ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

UNSUPERVISED AMHARIC NEWS CLASSIFICATION

LAKACHEW YAYEH

JULY 2011

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

UNSUPERVISED AMHARIC NEWS CLASSIFICATION

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

LAKACHEW YAYEH

JULY 2011

ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF INFORMATION SCIENCE

UNSUPERVISED AMHARIC NEWS CLASSIFICATION

By

LAKACHEW YAYEH

Name and signature of Members of the Examining Board

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

Declaration

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

_____

Date

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

_____

Advisor

CHAPTER ONE

INTRODUCTION

1.1 Background

As a result of the advancement of technologies such as the Internet and the World Wide Web (WWW), the amount of data generated in different private and public organizations is increasing from time to time. This huge collection of data is very important for its users in their day to day activity and for their success in today's information society. However, with the rapid growth of information resources (web pages, news corpora, e-mails, e-books, journals, online databases, etc), the task of finding relevant information becomes very complicated and challenging. To address these challenges, many data mining techniques such as document classification and document clustering have been introduced to structure such huge collection of documents. This is because finding or retrieval of relevant information from huge collection of documents is possible when the collection is organized in a systematic way or structure.

People make their own judgment to classify things in their every day life – they classify things based on similarities or likeliness of color, size, concept, ideas, and subject (Koller and Sahami, 1997).

Humans use classification techniques to classify text documents into categories. For instance, while reading a news story, we are rapidly able to assess whether it belongs to the domain of finance, politics or sports. However, manual classification of huge collection of documents to different categories is significantly more expensive in terms of time, cost and labor intensiveness. As a result, automatic text categorization has become one of the key techniques for handling and organizing textual document collection. This classification enables to increase efficiency of search, and accuracy of retrieval of documents relevant to the needs of the end user.

Automatic text classification can be done using the following two approaches (Sebastiani, 2002; Rasmussen, 1992): classification (supervised approach) and clustering (unsupervised approach). Text clustering is the automatic identification of a set of natural categories and the grouping of documents under them. Clustering is the operation by which similar objects are grouped together in an unsupervised manner (Jain, Murty and Flynn, 1999). In unsupervised approach, there is no need for human intervention or labeled documents at any point in the whole process. In a clustering technique, the properties and membership (composition) of classes is not known in advance.

Text classification, on the other hand, is the automatic assignment of documents to a predefined set of categories. In this approach, pre-defined category labels are assigned to documents based on the likelihood suggested by a training set of labeled documents. In a classification problem, the number of classes and their properties are known in advance, and documents are assigned to these classes before the classification has been done.

Nowadays, news items are produced every day in digital devices and organized in some order (Rennie, 2001). The amount of Amharic news item produced is also increasing from time to time which becomes a difficult task for news agencies to classify and manage such huge collection of news item manually. To alleviate this problem, a number of researches such as Zelalem (2001), Surafel (2003), Yohannes (2007), Worku (2009), Alemu (2010) and Zeleke (2010) were conducted in Amharic text documents using supervised text classification techniques.

1.2 Statement of the Problem and Its Justification

Due to the huge size, high dynamics, and large diversity of the web and of organizational intranets, it has become a very challenging task to find truly relevant content for some user or purpose (Beil, Ester and Xu, 2002). As a result, the need to classify information resources (newspapers, journals, magazines, thesis and dissertations, etc) has become an important issue as the production of such resources increase dramatically from time to time.

Currently, there are numerous electronic documents produced and stored in Amharic in different organizations. For instance, the Ethiopian News Agency (ENA) produces news items both in Amharic and English and releases the news on its own website. At present, the agency uses the ENASoft software for the management of news. However, the classification of news to their respective categories is done manually. Currently, there are 12 major categories available in ENA. Using manual classification is a challenging task for this large number of classes. To address this challenge, a number of researches were conducted using supervised approaches as discussed before. However, supervised text classification techniques have the following limitations.

First, supervised learning algorithms require a large, often prohibitive, number of labeled training documents for the accurate learning (Ko and Seo, 2000). Since the application area of automatic text categorization has diversified from articles and web pages to electronic mails and newsgroup postings, it is a difficult task to create training data for each application area (Nigam et al., 2000). According to Nigam et al. (2000), in supervised text classification, obtaining training labels is expensive in huge volume of document collection. This is because in supervised text classification labeling of training data is done by a person manually and this is a time consuming, cumbersome and error prone process.

Second, supervised learning algorithms require the use of a training set in which each element has already been correctly categorized. However, whenever a class or category is created or modified it is a must to train the classifier using sample documents taken from the newly created or modified classes. According to Veeramachaneni, Sona and Avesani (2005), this approach is impractical because of the necessity to provide labeled training examples whenever taxonomy is created or modified. The solution here is to automatically organize a given collection of documents according to the new category specification with out the need for training sample documents.

Third, Ozgur (2004) concludes that unsupervised text classification techniques perform better in terms of time complexity and the quality of clusters produced as compared to supervised techniques. This shows that the overall similarities of the clustering solutions obtained by the unsupervised techniques are higher than the supervised ones.
Fourth, supervised text classification algorithms are expensive and time consuming to organize documents in to their categories. As Nigam et al. (2000) suggests, text clustering is a useful and inexpensive way to organize vast text repositories into meaningful topic categories. Furthermore, text clustering offers a low cost alternative to supervised classification, which relies on expensive and difficult handwork to label training data (Massey, 2004).

Lastly, document classification is learning from examples and document clustering is learning from observation (Han and Kamber, 2001). Document clustering reveals the inherent organizational structure of the document corpus while document classification imposes a predefined organization scheme to the corpus (Yang, 2004). From this we can understand that the task of news classification is not a prediction based on training data set rather it is a description based on the internal relationships or similarity of news items. Therefore, it seems logical to apply unsupervised machine learning techniques for news classification.

In general, document clustering or unsupervised classification is an alternative to supervised classification with low cost and best quality of clustering solution. Therefore, this thesis concentrates on Amharic text clustering which is an unsupervised task where no human intervention at any point in the whole process and no labeled documents are provided.

Hence, the aim of this research is to explore the use of unsupervised text classification algorithms for grouping a large, heterogeneous collection of Amharic news text corpus into their natural groupings with low cost and best quality of clustering solutions.

To this end, the present study attempts to investigate and find solutions to the following research questions.

- Could an unsupervised learning approach using clustering algorithms perform better for automatic Amharic text news classification?
- What is the effect of the number of clusters and the size of documents used on the performance and efficiency of clustering algorithms?
- What is the effect of the number of clusters used on the performance and efficiency of clustering algorithms using the same data set?
- Which clustering algorithm performs best for Amharic text news clustering?
- To what extent categories revealed by the clustering algorithms match with the existing categories used by ENA? Why such variations if any?

1.3 Objective of the Study

The general and specific objectives of this research work are described here under.

1.3.1 General Objective

The main objective of this research is to design automatic Amharic text document classification using unsupervised machine learning approaches to cluster or group huge collection of Amharic text news to their natural grouping.

1.3.2 Specific Objectives

The specific objectives of this research are:

- To review literature on the concepts, techniques and tools of text classification particularly in the area of unsupervised learning.
- To preprocess the unstructured news items or documents to make them ready for the clustering process.
- To select suitable clustering technique for Amharic text news clustering.
- To evaluate the performances of clustering algorithms for Amharic text news clustering.
- To recommend further research for future work(s) in the area of automatic Amharic text classification.

1.4 Methodology

The methodology used in this study is the knowledge discovery in text (KDT) approach which is recommended by Karanikas et al. (2002). KDT is a multi-step process, which includes all the tasks from the gathering of documents to the visualization of the extracted information. In this research, the three phase KDT process was used to achieve the above stated general and specific objectives.

1.4.1 Literature Review

To understand the different approaches of automatic text classification and the methods used for document preprocessing, relevant literatures (books, journal articles, research works, materials on the Internet, etc) were reviewed. Interview and document analysis were also made to further understand the manual classification system of news in ENA.

1.4.2 Collection of Relevant Documents

The data source selected for this study is Ethiopian News Agency (ENA). As to the researcher knowledge, there is no standard Amharic news corpus ready for text classification task by other agencies and hence ENA was selected as a data source. An attempt was also made to use a corpus which has all news categories.

1.4.3 Document Preprocessing

One of the basic steps in this study is preprocessing of the unstructured text documents or news items. Document preprocessing is a very important step in text classification or clustering, because irrelevant and redundant features often degrade the performance of classification algorithms both in speed and classification accuracy and also its tendency to reduce over fitting (Khan et al., 2009).

Hence, document preprocessing activities were done to increase the performance of clustering algorithms both in speed and classification accuracy by removing irrelevant and redundant features of the Amharic text documents. Routines were written using Python 3.1 to tokenize, normalize, remove stop words and stem the documents.

1.4.4 Document Clustering Tools and Techniques

The main objective of clustering techniques is to compute a classification: The objects of one cluster or group should be similar to each other and objects of different clusters should be dissimilar. There are different text document clustering algorithms that have been proposed in the literatures. According to Steinbach, Karypis and Kumar (2000), the agglomerative hierarchical clustering algorithm and k-means algorithm are the two clustering techniques that are commonly used for document clustering.

Hence, in this study both the partitioning and the agglomerative hierarchical clustering algorithms were used for clustering purpose. From partitioning algorithms, the incremental k-means and its variant the bisecting k-means algorithms were used. This is because as discussed in Steinbach, Karypis and Kumar (2000) and Kumar et al (2005) the incremental k-means performs better than the standard version. i.e., incremental updates of centroid were more effective and the clustering results produced are with better overall similarity and lower entropy. In addition, the incremental version of K-means is also advocated by many researchers for text document clustering as stated in Steinbach, Karypis and Kumar (2000). Furthermore, the problems of the batch version of k-means can also be solved in the incremental version of k-means (Kumar et al., 2005).

The clustering software used in this study is gCLUTO which is a graphical interface to CLUTO (CLUstering TOolkit). We used the tool in this study because it is freely available and the software implements the clustering algorithms that the researcher wants to use in this study. Furthermore, gCLUTO offers improvements over existing tools by providing the following features that make clustering practical for a wide variety of applications (Rasmussen and Karypis, 2004).

1. gCLUTO provides an array of clustering algorithms and options through the use of the CLUTO clustering library as stated in Karypis (2003). The CLUTO library provides highly optimized implementations of agglomerative, k-means, and graph clustering, especially in the context of sparse high-dimensional data.
2. gCLUTO helps the user sort through the algorithm options and resulting data files by providing a intuitive graphical interface. Following the paradigm of most development tools, gCLUTO uses the concept of a "project" in order to organize the user's various datasets, clustering solutions, and visualizations.
3. gCLUTO provides both standard statistics and unique visualizations for interpreting clustering results. Therefore, additional effort has gone into visualizations that can facilitate analysis and comparisons.
4. The software package has been designed to suit a wide range of users from different problem domains that may or may not be knowledgeable about the subtleties of clustering.

