0.0074741193179041696

3) መንግስት ለአትሌቶች በአመቱ ስመ ጥር ለነበሩ 3 አትሌቶችም የ33 ግራም ባለ 21 ካራት ወርቅ ስጦታ የተበረከተላቸው ሲሆን በዚህ ውስጥ የተካተቱት አትሌት ማሬ ዲባባ፣ ገንዘቤ ዲባባና አልማዝ አያና ከኢፌዲሪ ጠ/ሚ/ር ከክቡር አቶ ኃ/ማርም ደሳለኝ እጅ ተበርክቶላቸዋል:0.006603422214588046

==========

For clustered document 2

How many sentences do you want select in doc2 to the text Summarization: 2?

Enter date of summary time Month-Year: Nov-2015

Total number posted text documents in doc2: 4

The compression ratio of doc2 is: 50.0

1)2-Nov-2015 ከሁለት ዓመታት በፊት በኖርዌይ በተደረገ ተደጋጋሚ የማራቶን ሩጫ ኢትዮጵያን ወክሎ በመወዳደር አንደኛ የወጣውና የአሸናፊነት ገመዱን ሲበጥስ ሁለት እጆቹን ወደ ላይ በማጣመር የተቃውሞ ምልክት ያሳየው አትሌት ሙሉጌታ ዘውዴ በከፍተኛ ችግርና በህመም ላይ እንደሚገኝ ገለጸ:0.020222867805150042

2)14-Nov-2015 የብሄራዊ ቡድኑ ተጫዋች ራምኬል ሎክ ተጫዋቹ ከከሳሾቹ ጋር እርቅ በማውረዱ እነሱም መታረቃቸውን ቀርበው በማስረዳታቸው ጉዳዮ ተዘግቶ ነገር ግን ከጥፋት እንዲማር በሚል ብቻ 3 ሺህ ብር ተወስኖበት ያለገደብ እንዲለቀቅ ሆኗል:0.003377092639424025

==========

For clustered document 3

How many sentences do you want select in doc3 to the summarization: 2?

Enter date of summary time Month-Year: Jul-2016

Total numbers of posted text documents in doc3: 6

The compression ratio of doc3 is: 66.66666666666666

1)4-Jul-2016 ኢትዮጵያዊ ፅጋቡ ግርማይ ኤርትራዊያን ዳንኤል ተኽለሃይማኖት እና ናትናኤል ብርሃን ስፖርት 103ኛ ቱርዲ ፍራንስ ከሃምሌ 2 እስከ 24 በሚካሄደው ለነዚህ ተወዳዳሪዎች መልካም እድል እንመኛለን:0.005135995055790703

2)16-Jul-2016 ኢትዮጵያ ለሪዮ ኦሎምፒክ የመረጠቻቸውን 43 አትሌቶች ስም ዝርዝር ይፋ አደረገች በሪዮ ከተማ ብራዚል የሚካሄደው 31ኛው የኦሎምፒክ ውድድር ሊጀመር የቀሩት ጥቂት ቀናት ናቸው:0.004402281476392031

========== the three documents are completely finished the process==========

**Floods posted texts to summaries by using the news items posted identified to date, month and year texts**

If the user selects the posted text of floods to need within the monthly to group only and to need for summarizing by entering the month and year into the system to originate automatically the system group posted text documents based on the user asked the month to group the same month and years, ranked each posted sentences list and also give to the summarize sentences for the reader easily. Example floods

For clustered document 1

How many sentences do you want select in doc1 to the summarization: 2?

Enter date of summary time Month-Year: Jul-2016

Total numbers of posted text documents in doc1: 54

The compression ratio of doc1 is: 96.29629629629629

1)5-Jul-2016 በአዳማ ከተማ በደረሰ የጎርፍ አደጋ የብዘ ሰዎች ተፈናቀሉ የአሜሪካ ድምፅ ያነጋገራቸው ከግማሽ ሚሊዮን ሜትሪክ ቶን በላይ የእርዳታ እህልን የጫኑት መርከቦት ካለፈው ወር አጋማሽ ጀምሮ የጫኑትን የእርዳታ እህል ማራገፍ እንዳልቻሉ ለመረዳት ተችሏል:0.010210645383053342

2) በአሁኑ ወቅት ከ10 በላይ መርከቦች የእርዳታ እህልን እንደጫኑ በወደቡ ወረፋን እየተጠባበቁ እንደሚገኝ የወደቡ ባለስልጣናት ይገልጻሉ:0.006123859807581863

==========

For clustered document 2

How many sentences do you want select in doc2 to the text Summarization: 1?

Enter date of summary time Month-Year: Jun-2016

Total number posted text documents in doc2: 3

The compression ratio of doc2 is: 66.66666666666666

1)6-Jun-2016 በተለያዩ አካቢዎች በኢትዮጵያ በደረሰው በጎርፍ አደጋ ከቤት ንብረታቸው እንደሚፈናቀሉም የብሄራዊ የአደጋ ስታት ስራ አመራር ኮሚሽን ተናግሯል:0.002923609665036853

==========

For clustered document 3

How many sentences do you want select in doc3 to the summarization: 2?

Enter date of summary time Month-Year: Apr-2016

Total numbers of posted text documents in doc3: 11

The compression ratio of doc3 is: 81.81818181818181

1)25-Apr-2016 ጎርፉ በሆቴሎችና ሌሎችም ንግድ ቤቶች ከባድ ጉዳት መድረሱን፣ጎርፉ በትምሕርት ቤቶች ገብቶ ብዛት ያላቸው የመማሪያ መጽሐፍት ማውደሙን በህይወት ላይ የደረሰ ጉዳት ግን የለም ወደ ከተማዋ የጤና ዘርፍ አካባቢ ደውለን ባገኘነው መረጃ መሰረት ቤቶቹ

እስከ ሃምሳ ሴንቲሜትር በሚደርስ ከፍታ በመጥለቅለቁ በደረሰው ጉዳት በአሁኑ ወቅት ለነዋሪዎች ምግብ የውሃና የኣልባሳት ርዳታ እየተደረገ ነው ለህጻናት የአልሚ ምግብ ርዳት እየተሰጠ መሆኑንም ገልጸውልናል 2:20PM: 0.03658466708791375

2)25-Apr-2016 ድሬደዋ በደረሰ የጎርፍ አደጋ ወደ ከተማዋ የጤና ዘርፍ ኣካባቢ ደውለን ባገኘነው መረጃ መሰረት ቤቶቹ እስከ ሃምሳ ሴንቲሜትር በሚደርስ ከፍታ በመጥለቅለቁ በደረሰው ጉዳት በአሁኑ ወቅት ለነዋሪዎች ምግብ የውሃና የኣልባሳት ርዳታ እየተደረገ ነው ለህጻናት ምግብ ርዳት እየተሰጠ መሆኑንም ገልጸውልናል 2:20PM: 0.023802072563220988

## 5.3. System Summarization for posting texts for each experiment

As to describe to the four experimentations for each post documents by summarizing extracted rate under each experiment, the four experiments are referred as E1, E2, E3 and E4. All of these experiments would be conducted for all the posted Amharic input texts for datasets with the summary extraction percentages of 10%, 20% and 30%. Those summarize extraction percentages are selected to be observed the effect of the summarization processes on different ranges of summarization posted text documents. The total numbers of automatic summaries of each post texts in protests, droughts, sports and floods are 120,862, 43,774, 10,299 and 1,209 respectively posted Amharic texts as training sets on social media, having 30 posts texts automatic summaries in each experiment for as test set for each clustered post text. For this is 30 posts texts are selected randomly in our training data set corpus of the clustered post documents as to a summary.

System summary is the number of sentences extracted for each post texts using the selected percentage are to determine by multiplying the number of sentences in the post texts in a sentence level by the percentage and rounding it's to the nearest zero post texts in a sentence. That is if the original one clustered posts texts has 67 sentences, within a 10% summary that has 7 sentences, within a 62 sentences in a 10% summary that has 6 sentences and so on to find for each sentence in a document. It is depending on the extraction rate post texts and the number of sentences in the document, this Summarizer selects the first *"n text file"* sentences of the documents.

$$n = extraction\ rate \times number\ of\ sentences$$

Those lists of the number of sentences extracted using each percentage rate for each posted text file are presented for each experiment in Table 5.21.

| File name | Post texts | No. of totals sentences | No. sentence in System and Ideal summary under E1, E2, E3 and E4 | | |
|---|---|---|---|---|---|
| Protests posts | | | 10% | 20% | 30% |
| Prot1.txt | 30 | 53 | 5 | 11 | 16 |
| Prot2.txt | 30 | 62 | 6 | 12 | 19 |
| Prot3.txt | 30 | 53 | 5 | 11 | 16 |
| **Average score** | **30** | **56** | **5** | **11** | **17** |
| Droughts posts | | | | | |
| Drot1.txt | 30 | 67 | 7 | 13 | 20 |
| Drot2.txt | 30 | 62 | 6 | 12 | 19 |
| Drot3.txt | 30 | 59 | 6 | 12 | 18 |
| **Average score** | **30** | **63** | **6** | **12** | **19** |
| Sport posts | | | | | |
| Spot1.txt | 30 | 42 | 4 | 8 | 13 |
| Spot2.txt | 30 | 38 | 4 | 8 | 11 |
| Spot3.txt | 30 | 34 | 3 | 7 | 10 |
| **Average score** | **30** | **38** | **4** | **8** | **11** |
| Floods posts | | | | | |
| Flod1.txt | 30 | 56 | 6 | 11 | 17 |
| Flod2.txt | 30 | 43 | 4 | 9 | 13 |
| Flod3.txt | 30 | 30 | 3 | 6 | 9 |
| **Average score** | **30** | **42** | **4** | **9** | **13** |

**Table 5.21.The number of sentences extracted using the selected percentages for each posted texts**

Based on the corresponding value in the above Table 5.21 for a given summary extraction percentage and also sentences are selected to appear in the summary. Before it is saved in a clustered text file separately to be used in objective evaluation/Recall for each post. Then ranked sentences are first arranged and extracted in the ascending order of rank into a ".txt" text file format. As discussed an individual ideal summary is also prepared manually for each posted texts in the above extraction percentages for the experiments in post texts. This is done based on the rank given to the sentences of each post by the evaluators. From this to prepare the manual summary for posting texts under each summary extraction percentages. A total of 30 posts Amharic texts to

take as manual summaries are created for the thirty posted documents for each clustered posts at the three extraction percentages. This manually prepared human summaries is used to automatically evaluate the quality of the automatic summary in all E1, E2, E3 and E4 experiments for posting texts.

The following table 5.22, to shows that the details of the file name, to take the testing set of training set to select randomly the 30 posted texts for each file's group, total numbers of the words contain the posted texts, the total numbers of sentences contain the posted texts, the total numbers of the evaluators in manually summary to rank the sentences in the given posted texts for the summary texts.

| **Files name** | **Posts texts** | **#of words in post texts** | **#of total sentences** | **#of manually ranked sentences** |
|---|---|---|---|---|
| Protests posts | | | | |
| Prot1.txt | 30 | 1,368 | 53 | 16 |
| Prot2.txt | 30 | 1,627 | 62 | 17 |
| Prot3.txt | 30 | 1,407 | 53 | 20 |
| **Average score** | **30** | **1467** | **56** | **18** |
| Drougts post | | | | |
| Drot1.txt | 30 | 1,454 | 67 | 25 |
| Drot2.txt | 30 | 1,275 | 62 | 17 |
| Drot3.txt | 30 | 1,099 | 59 | 13 |
| **Average score** | **30** | **1276** | **62** | **18** |
| Sport posts | | | | |
| Spot1.txt | 30 | 895 | 42 | 18 |
| Spot2.txt | 30 | 963 | 38 | 14 |
| Spot3.txt | 30 | 717 | 34 | 15 |
| **Average score** | **30** | **858** | **38** | **16** |
| Floods posts | | | | |
| Flod1.txt | 30 | 1,114 | 56 | 15 |
| Flod2.txt | 30 | 1,050 | 43 | 17 |
| Flod3.txt | 30 | 805 | 29 | 16 |
| **Average score** | **30** | **990** | **43** | **16** |

***Table 5.22.The testing set corpus posted texts for the experiments***

### 5.3.1. For the Experiments the Porter Stemmer for Amharic post texts

These experiments are worked by using the stemmer directly to count the root or stem words of frequently appears in the posted document for Amharic stemmer, for implementing the dictionary file which contains the keywords in Amharic language vocabulary(lexical) rules and lists of necessary words for the Summarizer as discussed in chapter 4. This experiment implements the term frequency, IDF and tf-IDF sentence ranking methods for posted Amharic texts on social media (table 5.6,5.7,5.8,5.10,5.11,5.12, 5.14,5.15,5.17 and 5.18)

respectively. It used for Porter stemmer which makes use of the Amharic dictionary prepared the lists of the rules for prefix and suffix removed from Amharic words.

From the extraction rate an automatic summary of 10%, 20% and 30% is generated for the summary each of the thirty posted text documents on social media. However, a total 30 posted Amharic texts automatic summaries should be generated at those three percentage rates for each summary post text. Those summaries post text files are saved in a text file separately under each percentage rates. The evaluations of these summaries posted texts and product results obtained are to discuss the following section 5.4 discussion in this chapter.

## 5.4. Evaluations and Discussion of the Results

For each experiment are different posts texts and for each clustered posted documents must have been one summary to be conducted with different methods (summary1, summary2 and summary3), this means we have three clustered documents to summarize each post text over a time period interval. A good summary must be easy to read and give a good overview of the content of the original text documents. Since summaries rise to be more and more oriented towards specific needs, it is necessary to adjust existing evaluation methods accordingly. Most summarization systems perform an extractive approach, selecting and copying important sentences from the higher ranked sentences in document to summarize.

As we had seen in chapter-2 of this thesis the literature reviewed, there are two main types of evaluation method linked to text summarizations. This is Intrinsic and extrinsic evaluation methods and each of them has it's a specific number of approaches which used in grouping in text summarize evaluations for posted Amharic texts. The Intrinsic evaluations are made by comparing for automatic summary within the human/ reference summary made manually prepared the summary posted texts on social media. It is measuring the automatic system summary performance on its own posted texts. We evaluated the process of the system an intrinsic method by using the subjective (qualitative) and objective (quantitative) for posting texts in social media to summaries by automatic and manual summary. The subjective evaluations are used to measure the informativeness, coherences structure and linguistic quality measuring criteria of the automatic summary generate for the posted documents. For the others evaluation method is objective evaluations that are measured the Summarizer of the performance in the extraction summary and also the identification of the post documents of salient sentences with the given post texts. For this to measure the performance of the standard for precision and recall measure in the given input texts, human summary and Summarizer extract percentages how to close the extracts for each other's for the system. Whereas, in the extrinsic evaluations is completed

based on the evaluating how the automatically summaries the systems are good enough to be accomplished the purpose of some other specific tasks. For this purpose for the evaluating the summaries of this research study, for intrinsic evaluations are too applied and implemented within the two ways of evaluation method to use for each experiments posted texts. The first one is Automatic/System summary evaluations which to compare the automatic summary within an ideal summary prepared by the evaluators for post texts and the second one is manual evaluations which are made by selecting the three linguistic experts to summarize the posted texts, who have been prepared the manually summary for posted text for each clustered document before, based on the chosen evaluation criteria. The evaluation to perform using the co-selection measures that are mentioned previously in chapter two are discussed, which assess the quality summary based on the number of sentences that commonly appear in the manual and automatic the system summary for posted texts. The most common of evaluation measures compression ratio (CR), precision, recall and F-measure for news items posted text summarization in the posted text documents over a time (section 2.6.2). The news item text summarization process is majorly evaluated using performance for compression ratio (CR) score, precision, recall and F-score for extraction the tasks. The compression ratio is the ratio of the size of the summarized text document to the total size of the original text documents to summarize automatically. The expression of calculating the CR is given in the equation.

Compression Ratio: CR = (length of the summary) / (length of the original text documents)

**Precision-** it measures the fraction of the system summaries that are correctly chosen to select. It reflects how many of the system's extracted sentences are automatically summarized. It is computed as the ratio of the relevant sentences extracted in the automatic summary (which is the intersection of the sentences that appear in both system summary automatically and summary ideal or reference/human summary) to the total the system summarizes selected sentences.

$$P = \frac{|system\ and\ human\ choice\ overlap|}{|sentences\ chosen\ by\ system|}$$

**Recall** –it gives the fraction of the sentences that the system has chosen from the whole of sentences found in the ideal/human summary. It reflects how many good sentences the automatic summary has not included or how much the system has missed. It is computed as the ratio of the relevant sentences extracted in the automatic summary of the total sentence extracted from the human summary selected sentences.

$$R = \frac{|system\ and\ human\ choice\ overlap|}{|sentences\ chosen\ by\ human|}$$

**F-measure** - is a good tradeoff between the P and R since it gives the same importance for both precision and recall, hence it is important to use it as an evaluation measure and it is the harmonic mean of the recall and precision. It is computed as:

$$F = \frac{2 * P * R}{P + R}$$

The Manual summary is based on the chosen six evaluations, quality measuring criteria to measure the summary for posting texts like informativeness, non-redundancy, coherence, grammatical correctness, overall observation and referential clarity to take as the evaluation for the summary texts. These evaluation measuring criteria used for the coverages, readability, flow, the level of information carried and fluency of the summaries is measured in terms of the linguistic quality and under the part of the linguistic quality is included in the above evaluation quality criteria is grammatical correctness, non-redundancy and referential clarity.

Based on the above evaluation, quality measure the summaries should be checked for post texts and for giving grade values established on non-redundancy, grammatical correctness and referential clarity of the post documents. Whereas, the informativeness of the summary measures for in the level of information carried and coverage in the automatically summaries the post texts. And in the flow of the system summary is measured in words/terms of the coherence and structure of the sentences in the summary texts. The finals, the grade value is given based on the Summarizer expectation and overall observation of the reader from the automatically summary texts in the documents.

The evaluator given grades value in the summary as, to what level of the summary information is found to be suitable in the overall examinations of the summary for the documents.

The evaluator given to a grade level from 1 up to 5 (where given to represent: 1- very poor, 2-poor, 3-fair, 4-good and 5-very good) is too used for each of posted text files to identify the above six evaluation quality measuring criteria to measure the quality of the evaluation in the automatically summaries the texts. The grading system is given to the evaluator from a total score of 30 marks for each automatically summary post text. From the grades value given under each quality measured criteria for each automatically summary posted texts are summed to get each value the total grade values for the summary texts. For this total grade values are converted to the percentage to show the evaluations results in the form of a percentage.

The objective evaluation methods are worked by comparing for the automatic summary generate post texts with an ideal reference summary for the evaluators for the sentences extraction technique of text summarization that could allow the use of compression ratio (CR), recall (R), precision (P) and F-measure (F-m) as an evaluation measure for automatic text summaries for a percentage value for the results. For the similarity of the contents in a sentence is measured for using the extraction units and the similarity of sentences between the automatically the system summary and ideal summaries for a given post text is determined using these three evaluation measures. Based on the numbers of similar sentences between system and human summary for precision, recall and also F-measure is calculated using the formula shown above in the given equation. The objective evaluations are done for all of the experiments on the 30 posted text summaries extracted rate at the three summarization percentage level. Due to the manual evaluations are time dependent and labors demanding to work for the summary, it is done for each experiments that have only done 30 posted texts summaries that would be made 40 summarize sentences, at the three different extraction rates of summary. These are testing set for each post texts in totals of 30 posted text summaries in Amharic texts in protests, droughts, sports and floods group should be evaluated using manually evaluation methods respectively for posted texts on social media. Yet, the objective evaluation is done for all the automatically summaries texts generated using all of the experimentations E1, E2, E3 and E4. For the discussions of the evaluations and also the results we observed as the following sections to present.

### 5.4.1. Manual Evaluation for each Experiments

From in the manual evaluations are prepared for 40 summaries texts to produce from 30 posted texts in news items posted Amharic texts on the social media in the protests, droughts, sports and floods post clusters texts at the three summary extraction percentage rate used in this the experiments. As discussed in the earlier sections, the six manual evaluation, quality measured criteria used to grades the summaries on the scales to give from 1-5. For these grades show that the level of the criteria's under the respect of the summary information is achieved in and for each summary in the documents.

The evaluation guideline the human and the system summary post texts is attached in appendix VI.

An average of the human summary and the system summary selected sentences grade values given by the evaluators (human +system summary) are calculated for each of the post tests to measure and convert into percentage score values to show the results in percent each cluster posts text files from hundred as tabular format. The comparison ratio is made for considering the size of the original poster texts using the protests, droughts, sports and floods groups and for seeing the size of the summary for a given post using in the three extraction percentage rate.

1. **Informativeness for the summary for post texts**

It is to what level of the summary informative and it measures the information on the contents of the automatically summary texts regarding to the topic of the posted documents in the social media. This to help and to understand for each post which of the summary is included the most important information's are presented in the post documents. In the summary should be included the key information related to the topics of the posted texts in Amharic sentences that are written about the topics related details. The evaluators to check the summarize post texts the informativeness for the summary especially, to check whether the summary is included best sentences that should be contained the most significant information is found about the topics for the posted texts and satisfies the information need of the readers or not in the summary.

The grades given in the automatically summaries post texts under this measured criteria the ranges from 40% until to 100%, where 1 got 40%, 2 got 60%, 15 got 80% and 22 got 100%. Since 37 of the summaries got 80 and 100%, for this it would be shown that summarize posted texts are more informative on the topics of the posted Amharic texts. As we had discussed in chapter 2, because the system summarize the most important information to kept and more ranked for the sentences about the topics in posted texts.

We are observing that the summary post texts in the protest post group become the average score rank of 86.67% for the summarizes rate of 10% and 20 % summaries, and 93.33% for the 30% summaries posted texts. From this the average for the protests post summaries for all the three summary and extraction rates, which is 93.33% highly compresses the post document. For the smallest size to summarize is 86.67% at 10% and 20%. This shows that post texts the size of the original post texts increases and the informativeness of the summary content increases and also summarize extracted rate is increase the number of sentences to be included in the summary texts is increasing. This show that a direct relationship between the summary size and information about the content in the summary post texts.

We are shown that the summary post texts in the droughts post clustered texts become the average score rank of 80% for the summarize rate for at 10% summaries, 66.67% for at 20% summarize and 86.67% for at the 30% summaries posted texts.

For the average score of the droughts posts summary is the average score for the three extraction and summary rates, which is 80%, 66.67% and 86.67% for at 10%, 20% and 30% extraction rate respectively. From this smallest size summary in droughts post is taken, it is 66.67% at 20% summary and 86.67% is the highest summary the posted texts at 30% extraction rate. This shows that the number of sentences in a summary post texts for a given post texts increases, the informativeness of summarizes also increases, that is a direct relation with the informativeness of the content texts and the size of the summary.

We are shown that the summary post texts in the sports posts grouped texts become the average values of 80% of the summarizes rate for at 10% and 20 % summaries, and 86.67% for the 30% summaries posted texts. For the average score values of the sports posts summary is the highest values in the three summary and extraction rates which is 86.67%. From the smallest size summary in the sports post texts is taken to 80% at 10% and 20 % summary. This indicated the relation between the size of the summary and informativeness of the summary contents in the document parallel increase each of them.

We are shown that the summary post texts in the floods post grouped texts become the average values of 93.33% of the summarizes rate for at 10% and 30 % summaries and 73.33% for the 20% summaries posted texts. For the average values of the floods posts summary is the highest score in the three summary and extraction rates, which is 93.33% at 10% and 30%. However, for the smallest size summary in the floods post texts is found that 20% is 73.33% summary. The overall ranked post texts which are presented in appendix VI.1 to show the informative for the post texts and the automatically summaries.

**2. Redundancy post texts in the summary of Amharic texts**

One of the linguistic quality measurement is non-redundant post texts in the extracted sentences for automatically summarized in Amharic posts texts in social media on Facebook and Twitter post texts. Thus, the experts to check for any repetition of post Amharic texts in sentence levels. The summary post texts should contain once sentences to express one score of posting text sentence, rather than having more than one sentence to express the same post texts and the same score values to ignore one of them for the summary. Having non-redundancy in the summary post Amharic texts makes the summary post texts more readable and interesting for the readers of the social media posts to get the best information each post item.

Based on the criteria quality measurement for the automatically summaries post texts are given scores: 8 of the summaries got to 60%, 10 of the summary got 80% and 22 of the summaries got 100% for using to evaluate this experiment. Based on this we could be the average score rank for the post texts in the protests posts group at 10%, 20% and 30 summary for it is score 93.33%%. This is constant within the all the three extraction rates. From in droughts posts, we could be obtained the average score in the droughts posts group at 10% of its score 86.67%, at 20% for extraction rate is 93.33% and also at 30% it is the score of 100%.

From in sport posts, we could be obtained in the average score for the summary in the sports posts group at 10% for it is 80%, at 20% for it is score 86.67%and 30 % summary of it scores 93.33%.

From in floods posts, we could be obtained in the average score rank for the floods posts group at 10% of it is score 86.67%, at 20% extraction rate is 80%% and also at 30% it is the score of 93.33%. From the above experiments for each posts texts which is the smallest size to summarize for each post texts the average score

to compare for their groups of posts, in the average score rank for droughts posts, sports posts and floods posts group are 86.67%, 80% and 80%, respectively, for all the three extraction rates, it scores small value to compress in the size of the summary relative to other post texts. These show that there is more redundancy in the group or clustered posts text summaries than others grouped post documents for summary texts. For this is shown that the size of the post texts increases, when the writers write the texts within the same idea post contents in different sentences repeatedly found in the document, this result in more redundancy post to create. Due to this, the extracted summary rate has been repeated sentences since the system uses word frequency and tf-idf to calculate and also to rank the sentences for each clustered post document. For all the summaries, rates except for these, prot1.txt at 10% extraction rate is 80% under and also flod3.txt at 20% and 30% summary the post texts are 80% under all the three extraction rates. For these rank score is given to the same under all the three extraction rates for the post documents. This measure, it is not linked the size of the summary within the extraction rate that is shown to more redundant. This shows that as the size of the posted text increases, summary extracted rate increases and also redundancy is increasing the post texts. The overall results for posting texts that show for the automatically summaries avoid redundancy very well for droughts posted texts that are 100% to keep for the extraction. The grade values are given for the summaries based on their criteria to show in appendix VI.2 to show the details.

**3. Coherence and structure in Amharic post texts**

This to consider in answering how well the summary is organized and also structured to concern the post texts. For this the measure for the criteria the connectedness of the post texts in the sentences in the summary and also the extraction structure of the summary post texts. For the extracted summaries should not be a heap of concatenated sentences in the document, but rather should build up the topic of the post texts in the successive sentences to a well-structured coherent summary in the posted text on social media. The system summaries are given grade values under this measuring criteria as, 10 of the summaries got to 60%, 14 of the summaries got to 80% and 16 of the summaries got to 100%. This shows that one fourth of the summary texts are less organized and structured in the post document. We observed that the average score of the grade in the protest post summaries differs from at 10% and 20% extraction rates toward to 80% and 30% extraction rate for the summary post texts for it is score 86.67%. For this (80% to 86.67%) to show that the size of the summary increase in a high extraction rate, the coherence and the structuredness increase at this condition.

In the droughts post texts cluster summaries it is 86.67% in all extraction rates are the same at 10%, 20% and 30% extraction rate for the summarizes post for each group. For this to show that the size of the summary is

directly related within the coherence and structuredness of the summary to increase proportionally in the post texts. From sport posts in the social media in the sport post group summaries it is scored in both rates to score the same values for 80% at 10 % and 20% extraction rates towards to increase into 86.67% in the 30% extraction rate for the summary. For this the average of the group of the summaries (80% to 86.67%) is shown that the coherence and structuredness increases, the same also the size of the summary increases, the resulting a more interesting, readable and attractive summary for the readers.

From post texts in the floods post texts in cluster summaries it is 66.67% at 10%, 80% at 20% and 93.33% at 30% extraction rates for summary posts for each group. For this show that the size of the summary is directly increase for the extraction rate also increase, so the relation within the coherence and structuredness of the summary to increase proportionally. Those are shown that the coherence and structuredness of a summary texts increases as the size of the summary post document increases, then it results in large size of the summaries to be created. The given grade values based on the given criteria, see under appendix VI.3 are presented in the table format.

**4. Overall Observation for the summary post texts**

This one of the linguistic quality measurement methods for the summary text document directly to observed the posts texts. If the readers could overall reading of the post texts in social media, based on this the reader to decide how you extracted rate summary for the post texts to find the summary. After reading the posts texts in the groups and the corresponding to extract the summary, the experts have been some subjective finding regard to the automatic summary post texts in the document. Based on this expectation, each system summary of groups post text is given to a grade value for the group of the text files. The grading given values for the evaluators or summarizers to measure the quality of the summary in this measuring criteria the range from 20% to 100% as; 1 summary got to 20%, 4 summaries got to 40%, 10 summaries got to 60%, 12 summaries got to 80% and 13 summaries got to100%.

For these the average score, grade values for the clustered summary post texts in the protest post group are 60% for at 10%, 73.33% at 20% and 86.67% of the extraction rates of 30% summary.

In the droughts post the group of summaries the average score values at the three extraction rates of 10%, 20% and 30% are 66.67%, 80% and 80% respectively.

For these the average score, grade values for the clustered summary post texts in the sport post clusters are 53.33% for at 10%, 80% at 20% summary and 30% of it is score 86.67% summary.

As the floods post the group of summaries the average score, grade values at the three extraction rates for 80%, for 10% summary, 86.67% for at 20% summary and 30% summary it is score 93.33%.

For these shows the size of the extracted summary increases, then the summaries for each group are more interesting for the users on posting texts. It also shows that the small size rate (10%) summaries for protest post, droughts post and sports group summaries are not good enough to meet the users or reader's expectation. But in the floods post for the summary for at 10% is good for the users /readers for compressing the summarize the post texts. We are observing that the average score of the grades for the summary groups of the post texts is shown that the different values in each posts at the three different extraction percentages or rates, then increasing in parallel by way of the extraction percentage increases. From this the score values are shown that the summary extracted rate using higher extraction percentages, that have more sentences in the summaries increase the achievement of the users' satisfaction and expectation for each the size of the post texts in the social media.

The given grade score values to give under this quality measure criteria by the evaluators are given in the table on the appendix VI.4 to show the details.

**5. Grammatical correctness**

This is one of the linguistic qualities to measure used to evaluate the automatic summary for posting texts. The evaluator to check any grammatical error for post texts with a sentence level, which is happening during the summarization processes to the fragment of sentences in post documents. We were observed in post documents in the grammar, in some of the summary post texts were concatenated to the sentences in the document followed by sentences from another group of post texts. From the experimentation to use the sentences extraction techniques, most of the results are exactly in terms of grammatical correctness to obtain the words in the documents. The summary for post texts are combined to directly extract sentences in their rank sentences, orders, thus by this measure of the criterion a good results are gotten in the automatically summaries post texts.

From these the 40 automatically summaries in the post texts in the documents, 3 summary got to 60%, 16summary got a score of 80% and the rest of the 21 summaries got to 100%.

We observed that the grammatical correctness is not related to the size of the source post texts and the size of the summary in post documents, therefor, each summary of the posted documents, it got to the same grade score values in all extractions percentages or rates in the texts files for each group of posts. For this measuring criteria's for the linguistic quality is dependent on their writing styles of the sources of post texts in the social media. From these we could be seen that the summary post texts in the sports post groups are better score

values (which is 93.33%) than those in the protests, droughts and floods post group in social media. For this is shown that the structure of the summaries of the original post texts in the sports post groups has less subtitles which are created the fragment sentences in the automatically summary texts than post texts in the others posts groups in the posted documents. The given grade values to give in the percentages are presented in the table showing appendix VI.5 to show the details.

**6. Referential clarity**

This measuring techniques for the last linguistic quality measuring criteria that are referential clarity measures which is to identify easier to read what or whom of the referential words or pronouns used to refer and also if the references are correct in consecutive sentences in post documents. This is additional references are used as the size of the post texts increases. For this the evaluator to check by using questions like "what? Or who?" could be easily too answered in the summary texts for the referential words in the posted documents is used. From the 40 summaries evaluated for posted documents using for measuring the criterion quality of summary, 6 summary got to 60%, 14 got 80% and 10 summary got to 100%.