1.4.5 Clustering Evaluation Techniques

In unsupervised classification algorithms, there are two types of measures to evaluate cluster quality, internal quality measures and external quality measures (Steinbach, Karypis and Kumar, 2000). Hence, in this study, both the internal and external evaluation measures were used to evaluate the performances of unsupervised learning algorithms for Amharic text document clustering.

1.5 Scope and Limitations of the Study

The scope of this study is limited to investigate the possibility of designing Amharic news text classification system using unsupervised machine learning approach. In this study, only the three clustering algorithms: incremental K-means (direct k-way), bisecting k-means (k-way through repeated bisection) and the agglomerative clustering algorithms were used. An evaluation was also made to compare their performance in the classification of Amharic text news. The study is limited only to classification of text

news items from ENA. HTML documents, image documents and others were not considered in this study.

According to the interview made with Ato Debebe, currently there are 12 major news categories in ENA. However, only 10 of these categories have predefined documents and the remaining 2 categories do not have pre-defined documents. For this reason only the 10 categories were considered to conduct the experiments.

1.6 Application and Significance of the Study

Document classification may appear in many applications: automatic indexing for Boolean information retrieval systems, document organization, text filtering, news monitoring, hierarchical categorization of web pages and word sense disambiguation (Sebastiani, 2002). One application of clustering is the analysis and navigation of big text collections such as Web pages. The basic assumption, called the cluster hypothesis, states that relevant documents tend to be more similar to each other than to non-relevant ones. If this assumption holds for a particular document collection, the clustering of documents based on the similarity of their content may help to improve the search electiveness (Carrasco, 2007).

Furthermore, one of the potential applications of document classification is text document classification in news agencies (Manning, Raghavan and Schütze, 2009). According to Hotho et al (2005), text classification in news agencies using text mining systems offers more consistent and faster annotation of news articles as compared to human annotators.

Hence, the findings of this study can be used to support the manual classification system of ENA to offer more consistent and faster classification of a large number of text news items. This would also facilitate the works of the agency by assisting journalists in producing timely and accurate news. In addition, this study can be extended for similar organizations. Furthermore, the different issues discussed and the results obtained in this

study can give an overview of automatic text document classification and can provide several starting points for further studies.

1.7 Thesis Organization

This thesis is organized into six chapters: Chapter 1 - Introduction; Chapter 2- Literature Review; Chapter 3 - Methodology; Chapter 4 – The Amharic Language and its Writing system; Chapter-5 Experiment and Performance Evaluation and Chapter 6 - Conclusion and Recommendations.

The first chapter contains the background, statement of the problem, objectives of the study, methodology, scope and applications of the study. Chapter 2 describes the different approaches of text classification, preprocessing and representation of documents, overview of the different clustering algorithms and their evaluation techniques. Chapter 3 provides the details of Amharic writing system.

The methodology adopted, including the document data set used, document preprocessing activities and the different clustering and evaluation techniques used in this study are presented in chapter 4. Chapter 5 presents the experimental results and findings of the study. In chapter 6, the works accomplished and the findings of this study are summarized and finally recommendations are suggested for further research.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

The advancement of technologies such as the Internet and the WWW accelerated the growth of document collections from time to time. We are living in an information age where there is an exponential growth in the volume of documents and information has a great value in our day to day life. However, with the increasing availability of huge collection of documents, the process of finding relevant information is becoming a difficult task. Searching through a large amount of collection is a challenge and can bring great loss in the form of productivity waste (workers spend about 65 percent of their time searching for information needed to complete their work (Eiring, 2002)); missed opportunities (failure to discover patterns and trends); and mismanaging or lack of managing knowledge (Zaghloul, 2005).

As the collection of documents increases, the task of automatic classification of documents became the key technology for organizing large collection of documents to provide the user with more relevant information. It is believed that grouping similar documents together into clusters will help the users find relevant information quicker, and will allow them to focus their search in the appropriate direction (Hammouda, 2001). Due to this fact, a large set of documents are organized into categories using different text classification approaches.

In this chapter the different approaches of text classification, document preprocessing and representation methods, the different unsupervised machine learning algorithms and evaluation techniques used to measure the quality of clustering solutions are discussed.

2.2 Text Classification Approaches

Text classification is the act of dividing a set of input documents into two or more classes where each document can be said to belong to one or multiple classes (Sebastiani, 2002). Text classification approaches are used as a tool to help people find, filter and manage huge collections of resources such as digital libraries, news sources and internal data of companies which are growing through time.

According to Blumberg and Atre (2003), there are four main approaches of text classification: manual classification, rule-based classification, supervised learning and unsupervised learning. Figure 2.1 shows the level of automation of these four approaches of text classification with their respective cost and control requirements.

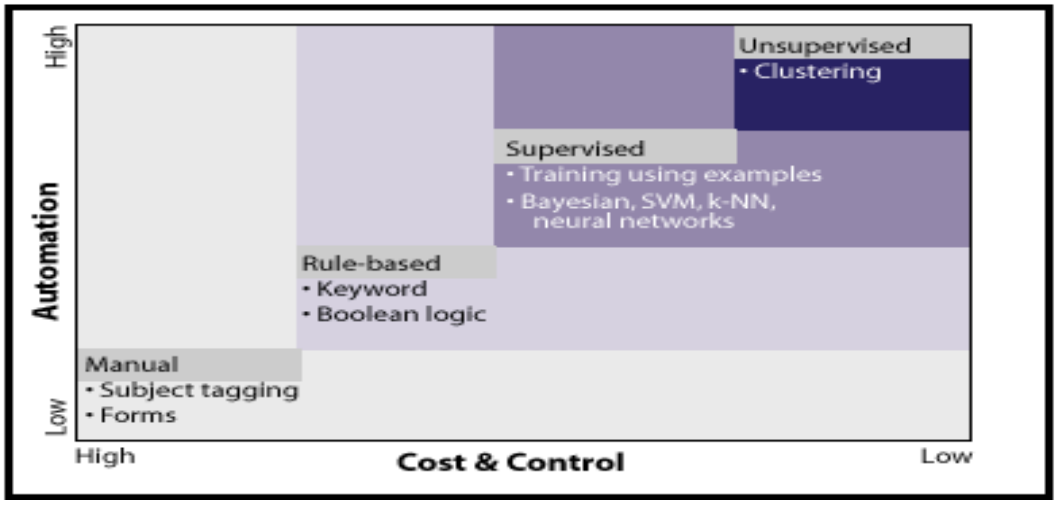

Figure 2. 1: Text classification approaches

2.2.1 Manual Classification

In manual classification, each document is assigned to one or more categories manually by domain experts who are knowledgeable and familiar about the category structure. It is mostly used in library and technical collections as well as in call centers and form-processing environments.

Even though domain experts occasionally disagree on how to categorize documents, manual classification can achieve a high degree of accuracy. However, it is more labor-

intensive, time consuming and most costly than automated techniques and thus its applicability is limited especially for very large document collections.

2.2.2 Rule-based Classification

In rule based classification, rules that define those categories are formulated manually and indexed. Keywords and Boolean expressions are used to classify document. This classification approach is used when a few words can sufficiently describe a category. For example, E-mail systems which typically provide rule based methods for routing messages to specific mailboxes. The e-mails are routed based either on the sender's name or the occurrences of specific words in the subject line.

Although this approach is effective for a limited number of categories with small document sets, defining rules is expensive in for large document sets with many categories and thus difficult to apply for large-scale classification systems.

2.2.3 Supervised Learning

This approach is similar to rule based classification, but the classification rules or the models are automatically constructed from a sample set of pre-classified documents. This classification method first analyses the statistical occurrences of each concept in the training documents and then constructs a model or classifier for each category that is used to classify documents (testing sets) automatically. In supervised learning, the classification is seen as supervised learning from training examples and the supervision took place when the data (observations, measurements, etc) are labeled with predefined categories. However, assigning of documents to predefined categories is done manually which is tedious and time consuming.

2.2.4 Unsupervised Learning

In an unsupervised learning there is no need to provide either the classification rules or the sample documents as a training set. The algorithms identifies a group or clusters of related documents based on their content similarities. This approach is commonly referred to as

clustering and it eliminates the need for training sets since it does not require a predefined category structure.

As shown in Figure 2.1 above, as the automation of the classification system moves from manual to unsupervised classification approach, the cost and control of the classification task always decreases. This shows that the unsupervised approach of text classification (clustering) perform the best clustering solutions with minimum cost and control as compared to the other three approaches.

2.3 Unsupervised Text Classification

Unsupervised classification, also known as clustering, is a process through which objects are classified into meaningful groups called clusters based on the level of similarity between the instances in a certain group, without any prior information (Carrasco, 2007). These sub-groups are called clusters, and hence the name “Clustering”. Document or text clustering is a subset of the larger field of data clustering, which borrows concepts from the fields of information retrieval (IR), natural language processing (NLP), and machine learning (ML) (Nicholas, Andrews and Edward, 2007).

In document clustering, a given set of documents are partitioned into a predetermined or an automatically derived number of clusters, such that the documents assigned to each cluster are more similar to each other than the documents assigned to different clusters. The algorithms’ goal is to create clusters that are coherent internally, but clearly different from each other. In other words, documents within a cluster should be similar and documents in one cluster should be dissimilar from documents in other clusters.

The process of clustering aims to discover natural groupings, and thus present an overview of the classes (topics) in a collection of documents. In the field of artificial intelligence, this is known as unsupervised machine learning. No supervision means that there is no human expert who has assigned documents to classes (Manning, Raghavan and Schütze, 2009). In addition, in a clustering problem, the number, properties and

membership (composition) of classes is not known in advance (Nicholas, Andrews and Edward, 2007). In clustering, it is the distribution and makeup of the data that will determine cluster membership. i.e., the result (the clustering, the partition) is based solely on the similarity between the documents (via the document representation) and the clustering algorithm.

2.4 Document Preprocessing and Representation

To apply the method of clustering or classification it is initially necessary to preprocess the original text documents using different document processing steps. Document preprocessing is a very important step in text clustering or classification, because irrelevant and redundant features often degrade the performance of classification algorithms both in speed and classification accuracy (Khan et al., 2009). In addition to document preprocessing, document representation is also one of the most important factors that influence the performance of clustering algorithms (Jain, 2008). According to Jain (2008), if the representation (choice of features) is good, the clusters are likely to be compact and isolated.

2.4.1 Document Representation

Document representation is the final task in document processing. After passing through a number of document preprocessing activities, the raw text document should be transformed into a representation that can be easily processed and analyzed by using machine learning techniques.

The representation of a set of documents as vectors in a common vector space is known as the vector space model and is fundamental to a host of information retrieval operations ranging from scoring documents on a query, document classification and document clustering (Manning, Raghavan and Schütze, 2009). The vector space model is the most widely used approach to represent textual documents as compared to the Boolean model and others (Amine, Elberrichi and Simonet, 2010). It is a model for representing the content of texts. According to Rosell (2006), Hammounda (2001) and Beil, Ester and Xu

(2002), most document clustering methods use the vector Space model to represent document objects.