We are observing that this quality of measuring the criteria is kept more in the posted text summaries for protests post groups that have the average score values at 10% and 30% it is score 86.67%, thus the three summary rate the better results are occurring in the group, then at 20% extraction rate is scoring 73.33%. For this post groups to score with 100% to one third (3 out of 9) of the protest post for the summaries got to 100% to record or score. For this average score of the referential clarity is the smallest size of summaries, posts at 20% extraction rate, we compared to the protest groups of post texts, which is scoring 73.33% than the other group of posts at extraction rates. From the post group of the drought posts have been the summary at 10% to 66.67%, at 20% it is score 80%, at 30% of it is score 93.33%, this the better result is occurring in the group. For this post groups to score with 100%, to score (2 out of 9) of the droughts post for the summaries got to 100% to score. For this average score value of the referential clarity is the smallest size of the summaries, posts in droughts groups at 10% extraction rate, we compared to the droughts groups of posts, which is scoring 66.67% less than the other group of post extraction rates.

In the sports post its average score for at 10% and 30% of it is score 93.33%, this is better results are occurring in the group and at 20% extraction rate for it score 86.67% summary. For this post groups to score with 100%, more than half (5 out of 9) of the sports post for the summaries got to 100% to score. For this average score of the referential clarity is the smallest size summaries, posts at 20% extraction rate, we compare to compression rate in the sport groups of post texts, which is scoring 86.67% less than the other group of posts at extraction rates for the summary.

Since in the floods post its average score for at 20% its score 80% summary, at 10% and 30% extraction rate for it is score 86.67%, this the better results occur in the groups. For this post groups to score with 100%, (2 out of 9) of the floods post for the summaries got to 100% to score. For this average score of the referential clarity is the smallest size for summaries the posts at 20% extraction rate, we compared to the floods groups of posts, which is scoring 80% than the other group of floods posts at extraction rates. Generally between all the three summary extraction rate of increasing the size of the summary within a given post text, we could be found more sentences to be included in the summarization post results and also the size of referential clarity is increases that are kept for more than in the smaller size of the summarize posts texts.

The given grade values of the average score based on this evaluation, quality measurement criteria are shown in the appendix VI.6.

The evaluators/summarizers given grades score values under each measuring quality criteria for the automatically summaries the post texts are summed to the totals grade score values in the given post texts. For this totals grade score is divided by 30, which is the total of the ranked the given grade under each measuring quality criteria and also to multiply in hundred for changes it's to percentages in the post texts. The given grade score value, the total grade rank and the average rank calculated for each summary is presented in appendix V.

The protests posts texts in the average score of those percentage values for the protests post group summaries are 78.89%, 81.11% and 83.33% for at 10%, 20% and 30% respectively for three extraction rates. This the extraction rate increase and also the size of the summary are increasing.

The droughts posts texts in the average score of those percentage values for the droughts post group summaries are 78.78%, 83.33% and 90% respectively for three the extraction rates.

The sports posts texts in the average score of those percentage values for the sports post group summaries are 82.22%, 84.44% and 88.89% for at 10%, 20% and 30% respectively for three extraction rates.

The floods posts texts in the average score of those percentage values for the protests post group summaries are 78.89%, 80% and 86.67% for at 10%, 20% and 30% respectively for three extraction rates.

Those of the average scores, grade evaluations of the automatically/system summary within using the manually evaluations are shown the quality of the system summaries increase as the size of the extracted summary increases within regard to the size of the sources of post text files.

Based on the manual evaluation, the more sentences are included in the extracted summary, the better the quality is for a given posts of any size text files. The following Table 5.23, shows the total evaluations, grade

given to each of the evaluated post summaries based on the six manual evaluation quality measuring criteria to be chosen for these experimentations.

The score values are presented for each of the 40 summaries selected for manual evaluations which are extracted from the 30 posts texts for each clustered posted texts.

| **Files name** | **No. Total sentences** | **Totals of the % average for each extraction rate** | | |
|---|---|---|---|---|
| | | **10% Summary** | **20% Summary** | **30% Summary** |
| Protests posts | | | | |
| Prot1.txt | 53 | 70% | 90% | 86.67% |
| Prot2.txt | 62 | 76.67% | 76.67% | 80% |
| Prot3.txt | 53 | 90% | 76.67 | 83.33% |
| **Average score** | **56** | **78.89%** | **81.11%** | **83.33%** |
| Droughts posts | | | | |
| Drot1.txt | 67 | 83.33% | 80% | 93.33% |
| Drot2.txt | 62 | 76.67% | 80% | 80% |
| Drot3.txt | 59 | 73.33% | 90% | 96.67% |
| **Average score** | **63** | **78.78%** | **83.33%** | **90%** |
| Sport posts | | | | |
| Spot1.txt | 42 | 83.33% | 86.67% | 83.33% |
| Spot2.txt | 38 | 80% | 76.67% | 96.67% |
| Spot3.txt | 34 | 83.33% | 90% | 93.33% |
| **Average score** | **38** | **82.22%** | **84.44%** | **88.89%** |
| Floods posts | | | | |
| Flod1.txt | 56 | 76.67% | 83.33% | 83.33% |
| Flod2.txt | 43 | 86.67% | 76.67% | 93.33% |
| Flod3.txt | 29 | 73.33% | 80% | 83.33% |
| **Average score** | **42** | **78.89%** | **80%** | **86.67%** |

***Table 5.23. The Subjective evaluation of the average score for each given post selected summary***

### 5.4.2. The Objective Evaluations post texts

#### 5.4.2.1. The Objective Evaluations for E1 protests texts

We made to test the protests 30 posted texts for the system automatically summaries to generate under the experiments are evaluated using the development of tools for this purpose in the Java programming language (NetBeans). For this it is completed by using for checking the presence of overlap similar sentences between human/ideal summary and automatically summarized that found in both prepared for each post texts.

Hence, in the 30 posted texts to find the 40 automatically summaries are generated within a ranked sentences are compared within the corresponding prepared the summary of evaluators 30 posted texts to take 40 human summaries texts.

Since we have seen in above table 5.23, for the F-measures are the weighted averages score values of the recall and precision are presented for each posted texts, it is used for as evaluation measure and to display the summary for the 40 automatic summaries under each extraction rate in percentage.

For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented in the protest post texts in the protest group, 74%, 85.35% and also 87.07% summarize the protest post texts in social media. For this experiment we observed that the increase rate of the performance of each summary of the posted texts of extraction rate changed to at 10% in 20% and also at 20% in 30% to increase in the extraction rate and the size of the summary for this the protest post group for it is 11.35% and 1.72% respectively.

For this is indicated that the system automatically summary increases for the extraction rate increases with the performance rate for the summary, this show that the direct relation between the performance of the systems and also the size of summary post texts. For the groups the highest summary of the three extraction rate is 87.07% to compress the summary, the extraction rate for the 30% extraction rate and the next highest average score performance is 85.35% to summarize for at 20% extraction rate. The change of protests post texts for each clustered post in the extraction rate is compared for each of the performances for the 11.35% for at extraction rate changes from the 10% to 20%) in the protest post texts group.

From this evaluation results, we could be seeing the summary that is becoming to better results in the extraction rate that is determined for more number of sentences extracted increases in the posted documents. This shows that the average score performance increased as well as the extraction rate increased in the posted texts, thus a summary of highest extraction rates get more similar to the original posts. That is the F-measure increased at an increasing rate as the extraction percentage increases constantly. Table 5.24 shows the

objective summary of the evaluation results of the 40 summaries extracted from the 30 posted texts using experiment one. See the details results for summary is presented in appendix VII. A.

| Clustered protests post files | Experiment 1 Summary files | **F-measure in (%)** |
|---|---|---|
| At 10% | | |
| Prot1.txt | E1Sm10Prot1.txt | 60% |
| Prot2.txt | E1Sm10Prot2.txt | 83.33% |
| Prot3.txt | E1Sm10Prot3.txt | 80% |
| Average score | | **74**% |
| At 20% | | |
| Prot1.txt | E1Sm20Prot1.txt | 81.81% |
| Prot2.txt | E1Sm20Prot2.txt | 83.33% |
| Prot3.txt | E1Sm20Prot3.txt | 90.90% |
| Average score | | **85.35**% |
| At30% | | |
| Prot1.txt | E1Sm30Prot1.txt | 87.5% |
| Prot2.txt | E1Sm30Prot2.txt | 73.68% |
| Prot3.txt | E1Sm30Prot3.txt | 100% |
| **Average score** | | **87.07**% |

**Table 5.24. The Summary of objective evaluation results for experiment 1**

### 5.4.2.2. The Objective Evaluations for E2 droughts texts

We made to test the droughts posts for the 30 posted texts in the system automatically summaries to generate under the experiments are evaluated using the development of tools for this purpose in the Java programming language (NetBeans). For this it is completed by using for checking the presence of overlap similar sentences between human/ideal summary and automatically summarized that found in both prepared for each post texts. Hence, in the 30 posted texts to find the 40 automatically summaries are generated within a ranked sentences are compared within the corresponding prepared the summary of evaluators 30 posted texts to take 40 human summaries posted texts.

Since we have seen in above table 5.23, for the F-measures are the weighted averages score values of the recall and precision are presented for each posted texts, it is used for as evaluation measured and summary of the evaluation values for the 40 automatic summaries under each extraction rate in percentage.

For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented in the droughts post texts in the droughts group, 73.81%, 83.54% and also 84% summarize the droughts post texts in social media.

For this experiment we observed that the increase rate of the performance of each summary of the posted texts of extraction rate changed to at 10% to 20% and also at 20% in 30% to increase in the extraction rate and the size of the summary for this the droughts post group for it is 9.73% and 0.46% respectively.

For this is indicated that the system automatically summary increases for the extraction rate increases with the performance rate for the summary, this show that the direct relation between the performance of the systems and also the size of summary post texts. See the details results for summary is presented in appendix VII.B

| Clustered droughts post files | Experiment 2 Summary files | **F-measure in (%)** |
|---|---|---|
| At 10% | | |
| Drot1.txt | E2Sm10Drot1.txt | 71.42% |
| Drot2.txt | E2Sm10Drot2.txt | 66.67% |
| Drot3.txt | E2Sm10Drot3.txt | 83.33% |
| **Average score** | | **73.81**% |
| At 20% | | |
| Drot1.txt | E2Sm20Drot1.txt | 92.30% |
| Drot2.txt | E2Sm20Drot2.txt | 83.33% |
| Drot3.txt | E2Sm20Drot3.txt | 75% |
| **Average score** | | **83.54**% |
| At30% | | |
| Drot1.txt | E2Sm30Drot1.txt | 75% |
| Drot2.txt | E2Sm30Drot2.txt | 89.47% |
| Drot3.txt | E2Sm30Drot3.txt | 88.89% |
| **Average score** | | **84**% |

***T*able 5.25.The Summary of objective evaluation results for experiment 2**

### 5.4.2.3. The Objective Evaluations for E3 sports texts

We made to test the sports posts for the 30 posted texts in the system automatically summaries to generate under the experiments are evaluated using the development of tools for this purpose in the Java programming language. For this it is completed by using for checking the presence of overlap similar sentences between human/ideal summary and automatically summarized that found in both prepared for each post texts.

Hence, in the 30 posted texts to find the 40 automatically summaries are generated within a ranked sentences are compared within the corresponding prepared the summary of evaluators 30 posted texts to take 40 human summaries posted texts. Since we have seen in above table 5.23, for the F-measures are the weighted averages score values of the recall and precision are presented for each posted texts, it is used to measure and a summary of the evaluation values for the 40 automatic summaries under each extraction rate in percentage.

For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented in the sports post texts in the sports group, 81%, 89.25% and also 91.37% summarize the sports post texts in social media. For this experiment we observed that the increase rate of the performance of each summary of the posted texts of extraction rate changed to at 10% to 20% and also at 20% to 30% to increase in the extraction rate and the size of the summary for this the sports post group for it is 8.25% and 2.12% respectively. For this is indicated that the system automatically summary increases for the extraction rate increases with the performance rate for the summary, this show that the direct relation between the performance of the systems and also the size of summary post texts. See the details results for summary is presented in appendix VII.C

| Clustered Sports post files | Experiment 3 Summary files | **F-measure in (%)** |
|---|---|---|
| At 10% | | |
| Spot1.txt | E3Sm10Spot1.txt | 75% |
| Spot2.txt | E3Sm10Spot2.txt | 100% |
| Spot3.txt | E3Sm10Spot3.txt | 66.67% |
| **Average score** | | **81**% |
| At 20% | | |
| Spot1.txt | E3Sm20Spot1.txt | 87.5% |
| Spot2.txt | E3Sm20SPot2.txt | 75% |
| Spot3.txt | E3Sm20Spot3.txt | 100% |
| **Average score** | | **89.25**% |
| At30% | | |

| | | |
|---|---|---|
| Spot1.txt | E3Sm30Spot1.txt | 92.30% |
| Spot2.txt | E3Sm30SPot2.txt | 81.81% |
| Spot3.txt | E3Sm30SPot3.txt | 100% |
| **Average score** | | **91.37**% |

***Table 5.26.The Summary of objective evaluation results for experiment 3***

### 5.4.2.4. The Objective Evaluations for E4 floods texts

We made to test floods posts for the 30 posted texts in the system automatically summaries to generate under the experiments are evaluated using the development of tools for this purpose in the Java programming language. For this it is completed by using for checking the presence of overlap similar sentences between human/ideal summary and automatically summarized that found in both prepared for each post texts.

Hence, in the 30 posted texts to find the 40 automatically summaries are generated within a ranked sentences are compared within the corresponding prepared the summary of evaluators 30 posted texts to take 40 human summaries posted texts.

Since we have seen in above table 5.23, for the F-measures are the weighted averages score values of the recall and precision are presented for each posted texts, it is used for as evaluation measure and displays the summary of the evaluation values for the 40 automatic summaries under each extraction rate in percentage. For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented in the floods post texts in the floods group, 86.11%, 89.56% and also 93.52% summarize the floods post texts in social media. For this experiment we observed that the increase rate of the performance of each summary of the posted texts of extraction rate changed to at 10% to 20% and also at 20% to 30% to increase in the extraction rate and the size of the summary for this the floods post group for it is 8.45% and 3.96% respectively. For this is indicated that the system automatically summary increases for the extraction rate increases with the performance rate for the summary, this show that the direct relation between the performance of the systems and also the size of summary post texts. See the details results for summary is presented in appendix VII.D.

| Clustered floods post files | Experiment 4 Summary files | **F-measure in (%)** |
|---|---|---|
| At 10% | | |
| Flod1.txt | E4Sm10 Flod1.txt | 83.33% |
| Flod2.txt | E4Sm10 Flod2.txt | 75% |
| Flod3.txt | E4Sm10 Flod3.txt | 100% |

| | | |
|---|---|---|
| **Average score** | | **86.11**% |
| At 20% | | |
| Flod1.txt | E4Sm20 Flod1.txt | 90.90% |
| Flod2.txt | E4Sm20 Flod2.txt | 77.78% |
| Flod3.txt | E4Sm20 Flod3.txt | 100% |
| **Average score** | | **89.56**% |
| At30% | | |
| Flod1.txt | E4Sm30 Flod1.txt | 88.24% |
| Flod2.txt | E4Sm30 Flod2.txt | 92.31% |
| Flod3.txt | E4Sm30 Flod3.txt | 100% |
| **Average score** | | **93.52**% |

***Table 5.27.The Summary of objective evaluation results for experiment 4***

### 5.4.3. Comparison of each experiment

From the four experiments are implemented in the posted texts on social media for Twitter and Facebook the users posted Amharic texts. We were tested for each post to take 30 posted texts for each text file and 40 summaries to select in ranked sentences for each experiment both the automatic summary and human summary. The main difference for each experiment for the use of the porter stemmer in the different post texts to remove the affixes by using the Java tools. For this each summary to evaluated by the human/ideal summary in all experiments to generate the summaries the post texts in the social media. During the experiments in the four experimentations that observed the difference between their performance on the effectiveness and efficiency to compared to extract the summary and extraction percentages. For the systems evaluated in 1st, 2nd, 3rd and 4th experiments are effective for all groups of post texts on the extraction percentages and the size of the summary the average score for each experiment are greater than 70% to score the performances. In all the experiments the evaluation results that could be shown the automatic summaries for using the Amharic stemmers to perform the extraction rate of the size of the posted texts with the better consistency in higher rates. In the first experiment (E1) the average results of the performances as well as in the table 5.24, the extraction rates at 10% to compare the other's experiments, it performs better results than E2 in the groups to show that as 0.19%, 7%, 12.11% for protests post group and the droughts group 7.19%, 12.3% and also in the sports group 5.11% posted texts to change. We compared for the first experiments to the second experiments are more condensed or compress for the summary, but in the third and the fourth experiments are less

compressing the posted documents at 10% extraction rate. For the others at 20% extraction rates for the size of the summary in the first experiments (E1) is better results than E2 in the group of post texts, such as 1.81%, 3.9%, 4.21% and in the droughts 5.71%, 6.02% and for in sports post groups is 0.31%. We compare the first experiments for the others experiments in the extraction rate at 20%, it is greater than E2 for compress the summary, but it is less than others 20% extraction rate for the post texts. The size of the summary increase, the extraction rate is increasing in the summary. In all the experiments at 20% summary the highest performance to compress the summary in the E4, E3, E1, E2 and 0.31%, 5.71%, 1.81% and 3.9% respectively. At 30% extraction rate for the summary to compare for each experiment (E1) is the best result outperformed than E2 in the groups of posts for 3.07%, 4.3%, 6.45% and in the droughts post 7.33%, 9.52% and in the sports 2.15%. In the experiments at 30% summary the highest performance to compress the summary in the E4, E3, E1, E2 2.15%, 7.37%, 3.07% and 4.3% respectively. In the average score for experiment E4 is more condensed the posted documents for the summary than the others experiments, which is 93.52% at 30% extraction rate.

| Group File | Average of F-measure in percentage (%) for all E1,E2,E3,E4 | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| At10% | E1 | E2 | Differenc | E1 | E3 | Diff | E1 | E4 | difference | E2 | E3 | Differen | E2 | E4 | differenc | E3 | E4 | differenc |
| Protests(E1) | 74% | 73.81 | -0.19 | 74 | 81 | 7 | 74 | 86.11 | 12.11 | | | | | | | | | |
| Droughts(E2) | | | | | | | | | | 73.81 | 81 | 7.19 | 73.81 | 86.11 | 12.3 | | | |
| Sports (E3) | | | | | | | | | | | | | | | | 81 | 86.11 | 5.11 |
| Floods (E4) | | | | | | | | | | | | | | | | | 86.11 | |
| At20% | | | | | | | | | | | | | | | | | | |
| Protests(E1) | 85.35 | 83.54 | -1.81 | 85.35 | 89.25 | 3.9 | 85.35 | 89.56 | 4.21 | | | | | | | | | |
| Droughts(E2) | | | | | | | | | | 83.54 | 89.25 | 5.71 | 98.25 | 89.56 | 6.02 | | | |
| Sports (E3) | | | | | | | | | | | | | | | | 89.25 | 89.56 | 0.31 |
| Floods (E4) | | | | | | | | | | | | | | | | | 89.56 | |
| At30% | | | | | | | | | | | | | | | | | | |
| Protests(E1) | 87.07 | 84 | -3.07 | 87.07 | 91.37 | 4.3 | 87.07 | 93.52 | 6.45 | | | | | | | | | |
| Droughts(E2) | | | | | | | | | | 84 | 91.37 | 7.37 | 84 | 93.52 | 9.52 | | | |
| Sports (E3) | | | | | | | | | | | | | | | | 91.37 | 93.52 | 2.15 |
| Floods (E4) | | | | | | | | | | | | | | | | | 93.52 | |

**Table 5.28.The average performance comparison results for all experiments**

### 5.4.4. Discussions results

In this paper, we reported the evaluation of the results our research news items posted Amharic text summarization on social media for posting text on Twitter and Facebook and how to process, the steps minimize the error and to increase the products of the results to summarize Amharic text our research algorithm is hybrid tf-IDF. We described the experiment we showed that the results of sentence similarity (table 4). First, the sentence similarity is computed as a linear combination of sentence similarity and word order similarity. Sentence similarity is computed as the Cosine similarity between the sentences vectors for the two sentences and to build the sentences vectors, the union of words in the two sentences is treated as the vocabulary (characters). If the two compare sentences [55]are similar all words occur both in the sentence, its given value for the results is 1. If it doesn't, the similarity for all the words it returns the results is 0. Similarity posting documents to calculate the similarity and cluster based on similarity results and also used for clustering technique is presented in this section (2.9.1). After a similarity calculated for each post in a pair of sentences, the document clustering is performed using K-means clustering algorithm on news items posted Amharic text summarization. For each posted document, it finds closest to group and its similar post sentences on cluster groups of documents and produces the single output key sentences as document without duplicate tweets.

According to similar measured, the clustering technique that used by Kmeans algorithm first the data input files must be convert the text files (. txt) by import the data into excel, both the two sentences were implemented and the results of a similarity pair of sentences to obtain the best order for the preprocess. K-Mean Clustering uses the Java tool and to cluster documents with similarity, after doing preprocessing tasks we have to form a file which is well-matched with the similarity text documents using separate by delimiter comma (,) to distinguish one sentence to another's sentences on excel to arrange by column and save as by comma separated format (. CSV) format files. So, we import the CSV file through Java tool to form clusters for those pairs of sentences values and also to give k number, that number to decide the cluster to draw the graph, the group of similar items on Java interface for k clusters are to randomly to groups within the nearest value of the centroid use of K-means algorithm and the three field's column in excel (Similar id, sentence1, sentence2) from this application i.e. preprocess and cluster fields are used. See below figure 5, an excel column A, B, C respectively.

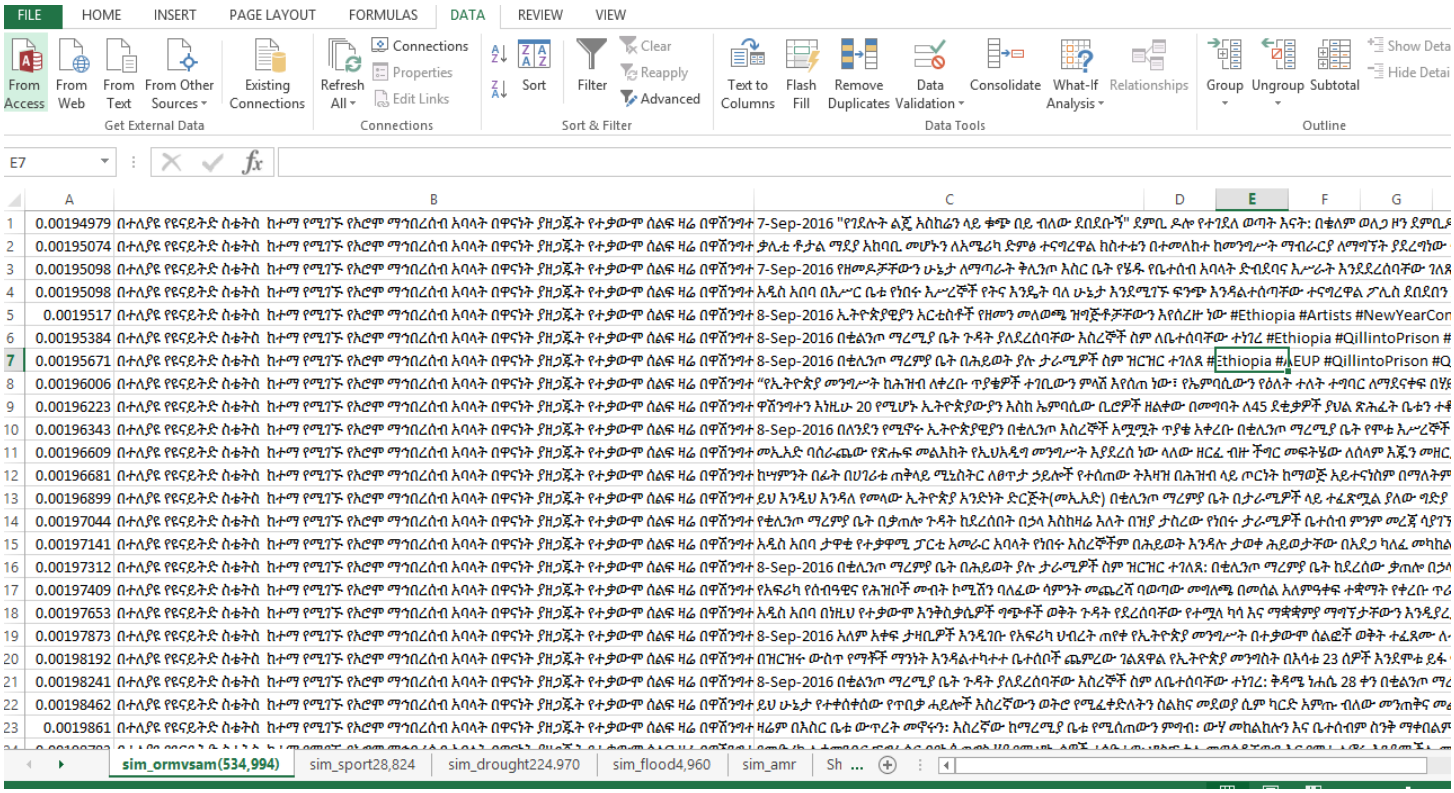

*Figure 5.Preprocess clustering sentence similarity*

In the above figure 5, we identified the similarity of the sentences to check similarity one sentence to others sentences after that, each sentences find to the similarity calculate for all the sentences in the give text documents that we should be saved as CSV format to ready for clustering documents.

From the Hybrid TF-IDF process is its ability to assign meaningful term frequency and inverse-document frequency values to a very short document type while also being able to carefully control the overall weight and length of the target posted Amharic texts (as shown the table 5.7 and others table for tf-idf).If we computing the term frequencies of the words for each document, we assume a single or multi document is containing the entire collection of Tweets. This way, we have differentiated term frequencies of words, but also do not lose the IDF component. For this purpose, a term is a single word in a posted documents that can be directed to display in the results. In defining the term frequencies in terms of the entire set of words within all of the candidates Twitter and Face book posted documents, we have a much more representative set in order to judge a word's natural frequency compared to using a single document. Using only a single Twitter and Facebook document would have resulted in all of the words having about the same small term frequency since it is unlikely a word would ever occur more than once within a single document or two. Conversely, by

choosing to use individual documents for defining the inverse document frequency, we are able to measure and weight terms based on how often those terms occur across documents. Words that occur too frequently across the majority of documents are most likely either the candidate words, phrase and sentences (which by definition is in every document) or unknown stop words. In either of these cases, these types of words do not provide much discriminatory power. Alternatively, words that occur with some higher frequency, but not in every document provide much greater biased influence. Therefore, by defining document in a hybrid way, we have maximized the amount of information available for both the term frequency (table 5.5) and inverse document frequency components within the established TF-IDF equation.

Finally, the combination of meaningful term frequencies, inverse document frequencies, and deliberate control of the overall goal summary length give the Hybrid TF-IDF algorithm to differentiate the most relevant documents of a desired length within the collection of available documents for a given document. The sentence is again to create a summarization system that produces summaries with as much content as possible that satisfies the user (table 5.24, 5.25, 5.26 and 5.27), in is given a set post text. Since the summary document produced by TF-IDF summarization process alone are extractive based on clustering documents, we view sentence simplification (also known as sentence shortening or sentence compression) as a means of creating more space within which to capture important content. Extracting and summarization, we would be using the first extracted set, represented news items posted Amharic text summarization that consists of the top score sentences from a high precision retrieval for each user post and also this corpus is still high volume posts and contains much irrelevant content found on social media post data. If users have allowed post extracting and summarization for Amharic tweets without redundant posting to take one or single posting user posts that could be used for generating summaries in order to generate Amharic tweets summary the algorithm looks for the most overlapping Amharic tweets the news items posted documents user posts on the social media within the input documents. In many cases where an algorithm was evaluated, the input data corpus is either manually prepared, semi-automatic or automatically produced and then manually post-edited, in order to have a perfect input to the algorithm. Three major requirements for single and multi-document summarization are clustering, coverage and anti-redundancy. Clustering is the ability to cluster similar posted documents and passages to find related information, coverage is the ability to find and extract the main points for posting documents to the summary and anti-redundancy is the ability to minimize redundancy user posts document between passages in the summary. Coverage and anti-redundancy is achieved with the help of sentence filtering while generating the final summary[56].

After, preprocessing the summary has gone its post formatting, since these features are removed to increase the amount of overlap post documents and also the set of input documents to find a matching document (similar posted documents) that take to one tweet documents to condense add to together with word similarity documents to cluster and finally input cluster documents process to summary to decrease or compress the single post to display for the results. Since the algorithm only generate summaries from common input documents for Amharic posts to be summarized[28]. A summary can be wonderfully written, but its content can be completely misleading and also in view of this, whenever a summary is assessed, in addition to evaluating its quality, its informativeness should also be judged. To evaluate the informativeness, humans are required to read both the source and the automatic summary, and compare the input corpus information contained in them making the process in the original input data set and the output data set to produce the results. Summary extraction is after ranking the sentences based on their total score each sentence to the higher value sentence to come first and to pick the sentence into summary is produced selecting a certain number of top ranked sentences where the number of sentences required is provided by the user. For the reader's suitability, the selected sentences in the summary are rearranged according to their original positions in the document. We have discussed the previous chapters the basic practice behind tf-IDF, the general process we would follow to summarize a texts of steps needed preprocess the document, import a corpus used for training, create a count vector for the term, build a tf-IDF score matrix, score each sentence, summarize using top ranking sentences. We could be using this research the following steps, we can extract important sentences from a set of documents the process. Sentence identification transfers the documents into sentences, similarity calculation is calculate the similarity between sentences, tokenization is split each sentence into a set of stem words, build similar sentences into graph of clusters is build a graph of the sentences for the similarity posted sentence in the matrix (similar id, sentence1, sentence2) and after cluster to summarize each clustered post documents individually to need one summary as one cluster document.

As evaluation of the experiments for each training data sets to compress the text summarization by automatically if the user selected the size of the summary of the document and to condense the document. We tested the best compression rate for each training data for protests posts higher or better compress ratio sequentially to condense the objective summary (table 5.24). For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented for the protests post texts in the protest group 74%, 85.35% and also 87.07% summarize the protests post texts in social media.

The droughts training data sets, we tested the best summarize document sequentially to condense the objective summary (table 5.25). For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented for the droughts post texts in the droughts group 73.81%, 83.54% and also 84% summarize the droughts post texts in social media.

Sports training data sets, we tested the summarize cluster document sequentially to condense the objective summary (table 5.26) for text summarization in this post texts. For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented for the sports post texts in the sports group 81%, 89.25% and also 91.37% summarize the sports post texts in social media.

In Floods training data sets, we tested the summarize document sequentially to condense the objective summary (table 5.27). For the average score for the performance measuring the three extraction percentages at 10%, 20% and 30% are presented for the floods post texts in the floods group 86.11%, 89.56% and also 93.52% summarize the floods post texts in social media.

# CHAPTER SIX

## 6. Conclusion and Recommendation for the research study

### 6.1. Conclusion

In the information overload of social media are different news items posted in Amharic texts. For this is a single or multi document to need summarization that contain in the frequent terms or repeated posting documents to summarize using hybrid TF-IDF algorithm is introduced. As discussed in the previous chapters, in this research, hybrid TF-IDF algorithm was recommended for Amharic and for under resourced languages in social media posted text documents. The TF-IDF is used to process for counting words in a document as well as throughout a corpus of documents to the end of sorting documents in statistically relevant ways. To this end, we have forwarded the conclusion and recommendation of the research study. The growing of the web for social media a huge amount of information is posted and the posted documents coming from different sources at this time, it becomes very difficult for summarizing the documents to find specific information they want to summarize. It provides support for those users to get that relevant information in posted Amharic texts and summary is today the mandatory to get relevant important information for the documents for social media. As the, text summarization is one of the natural language processing (NLP) application that propose to extract the most important information from a posted document to produce a condensed version for a particular user selected the tasks to summarize [30]. The goal of text summarization is to automatically condense unstructured text documents or recently posted texts into a summary containing the most important information over a time to summarize post texts. Instead of a human having to read entire documents, we could use automatically to summarize the most important sentences into the summary more manageable to the reader. The contribution of NLP in achieving such goal of text summarization posted texts have been pointed out and the news items posted text summarization on the social media for Twitter and Facebook user posted texts are important towards text summarization to understand for summary and an essential for many current NLP applications tasks such as, text summarization, information retrieval, information extraction and question answering focus on the need for systems that are news items posted Amharic texts aware. News items posted Amharic text summarization can be described as the problem of automatically summarizing information over time and the easiest way to extract Amharic tweets related to a post is through a query for the news items user posts in Amharic text. The overall focus of this research is to investigate the Amharic text summarization news items posted on social media over a time posts for Facebook and Twitter user to find the relevant information to

summarize monthly to groups and enter the month and year, which address the problem of deciding the correct or the target summary to the users.