In this model text documents are represented by a numerical vector obtained by counting the most relevant features present in the text after document preprocessing. This numerical value stored in a vector representation is used to define the importance of a word in representing the content of the given document. As discussed in Amine, Elberrichi and Simonet (2010), all document $\mathbf{d_j}$ will be transformed into a vector:

$$\mathbf{d_j = (w_{1j}, w_{2j}, ..., w_{|T|j})}$$

where T is the whole set of terms (or descriptors) which appear at least once in the corpus (|T| is the size of the vocabulary), and w_{kj} represents the weight (frequency or importance) of the term t_k in the document d_j. The documents are represented using a document - term matrix as shown in Table 2.1(Amine, Elberrichi and Simonet, 2010).

Documents	Terms or Descriptors				
D1	W11	W21	W31	…	Wj1
D2	W12	W22	W32	…	Wj2
…	…	…	…	…	…
Dm	D1m	D2m	D3M	…	Wjm

Table 2. 1: Document-term matrix

A number of text representation methods such as “bag of words”, “bag of phrases”, “ontology or concept based” have been introduced within the framework of the vector space model (Amine, Elberrichi and Simonet, 2010). However, in most existing document clustering algorithms, documents are represented using the vector space model which treats a document as a bag of words (Thangamani and Thangaraj, 2010).

According to Ozgur (2004), bag of words representation is the most frequently used method for defining terms and it is computationally more efficient than the phrase representation. In a bag of words representation a term is defined as a distinct single word and it transform texts into vectors where each component represents a word. This

representation of texts excludes any grammatical analysis and any irrelevant and redundant features of the original text through document preprocessing to make them more understandable to the learning algorithms.

It has been hypothesized that clustering accuracy can be improved by encoding word order information, resulting in a number of alternative document models such as phrase based models (Nicholas, Andrews and Edward, 2007) . Consider the phrases “the dog chased a cat" and “the cat chased a dog." Although their vector representation is identical, d = {chase; cat; dog}, the meaning is obviously different. The generation of legible cluster labels is an important motivation for phrase-based models (Nicholas, Andrews and Edward, 2007). The vector space model describe each cluster with its most significant terms (for example, the three most significant), while Phrase-based models can naturally describe clusters by phrases, and it is generally agreed that these are more descriptive of the cluster contents.

2.4.2 Documents Preprocessing

All methods of text clustering require several steps of preprocessing of the data (Beil, Ester and Xu, 2002). These document preprocessing steps transform the raw text documents into a representation suitable for applying the learning algorithms.

Document Preprocessing consists of steps that take a plain text document as input and output a set of tokens (which is single term or word in this case) to be included in the vector model. According to Nicholas, Andrews and Edward (2007) and Manning, Raghavan and Schütze (2009), these steps typically consist of: tokenization, normalization, stemming, removal of stop words and term weighting.

Tokenization: A document is treated as a string, and then the sentences are partitioned into a list of individual tokens, typically words. Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens,

perhaps at the same time throwing away certain characters, such as punctuation marks (Manning, Raghavan and Schütze, 2009).

Normalization (equivalence classing of terms): Term normalization is the process of removing superficial differences that prevent words in different forms that convey the same information, to be matched (Rossel, 2009). More interesting is how to treat the morphology of the language(s) used such as capitalization and compound words in a suitable and consistent way. For instance, if the tokens “anti-democratic” and “antidemocratic” are both mapped into the term “antidemocratic”, in both the document text and queries, then searches for one term will retrieve documents that contain either.

Stemming: is **t**he process of reducing words to their base form or stem. For example, the words “connected, "connection", “connections" are all reduced to the stem “connect." The goal of stemming is to reduce derivationally related forms of a word to a common base form.

Stop word removal: A stop word is defined as a term which is not thought to convey any meaning as a dimension in the vector space (i.e. without context). They are words which appear frequently and are insignificant words in discriminating one document from another. A typical method to remove stop words is to compare each term with a collection of known stop words.

2.4.3 Term weighting

Different term weighting approaches have been proposed in the literature to measure the weight of the ‘importance’ of representative terms in a document. As Ozgur (2004) discussed most of the terms weighting approaches are based on the following observations:

- The relevance of a word to the topic of a document is proportional to the number of times it appears in the document.

- The discriminating power of a word between documents is less, if it appears in most of the documents in the document collection.

According to Baeza-Yates and Ribeiro-Neto (1999), the three common term weighing methods used to show the importance of a term are Term Frequency (TF), Inverse Document Frequency (IDF) and Term Frequency by Inverse Document Frequency (TF*IDF).

Term Frequency Weighting (TF): In this method, the weight of a term in a document is equal to the number of times the term appears in the document, i.e. to the raw frequency of the term in the document. The value of TF is zero if the term does not appear in document. The weight of term i in document d, is given by:

$$\boldsymbol{w_i = tf_i} \qquad (\mathbf{2.1})$$

where, tf_i is the raw frequency of term i in document d;

Inverse document frequency (IDF): is a measure of the general importance of the term. IDF decreases as the number of documents in which the term occurs increases in a given collection. So terms that are found only in small documents receive a higher weight.

$$IDF = \log_2 \frac{N}{N_i} \qquad (2.2)$$

where N is the total number of documents in the collection, N_i is the number of documents in which term i occurs.

Term Frequency × Inverse Document Frequency Weighting (TF*IDF): TF weighting do not consider the frequency of the term throughout all the documents in the document corpus. TF*IDF weighting is the most common method used for term weighting that takes into account this property. In this approach, the weight of term i in document d is assigned proportionally to the number of times the term appears in the document, and in inverse proportion to the number of documents in the corpus in which the term appears.

$$W_i = tf_i . \log_2 \left(\frac{N}{N_i}\right) \qquad (2.3)$$

where, N is the total number of documents in the document corpus; Ni is the number of documents in the corpus where term i appears;

TF*IDF weighting approach weights the frequency of a term in a document with a factor that discounts its importance if it appears in most of the documents, as in this case the term is assumed to have little discriminating power. In this approach, to account for documents of different lengths each document vector is normalized so that it is of unit length.

In text clustering, a text document may partially match many categories. So we should find the best matching category for the text document. According to Khan et al. (2009), the TF*IDF approach is commonly used to weight each word in the text document according to how unique it is and hence this approach captures the relevancy among words, text documents and particular categories. Therefore, the TF*IDF weighing method is used in this study to find the best matching cluster for the documents.

2.4.4 Dimension Reduction

Document representation using bag of words create a problem in that the feature space becomes very high dimensional which imposes a big challenge on the performance of clustering algorithms. The computational complexity of any operations with such feature vectors will be proportional to the size of the feature vector (Yang and Pedersen, 1997). In addition, it has been shown that some specific words in specific languages only add noise to the data and removing them from the feature vector actually improves classification performance (Yang and Pedersen, 1997).

Feature selection not only reduces the high dimensionality of the feature space, but also provides better data understanding, which improves the clustering result (Sebastiani, 2002). Hence it is important to reduce the size of the feature vector by selecting only relevant terms that leads to better clustering performance. According to Liu and Motoda (1998) and Sebastiani (2002), the set of feature reduction operations involves a

combination of three general approaches: Stop words removal, Stemming and Statistical filtering.

Statistical filtering practices are used to select those words that have higher statistical significance. According to Sebastiani (2002), there are various statistical filtering approaches applied for dimensionality reduction in document classification. Among feature selection methods Document Frequency thresholding (DF) and Information Gain (IG) are frequently used methods for choosing a subset of the available features (Krishnakumar, 2006). Furthermore, IG is one of statistical filtering method for supervised techniques where the class label information is required for each document. Hence, IG is not suitable to be applied in unsupervised classification and only DF is used in this study.

In DF approach, the document frequency of each unique term is computed and terms whose document frequencies are less than a predetermined threshold are eliminated. The basic assumption behind this technique is that rare terms are either non-informative for document clustering or they do not have much weight in global performance. This technique can also lead to improvement in classification accuracy in case rare terms are noise terms. However, DF is usually not used for aggressive term elimination because there is another widely accepted assumption in information retrieval that low-DF terms are distinctive and thus relatively informative and for this reason should not be removed aggressively (Yang and Pedersen, 1997).

2.4.5 Document Similarity Measure

A key factor in the success of any clustering algorithm is the similarity measure adopted by the algorithm (Hammounda, 2001). In order to group similar text documents a proximity measure has to be used to find which documents or clusters are similar. The similarity or dissimilarity between documents is measured by a function calculating the distance between the vectors of these documents. Two close documents according to this distance are regarded as similar in content.

There are a number of possible measures for computing the similarity between documents such as Euclidian distance and Manhattan distance, but according to Steinbach, Karypis and Kumar (2000), Ozgur (2004) and Hammounda (2001), the most common similarity measure that is used in document clustering is the cosine measure, which is applied in this study. To measure the similarity between two documents d_1 and d_2 represented in the vector space model, the cosine measure is defined by the cosine of the angle between the two vectors as:

$$cosine(d_1, d_2) = \frac{d_1 \bullet d_2}{\|d_1\| \|d_2\|} \quad (2.4)$$

where • indicates the vector dot product and ||d|| is the length of vector d.

The cosine value is 1 when two documents are identical and 0 if there is nothing in common between them. The larger cosine value indicates that these two documents share more terms and are more similar.

2.5 Machine Learning Algorithms

Machine Learning (ML) can be considered as a sub-category of the broader field of Artificial Intelligence (AI). Michie (1991) defined ML as; "a learning system that uses sample data to generate an updated basis for improvement on subsequent data from the same data source and express the new basis in intelligible symbolic form." Simon (1983) also defined ML as; "Learning denotes changes in the systems that are adaptive in the sense that they enable the system to do the same task or tasks drawn from the same population more effectively the next time."

According to Aha (1995), Machine Learning algorithms can be categorized into three sub-types; unsupervised, supervised and reinforcement learning.

1. Supervised learning algorithms: These input (or request) instances described by a set of attributes where at least one attribute is a predefined class attribute. This special attribute can be either nominal or numeric-valued. Most work in supervised learning in ML has concerned classification, which concerns learning concept descriptions described by nominal-valued target attributes.

2. Unsupervised learning algorithms: These have the same inputs as supervised learning algorithms, but have no special target attribute. In some sense, the target attribute's values are all missing and the objective of unsupervised learning is to cluster the given instances according to their presumed class.
3. Reinforcement learning algorithms: These are distinguished from the other approaches as they rely on learning from direct interaction with the environment but do not rely on explicit supervision or complete models of the environment. Learning is performed using the concept of trial and error search, with delayed rewards.

2.6 Unsupervised Learning Algorithms for Document Clustering

Many different unsupervised learning (clustering) algorithms have been proposed and tried for document clustering (Rosell, 2006). Each clustering technique adopts a certain strategy for detecting the grouping in the data. However, as discussed in Hammouda (2001) most of the reported methods have some common features as:

- there is no explicit supervision effect
- patterns are organized with respect to an optimization criterion
- they all adopt the notion of similarity or distance

Hammouda (2001) also mentioned that some algorithms, however, make use of labeled data to evaluate their clustering results, but not in the process of clustering itself.