To this end the research question to answers, we consistent on the techniques which include measuring the similarity between each posted documents for sentences level to calculate the cosine similarity for each sentences and after similarity results to use for clustering techniques (kmeans algorithm) to cluster related similar posted sentence based on values to groups the similarity sentences and to extract the summarize each clustered grouped post texts. In this study, we applied the summarize approach is extractive summarization which is extract the relevant sentences to extract from the input corpus and finally to summarize the user selected sentences in the document.

The evaluation methodology for news items posted text summarization measuring the compression rate, precision, Recall and F-measure discovered results are interesting and meaning of the summarized document. For all experiments are done for using the porter stemmer that used the dictionary files to contain the experiments in developed rules for removing the Amharic affixes in the morphological rich Amharic words and inflectional words to stem, for this evaluation of the stemmer to increase the performance of the system. There are four grouped posted texts based on this the representative files group protests post, droughts post, sports post and floods post. The system summary generated by two ways in the automatic and ideal summary to prepared 30 posted texts to evaluate results for each experiments to use the three extraction rates for at 10%, 20% and maximum rate 30% to evaluate the summary posted Amharic texts. Based on this to evaluate the performance of the system that should be generated the summaries to compare with the manual summary of the human prepared for the summarizer/evaluators for the objective evaluation of posted documents. However, the text summarization process evaluations are challenged to evaluate, it used for subjective, objective evaluation method and also f-measure evaluation systems are used for this research study. For this evaluation is challenged due to the real fact of the human summaries, there is not perfect to evaluate the performance of the text summarization that should be several human/ideal summary can be made over a given text documents. In this work the hybrid TF-IDF achieving the compresses rate in news items posted text summarization posted texts documents on social media in protests post (E1), droughts post (E2), sports post (E3) and floods post (E4) are evaluated the results for the experiments. In the experiment one the highest F-measure score is 87.07% for extraction rate at 30%, in the clustered group of protests posts. The second experiment the highest F-measure score is 84% for extraction rate at 30%, in droughts post groups. In the third experiment the highest F-measure score is 91.37% for extraction rate at 30%, in the sports post groups and also the fourth experiments the highest F-measure score is 93.52% for extraction rate at 30% to generate the summary post texts. If the

system to generate the size of summary is increased, the extraction rate also increased in posted texts. For this the evaluation system shown that a very good results to summaries the posted texts on social media. Based on the results of the experiments the TF-IDF algorithms are the best of all to identify the frequently terms in the document to find the important sentences found in the posted texts to summaries the sentences scores and top ranked sentences to pick the higher score sentences from top score to low score value sentences in the document depend on the user to decide the size of the summary. The results obtained were encourage as there is lack of resources for Amharic texts and tools of the Amharic language because of lack of labeled corpus for social media and summary tools.

### 6.2. Recommendation and future work

The following recommendations and future work to advance based on our findings with regards to the developments of resources on social media for Facebook and Twitter posted texts and future research directions for news items posted text summarization on Amharic language some of the items that could be easily done to build upon this system are:

- All the results showed that the methods are properly working for creating news items posted text summarization for used the methods for hybrid TF-IDF algorithm. Apply proposed algorithm to identify the news item posted texts automatically grouped on social media post news rather than to prepare manual the corpus separated and further research in the area of Amharic language on social media and also it can be done the news items posted text summarization for others local language.
- For applying the algorithm with the topic identification summarization the news items on social media posted texts on Facebook and Twitter to link Amharic posting news texts sites of the social media. These news item post information summarizes systems on social media are highly connected with human daily activities.
- The stemming algorithm used in our work identifies inflectional morphology Amharic words to increase in co-occurrence counts that results from the use of derivation morphology could result in increased performance of the system and to apply more Amharic lexicon rules and also lists into the dictionary file to use a dictionary to control over and under stemming the Amharic words.
- Applying the proposed algorithm for the tasks of multi-documents posted Amharic texts and sequential update summarization tasks are one possible future work to do the research on social media post texts. These summarization tasks usually contain the summarization of a post texts of topically related posted texts. This means that redundant news posted texts are likely to exist during the news item posted whose summary is required.

- For other foreign language a standard annotated corpora are available for news items posted events for training and testing a system. But in Ethiopia, we could not have such data for Amharic language. So, there needs to be an initiative to prepare the data from Facebook and Twitter in different sources.
- In order to improve our results the posted clustered posted document for summarization may be required to compress the documents to make a positive summary well done. In addition, if the assigned threshold for summarization is changed according to the document already containing in a summary better results will follow. In order to determine precise extracted summary and I suggested to them will try sentence role based summarization work.

# REFERENCES


[1] Q. Guo, F. Diaz, and E. Yom-Tov, "Updating users about time critical events," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7814 LNCS, pp. 483–494, 2013.

[2] E. N. Agency, "Yonnaness Afwork "Amharic Text classification," 2000.

[3] T. Xu, P. Mcnamee, and D. W. Oard, "HLTCOE at TREC 2013 : News items Summarization," 2013.

[4] "Ethiopian History, "https: //www.ethiotube.net/video/37851" .

[5] D. Chakrabarti and K. Punera, "Event Summarization Using Tweets," pp. 66–73, 2011.

[6] F. Chong and T. Chua, "Automatic Summarization of Events from Social Media," 2009.

[7] H. Sayyadi, M. Hurst, A. Maykov, and M. Livelabs, "Event Detection and Tracking in Social Streams ∗," pp. 311–314, 2009.

[8] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: real-time event detection by social sensors," *WWW '10: Proceedings of the 19th international conference on World wide web*, p. 851, 2010.

[9] E. Yulianti, S. Huspi, and M. Sanderson, "Tweet-Biased Summarization," pp. 1–17.

[10] H. Dang and K. Owczarzak, "Overview of the TAC 2008 update summarization task," *Tac*, pp. 1–16, 2008.

[11] H. L. Eidheim, "News items Summarization of Time Critical Events," no. June, 2015.

[12] P. W. Mcburney and C. Mcmillan, "Automatic Source Code Summarization of Context for Java Methods," vol. 5589, no. c, pp. 1–18, 2015.

[13] J. Yang and S. Counts, "Predicting the Speed , Scale , and Range of Information Diffusion in Twitter," 2009.

[14] D. Metzler, C. Cai, E. Hovy, A. Way, and M. Rey, "Structured Event Retrieval over Microblog Archives," pp. 646–655, 2012.

[15] M. Imran, C. Castillo, F. Diaz, and S. Vieweg, *A Processing Social Media Messages in Mass Emergency: A Survey*. 2012.

[16] K.R.Premlath and T.V.Geetha, "Extracting Temporal Patterns and Analyzing Peak Events." Vol2, July- Dec 2010.

[17] J. Makkonen and H. Ahonen-myka, "Utilizing News items Information in Topic Detection and Tracking."

[18] A. T. S. To, "SCHOOL OF GRADUATE STUDIES AUTOMATIC AMHARIC TEXT SUMMARIZATION USING LATENT SEMANTIC ANALYSIS By : Melese Tamiru AUTOMATIC AMHARIC TEXT SUMMARIZATION USING LATENT SEMANTIC ANALYSIS By : Melese Tamiru," 2009.

[19] "A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Science By," 2013.

[20] F. O. F. Computer and M. Science, "SCHOOL OF GRADUATE STUDIES FACULTY OF COMPUTER AND MATHEMATICAL SCIENCE Topic-based Amharic Text Summarization Topic-based Amharic Text Summarization," no. March, 2011.

[21] K. D. Dessalegn and M. Y. Tachbelie, "Graph-based Automatic Amharic Text Summarizer," vol. 8.

[22] H. Saggion, D. Radev, S. Teufel, and S. M. Strassel, "Developing Infrastructure for the Evaluation of Single and Multi-document Summarization Systems in a Cross-lingual Environment," no. 2000, 2001.

[23] D. R. Radev and A. Arbor, "Centroid-based summarization of multiple documents : sentence extraction , utility-based evaluation , and user studies."

[24] R. Barzilay and K. R. Mckeown, "Inferring Strategies for Sentence Ordering in Multidocument News Summarization," vol. 17, pp. 35–55, 2002.

[25] "COMPARATIVE EVALUATION OF MODULAR AUTOMATIC SUMMARISATION SYSTEMS USING CAST Constantin Or ˇ," 2006.

[26] D. Marcu and M. Rey, "Discourse-Based Summarization in DUC-2001 Summarizing document," 2001.

[27] D. Inouye and J. K. Kalita, "Comparing Twitter Summarization Algorithms for Multiple Post Summaries," 2010.

[28] B. P. Sharifi, D. I. Inouye, and J. K. Kalita, "Summarization of Twitter Microblogs," vol. 57, no. 3, 2014.

[29] Z. He, C. Chen, J. Bu, C. Wang, and L. Zhang, "Document Summarization Based on Data Reconstruction," pp. 620–626, 2011.

[30] P. Paroubek, S. Chaudiron, L. Hirschman, L. Cnrs, B. Université, and P. Xi, "Principles of Evaluation in Natural Language Processing," vol. 48, pp. 7–31, 2007.

[31] K. Tao, F. Abel, Q. Gao, and G. Houben, "TUMS : Twitter-based User Modeling Service," pp. 1–15, 2010.

[32] W. Chung, H. Chen, L. G. Chaboya, C. D. O. Toole, and H. Atabakhsh, "Evaluating event visualization : a usability study of COPLINK spatio-news items visualizer," vol. 62, pp. 127–157, 2005.

[33] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, and J. Lin, "Earlybird : Real-Time Search at Twitter," 2011.

[34] M. Potthast, B. Stein, F. Loose, and S. Becker, "Information Retrieval in the Commentsphere," vol. V, no. January, 2012.

[35] I. Novalija, M. Papler, and D. Mladeni, "TOWARDS SOCIAL MEDIA MINING : TWITTEROBSERVATORY," pp. 2–5.

[36] B. Sharifi, M. Hutton, and J. K. Kalita, "Experiments in Microblog Summarization."

[37] P. Meladianos and I. R. C. Athena, "Degeneracy-based Real-Time Sub-Event Detection in Twitter Stream," 2015.

[38] F. Ibekwe-sanjuan, S. Fernandez, E. Sanjuan, and E. Charton, "Annotation of Scientific Summaries for Information Retrieval," pp. 1–14, 2002.

[39] K. Bontcheva, L. Derczynski, A. Funk, M. A. Greenwood, D. Maynard, and N. Aswani, "TwitIE : An Open-Source Information Extraction Pipeline for Microblog Text," 2013.

[40] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter , a Social Network or a News Media ? Categories and Subject Descriptors."

[41] B. Truong, C. Caragea, A. Squicciarini, and A. H. Tapia, "Identifying Valuable Information from Twitter During Natural Disasters," 2014.

[42] T. N. Dao and T. Simpson, "Measuring Similarity between sentences."

[43] L. Huang, H. Li, and L. Huang, "Comments-Oriented Document Summarization Based on Multi-aspect Co-feedback Ranking," pp. 363–374, 2013.

[44] Y. Seki, "Sentence Extraction by tf / idf and Position Weighting from Newspaper Articles," 2003.

[45] N. K. Nagwani, "A Frequent Term and Semantic Similarity based Single Document Text Summarization Algorithm," vol. 17, no. 2, pp. 36–40, 2011.

[46] H. Saggion, T. Poibeau, H. Saggion, T. Poibeau, A. Text, and S. Past, "Automatic Text Summarization : Past , Present and Future To cite this version : Automatic Text Summarization : Past , Present and Future," 2016.

[47] J. L. Neto, A. A. Freitas, and C. A. A. Kaestner, "Automatic Text Summarization using a Machine Learning Approach," no. i.

[48] S. Eyassu, "Classifying Amharic News Text Using Self-Organizing Maps," 2000.

[49] N. African and L. Resource, "Language & Culture," no. 608.

[50] S. Abebe, T. Seyum, S. Atnafu, and S. K. Kassegne, "Ethiopic Keyboard Mapping and Predictive Text Inputting Algorithm in a Wireless Environment," no. 760.

[51] T. Thesis, I. S. My, O. Work, H. A. S. N. O. T. Been, Z. Sintayehu, T. H. E. Thesis, H. A. S. Been, S. For, and E. With, "No Title."

[52] J. Macqueen, "SOME METHODS FOR CLASSIFICATION AND ANALYSIS OF MULTIVARIATE OBSERVATIONS," vol. 233, no. 233, pp. 281–297.

[53] T. Schubotz and R. Krestel, "Online News items Summarization of News Events," no. 2.

[54] R. Barzilay and L. Lee, "Catching the Drift : Probabilistic Content Models , with Applications to Generation and Summarization," 2003.

[55] M. Yousfi-monod, M. Cedex, V. Prince, and M. Cedex, "Sentence Compression as a Step in Summarization or an Alternative Path in Text Shortening," no. August, pp. 139–142, 2008.

[56] V. M. Khanapure and P. C. V R, "Multi-document Summarization Based on," pp. 8318–8325, 2014.

# Appendixes

## I. Amharic character sets

| Ordere | | | | | | |
|---|---|---|---|---|---|---|
| 1st | 2nd | 3rd | 4th | 5th | 6th | 7th |
| ሀ | ሁ | ሂ | ሃ | ሄ | ህ | ሆ |
| ለ | ሉ | ሊ | ላ | ሌ | ል | ሎ |
| ሐ | ሑ | ሒ | ሓ | ሔ | ሕ | ሖ |
| መ | ሙ | ሚ | ማ | ሜ | ም | ሞ |
| ሠ | ሡ | ሢ | ሣ | ሤ | ሥ | ሦ |
| ረ | ሩ | ሪ | ራ | ሬ | ር | ሮ |
| ሰ | ሱ | ሲ | ሳ | ሴ | ስ | ሶ |
| ሸ | ሹ | ሺ | ሻ | ሼ | ሽ | ሾ |
| ቀ | ቁ | ቂ | ቃ | ቄ | ቅ | ቆ |
| በ | ቡ | ቢ | ባ | ቤ | ብ | ቦ |
| ተ | ቱ | ቲ | ታ | ቴ | ት | ቶ |
| ቸ | ቹ | ቺ | ቻ | ቼ | ች | ቾ |
| ኀ | ኁ | ኂ | ኃ | ኄ | ኅ | ኆ |
| ነ | ኑ | ኒ | ና | ኔ | ን | ኖ |
| ኘ | ኙ | ኚ | ኛ | ኜ | ኝ | ኞ |
| አ | ኡ | ኢ | ኣ | ኤ | እ | ኦ |
| ወ | ዉ | ዊ | ዋ | ዌ | ው | ዎ |
| ዐ | ዑ | ዒ | ዓ | ዔ | ዕ | ዖ |
| ከ | ኩ | ኪ | ካ | ኬ | ክ | ኮ |

We could import the Unicode characters of Amharic filed ሀ - ፐ into java code.

/// Unicode representation of Ethiopic Amharic symbols

We import all Amharic letter into java

```
Prívate final char [][] UNICODECHARS = {
  {'\u1200', '\u1201', '\u1202', '\u1203', '\u1204', '\u1205', '\u1206'}, // ሀ = '\u1200'
  {'\u1208', '\u1209', '\u120A', '\u120B', '\u120C', '\u120D', '\u120E'},
  {'\u1210', '\u1211', '\u1212', '\u1213', '\u1214', '\u1215', '\u1216'},
  {'\u1218', '\u1219', '\u121A', '\u121B', '\u121C', '\u121D', '\u121E'},
  {'\u1220', '\u1221', '\u1222', '\u1223', '\u1224', '\u1225', '\u1226'},
  {'\u1228', '\u1229', '\u122A', '\u122B', '\u122C', '\u122D', '\u122E'},
  {'\u1230', '\u1231', '\u1232', '\u1233', '\u1234', '\u1235', '\u1236'},
  {'\u1238', '\u1239', '\u123A', '\u123B', '\u123C', '\u123D', '\u123E'},
  {'\u1240', '\u1241', '\u1242', '\u1243', '\u1244', '\u1245', '\u1246'},
  {'\u1260', '\u1261', '\u1262', '\u1263', '\u1264', '\u1265', '\u1266'},
  {'\u1268', '\u1269', '\u126A', '\u126B', '\u126C', '\u126D', '\u126E'},
  {'\u1270', '\u1271', '\u1272', '\u1273', '\u1274', '\u1275', '\u1276'},
  {'\u1278', '\u1279', '\u127A', '\u127B', '\u127C', '\u127D', '\u127E'},
  {'\u1280', '\u1281', '\u1282', '\u1283', '\u1284', '\u1285', '\u1286'},
  {'\u1290', '\u1291', '\u1292', '\u1293', '\u1294', '\u1295', '\u1296'},
  {'\u1298', '\u1299', '\u129A', '\u129B', '\u129C', '\u129D', '\u129E'},
  {'\u12A0', '\u12A1', '\u12A2', '\u12A3', '\u12A4', '\u12A5', '\u12A6'},
  {'\u12A8', '\u12A9', '\u12AA', '\u12AB', '\u12AC', '\u12AD', '\u12AE'},
  {'\u12B8', '\u12B9', '\u12BA', '\u12BB', '\u12BC', '\u12BD', '\u12BE'},
  {'\u12C8', '\u12C9', '\u12CA', '\u12CB', '\u12CC', '\u12CD', '\u12CE'},
```

**{'\u12D0', '\u12D1', '\u12D2', '\u12D3', '\u12D4', '\u12D5', '\u12D6'},**

**{'\u12D8', '\u12D9', '\u12DA', '\u12DB', '\u12DC', '\u12DD', '\u12DE'},**

**{'\u12E0', '\u12E1', '\u12E2', '\u12E3', '\u12E4', '\u12E5', '\u12E6'},**

**{'\u12E8', '\u12E9', '\u12EA', '\u12EB', '\u12EC', '\u12ED', '\u12EE'},**

**{'\u12F0', '\u12F1', '\u12F2', '\u12F3', '\u12F4', '\u12F5', '\u12F6'},**

**{'\u12F8', '\u12F9', '\u12FA', '\u12FB', '\u12FC', '\u12FD', '\u12FE'},**

**{'\u1300', '\u1301', '\u1302', '\u1303', '\u1304', '\u1305', '\u1306'},**

**{'\u1308', '\u1309', '\u130A', '\u130B', '\u130C', '\u130D', '\u130E'},**

**{'\u1320', '\u1321', '\u1322', '\u1323', '\u1324', '\u1325', '\u1326'},**

**{'\u1330', '\u1331', '\u1332', '\u1333', '\u1334', '\u1335', '\u1336'},**

**{'\u1338', '\u1339', '\u133A', '\u133B', '\u133C', '\u133D', '\u133E'}**

**{'\u1340', '\u1341', '\u1342', '\u1343', '\u1344', '\u1345', '\u1346'},**

**{'\u1348', '\u1349', '\u134A', '\u134B', '\u134C', '\u134D', '\u134E'},**

**{'\u1350', '\u1351', '\u1352', '\u1353', '\u1354', '\u1355', '\u1356'}//ፐ = '\u1355' //the 6th order Fidel**

**};** // the Unicode values of Amharic script symbols.

## II. List of Affixes for Amharic words

Prefixes removal as the following another in one, two, three, four prefixes to develop rules to read and remove the words in the beginning of words.

| Individual Prefixes | | | Combined Prefixes | | |
|---|---|---|---|---|---|
| በ | ለ | እየ | እንደ | እስኩ | በየ |
| ከ | ይ | እን | እንዲ | እስከን | ከየ |
| የ | መ | የሚ | እስከ | እንደየ | የእነ |
| ሲ | ስለ | ከነ | እንድ | እንድን | በእነ |
| ን | እነ | | እስኪ | ለእነ | ከእነ |

Suffixes words to remove the affixes at end of the words or postfixes to eliminate the words like the following affixes.

| Individual Suffixes | | | Combined Suffixes | | |
|---|---|---|---|---|---|
| ን | ያል | ላችሁ | ዎች | ዎቿን | ምና |
| ና | የዋ | ላቸው | ዎችን | ናም | ምናም |
| ም | ቸው | ባቸው | ዎችንና | ናምና | ምናን |
| ች | ተው | ባችሁ | ዎችንም | ናምን | ምቹ |
| ዬ | የው | | ዎችና | ናን | ምን |
| ዎ | በት | | ዎችናም | ናንና | ምንና |
| ህ | ባት | | ዎችናን | ናንም | ምንም |
| ሽ | ነት | | ዎችም | ናንን | ምንን |
| ዋ | ለት | | ዎችምና | ናው | ምዉና |
| ቹ | ላት | | ዎቿና | ናዉና | ምዉም |
| ው | ይቱ | | ዎቿናም | ናውና | ምውም |
| ቱ | ያዊ | | ዎቿናን | ናዉም | ምዉን |
| ሁ | ኞች | | ዎቿም | ናውም | ዉም |
| ኝ | ዎቹ | | ዎቿምና | ናዉን | ዉምና |
| ኛ | አዊ | | ዎቿምን | ንና | ያውያኑ |
| ዊ | | | ዉናን | ዉና | ያውያንና |
| ት | | | ዉናዉ | ዉናም | ያውያን |

## III. Guideline for Manual Summary for post texts

The purpose of this research is to develop a computerized news item Amharic posted text Summarizer over a time (date, month, and year) post to identify the texts. The performance of this summarized will be evaluated using by comparing it with the summaries with ideal or human prepared manually summaries. In this, you are kindly requested to prepare a manual summary based on the following guideline for each posted Amharic texts on social media provide to you and you will be given. You prepare a manually summarize, all you will

have to do the major action that will be taken and ranking each sentence of Amharic texts according to their importance information in describing the main information flows of the posted texts in the documents. A well organized and well-structured, meaningful summarize can be created combining these ranked sentences, one after the other in their orders of rank. From this most important sentences will be given an importance score equal to one and the least important sentence will be assigned n, where n is the total number of sentences in the document. The sentence ranking guidelines will be based on the quality measuring criteria that is included Informativeness, Non-Redundancy, Coherence, Coverage, and Referential clarity to describe as following.

1. Informativeness: is the sentences should be contain the core idea of the posted texts in the document. This to get higher ranked sentences would be more informative sentences for containing the key ideas of the post texts to obtain for the summary.
2. Non-Redundancy: is a repetition of post text ideas and the same contents using sentences should completely be avoided in the summary post texts.
3. Coherence: is the flow of information in the given text and about a topic in consecutive sentences in the posted documents and also the sentences are read for organizing the idea of the texts must have a flow information rather than having unrelated sentences in the posted texts.
4. Coverage: is a summary should cover all of the important topics presented in the post Amharic texts. For this the sentence to contain the core topic that is highly ranked sentences should contain the maximum possible information about the topic of the posts.
5. Referential clarity: when the reading of post text in the summary in the ranked order of the sentences, those sentences that have referential words like noun phrases and others should be easy for the user or reader to identify to whom or what they are referring to in the sentence.

Thank you in advance, for taking your time to prepare the texts to the summaries for post texts!

Example some sample of testing data set in the protest post within a manual summary, you evaluate the following post within the sentences.

18-Dec-2015 "ወጣቱ ይህ መንግሥት በሚያደርገው ነገር ሁሉ ተንገሽግሿል" #OromoProtests #Ethiopia ላለፉት ሦስት ሳምንታት የኦሮሞ ብሔረሰብ አባላት ተማሪዎች የአዲስ አበባ እና የፊንፊኔ ዙሪያ የኦሮሚያ ልዩ ዞን የተቀናጀ ማስተር ፕላን ተግባራዊ ይደረጋል መባሉን ሲቃወሙ ሰንብተዋል።

2-Feb-2016 ኢትዮጵያውያን ዛሬ በዋሽንግተን ዲሲ ሰላማዊ ሰልፍ አደረጉ "በኦሮሚያ፥ በጎንደርና በጋምቤላ በዜጎች ላይ እየደረሰ ያለው በደል ይቁም" "አምባገነኖች የልማት ሃጋር አይደሉም" የዛሬው የዋሽንግተን ሰልፈኞች የኦሮሚያን ክልላዊ ምክር ቤት እንበትናለን ባሉት መሰረት በተደረገው ግምገማ የክልሉ ምክር ቤት ፕረዚዳንት እና ኣባላትን ጨምሮ የፊደራል መንግስታቸውን ባለስልጣናት እና የኡሕዴድ ኣመራሮች እንደሚያሰናበቱ እየተጠበቀ ይገኛል።

12-Sep-2016 ህዝቡ የስልጣን ባለቤት አለመሆኑ አገሪቷን ለችግር እንደዳረጋት የሙስሊም የመፍትሄ አፈላላጊ ኮሚቴ ገለጸ ኢሳት (መስከረም 2 ፥ 2009) የሙስሊም የመፍትሄ አፈላላጊ ኮሚቴ በኢትዮጵያ የህዝብ ስልጣን ባለቤትነት አለመከበር እንዲሁም የዕኩልነትና የእኩል ተጠቃሚነት መርህ አለመተግባርና ዜጎችን ሁሉ ሊያሳትፍ የሚችል ዴሞክራሲያዊ ስርዓት እጦት ሃገሪቱ በአሁኑ ወቅት ላለችበት አሳሳቢ ሁኔታ መንስዔ መሆኑን ገልጿል።

12-Sep-2016 በጎንደር የተሰባሰቡ የሙስሊሙ ማህበረሰብ አባላት ቀይ ፊኛ ከመልቀቃቸው በተጨማሪ፣ እጃቸውን አጣምረው ተቃውሞ አሰሙ ኢሳት (መስከረም 2 ፥ 2009) የዘንድሮውን የኢድ አል አድሃ በዓል ለማክበር በጎንደር ከተማ የተሰባሰቡ የሙስሊሙ ማህበረሰብ በከተማዋ ባለስልጣናት ባለኮከብ ባንዲራን ይዘው እንደወጡ የተሰጣቸውን መልዕክት ሳይቀበሉ መቅረታቸው ታውቋል።

12-Sep-2016 ባለፈው ሳምንት በኦሮሚያ የተጀመረው ተቃውሞ ዛሬም ቀጥሎ እንደሚገኝ ተገለጸ ኢሳት (መስከረም 2 ፥ 2009) ባለፈው ሳምንት በተለያዩ የኦሮሚያ የክልል ከተሞች ሲካሄድ የቆየው ከቤት ያለመውጣትና የንግድ እንቅስቃሴ የማቆም አድማ ሰኞ ድረስ መቀጠሉን ከሃገር ቤት የተገኘ መረጃ አመልክቷል።

24-Feb-2016 በምዕራብ ሐረርጌ መቻራ፣ በወሊጋ ነቀምትና በሊበን ነጌሌ ከተሞች ውስጥ የተለያዩ የተቃውሞና የግጭት እንቅስቃሴዎች ትናንትናና ዛሬ እንደነበሩ የየአካባቢው ነዋሪዎች ገልፀዋል። የአከባቢው ነዎሪዎች የመንግስትን ሃላፊዎች አንቀበልም የእኛ መሪ ዶክተር መረራ ጉዲና ናቸው ብለዋል በወሊጋ በበርካታ አከባቢዎች መረጋጋት ከጠፋ ሳምንታት አልፈዋል በእስር ቤት ህይወቱ ያለፈው የወጣት ሙባረክ ይመር የቀብር ስነስርዓት ቅዳሜ በደሴ ከተማ ተፈጸመ።

18-Feb-2016 አሜሪካ፣ ኖርዌይና እንግሊዝ በኦሮምያ የጉዞ ማስጠንቀቂያ አወጡ ዋሽንግተን ዲሲ ኦሮምያ ውስጥ በምሥራቅ ሐረርጌና በምዕራብ አርሲ አካባቢዎች ሰሞኑን የታየው የተቃውሞ እንቅስቃሴ ዛሬም ቀጥሎ በተለይ በምሥራቅ ሃረርጌ ጉራዋ ሕይወት መጥፋቱ ተሰምቷል።በኦሮሚያና በአማራ ክልሎች በመንግስት የሚፈጸሙ ግድያዎችንና የጅምላ እስራቶችን በመቃወም 14 አለም አቀፍ የሰብዓዊ መብት ተሟጋች ተቋማት የተባበሩት መንግስታት የሰብዓዊ መብት ምክር ቤት እርምጃን እንዲወስዱ መጠየቃቸው ይታወሳል።

12-Sep-2016 ጋዜጠኛ ዩሱፍ ጌታቸውን መንግስትን ይቅርታ አልጠየቅሁም አለ ኢሳት (መስከረም 2 ፥ 2009)የህዝበ ሙስሊሙን ሰላማዊ እንቅስቃሴ ሲዘግብ በአሸባሪነት ተከሶ በእስር ቤት ላይ የቆየው ጋዜጠኛ ዩሱፍ ጌታቸው መንግስትን ይቅርታ አለመጠየቁን ገለጸ። ጋዜጠኛ ዩሱፍ ሙያውን ሽፋን በማድረግ ህዝብን ለአመጽ በማነሳሳት በሚል ክስ 7 አመት ተፈርዶበት በእስር ቤት አመታትን አስቆጥሯል። ከ4 አመታት በላይ በእስር የቆየው ጋዜጠኛ ዩሱፍ ጌታቸው በመከሮ ሊፈታ ወራት ሲቀሩት ከአዲሱ ዓመት ጋር በተየያዘ በምህረት በሚል ከተለቀቁት መካከል ነው። ወደፊት ስራውን እንደሚቀጥልና መንግስት ዜጎችን በማፈን ሃሳባቸውን እንዳይገልጹ የሚያደርገውን ተፅዕኖ እንዲያቆምም ጥሪ አቅርቧል።

## IV. Guidelines for Summary Subjective Evaluation posted texts

This research the Summarizer guidelines are prepared to help you, for to do, the evaluator, in the evaluation of the quality of the system summary prepared for the selected post Amharic texts on social media. Since you have read the posted texts before, you will be preparing an ideal summary your participation in the evaluation of the automatic summary is important sentences to rank and you are provided within the automatic summary equal of some of the posted texts in the document that you have been summarized to evaluate the system

performance subjectively. So, you will be provided a summary of the extraction rate posts document at the three different summarizing rates (10%, 20% and 30% for each of posted Amharic texts) that will be randomly selected post texts in the clustered document.

From these evaluations measure quality criteria that you should follow are described as follows.

1. The Linguistic Quality: This to consider how easy and readable post texts are summarized. A summary should have a good quality when it comes to linguistics experts. The focus here is on readability, fluency and quality of the summary. The evaluation includes quality measuring criteria like to grammatical correctness, referential clarity and non-redundancy. In grammatical correctness you have to check the occurrence of grammatically correct or incorrect sentences like missing words, fragments of the sentences of the summary in posted texts. The referential clarity is measured the easiness to identify answers to "what, who whose and others" in the sentences for posting documents. And also in non-redundancy is checking for repetition of sentences in posted texts, you should look that to avoid unnecessary repetition of sentences in the post texts in the given texts. Based on this, to evaluate the quality measuring levels of the summary in the give texts, please you give rank to each of the summaries posted texts that you are provided with a grade level from 1 up to 5 (which you will be converted gradually to 20% up to 100%) in the posted texts.

2. Informativeness: for which of the summaries includes the most important concept of the posted text?

This is the level of satisfying the users in posted texts the information to need by each summary text and containing the topic concept is different post texts of the given document. Then, you are asked to identify this level of the post texts. For this, please read the topic statements and the systems automatically summary extracted for each posted texts.

Based on the informativeness level of the summary, please give rank to each of the summaries that you are provided with a grade level from 1 up to 5.

3. Coherence and Structure: it is considered to how well is the summary posted texts for organized and structured the texts. The summary of the coherence that it should build up the topic of the posted texts in the sequential sentences to a coherent summary on the posted texts. Then, you should be observed at the buildup for sentences that should be comprised the automatic summary generated the posted texts. Based on this the structured and coherence level of the summary information in the given texts, please give rank to each of the summary post texts that you provided with a grade level from 1 up to 5.