Many of the clustering algorithms were motivated by a certain problem domain. Accordingly, there is a variation on the requirements of each algorithm, including data representation, similarity measures, and running time. Each of these requirements more or less has a significant effect on the usability of any algorithm.

Clustering algorithms can be broadly divided into two groups: partitional clustering and hierarchical clustering algorithms (Jain, 2008). According to Steinbach, Karypis and Kumar (2000), k-means and agglomerative hierarchical clustering algorithms are the two clustering techniques that are commonly used for document clustering.

2.6.1 Partitional Clustering Techniques

These methods are also called flat clustering. In contrast to hierarchical clustering techniques, partitional clustering techniques create a one-level (un-nested) partitioning of the data points. If K is the desired number of clusters, then partitional approaches typically find all K clusters at once rather than bisecting a cluster to get two clusters or merge two clusters to get one.

There are a number of partitional clustering techniques, but the K-means algorithm is the most widely used in document clustering (Steinbach, Karypis and Kumar, 2000). Accordingly, the k-means algorithm and its variant bisecting k-means are discussed in the following sections.

2.6.1.1 K-Means Clustering

The most known class of partitional clustering algorithms are the k-means algorithm and its variants. Even though K-means was first proposed over 50 years ago, it is still one of the most widely used algorithms for clustering. Ease of implementation, simplicity, efficiency, and empirical success are the main reasons for its popularity (Jain, 2008).

K-means create a one level partition of data objects into a user specified number of clusters (K), which are represented by their centroid, which is usually the mean of a group of points, and is typically applied to objects in a continuous n-dimensional space. The mean or the centroid almost never corresponds to the actual data point and it is a cluster representative of the objects in the cluster. The centroid vector **c** of cluster *C* of documents is defined as follows:

$$c = \frac{\sum_{d \in c} d}{|C|} \tag{2.5}$$

So, c is obtained by averaging the weights of the terms of the documents in C. Analogously, the similarity between a document d and a centroid vector c by cosine similarity measure is defined as

$$\cos(d, c) = \frac{d.c}{\|d\|\|c\|} \tag{2.6}$$

Note that although documents are of unit length, centroid vectors are not necessarily of unit length.

As discussed by Berkhin (2002), there are two versions of k-means algorithm. The first algorithm is the batch version or original k-means and the second version of k-means is known as online or incremental k-means. The batch k-means algorithm works in the following way (Rosell, 2006).

Input: *N documents to be clustered, the cluster number k and distance measure*

Output: *K clusters, and each document will be assigned to one cluster*

1. *Pick k objects at random and let them define k clusters.*
2. *Calculate cluster representatives.*
3. *Make new clusters, one per cluster representative. Let each text belongs to the cluster with the most similar cluster representative.*
4. *Repeat from 2 until a stopping criterion is reached.*

Basic k-means Algorithm for finding *K* clusters.

The algorithm requires K number of clusters, N documents to be clustered and distance measure as input from user. The algorithm starts by randomly selecting K initial cluster centroid to represent initial cluster centers, where k is the number of clusters specified by the user. Then each document is assigned to a particular closest centroid, depending on a proximity measure that quantifies the notion of closest for the specific data like cosine similarity measure and each collection of documents assigned to a centroid is a cluster. The centroid of each cluster is then updated based on the new documents assigned to the cluster. The assignment and the update steps are repeated until no document changes clusters or until the centroids remain the same.

For some conditions of proximity functions and types of centriods, k-means always converges to a solution; i.e., k-means reaches a state in which no documents are shifting from one cluster to another, and hence, the centroids don't change (Kumar et al., 2005). This is normally the stopping criterion or the termination condition in which no documents change clusters, or very few change clusters between iterations. It is also possible to stop after a predefined number of iterations, since most quality improvement usually is gained during the first iterations (Rosell, 2006). Therefore, in this thesis the termination condition is applied by using sufficient number of iterations after evaluating the performance of the clustering results of different iterations.

Although the batch or basic k-means algorithm is popular, in part due to its ease of implementation, it has a number of drawbacks or problems as stated by Kumar et al (2005) and (Nicholas, Andrews and Edward, 2007).

1. The outcome of clustering in *K*-means depends on the initial seeds, i.e. the value of K must be determined beforehand and the initial document seeds need to be selected randomly. These initial setting, namely, k value and document seeds will have impacts on the clustering results (Chen et al., 2010). K-means is greedy algorithm in essence; it relies on random initialization and it can converge to suboptimal local minima (Nicholas, Andrews and Edward, 2007) and it is hard to attain the global optimum clustering results as discussed in Chen et al. (2010).

2. When outliers or noisy are present, the resulting cluster centroid may not be representative enough and influence the clusters that are found. If a document set contains many outliers, documents that are far from any other documents and therefore do not fit well into any cluster. Frequently, if an outlier is chosen as an initial seed, then no other vector is assigned to it during subsequent iterations. Thus, we end up with a singleton cluster (a cluster with only one document) (Manning, Raghavan and Schütze, 2009) and (Kumar et al., 2005).

Given the aforementioned problems, there are many ways to enhance the basic K-means algorithm (Steinbach, Karypis and Kumar, 2000). A large number of clustering

algorithms have been developed to efficiently handle large size data sets (Jain, 2008). One of the most studies conducted is the incremental Clustering. Approaches in this category are designed to operate in a single pass over data points to improve the efficiency of data clustering (Bradley and Fayyad, 1998).

In incremental clustering, instead of updating cluster centroids after all points have been assigned to a cluster, the centroids are updated incrementally, after each assignment of a document to a cluster. At each step or iteration, the documents are visited in random order and a document either moves to a new cluster (if it lead to an improvement in the value of the criterion function) or stay in its current cluster. Iterations stop, as soon as the iteration is performed in which no documents moved between clusters.

Kumar et al (2005) also discussed that incremental updating of centroids is used as the strategy to solve the problems of the basic k-means algorithms. Using an incremental update strategy guarantees that empty clusters are not produced since all clusters start with a single point, and if a cluster ever has only one point, then that point will always be reassigned to the same cluster. In addition, if incremental updating is used, the relative weight of the point being added may be adjusted; e.g., the weight of points is often decreased as the clustering proceeds which result with a better accuracy and faster convergence (Kumar et al., 2005).

2.6.1.2 Bisecting K-Means

The bisecting k-means algorithm is one of the fast text clustering algorithms which is used for large size of the textual data. In Steinbach, Karypis and Kumar (2000) it was shown that Bisecting k-means is a fast and high-quality clustering algorithm for text documents which is frequently outperforming both the standard k-means and the agglomerative clustering techniques.

Bisecting k-means is the variant of k-means algorithm that is based on a simple idea; to obtain k clusters, split the set of all points in to two clusters (using k-means), select one of these clusters to split, and so on, until K clusters have been produced. This algorithm

starts with a single cluster of all the documents and works in the following way (Steinbach, Karypis and Kumar, 2000):

1. *Pick a cluster to split.*
2. *Find 2 sub-clusters using the basic K-means algorithm. (Bisecting step)*
3. *Repeat step 2, the bisecting step, for ITER times and take the split that produces the clustering with the highest overall similarity.*
4. *Repeat steps 1, 2 and 3 until the desired number of clusters is reached.*

Bisecting k-means Algorithm

Initially the whole documents are considered as a single cluster. Then, there are a number of different ways to choose which cluster is split. According to Steinbach, Karypis and Kumar (2000), we can choose the largest cluster at each step, the one with the least overall similarity, or use a criterion based on both size and overall similarity. Steinbach, Karypis and Kumar (2000) did numerous runs and determined that the differences between the two methods were small. Hence, in this thesis we split the largest remaining cluster.

The bisecting K-means algorithm can produce either an un-nested (flat) clustering or a divisive hierarchical clustering (Steinbach, Karypis and Kumar, 2000). To avoid confusion in this paper it is used as a variant of k-means algorithm. For un-nested (partitioning) clusters we will often “refine” the clusters using the K-means algorithms, but we do not refine the nested clusters. The clustering results are refined by using their centroids as the initial centroids for the K-means algorithm.

2.6.2 Hierarchical Clustering Techniques

Hierarchical clustering algorithms recursively find nested clusters either in agglomerative mode (starting with each data point in its own cluster, merge the most similar pair of clusters successively to form a cluster hierarchy) or in divisive mode (starting with all the data points in one cluster, recursively divide the cluster into smaller clusters) (Jain,2008).

The result of a hierarchical clustering algorithm can be viewed as a tree, called a dendogram. This tree graphically represents the repeated merging of clusters and the intermediate clusters. Clusters at an intermediate level encompass all the clusters below them in the hierarchy. The dendogram below shows how five points can be merged into a single cluster. For document clustering, this dendogram provides a taxonomy, or hierarchical index (Amine, Elberrichi and Simonet, 2010).

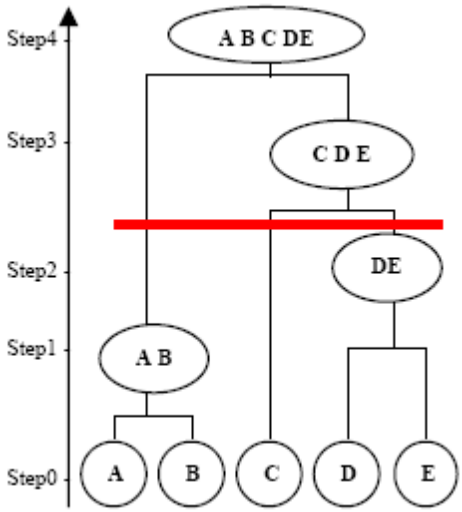

Figure 2. 2: A hierarchical clustering of five points shown as a dendogram, the tree is cut by a horizontal line at level 3.

Hierarchical clustering does not require a pre-specified number of clusters. However, in some applications we want a partition of disjoint clusters just as in flat clustering. In those cases, the hierarchy needs to be cut at some point. To determine the cutting point, as in flat clustering, we can also pre-specify the number of clusters K and select the cutting point that produces K clusters (Manning, Raghavan and Schütze, 2009). The stopping criterion for hierarchical algorithms may be that the desired number of cluster is

reached or some limit on an objective function or any internal evaluation measure (Rosell, 2009).

Depending on the direction of building the hierarchies there are two methods of hierarchical clustering: Agglomerative (bottom-up) and Divisive (top-down). The agglomerative approach is the most commonly used in hierarchical clustering (Hammouda, 2001).

2.6.2.1 Agglomerative Hierarchical Clustering

These approach starts with the points as individual clusters and, at each step, merge the most similar or closest pair of clusters. This requires definition of cluster similarity or distance. Agglomerative algorithms find the clusters by initially assigning each object to its own cluster and then repeatedly merging pairs of clusters until a certain stopping criterion is met. The traditional agglomerative hierarchical clustering procedures are summarized as follows (Steinbach, Karypis and Kumar, 2000):

1. *Compute the similarity between all pairs of clusters, i.e., calculate a similarity matrix whose* ij^{th} *entry gives the similarity between the* i^{th} *and* j^{th} *clusters.*
2. *Merge the most similar (closest) two clusters.*
3. *Update the similarity matrix to reflect the pairwise similarity between the new cluster and the original clusters.*
4. *Repeat steps 2 and 3 until only a single cluster remains.*

Simple agglomerative clustering Algorithm

A number of different methods have been proposed for determining the next pair of clusters to be merged by computing the distance between two clusters. According to (Hammounda, 2001) and (Zhao and Karypis, 2002), the three most commonly used methods for computing this distance are discussed below.

Single Linkage Method:

The single-link scheme measures the similarity of two clusters by the maximum similarity between the documents from each cluster. That is, the similarity between two clusters S_i and S_j is given by

$$sim(S_i, S_j) = \underset{d_i \in S_i, d_j \in S_j}{Max} \{(\cos(d_i, d_j))\} \quad (2.7)$$

where cos (d_i , d_j) is the similarity between two documents d_i and d_j

Complete Linkage Method:

In contrast to single link, the complete-link scheme uses the minimum similarity between a pair of documents to measure the same similarity. That is, the similarity between two clusters S_i and S_j is given by

$$sim(S_i, S_j) = \underset{d_i \in S_i, d_j \in S_j}{Min} \{(\cos(d_i, d_j))\} \quad (2.8)$$

In general, both the single- and the complete-link approaches do not work very well because they either base their decisions on limited amount of information (single-link) or they assume that all the documents in the cluster are very similar to each other (complete-link approach) (Zhao and Karypis, 2002).

Average Linkage Method:

This method takes into account all possible pairs of distances between the objects in the clusters, and is considered more reliable and robust to outliers (Hammouda, 2001). The average linkage (also known as UPGMA (Unweighted Pair- Group Method using Arithmetic averages) overcomes the problems of single link and complete link approaches by measuring the similarity of two clusters as the average of the pairwise similarity of the documents from each cluster. That is, the similarity between two clusters S_i and S_j is given by

$$sim(S_i, S_j) = \frac{1}{n_i n_j} \sum_{d_i \in S_i, d_j \in S_j} \cos(d_i, d_j) = \frac{D_i^t D_j}{n_i n_j} \quad (2.9)$$

where $D_i^t D_j$ is the normalized cosine similarity when the document vectors are of unit length, n_i and n_j denotes the sizes of the corresponding clusters S_i and S_j respectively.

Agglomerative techniques are usually with quadratic running time $O(n^2)$ due to their global nature since all pairs of inter-group similarities are considered in the course of selecting an agglomeration(Hammouda, 2001).

2.6.2.2 Divisive Hierarchical Clustering

These methods work from top to bottom, starting with the whole data set as one cluster, and at each step split a cluster until only singleton clusters of individual objects remain. They basically differ in two things: (1) which cluster to split next, and (2) how to perform the split. Usually an exhaustive search is done to find the cluster to split such that the split results in minimal education in some performance criterion. A simpler way would be to choose the largest cluster to split, the one with the least overall similarity, or use a criterion based on both size and overall similarity.

2.7 Clustering Evaluation measures

Measuring the quality of a clustering algorithm is a common problem in text as well as data mining. This is because in clustering labeling of clusters is not known in advance which makes the interpretation of clustering results more difficult.

The difficulty mainly comes from the fact that this evaluation is subjective by nature because there are often various possible relevant groupings for the same data set. The four criteria most commonly used to evaluate an unsupervised classification of textual documents are (Amine, Elberrichi and Simonet, 2010):

- Ability to process very large volumes of unstructured data.
- Easy reading of results: the system must offer various modes of visualization of the results.
- The data must be as homogeneous as possible within each group, and the groups as distinct as possible. This amounts to choosing the best adapted similarity measure.
- A good representation unquestionably influences the clustering.

The choice of a clustering quality evaluation metric usually depends on the application (Rosell, 2009). In the context of text clustering, several options are available. In order to investigate cluster validity, the two commonly used approaches are internal quality measures and external quality measures.

2.7.1 Internal Measures

Typical objective functions in clustering formalize the goal of attaining high intra cluster similarity (objects within a cluster are similar) and low inter-cluster similarity (objects from different clusters are dissimilar). This is an internal criterion for the quality of a clustering. This approach is based on the information intrinsic to the data set alone and used when the classes are not known. According to Kumar et al (2005), it measures the goodness of a clustering structure without respect to external information and it is also called unsupervised measure. This is to measure the quality of clustering as how well clusters are separated and how compact they are.

2.7.1.1 Overall Similarity

Overall similarity is an internal quality measure that uses weighted similarity of internal cluster similarities to measure the cohesiveness of the produced clusters. Internal cluster similarity I for cluster C_j can be computed as:

$$I_j = \frac{1}{n_j^2} \sum_{d \in C_j d' \in C_j} \cos(d, d') \tag{2.10}$$

where n_j is number of documents in cluster j. We can rewrite I_j as:

$$I_j = \left(\frac{1}{n_j}\sum_{d \in C_j} d\right) \bullet \left(\frac{1}{n_j}\sum_{d' \in C_j} d'\right) = c \bullet c = \|c\|^2 \tag{2.11}$$

So, I_j the average pairwise similarity between all points in cluster C_j is equal to the square of the length of the centroids of that cluster. Overall similarity of the clustering solution is:

$$overall\ Similarity = \sum_j \frac{n_j}{N} I_j \tag{2.12}$$

where N is the total number of documents in the corpus.

2.7.2 External Measures

This approach is based on a previous knowledge of classes of the data set. In this approach the clustering results are compared to an existing solution prepared manually by professional indexers. i.e., Clustering algorithms can be evaluated by comparing clustering output with known classes as answer keys. According to Kumar et al (2005), this approach measures the extent to which the clustering structure discovered by a clustering algorithm matches some external structure or class labels and they are often called supervised measures. These measures evaluate the extent to which a cluster contains objects of a single class. There have been a number of evaluation measures, such as Entropy, purity and F-measure.

2.7.2.1 Purity

Purity is a simple and transparent evaluation measure. Purity measures the extent to which each cluster contains documents from primarily one class. According to Ozgur (2004), for a particular cluster j of size nj, purity of this cluster is defined to be:

$$P_j = \frac{1}{n_j} Max_i n_{ji} \tag{2.13}$$

where n_{ji} is number of documents of class i that are assigned to cluster j. So, Pj is the fraction of overall cluster size that the largest class of documents assigned to that cluster constitutes. The overall purity of the clustering solution is obtained by the weighted sum of individual cluster purities.

$$P = \sum_j \frac{n_j}{N} P_j \tag{2.14}$$

where N is total number of documents in the document collection

The values of purity are with the interval 0 and 1.0. Bad clustering has purity values close to 0, whereas a perfect clustering has a purity of 1.0. In general, the larger are the values of purity; the better is the clustering solution.

High purity is easy to achieve in case of large number of clusters, in particular purity is 1.0 if each object gets its own cluster. Thus, we cannot use this criterion to trade off the quality of the clustering against the number of clusters.

2.7.2.2 Entropy

One external measure is **entropy**, which provides a measure of "goodness" for unnested clusters or for the clusters at one level of a hierarchical clustering. Entropy tells us how homogeneous a cluster is. The higher the homogeneity of a cluster, the lower the entropy is, and vice versa. The entropy of a cluster containing only one object (perfect homogeneity) is zero. Entropy can be used to find the best single cluster in a clustering (Rosell, 2009).

After computing the class distribution of the data, the total entropy E for a set of clusters is obtained by summing the entropies Ej of each cluster j weighted by it size (Ozgur, 2004):

$$E_j = \sum_i P(i,j)\,.\,logP(i,j) \tag{2.15}$$

$$E = \sum_j \frac{n_j}{N} E_j \tag{2.16}$$

P (i, j) is the probability that a document has class label i and is assigned to cluster j, nj is size of cluster j and N is the total number of documents in the corpus.

Entropy is a more comprehensive measure than purity. It considers the distribution of classes in a cluster. Note that, in this case, entropy is normalized to take values between 0 and 1. An entropy value of 0 means the cluster is comprised of documents only from one category, while an entropy value near 1 is bad since it implies that the cluster contains a uniform mixture of classes.

2.7.2.3 F -Measure

F measure provides a good balance between precision and recall, which is excellent in the context of information retrieval. To extend this to clustering, we assume the existence of a set of reference classes, and the found clusters (the output of the clustering algorithm) are treated as retrieved documents from these classes. Each cluster is considered as if it were the result of a query and each class as if it were the desired set of documents for the query.

According to Massey (2004), the F measure is specifically proposed to be used to evaluate clustering quality. This measure is widely used in supervised text categorization but also in text clustering. According to Steinbach, Karypis and Kumar (2000), for cluster *j* and class *I the* Recall and precision for each cluster j and class i are calculated as follows:

$$Recall(i,j) = n_{ij}/n_i \, , precission(i,j) = n_{ij}/n_j$$

Here, n_{ij} is the number of documents with class label i in cluster j, n_i is the number of documents with class label i and n_j is the number of documents in cluster j. The F-measure of cluster j and class i is calculated as follows:

$$F(i,j) = \frac{2Recall\ (i,j)precision(i,j)}{Recall(i,j) + \ precision(i,j)} \quad (2.17)$$

The F-measure values are in the interval (0,1) and larger F-measure values correspond to higher clustering quality.

2.8 Review of Related Research Works on Amharic Text Classification

As to the knowledge of the researcher, the following researches were conducted in Amharic news classification using different machine learning approaches. Table 2.2 summarizes the machine learning approaches used, categories considered and the performance achieved by different researchers.

Researcher	Categories considered	Approach used	Accuracy achieved
Zelalem S. (2001)	3	Cosine Similarity	85.05%
Surafel T. (2003)	3	KNN (K-Nearest Neighbor)	89.61%
		Naïve Bayes	95.73%
	4	KNN	84.51%
		Naïve Bayes	93.86%
	7	KNN	75.27%
		Naïve Bayes	89.93%
	16	KNN	64.4%
		Naïve Bayes	78.48%
Yohannes A.(2007)	5	LMT	93.45%
		LibSVM (support vector machine)	95.21%
	10	LMT (Logic model Tree)	89.98%
		LibSVM	91.36%
	15	LMT	79.72%
		LibSVM	81.15%
Worku K. (2009)	9	ANN(Artificial Neural Network)	70.8%
Alemu k.(2010)	8	libVSM (Hierarchical approach)	80.34%
Zeleke A. (2010)	12	LibVSM(Phrase based approach)	72.01%

Table 2. 2: Summary of previous researches done in Amharic news classification

As shown in Table 2.2, all of these researches were conducted using supervised text classification techniques. However, supervised text classification approaches have a number of problems as discussed in Section 1.2 of Chapter One.

As shown in Table 2.2, Surafel and Yohanes have done experiments at increasing number of categories and they found that the accuracy decreases as the number of categories increase.