4. How do you compress rates the summary for post texts?

For this is to say, to what level of measure does the summary quality measuring in your observation for the post texts. After reading the posted texts to generate the automatic summary texts and the posted texts in the

documents, since the overall observation to determine for what do you think the texts for the level of the automatic summary that should be given five level of grading rank to present the score values to give for this: 1→Very Poor, 2→Poor, 3→Fair, 4→Good and 5→Very Good. Based on this, please the summarizer to rank each automatically summary based on the following six quality measuring criteria to evaluate the summary post texts. Like Informativeness, coherence and structure, overall observation, non-redundancy, grammatical correction and referential clarity.

| **Quality measuring** | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1.Informativeness | | | | | |
| 2.Non-redundancy | | | | | |
| 3.Coherance and structure | | | | | |
| 4.Overal Observation | | | | | |
| 5.Gramatical correctness | | | | | |
| 6.Referential clarity | | | | | |

Thank in advance, you are taking your time to evaluate the summaries the post texts!

## V. The grade scores given under for the Subjective Evaluations for the summaries

From the grade score given under for the Subjective Evaluations for 40 summaries for selected in to post texts

| **File name for each clustered posts** | **Evaluation grade given for the summary** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **totals sentences** | **Informativeness** | **Non-red Undancy** | **Coherence &Structure** | **Overall observation** | **Grammatical correctness** | **Referential clarity** | **Total grades scores from 30** | **Total grade value in % average** |
| At 10% Protests posts | | | | | | | | | |
| Prot1.txt | 53 | 4 | 4 | 4 | 2 | 4 | 3 | 21 | 70% |
| Prot2.txt | 62 | 5 | 5 | 3 | 3 | 3 | 4 | 23 | 76.67% |
| Prot3.txt | 53 | 4 | 5 | 5 | 4 | 5 | 4 | 27 | 90% |
| **Average score** | **56** | **4** | **5** | **4** | **3** | **4** | **4** | **23.67** | **78.89%** |
| Drought post | | | | | | | | | |
| Drot1.txt | 67 | 4 | 4 | 5 | 4 | 4 | 4 | 25 | 83.33% |
| Drot2.txt | 62 | 4 | 4 | 5 | 3 | 4 | 3 | 23 | 76.67% |
| Drot3.txt | 59 | 4 | 5 | 3 | 3 | 4 | 3 | 22 | 73.33% |
| **Average score** | **63** | **4** | **5** | **4** | **3** | **4** | **3** | 23.33 | 78.78% |
| Sport posts | | | | | | | | | |
| Spot1.txt | 42 | 5 | 4 | 4 | 3 | 5 | 4 | 25 | 83.33% |
| Spot2.txt | 38 | 3 | 5 | 4 | 3 | 4 | 5 | 24 | 80% |
| Spot3.txt | 34 | 4 | 5 | 4 | 2 | 5 | 5 | 25 | 83.33% |
| **Average core** | **38** | **4** | **5** | **4** | **4** | **5** | **5** | **24.67** | **82.22%** |
| Floods posts | | | | | | | | | |
| Flod1.txt | 56 | 4 | 5 | 3 | 4 | 3 | 4 | 23 | 76.67% |
| Flod2.txt | 43 | 5 | 4 | 4 | 3 | 5 | 5 | 26 | 86.67% |
| Flod3.txt | 29 | 5 | 4 | 3 | 2 | 4 | 4 | 22 | 73.33% |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Average score** | **42** | **5** | **4** | **3** | **3** | **4** | **4** | 23.67 | **78.89%** |
| At 20% Protests posts | | | | | | | | | |
| Prot1.txt | 53 | 5 | 5 | 5 | 3 | 4 | 5 | 27 | 90% |
| Prot2.txt | 62 | 5 | 5 | 3 | 4 | 3 | 3 | 23 | 76.67% |
| Prot3.txt | 53 | 3 | 4 | 4 | 4 | 5 | 3 | 23 | 76.67 |
| **Average score** | **56** | **4** | **5** | **4** | **4** | **4** | **4** | **27.33** | **81.11%** |
| Droughts post | | | | | | | | | |
| Drot1.txt | 67 | 4 | 4 | 4 | 3 | 4 | 4 | 24 | 80% |
| Drot2.txt | 62 | 3 | 5 | 4 | 4 | 4 | 4 | 24 | 80% |
| Drot3.txt | 59 | 4 | 5 | 5 | 5 | 4 | 4 | 27 | 90% |
| **Average score** | **63** | **4** | **5** | **4** | **4** | **4** | **4** | **25** | **83.33%** |
| Sport posts | | | | | | | | | |
| Spot1.txt | 42 | 4 | 5 | 4 | 4 | 5 | 4 | 26 | 86.67% |
| Spot2.txt | 38 | 3 | 4 | 5 | 3 | 4 | 4 | 23 | 76.67% |
| Spot3.txt | 34 | 5 | 4 | 3 | 5 | 5 | 5 | 27 | 90% |
| **Average score** | **38** | **4** | **4** | **4** | **3** | **5** | **4** | **25.33** | **84.44%** |
| Floods posts | | | | | | | | | |
| Flod1.txt | 56 | 4 | 5 | 5 | 4 | 3 | 4 | 25 | 83.33% |
| Flod2.txt | 43 | 3 | 3 | 4 | 5 | 5 | 3 | 23 | 76.67% |
| Flod3.txt | 29 | 4 | 4 | 3 | 4 | 4 | 5 | 24 | 80% |
| **Average score** | **42** | **4** | **4** | **4** | **4** | **4** | **4** | 24 | **80%** |
| **At 30%** Protests posts | | | | | | | | | |
| Prot1.txt | 53 | 5 | 5 | 3 | 5 | 4 | 4 | 26 | 86.67% |
| Prot2.txt | 62 | 5 | 5 | 3 | 4 | 3 | 4 | 24 | 80% |
| Prot3.txt | 53 | 4 | 4 | 3 | 4 | 5 | 5 | 25 | 83.33% |
| **Average score** | **56** | **5** | **5** | **3** | **4** | **4** | **4** | **25** | **83.33%** |
| Drought post | | | | | | | | | |
| Drot1.txt | 67 | 5 | 5 | 5 | 4 | 4 | 5 | 28 | 93.33% |
| Drot2.txt | 62 | 3 | 5 | 3 | 5 | 4 | 4 | 24 | 80% |
| Drot3.txt | 59 | 5 | 5 | 5 | 5 | 4 | 5 | 29 | 96.67% |
| **Average score** | **63** | **4** | **5** | **4** | **5** | **4** | **5** | 27 | 90% |
| Sport posts | | | | | | | | | |
| Spot1.txt | 42 | 4 | 4 | 4 | 4 | 5 | 4 | 25 | 83.33% |
| Spot2.txt | 38 | 5 | 5 | 5 | 5 | 4 | 5 | 29 | 96.67% |
| Spot3.txt | 34 | 4 | 4 | 4 | 4 | 5 | 5 | 26 | 93.33% |
| **Average score** | **38** | **4** | **4** | **4** | **4** | **5** | **5** | 26.67 | 88.89% |
| Floods posts | | | | | | | | | |
| Flod1.txt | 56 | 5 | 5 | 4 | 4 | 3 | 4 | 25 | 83.33% |
| Flod2.txt | 43 | 4 | 5 | 4 | 5 | 5 | 5 | 28 | 93.33% |
| Flod3.txt | 29 | 5 | 4 | 3 | 5 | 4 | 4 | 25 | 83.33% |
| **Average score** | **42** | **5** | **5** | **4** | **5** | **4** | **4** | 26 | **86.66%** |

## VI. Manual Evaluation Posts text Results

### 1. Informativeness

| **Files name** | **10% Summary** | | **20% Summary** | | **30% Summary** | |
|---|---|---|---|---|---|---|
| | **Grade** | **In the %** | **Grade** | **In the %** | **Grade** | **In the %** |
| Protests posts | | | | | | |
| Prot1.txt | 4 | 80% | 5 | 100% | 5 | 100% |
| Prot2.txt | 5 | 100% | 5 | 100% | 5 | 100% |

| | | | | | | |
|---|---|---|---|---|---|---|
| Prot3.txt | 4 | 80% | 3 | 60% | 4 | 80% |
| **Average score** | **4** | **86.67%** | **4** | **86.67%** | **5** | **93.33%** |
| Droughts post | | | | | | |
| Drot1.txt | 4 | 80% | 4 | 80% | 5 | 100% |
| Drot2.txt | 4 | 80% | 3 | 60% | 3 | 60% |
| Drot3.txt | 4 | 80% | 4 | 60% | 5 | 100% |
| **Average score** | **4** | **80%** | **4** | **66.67%** | **4** | **86.67%** |
| Sport posts | | | | | | |
| Spot1.txt | 5 | 100% | 4 | 80% | 4 | 80% |
| Spot2.txt | 3 | 60% | 3 | 60% | 5 | 100% |
| Spot3.txt | 4 | 80% | 5 | 100% | 4 | 80% |
| **Average score** | **4** | **80%** | **4** | **80%** | **4** | **86.67%** |
| Floods posts | | | | | | |
| Flod1.txt | 4 | 80% | 4 | 80% | 5 | 100% |
| Flod2.txt | 5 | 100% | 3 | 60% | 4 | 80% |
| Flod3.txt | 5 | 100% | 4 | 80% | 5 | 100% |
| **Average score** | **5** | **93.33%** | **4** | **73.33%** | **5** | **93.33%** |

2. **Non-redundancy**

| **Files name** | **10% Summary** | | **20% Summary** | | **30% Summary** | |
|---|---|---|---|---|---|---|
| | **Grade** | **In the %** | **Grade** | **In the %** | **Grade** | **In the %** |
| Protests posts | | | | | | |
| Prot1.txt | 4 | 80% | 5 | 100% | 5 | 100% |
| Prot2.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| Prot3.txt | 5 | 100% | 4 | 80% | 4 | 80% |
| **Average score** | **5** | **93.33%** | **5** | **93.33%** | **5** | **93.33%** |
| Droughts post | | | | | | |
| Drot1.txt | 4 | 80% | 4 | 100% | 5 | 100% |
| Drot2.txt | 4 | 80% | 5 | 100% | 5 | 100% |
| Drot3.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| **Average score** | **5** | **86.67%** | **5** | **93.33%** | **5** | **100%** |
| Sport posts | | | | | | |
| Spot1.txt | 3 | 60% | 5 | 100% | 5 | 100% |
| Spot2.txt | 5 | 100% | 4 | 80% | 5 | 100% |
| Spot3.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| **Average score** | **4** | **80%** | **4** | **86.67%** | **4** | **93.33%** |
| Floods posts | | | | | | |
| Flod1.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| Flod2.txt | 4 | 80% | 3 | 60% | 5 | 100% |
| Flod3.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| **Average score** | **4** | **86.67%** | **4** | **80** | **5** | **93.33%** |

**3. Coherence and structure**

| Files name | 10% Summary | | 20% Summary | | 30% Summary | |
|---|---|---|---|---|---|---|
| | Grade | In the % | Grade | In the % | Grade | In the % |
| Protests post | | | | | | |
| Prot1.txt | 4 | 80% | 5 | 100% | 4 | 80% |
| Prot2.txt | 3 | 60% | 3 | 60% | 5 | 100% |
| Prot3.txt | 5 | 100% | 4 | 80% | 4 | 80% |
| **Average score** | **4** | **80%** | **4** | **80%** | **3** | **86.67%** |
| Droughts post | | | | | | |
| Drot1.txt | 5 | 100% | 4 | 80% | 5 | 100% |
| Drot2.txt | 5 | 100% | 4 | 80% | 3 | 60% |
| Drot3.txt | 3 | 60% | 5 | 100% | 5 | 100% |
| **Average score** | **4** | **86.67%** | **4** | **86.67%** | **4** | **86.67%** |
| Sport posts | | | | | | |
| Spot1.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Spot2.txt | 4 | 80% | 5 | 100% | 5 | 100% |
| Spot3.txt | 4 | 80% | 3 | 60% | 4 | 80% |
| **Average score** | **4** | **80%** | **4** | **80%** | **4** | **86.67%** |
| Floods posts | | | | | | |
| Flod1.txt | 3 | 60% | 5 | 100% | 4 | 80% |
| Flod2.txt | 4 | 80% | 4 | 80% | 5 | 100% |
| Flod3.txt | 3 | 60% | 3 | 60% | 5 | 100% |
| **Average score** | **3** | **66.67%** | **4** | **80%** | **4** | **93.33%** |

**3. Over all observation**

| Files name 3. | 10% Summary | | 20% Summary | | 30% Summary | |
|---|---|---|---|---|---|---|
| | Grade | in the % | Grade | in the % | Grade | in the % |
| Protests posts | | | | | | |
| Prot1.txt | 2 | 40% | 3 | 60% | 5 | 100% |
| Prot2.txt | 3 | 60% | 4 | 80% | 4 | 80% |
| Prot3.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| **Average score** | **3** | **60%** | **4** | **73.33%** | **4** | **86.67%** |
| Droughts post | | | | | | |
| Drot1.txt | 4 | 80% | 3 | 60% | 3 | 60% |
| Drot2.txt | 3 | 60% | 4 | 80% | 5 | 100% |
| Drot3.txt | 3 | 60% | 5 | 100% | 4 | 80% |
| **Average score** | **3** | **66.67%** | **4** | **80%** | **4** | **80%** |
| Sport posts | | | | | | |
| Spot1.txt | 3 | 60% | 4 | 80% | 4 | 80% |
| Spot2.txt | 3 | 60% | 3 | 60% | 5 | 100% |
| Spot3.txt | 2 | 40% | 5 | 100% | 4 | 80% |
| **Average score** | **3** | **53.33%** | **4** | **80%** | **4** | **86.67%** |

| | | | | | | |
|---|---|---|---|---|---|---|
| Floods posts | | | | | | |
| Flod1.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Flod2.txt | 3 | 60% | 5 | 100% | 5 | 100% |
| Flod3.txt | 2 | 40% | 4 | 80% | 5 | 100% |
| **Average score** | **3** | **80%** | **4** | **86.67%** | **5** | **93.33%** |

**4. Grammatical correctness**

| **Files name** | **10% Summary** | | **20% Summary** | | **30% Summary** | |
|---|---|---|---|---|---|---|
| | **Grade** | **In the %** | **Grade** | **In the %** | **Grade** | **In the %** |
| Protests posts | | | | | | |
| Prot1.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Prot2.txt | 3 | 60% | 3 | 60% | 3 | 60% |
| Prot3.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| **Average score** | **4** | **80%** | **4** | **80%** | **4** | **80%** |
| Droughts posts | | | | | | |
| Drot1.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Drot2.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Drot3.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| **Average score** | **4** | **80%** | **4** | **80%** | **4** | **80%** |
| Sport posts | | | | | | |
| Spot1.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| Spot2.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Spot3.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| **Average** | **5** | **93.33%** | **5** | **93.33%** | **5** | **93.33%** |
| Floods posts | | | | | | |
| Flod1.txt | 3 | 60% | 3 | 60% | 3 | 60% |
| Flod2.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| Flod3.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| **Average score** | **4** | **80%** | **4** | **80%** | **4** | **80%** |

**5. Referential Clarity**

| **Files name** | **10% Summary** | | **20% Summary** | | **30% Summary** | |
|---|---|---|---|---|---|---|
| | **Grade** | **In the %** | **Grade** | **In the %** | **Grade** | **In the %** |
| Protests posts | | | | | | |
| Prot1.txt | 4 | 80% | 5 | 100% | 4 | 80% |
| Prot2.txt | 4 | 80% | 3 | 60% | 4 | 80% |
| Prot3.txt | 5 | 100% | 3 | 60% | 5 | 100% |
| **Average score** | **4** | **86.67%** | **4** | **73.33%** | **4** | **86.67%** |
| Droughts post | | | | | | |
| Drot1.txt | 4 | 80% | 4 | 80% | 5 | 100% |
| Drot2.txt | 3 | 60% | 4 | 80% | 4 | 80% |
| Drot3.txt | 3 | 60% | 4 | 80% | 5 | 100% |
| **Average score** | **3** | **66.67%** | **4** | **80%** | **5** | **93.33%** |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sport posts | | | | | | |
| Spot1.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Spot2.txt | 5 | 100% | 4 | 80% | 5 | 100% |
| Spot3.txt | 5 | 100% | 5 | 100% | 5 | 100% |
| **Average score** | **5** | **93.33%** | **4** | **86.67%** | **5** | **93.33%** |
| Floods posts | | | | | | |
| Flod1.txt | 4 | 80% | 4 | 80% | 4 | 80% |
| Flod2.txt | 5 | 100% | 3 | 60% | 5 | 100% |
| Flod3.txt | 4 | 80% | 5 | 100% | 4 | 80% |
| **Average score** | **4** | **86.67%** | **4** | **80%** | **4** | **86.67%** |