CHAPTER THREE

THE AMHARIC LANGUAGE AND ITS WRITING SYSTEM

3.1 The Amharic Language

Ethiopia is a multi-lingual country where more than 80 languages are used in day-to-day Communication (Tessema, Meron and Teshome, 2009). Although many languages are spoken in Ethiopia, according to the Federal Democratic Republic Population census commission, Amharic is dominant in that it is spoken in the country as a mother tongue language in more than 21 million people (29.3% of the total population) (Census Summary of Ethiopia, 2007).

3.2 The Amharic Writing System

The Amharic writing system is taken from Gee'z alphabet and it is written using the Gee'z script horizontally from left-to-right. This writing style is one of the major differences between Amharic and the other Semitic languages like Arabic and Hebrew (Wapedia, 2009).

3.2.1 Amharic Characters

In Amharic langue, each characters have seven different forms called orders that reflect the seven vowel sounds (e, u, i, a, e, i, o). The seven orders represent syllable combinations consisting of a consonant followed by vowel. The first order is the basic form which contains 34 base characters or basic forms; and the other six orders are non-base characters derived from the base forms by changing the different vowels. In other words, the six non-base forms or orders show the different forms of the basic characters (first orders). The 34 basic characters and their respective six derived orders give a total of (34 * 7) 238 distinct characters (fidels). Sample lists of orders for Amharic characters are shown in Table 3.1.

Orders	1st order	2nd order	3nd order	4th order	5th order	6th order	7th order
V / C	e	U	i	a	e	i	o
h	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ
l	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
m	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ
s	ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ

Table 3. 1: Sample lists of orders for Amharic characters

In addition, the writing system includes 4 symbols with five character orders, which contain a special feature usually representing labialization. The eighteen labialized consonants, which have only one order such as ሏ, ሟ and ሯ, are also included in the writing system. The complete list of Amharic characters is shown in appendix 2.

3.2.2 Amharic Punctuation Marks

Punctuation marks are symbols that are used in written language to separate sentences and parts of sentences in order to make their meaning clear. The Amharic language uses punctuation marks in its writing system. Some of these punctuation marks are unique to the Amharic language and others are taken from foreign languages.

'Hulet netb' (፡), 'arat netb'(::), 'netela serez'(፤), 'drb serez'(፧), question mark(?) and exclamation mark(!) are some of the common punctuation marks that are used in Amharic language. The complete list of the punctuation marks and their use is presented in appendix 3.

3.2.3 Amharic Number System

The Amharic number system consists of twenty single characters which are derived from Greek letters, and some were modified to look like Amharic 'Fidel' (Bender et al., 1976). These twenty single characters represent numbers from one to ten, for multiples of ten,

hundred and thousand (see appendix 4 for the list). The number system has no representation for zero (0) and it is not suitable for arithmetic computation. Hence the Amharic number system is used to write dates specially calendar and the Hindu-Arabic numeral system is used for arithmetic purposes (Bender et al., 1976).

3.3 Problems of Amharic Writing System

Due to the various problems of the Amharic writing system, it is difficult to automate information retrieval system for Amharic language. These writing problems also have a negative effect on the performance of different machine learning approaches in text classification and text clustering. Some of the problems are discussed in the following sections.

3.3.1 Redundancy of Some Characters

In Amharic writing system, there is unnecessary redundancy of some characters (fidels) with the same pronunciation (sound) and meaning. Although these different characters have the same pronunciation and meaning, they are represented with different symbols. These various forms of a character have their own meaning in Ge'ez. However, there is no clear rule that shows its purpose and use in Amharic writing system (Bender et al., 1976). These characters (fidels) are **ሀ**, **ሐ**, and **ኀ**, **ሰ** and **ሠ**, **አ** and **ዐ**, and **ጸ** and **ፀ**.

Because of these different forms of a single character, a single word such as **"religion"** can be written in different forms as "**ሀይማኖት**", "**ሃይማኖት**", "**ሐይማኖት**", and "**ኀይማኖት**" although they all have the same meaning. This result in an increase in the number of words representing a document and an increase in vector dimension which is a challenge in Amharic document retrieval, document classification and document clustering.

3.3.2 Inconsistency of Compound Words

In Amharic writing system, two different words can be joined by hyphen, forward slash or space to form compound words. This shows that, there is inconsistency in

representing or writing of compound words in Amharic writing system. This is because compound words are sometimes written as two separate words and considered as two independent words. In addition, compound words are also treated as a single word by fusing the two words or by inserting a hyphen between them or by using their short form. For instance a word can be written as "ጽህፈት ቤት", "ጽህፈት-ቤት" or "ጽ/ቤት" all having the same meaning.

This shows that a single compound word is written in several ways which result in an increase in the dimension of the vector space. In addition, considering compound words as two separate words would result in loss of the original meaning of compound words. For example, if the compound word "ጽህፈት ቤት" is treated as two separate words "ጽህፈት"and "ቤት". The result would be two independent words with different meaning and the original meaning of the compound word has been lost. The list of some compound words and their abbreviations is presented in appendix 5.

3.3.3 Inconsistency of Abbreviations

In Amharic language, using forward slash ("/") or period (".") is common to write some Amharic words in abbreviation or shorter form. For example the short form of the word ዓመተ ምህረት can be written as "ዓ/ም", "ዓ.ም" or "ዓም " which result in an inconsistency of abbreviating Amharic words. These different representations of the same word create high dimensional vector space and it has a negative effect on the performance of learning algorithms.

3.3.4 Transliterations Problem

In Amharic language there are some words that are taken from foreign languages. In Amharic writing system, such types of words are written in different ways with spelling variation. For instance, the word "Computer" can be written as "ኮምፒዩተር" or "ኮምፒውተር".

All the above mentioned features or problems of the language have a negative effect in applying machine learning approaches for document classification and clustering. Hence, to solve such problems and to increase the performance of learning algorithms in document classification and clustering, a number of document preprocessing activities were made before an index term is selected (see chapter 4).

3.4 System for Ethiopic Representation in ASCII (SERA)

SERA is a scheme for transliterating Amharic characters. The fundamentals of SERA are discussed in Daniel (1996). SERA is a convention for transliteration of Amharic characters (Fidel) script into Latin script that insures the integrity of the format and content of the original document, and that can be fully transportable across all computer mediums.

For ease of preprocessing and compatibility reasons, the Amharic text was transliterated into an ASCII representation using SERA. The SERA transliteration table is presented in appendix 6.

CHAPTER FOUR

METHODOLOGY

4.1 Introduction

In this chapter, the methodology adopted in this study including the actual tasks done at each phases of the methodology are discussed in detail.

Knowledge Discovery in Text (KDT) and Text Mining (TM) are emerging research areas that try to resolve the problem of information overload by using techniques from data mining, machine learning, NLP, Information Retrieval (IR) and knowledge management (Karanicas et al., 2002). KDT and TM are mostly automated techniques that aim to discover high-level information in huge amount of textual data and present it to the potential user (analyst, decision-maker, etc.). KDT is the non-trivial process of identifying valid, novel, potentially useful and ultimately understandable patterns in unstructured textual data (Karanicas et al., 2002).

KDT is a multi-step process, which includes all the tasks from the gathering of documents to the visualization of the extracted information. The main goal of KDT is to make patterns understandable to humans to facilitate a better understanding of the underlying data (Fayyad and Piatetsky, 1996). TM is a step in the KDT process consisting of particular data mining and NLP algorithms that under some acceptable computational efficiency limitations produce a particular enumeration of patterns over a set of unstructured textual data (Karanicas et al., 2002).

The methodology adopted in this study has the following three major steps that are performed in the process of discovering knowledge from text as discussed in Karanicas et al., (2002).

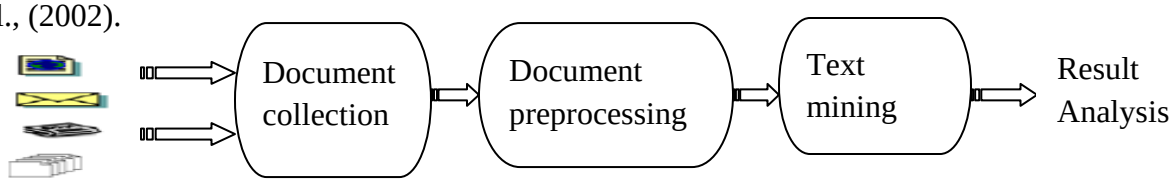

Figure 4. 1: Major stages of the KDT process.

In the first stage of the approach, relevant text documents were collected from ENA for use in this study. Then, the collected Amharic text documents were preprocessed before using for the experimentations process. The third phase of the approach used is text mining which is document clustering and cluster analysis in this particularly study. In this stage of the KDT process, experimentations were conducted using the most commonly used unsupervised machine learning algorithms and finally the outputs produced were evaluated using the internal and external evaluation metrics.

The subsequent sections discussed the architecture of this study and the major tasks performed at each phases of the methodology

4.2 Architecture of Amharic Text News Clustering

The architecture for Amharic text news clustering consists of four basic phases, namely, document collection, document preprocessing and representation, clustering and evaluating the clustering results. The architecture of Amharic document clustering system is described in Figure 4.2.

As shown in Figure 4.2, the document preprocessing techniques such as transliteration, tokenization, normalization, stop words and numbers removal, stemming and dimension reduction were done on Amharic documents. The TF*IDF term weighting approach was used to represent text documents with representative features.

Once all these document preprocessing and representation activities were done, the datasets were prepared in an appropriate format and given to the learning algorithms. The learning algorithms process this dataset and group them into the appropriate clusters and finally the performances of those clustering algorithms were evaluated using different clustering evaluation measures. The details of each phase are discussed in the subsequent sections.

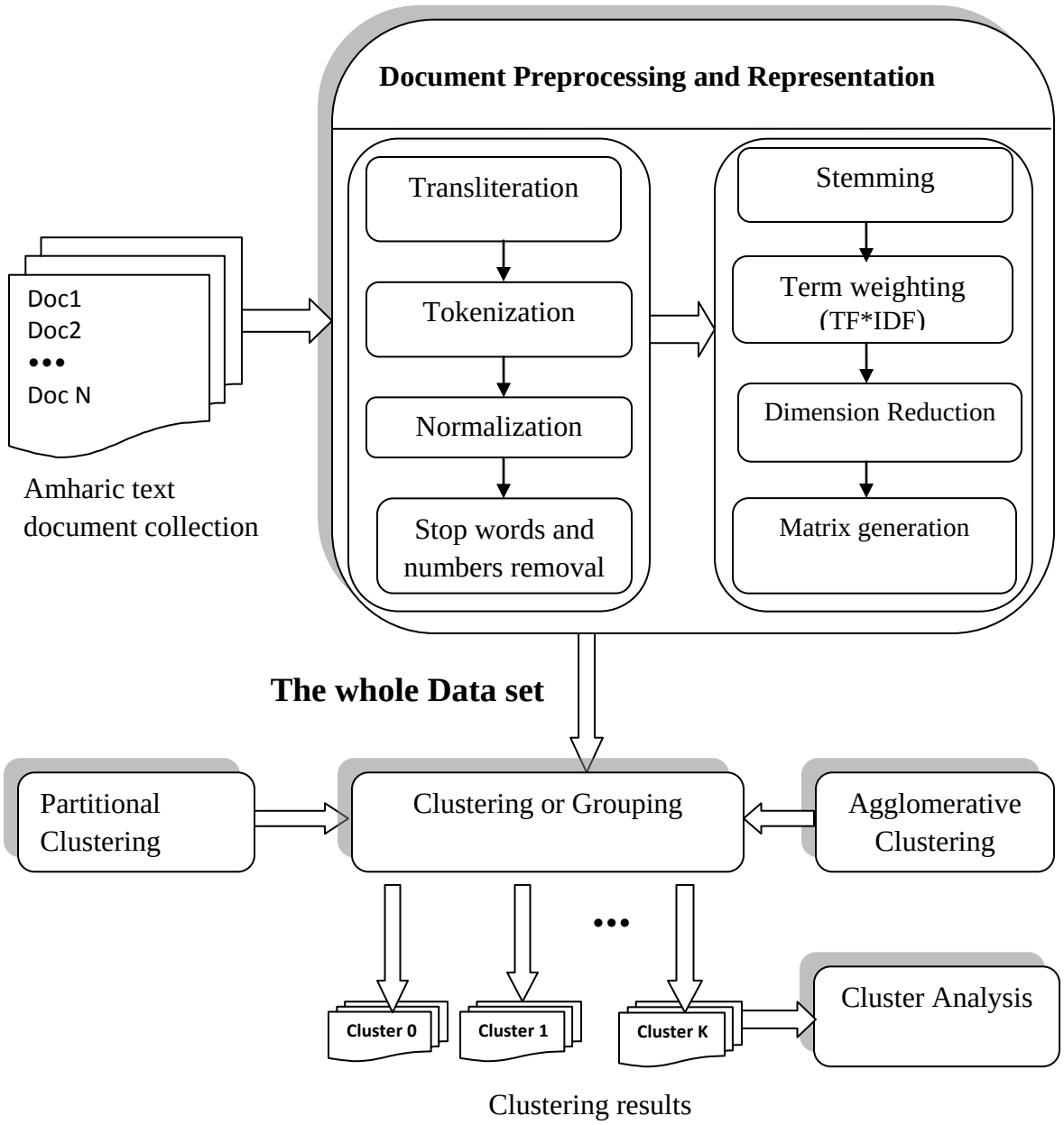

Figure 4.2: Architecture of Amharic text document clustering

4.3 Document Collection

The document data set that was used for the experiments was Amharic text News which was collected from ENA. Although the classification of news items is done manually, ENA uses software called ENASoft to make the management of news items easy. Once the classification task is done manually, ENASoft is used to dispatch news items into different Media such as Ethiopian Radio and Television, Addis Zemen, Sheger FM and others. The total number of categories collected and considered in this study are 10; with a total of 3,047 Amharic news items or documents.

4.4 Document Preprocessing

Document preprocessing is a process of identifying and determining relevant terms (vocabulary of terms) that can represent and discriminate one document and even a particular category from the other by removing irrelevant and redundant features of the document. Before clustering or grouping text documents, text document pre-processing operations including normalization, tokenization, stemming, dimension reduction and term weighting were performed to identify and extract representative terms from Amharic text documents. All these document preprocessing activities were performed to generate a sequence of representative terms and their TF*IDF weight values which is suitable format for clustering tool.

The process of tokenization, normalization and stemming is language-dependent and in this thesis the different characteristics or features of the Amharic language were considered in the development of the algorithms. The document preprocessing task was implemented using python programming language (Python 3.1). The document preprocessing activities done in this thesis are presented in the following subsections.

4.4.1 Amharic Document Transliteration

The original Amharic text was transliterated into an ASCII representation using SERA as discussed in Section 3.4 of Chapter Three. Both document preprocessing activities and the experiments were done using the transliterated form in order to simplify spelling

normalization of Amharic characters and to make it compatible with the clustering tool used for the experiments. The transliteration was done using algorithm 4.1.

```
Read document file
Read SERA mapping table
For each characters in file
      If dictionary contains character
            Replace the Ethiopic characters to ASCII
      End if
End for
```

Algorithm 4.1: Amharic document transliteration

4.4.2 Tokenization

In the process of tokenization, documents are broken into individual tokens. Typically words and sequence of words are extracted from documents to enhance the performance of learning algorithms.

In the tokenization process irrelevant and noisy features for the clustering process such as punctuation marks and any non Amharic characters were removed from documents in the collection using Algorithm 4.2. This is because these features are not relevant to represent the content of documents and they have no contribution in discriminating one document or category from the other.

```
Read document file
Read punctuations list
Read unnecessary characters list
For each token in file
      If token ends with punctuation then
            Remove punctuation from file
      End if
      If token is in characters list
            Remove token from file
      End if
End for
```

Algorithm 4.2: Tokenization

4.4.3 Normalization

In normalization, the different forms or representations of a single word were removed by changing spelling variation of the same sound word to one common form. In addition, abbreviations are expanded into their full form. Compound words have also been written in one common standard form by concatenating them as a single word to keep their original meaning and to reduce the dimension of the vector space.

1. Changing spelling variation of a word into one common form

As discussed in Section 3.3.1 of Chapter Three, the Amharic writing system has redundant characters which lead to spelling variation of words with the same sound which is a problem in text classification and clustering. To solve this problem, all similar characters with the same pronunciation were converted into one common form during transliteration using SERA to bring the spelling variation of a word into one. Hence, words of different character sequences with the same meaning are matched into one common form.

For example, the different forms of a word "ሀይማኖት", "ሃይማኖት", "ሐይማኖት", and "ኃይማኖት" created by spelling variation are changed into their equivalent of a single word "ሀይማኖት" by changing "ሃ", "ሐ", and "ኃ" into "ሀ". This shows that the different forms of words that convey the same information are reduced into a single term. Normalization of such spelling variations of words reduces the dimension of feature space and enhances the performance of learning algorithms.

In similar fashion, the orders of 'ሠ', 'ዐ' and 'ፀ' were also converted to the corresponding orders of 'ሰ', 'አ' and 'ጸ' respectively during transliteration (see appendix 7 for the transliteration).

2. Expanding short form of compound words and abbreviations.

In Amharic writing system, compound words and abbreviations are written with variety of ways leading to inconsistency in writing. This different and inconsistent representation of compound words and abbreviations was solved by expanding all the short forms into their expanded form.

3. Concatenation of Compound Words

As we have seen in Section 3.3.2 of Chapter Three, there are different representations of compound words in Amharic writing which result in an increase in the dimension of the vector space. Hence, to solve this problem, algorithm 4.4 was used to convert the expanded form into a single common standard form after creating a list that contained such type of words.

```
Read document file
Read compound words list
For each token in file
     If token is in list then
          Concatenate token with the next token
     End if
End for
```

Algorithm 4.3: Concatenation of compound words

4.4.4 Stop Words and Numbers Removal

As discussed in the previous chapters, not all terms or words in a document are relevant equally to represent the contents of the document. Some extremely common words that would appear almost in all documents of the collection are not relevant to represent the content of documents and discriminate one document from others.

Like other languages, there are some common Amharic words that appear in almost all the documents in the collection. These words which are encountered very frequently and carry no useful information about the content and thus the category of documents are called stop words. These stop words have a little value in representing and discriminating one news item (cluster) from the other. Hence these stop words should be excluded from

the vocabulary entry, which will lead to a drastic reduction in the dimensionality of the feature space.

It is common practice to exclude stop words from feature vector. To remove stop words a stop word lists can be used or the stop word can be determined from their frequency, which is said to be more efficient and language independent (Ho, 1999; Wilbur and Sirotkin, 1992). Hence, in this work, stop word removal was performed using the two approaches; stop word list and term frequency thresholding.

Stop words are language specific and often domain specific. However, for Amharic language there is no standard stop word list. Hence, in this study, two kinds of stop words lists were prepared by considering the stop word lists used by previous researchers such as Alemu (2010).

The first stop word list consists of some Amharic words such as "**ነበር**", "**ነው**", "**ሆነ**", etc which appear almost in all documents and are used to provide structure in the language rather than content. The second stop word list contains news specific Amharic such as "**ዘገበ**", "**ገለፀ**", etc. Such words have no discriminating power among Amharic documents in the collection and removed from the document collection.

After preparing such stop word lists, algorithm 4.5 was used to remove such stop words and numbers from Amharic text news collection.

```
Read document file
Read stop word list
For each token in file
     If token is in stop word list then
          Remove token from file
     End if
     If token is number then
          Remove token from file
     End if
End for
```

Algorithm 4.4: Stop words and numbers removal

4.4.5 Stemming

Stemming is the process of reducing words to their base form, or stem by removing suffixes or prefixes. For example, the Amharic words **የግብርና**, **ግብርናዎች**, **የግብርናዎች** are all reduced to the stem or root word **ግብርና**. Notice that one effect of stemming is to reduce the number of distinct words in a text corpus and to increase the frequency of occurrence of some individual words. As shown in the above example, if the terms **ግብርና**, **የግብርና**, **ግብርናዎች**, **የግብርናዎች** occur in the document with the frequency of one, the frequency of **ግብርና** increases from one to four by stemming the other three terms into their base form.

An exceptional list of words on which the affix (suffix and prefix) removal algorithms can not be applied was prepared. This is because removing suffix or prefix from some Amharic words result in loss of the original meaning or for that matter any meaning. This may mislead the learning or clustering algorithms to produce a different clustering solution. For example if we remove the suffix '**ን**' ('n') from the word '**ኮንስትራክሽን**' ('construction'), the result is **ኮንስትራክሽ** ('konstrakxi') which has no meaning in Amharic language. Similarly if we remove the prefix '**ከ**' ('ke') from the word '**ከተማ**' ('town') we get a combination of characters '**ተማ**' (tema) which is not in the Amharic lexicon.

Although, Nega and Willett (2002) have developed a stemming algorithm for Amharic language, it was found to be hard to replicate. Hence, a stemming algorithm that can remove common Amharic prefixes and suffices was developed in collaboration with other research group members[1].