## VII. The Objective Evaluations for posting texts for all experiments

### A. Protests posted texts

| Clustered protests post files | Experiment1 Summary files | No.of totals sentences | #of Overlap sentences between SA&SI | F-measure in (%) |
|---|---|---|---|---|
| At 10% | | | | |
| Prot1.txt | E1Sm10Prot1.txt | 5 | 3 | 60% |
| Prot2.txt | E1Sm10Prot2.txt | 6 | 5 | 83.33% |
| Prot3.txt | E1Sm10Prot3.txt | 5 | 4 | 80% |
| **Average score** | | **5** | **4** | **74%** |
| At 20% | | | | |
| Prot1.txt | E1Sm20Prot1.txt | 11 | 9 | 81.81% |
| Prot2.txt | E1Sm20Prot2.txt | 12 | 10 | 83.33% |
| Prot3.txt | E1Sm20Prot3.txt | 11 | 10 | 90.90% |
| **Average score** | | **11** | **10** | **85.35%** |
| At30% | | | | |
| Prot1.txt | E1Sm30Prot1.txt | 16 | 14 | 87.5% |
| Prot2.txt | E1Sm30Prot2.txt | 19 | 14 | 73.68% |
| Prot3.txt | E1Sm30Prot3.txt | 16 | 16 | 100% |
| **Average score** | | **17** | **14** | **87.07%** |

### B. Droughts posted texts

| Clustered droughts post files | Experiment2 Summary files | No.of totals sentences | Number of Overlap sentences between SA&SI | F-measure in (%) |
|---|---|---|---|---|
| At 10% | | | | |
| Drot1.txt | E2Sm10Drot1.txt | 7 | 5 | 71.42% |
| Drot2.txt | E2Sm10Drot2.txt | 6 | 4 | 66.67% |
| Drot3.txt | E2Sm10Drot3.txt | 6 | 5 | 83.33% |
| **Average score** | | **6** | **5** | **73.81%** |
| At 20% | | | | |
| Drot1.txt | E2Sm20Drot1.txt | 13 | 12 | 92.30% |
| Drot2.txt | E2Sm20Drot2.txt | 12 | 10 | 83.33% |
| Drot3.txt | E2Sm20Drot3.txt | 12 | 9 | 75% |
| **Average score** | | **12** | **10** | **83.54%** |

| | | | | |
|---|---|---|---|---|
| At30% | | | | |
| Drot1.txt | E2Sm30Drot1.txt | 20 | 15 | 75% |
| Drot2.txt | E2Sm30Drot2.txt | 19 | 17 | 89.47% |
| Drot3.txt | E2Sm30Drot3.txt | 18 | 16 | 88.89% |
| **Average score** | | **19** | **16** | **84**% |

### C. Sports posted texts

| Clustered Sports post files | Experiment3 Summary files | Number of totals sentences | Number of Overlap sentences between SA&SI | **F-measure in (%)** |
|---|---|---|---|---|
| At 10% | | | | |
| Spot1.txt | E3Sm10Spot1.txt | 4 | 3 | 75% |
| Spot2.txt | E3Sm10Spot2.txt | 4 | 4 | 100% |
| Spot3.txt | E3Sm10Spot3.txt | 3 | 2 | 66.67% |
| **Average score** | | **4** | **3** | **81**% |
| At 20% | | | | |
| Spot1.txt | E3Sm20Spot1.txt | 8 | 7 | 87.5% |
| Spot2.txt | E3Sm20SPot2.txt | 8 | 6 | 75% |
| Spot3.txt | E3Sm20Spot3.txt | 7 | 7 | 100% |
| **Average score** | | **8** | **7** | **89.25**% |
| At30% | | | | |
| Spot1.txt | E3Sm30Spot1.txt | 13 | 12 | 92.30% |
| Spot2.txt | E3Sm30SPot2.txt | 11 | 9 | 81.81% |
| Spot3.txt | E3Sm30SPot3.txt | 10 | 10 | 100% |
| **Average score** | | **11** | **15** | **91.37**% |

### D. Floods posted texts

| Clustered floods post files | Experiment 4 Summary files | Number of totals sentences | Number of Overlap sentences b/n SA&SI | F-measure in (%) |
|---|---|---|---|---|
| At 10% | | | | |
| Flod1.txt | E4Sm10 Flod1.txt | 6 | 5 | 83.33% |
| Flod2.txt | E4Sm10 Flod2.txt | 4 | 3 | 75% |
| Flod3.txt | E4Sm10 Flod3.txt | 3 | 3 | 100% |
| **Average score** | | **4** | **4** | **86.11**% |
| At 20% | | | | |
| Flod1.txt | E4Sm20 Flod1.txt | 11 | 10 | 90.90% |
| Flod2.txt | E4Sm20 Flod2.txt | 9 | 7 | 77.78% |
| Flod3.txt | E4Sm20 Flod3.txt | 6 | 6 | 100% |
| **Average score** | | **9** | **8** | **89.56**% |
| At30% | | | | |
| Flod1.txt | E4Sm30 Flod1.txt | 17 | 15 | 88.24% |
| Flod2.txt | E4Sm30 Flod2.txt | 13 | 12 | 92.31% |
| Flod3.txt | E4Sm30 Flod3.txt | 9 | 9 | 100% |
| **Average score** | | **13** | **12** | **93.52**% |

## VIII. Sample source codes

**// Similarity between posted texts in a pair of sentences**

```
Package AmharicSimilarity;

Public class AmharicSimilarity {

Public static HashMap<String, Double> map1 = new HashMap<>();

Public static HashMap<String, Double> map2 = new HashMap<>();

Public double SentenceamharicSim (String sentence1, String Sentence2) {

double sim = 0.0;       int i = 0;

String[] words1 = sentence1.split(" ");

String[] words2 = Sentence2.split(" ");

while (i < words1.length - 1) {

if (map1.containsKey(words1[i])) {

doublefr = map1.get(words1[i]);

map1.put(words1[i], fr + 1);              }

else {          map1.put(words1[i], 1.0);          }

i++;       }

int j = 0;

while (j < words2.length - 1) {

if (map2.containsKey(words2[j])) {

double fr = map2.get(words2[j]);

map2.put(words2[j], fr + 1);              }

else {          map2.put(words2[j], 1.0);          }

j++;       }
```

```
        Set s = map1.entrySet  ();
        Iterator it = s.iterator();
int count = 0;String  word = "";      double total = 0.0;
while  (it.hasNext())  {
Map.Entry  ob = (Map.Entry)  it.next();  // System.out.println(ob);
word = ob.getKey().toString();
double  frq = Double.parseDouble(ob.getValue().toString());
if (map2.containsKey(word))  {
double  fr = map2.get(word);
                total = total + 1;          }        }
// the union  all the two sentence divide  the sentence1&2  sum  //similarity(S1,S2)=  match  (S1 n S2)/ (S1 U S2)
return sim;     }
public  double amharicSim(String  word1, String  word2) {
double  sim = 0.0;        int  i = 0;
while  (i < word1.length())  {
if (map1.containsKey(word1.substring(i,  i + 1))) {
double  fr = map1.get(word1.substring(i,  i + 1));
map1.put(word1.substring(i,  i + 1), fr + 1);            }
else {         map1.put(word1.substring(i,  i + 1), 1.0);            }
i++;        }
int  j = 0;        while  (j < word1.length())  {
if (map2.containsKey(word1.substring(j,  j + 1))) {
double  fr = map2.get(word1.substring(j,  j + 1));
```

```
map2.put(word1.substring(j,  j + 1), fr + 1);            }
else {       map2.put(word1.substring(j,  j + 1), 1.0);       }
j++;          }
       Set s = map1.entrySet();
       Iterator it = s.iterator();
int count = 0;    String word = "";
double total = 0.0;
while (it.hasNext()) {
Map.Entry ob = (Map.Entry) it.next();
System.out.println(ob);
word = ob.getKey().toString();
double frq = Double.parseDouble(ob.getValue().toString());
if (map2.containsKey(word)) {
double fr = map2.get(word);
              total = total + 1;          }       }
sim = (total intersection / (map1.size() + map2.size()));
return sim;    } }
```

**#Stemmer coding for Amharic texts based on rules to developed as follow to remove affixes**

```
Public void step1 ( ) {
if (b[k] == ኅ) //-ኅ
{     k -= 1;     }    }
Public void step2 ( ) {
try {       //SUFFIX-5
```

**if (k > 6&&b[k -4]==አ&& b[k - 3]== ች&&b[k - 2]== ኋ&&b[k - 1]==ለ&& b[k]==ሁ){k -= 5; }//-አችኋለሁ**

**else if (k >6&&b[k - 4]==አ&&b[k - 3] == ቸ&& b[k - 2]== ዋ&&b[k - 1]== ለ&& b[k] == ሁ) { k -= 5; }**

**else if (k > 6&& b[k - 4]== ያ &&b[k - 3]== ው && b[k - 2]== ያ&&b[k - 1]== ን&& b[k] == ና){k -= 5; }**

**else if (k >6&& b[k - 4]==አ && b[k - 3] == ቸ&& b[k - 2]== ዋ&&b[k - 1]==ለ && b[k] == ሽ){ k -= 5; }**

**else if (k > 6&&b[k - 4]== ዎ&& b[k - 3] == ቻ&& b[k - 2]== ች&& b[k - 1]== ሁ&& b[k]== ን) {k -= 5; }**

**else if (k > 6&&b[k - 4]== ይ&& b[k - 3] == ታ&& b[k - 2] == ች&& b[k - 1]== ን&& b[k]== ን){k -= 5; }**

**///SUFFIX-4**

**else if (k > 5 && b[k - 3] == ዎ && b[k - 2]== ች && b[k - 1] == ን && b[k] == ም) { k -= 4; }//- ዎችንም**

**else if (k > 5 && b[k - 3] == ዎ && b[k - 2] == ች && b[k - 1] == ና && b[k] == ም) { k -= 4; }**

**else if (k > 5 && b[k - 3] == ዎ && b[k - 2] == ች && b[k - 1] == ን && b[k] == ና) { k -= 4; }**

**else if (k > 5 && b[k - 3] == ዎ && b[k - 2] == ች && b[k - 1] == ና && b[k] == ን) { k -= 4; }**

**else if (k > 5&& b[k - 3] == ው&& b[k - 2]== ያ&& b[k - 1] == ን && b[k] == ን){ k -= 4; }**

**else if (k > 5 && b[k - 3] == ው && b[k - 2] == ይ && b[k - 1] == ን&& b[k] == ና) {k -= 4; }//-ውይንና**

**//Suffix -3**

**else if (k > 3 && b[k - 2] == ና && b[k - 1] == ም && b[k] == ና) { k -= 3; }**

**else if (k > 3 && b[k - 2] == ና && b[k - 1] == ም && b[k] == ን) { k -= 3; }**

**else if (k > 3 && b[k - 2] == ዎ && b[k - 1] == ች && b[k] == ን) { k -= 3; }**

**else if (k > 3 && b[k - 2] == ው && b[k - 1] == ያ && b[k] == ን) {k -= 3; }//-ውያን**

**else if (k > 3 && b[k - 2] == ው && b[k - 1] == ያ && b[k] == ኑ) { k -= 3; }**

**else if (k > 3 && b[k - 2] == ና && b[k - 1] == ን && b[k] == ን) { k -= 3; }//-ናንን**

**else if (k > 3 && b[k - 2] == ና && b[k - 1] == ዉ && b[k] == ና) { k -= 3; } //-ናዉን**

**//Suffix-2**

```
else if (k > 2 && b[k - 1] == ቸ && b[k] == ን) {   k -= 2;     } //-ቸን

else if (k > 2 && b[k - 1] == ን && b[k] == ህ) {   k -= 2;    }

else if (k > 2 && b[k - 1] == ነ && b[k] == ው) {   k -= 2;    }

else if (k > 2 && b[k - 1] == ና && b[k] == ት) {     k -= 2;   }

else if (k > 2 && b[k - 1] == ዋ && b[k] == ት) {   k -= 2; }

else if (k > 2 && b[k - 1] == ዎ && b[k] == ቹ) {    k -= 2; }

else if (k > 2 && b[k - 1] == የ && b[k] == ው) {    k -= 2;   }

else if (k > 2 && b[k - 1] == ዎ && b[k] == ች) {k -= 2;    } //-ዎች

else if (k > 2 && b[k - 1] == ነ && b[k] == ት) {  k -= 2;    } //-ነት

else if (k > 2 && b[k - 1] == አ && b[k] == ዊ) {     k -= 2;    } //-አዊ

else if (k > 2 && b[k - 1] == ያ && b[k] == ዊ) { k -= 2;      } //-ያዊ

else if (k > 2 && b[k - 1] == ነ&& b[k] == ቱ) {   k -= 2;      }//-ነቱ

//SUFFIX  ONE

else if (k > 1 && b[k] == ኝ) {    k -= 1; } //-ኝ

else if (k > 1 && b[k] == ኛ) { k -= 1;} //-ኛ

else if (k > 1 && b[k] == ዋ) {  k -= 1;    }//-ዋ

else if (k > 1 && b[k] == ቹ) {         k -= 1;   } //-ቹ

else if (b[k] == ን) {          k -= 1;       }//-ን

else if (k > 1 && b[k] == ዊ) {  k -= 1;   } //-ዊ

else if (k > 1 && b[k] == ች) {    k -= 1;   } //-ች

} Catch (ArrayIndexOutOfBoundsException  e) {

System.out.println  ("Array  exception");
```

**} }**

**Public void step3 ( ) { //Prefix-4**

**if (b[l] == አ && b[l + 1] == ስ && b[l + 2] == ክ && b[l + 3] == ት) { l += 4; }//አስክት-**

**if (b[l] == አ && b[l + 1] == ስ && b[l + 2] == ክ && b[l + 3] == ን) {l += 4; }//አስክን-**

**if (b[l] == አ && b[l + 1] == ን && b[l + 2] == ደ && b[l + 3] == የ) { l += 4; }//አንደየ-**

**if (b[l] == አ && b[l + 1] == ን && b[l + 2] == ደ && b[l + 3] == ሚ) { l += 4; }//አንደሚ-**

**if (b[l] == አ && b[l + 1] == ን && b[l + 2] == ድ && b[l + 3] == ን) { l += 4; }//አንድን-**

**//PREFIX 3**

**if (b[l] == አ && b[l + 1] == ን && b[l + 2] == ድ) { l += 3; }//አንድ-**

**else if (b[l] == አ && b[l + 1] == ን && b[l + 2] == ደ) { l += 3; }//አንደ-**

**else if (b[l] == በ && b[l + 1] == እ && b[l + 2] == ነ) { l += 3; }//በእነ-**

**else if (b[l] == ከ && b[l + 1] == እ && b[l + 2] == ነ) { l += 3; }// ከእነ-**

**//Prefix 2**

**else if (b[l] == አ && b[l + 1] == ይ) { l += 2; }//አይ-**

**else if (b[l] == በ && b[l + 1] == ሚ) { l += 2; }//በሚ-**

**else if (b[l] == ከ && b[l + 1] == ነ) { l += 2; }//ከነ-**

**else if (b[l] == ከ && b[l + 1] == የ) { l += 2; } //ከየ-**

**else if (b[l] == ስ && b[l + 1] == ለ) { l += 2; }/ስለ-**

**//PREFIX 1**

**else if (b[l] == አ) { l += 1; }//አ**

**else if (b[l] == ሲ) { l += 1; }//ሲ-**

**else if (b[l] == ከ) { l += 1; }//ከ-**

```
 else if (b[l] == ʌ) { l += 1; }//ʌ-
else if (b[l] == ʔ) {l += 1;} //ʔ-
    }   Public  void step4 ( ) {
        Boolean found;
        for (int w = l; w <= k; w++) {
           found  = false;
           for (i = 0; i < UNICODECHARS.length;  i++) {
              for (j = 0; j < UNICODECHARS[i].length;  j++) {
                 if (b[w]  == UNICODECHARS[i][j)  {
                    b[w]  = UNICODECHARS[i][5];
                    found  = true;            break;      }
                 if (found)  {     break;    }          }        }
           Continue;        }      }
  Public  void stem ( ) {        k = i - 1;         l = 0;
        if (k > 1) {  step1 ( );  step2 ( );    step3 ( );  step4 ( );   step5 ( );        }
        i_end = k + 1 - l;
        i = 0;         i_beg = l;       }
```