Removal of Prefixes and suffixes:

Prefixes are characters that are attached at the beginning of the word depending on the context of a sentence**.** Suffixes are also characters attached at the end of the word. Common Amharic prefixes such as "**የ**", "**ከ**", "**በ**", "**ለ**" and "**እንደ**" and common Amharic suffixes such as "**ም**", "**ን**", "**ዎች**" and "**ዊነት**" were removed from the whole document

[1] *Gedefew Mehari, Lakachew Yayeh and Solomon Asemu*

collection except from the words in the exception list using algorithm 4.6. The list of common prefixes and suffixes is presented in appendix 8.

```
Read document file
Read exception list
Read prefix list
Read suffix list
Assign the first 1, 2, 3,… character(s) of the token to prefix
Assign the last 1, 2, 3,… character(s) of the token to suffix
For each token in file
     If token is not in exception list and prefix is in
prefix list
          Remove prefix from token
     End if
     If token is not in exception list and suffix is in
suffix list
          Remove suffix from token
     End if
End for

Update file
```

Algorithm 4.5: Removal of prefixes and suffixes

4.4.6 Term Weighting

As discussed in the previous chapters, not all terms or words with in a document are relevant equally to represent the contents of the document. Term weighing is used to weight representative terms that describe and summarize document content based on the importance of terms with in a document. Hence, in order to define the importance of a word within Amharic text documents, a vector representation was used, where for each word a numerical importance value is stored using the TF*IDF term weighting approach. The TF*IDF values were generated Using doc2matrix, a utility which comes with gCLUTO. It also converts raw documents into the matrix format required by gCLUTO.

4.4.7 Dimension Reduction

A number of document preprocessing activities including tokenization, normalization, stop words removal and stemming were preformed to reduce the number of tokens to be used as features. But to identify the most important representative attributes of documents or categories, the dimension has to be further reduced so that the performance of the learning algorithms is enhanced. Hence, to reduce the dimensionality of the data, predetermined DF threshold value was defined as 1 and hence the terms that appear in only one document were removed.

4.4.8 Document Representation and Matrix Generation

Document representation is the final task in document processing to generate document vector with a sequence of representative terms and their TF*IDF weight values. Once the Amharic news documents are preprocessed, all the documents are represented with a document-term matrix format. In the document term matrix, each column represents the TF*IDF value of a unique term and each individual row represents an individual document. The document-term matrix was also generated Using doc2matrix.

4.5 Document Clustering and Evaluation

The final phase of the methodology adopted in this study is text mining which is document clustering and result analysis. In this study, the two incremental partitioning clustering incremental k-means and bisecting k-means and the three agglomerative hierarchical clustering algorithms: single link, complete link and average link were tested for Amharic text document clustering.

Furthermore, the average link from the agglomerative clustering technique was used to check whether the categories classified and currently used by ENA matches with the categories discovered by the clustering algorithm.

4.5.1 gCLUTO

The clustering software that was used in this study is gCLUTO which is a graphical interface to CLUTO (CLUstering TOolkit). gCLUTO is a stand-alone clustering software package which combines clustering algorithms along with a number of analysis, reporting, and visualization tools to aid in interactive exploration and clustering-driven analysis of large datasets. gCLUTO provides a wide-range of algorithms that are capable of analyzing different types of datasets and finding clusters with different characteristics (Rasmussen and Karypis, 2004).

The solution reports generated for clustering solutions contain information about the clustering options used and statistics about the discovered clusters. These statistics include the number of clusters, cluster sizes, the average internal and external similarities (ISim and ESim), and the average internal and external standard deviations of these similarities (ISdev and (ESdev) and a list of the most discriminating and descriptive features for each cluster. The entropy, purity, and class conservation statistics are also displayed when the predefined classes are specified for the documents.

4.5.2 Criterion Functions in gCLUTO

Objective or criterion functions are used to drive the clustering process and to measure various aspects of intra-cluster similarity, inter-cluster dissimilarity, and their combinations (Rosell, 2009).

The overall clustering process is repeated a number of times and a collection of objects can be clustered in many different ways. That is, N different clustering solutions (i.e., initial clustering followed by cluster refinement) are computed. The problem here is in selecting the best clustering solution among those revealed by the clustering algorithms. To solve this problem, clustering have criterion functions that they seek to optimize. i.e, the clustering solution that achieves best value for the particular criterion function is selected out of these N potentially different solutions (Zhao and Karypis, 2002).

In Zhao and Karypis (2002) several objective functions for partitioning algorithms are evaluated to study their performance in the context of text document clustering. After comparing the different criterion functions implemented in CLUTO, they concluded that for k-means, the criterion functions I_2 and H_2 always produced the best clustering solutions; while for bisecting k-means, the H_2 produced the best overall clustering solutions.

Hence in this thesis the H_2 criterion function was used for both k-means and bisecting k-means which helps us to compare the similarity and dissimilarity of objects with in the clusters and the performance of the two clustering algorithms. The criterion functions for partitioning algorithms are presented in Table 4.1, where *k* is the total number of clusters, *S* is the total objects to be clustered, S_i is the set of objects assigned to the i^{th} cluster, n_i is the number of objects in the i^{th} cluster, *v* and *u* represents two objects and *sim (v,u)* is the similarity between two objects.

Criterion Function	**Optimization Function**
I_1	$maximaise \sum_{i=1}^{k} \frac{1}{n_i} \left(\sum_{v,u \in S_i} sim(v,u) \right)$
I_2	$maximise \sum_{i=1}^{k} \sqrt{\sum_{v,u \in S_i} sim(v,u)}$
E_1	$minimise \sum_{i=1}^{k} n_i \frac{\sum_{v \in S_i, u \in S} sim(v,u)}{\sum_{v,u \in S_i} sim(v,u)}$
H_1	$maximise \frac{I_1}{E_1}$
H2	$maximise \frac{I_2}{E_1}$

Table 4. 1: The mathematical definition of gCLUTO's clustering functions.

4.5.3 Document Clustering Evaluation Techniques

Several ways of measuring the quality of clustering, especially text clustering have been proposed in the literature. For evaluation purpose, the class or label of news category currently used by ENA and the associated number of documents in each category were used to compare with the results discovered by the clustering algorithms. Therefore, in addition to internal measures, the different external measures which consider the degree of agreement or overlap between the classes and the computed clusters can also be used.

In this thesis, the clustering results of the different clustering algorithms were evaluated and compared using both the internal quality measures and the external quality measures such as purity and entropy.

CHAPTER FIVE

EXPERIMENT AND PERFORMANCE EVALUATION

5.1 Introduction

This chapter presents the experimentations, analysis and interpretation of the results revealed by the different clustering algorithms. The experiments were done using three document clustering algorithms: k-means, bisecting k-means and agglomerative hierarchical clustering algorithms. The results obtained from these clustering algorithms are discussed and a comparison of these clustering algorithms was done to select the best clustering solution among the three algorithms.

In the following sections, the experimentations, evaluation and comparison of different document clustering algorithms are discussed.

5.2 Experimentations Plan

As shown in Table 5.1, a total of 10 classes and 3047 documents were used in the experimentation process. The list of pre-defined classes with the corresponding documents is already provided by ENA. These pre-classified documents were first merged into one before clustering. However, these pre-defined classes were used as an evaluation benchmark or gold standard to compare against the clusters discovered by clustering algorithms.

To test the performances of k-means, bisecting k-means and agglomerative clustering algorithms at increasing number of clusters and documents, the different pre-defined number of clusters and the corresponding pre-classified documents were used to conduct the experiments. The predefined classes were arranged in alphabetical order since the size of documents (data sets) is balanced at increasing number of clusters. The 10 categories were divided into three and the experiments were done on 4, 7 and 10 number of clusters using 1209, 2157 and 3047 documents respectively as depicted in Table 5.1.

Experiments	No.	List of pre-defined Classes used	Number of documents		Algorithms used
On 4 Clusters	***1.***	Accident	179	1209	• K-means • Bisecting k-means • Agglomerative o Single link o Complete link o Average link
	2.	Culture and Tourism	263		
	3.	Economy	446		
	4.	Education	321		
On 7 Clusters	**1.**	Accident	179	2157	• K-means • Bisecting k-means • Agglomerative o Single link o Complete link o Average link
	2.	Culture and Tourism	263		
	3.	Economy	446		
	4.	Education	321		
	5.	Health	299		
	6.	Law and Justice	197		
	7.	Politics	452		
On 10 Clusters	**1.**	Accident	179	3047	• K-means • Bisecting k-means • Agglomerative o Single link o Complete link o Average link
	2.	Culture and Tourism	263		
	3.	Economy	446		
	4.	Education	321		
	5.	Health	299		
	6.	Law and Justice	197		
	7.	Politics	452		
	8.	Science & Technology	361		
	9.	Social	234		
	10.	Sport	295		

Table 5. 1: Experimentations set up

5.3 K-means clustering Algorithm

The performance of k-means was tested for number of clusters equal to the number of pre-defined classes in each document collection. To test its performance at increasing number of clusters and documents, the experiments were done on four, seven and ten number of clusters using the pre-defined data sets from each class.

5.3.1 Experiment on Four Clusters

In this experiment, the four classes namely, accident (Acc), culture and tourism (CT), economy (Eco) and education (Educ) were considered. The descriptive and discriminating features of each cluster are shown in Table 5.2.

Descriptive & Discriminating Features

Cluster 0	**Size:** 235	**ISim:** 0.027		**ESim:** 0.006				
Descriptive:	buna	7.0%	aedega	2.8%	waga	1.9%	gorf	1.7%
Descriminating:	buna	4.7%	aedega	1.8%	waga	1.2%	negadE	1.1%
Cluster 1	**Size:** 327	**ISim:** 0.024		**ESim:** 0.007				
Descriptive:	tmhrt	5.0%	weyra	3.0%	memhr	2.6%	temari	2.3%
Descriminating:	tmhrt	3.6%	weyra	2.4%	memhr	2.0%	temari	1.7%
Cluster 2	**Size:** 299	**ISim:** 0.022		**ESim:** 0.006				
Descriptive:	hotEl	5.0%	wbet	2.4%	turist	2.3%	bololo	1.8%
Descriminating:	hotEl	3.8%	wbet	1.8%	turist	1.8%	bololo	1.3%
Cluster 3	**Size:** 348	**ISim:** 0.023		**ESim:** 0.007				
Descriptive:	whe	2.8%	meTeT	2.7%	menged	2.5%	bdr	1.9%
Descriminating:	whe	2.0%	meTeT	2.0%	menged	1.8%	bdr	1.5%

Table 5. 2: Descriptive and discriminating features for 4 clustering solution.

From Table 5.2, we can see that the descriptive and discriminating features of cluster 0 are ‘aedega’, ’gorf’, ’buna’, ‘waga’ and ‘negadE’. This shows that cluster 0 consists of documents which deal with accident and economic issues. Cluster 1, whose descriptive and discriminating features are ‘tmhrt’, ’memhr’, and ‘temari’ is about educational issues. Cluster 2 with descriptive and discriminating features of ‘hotEl’, ‘wbet’ and ‘turist’ is about culture and tourism, while cluster 3 with descriptive and discriminating features of ‘whe’, ‘meTeT’, ‘menged’ and ‘bdr’ is about economic issues.

The class distribution of the documents for each cluster is shown in Table 5.3.

Class Distribution				
Cluster	Acc	CT	Eco	Educ
0	163	0	70	2
1	8	1	3	315
2	2	261	34	2
3	6	1	339	2

Table 5. 3: Confusion matrix or class distribution of K-Means over the 4 clusters.

As shown in Table 5.3, most of the documents that belong to accident, culture and tourism, economy and education sections are assigned to cluster 0, 2, 3 and 1 respectively. However, some of the documents from economic section are also distributed to other clusters.

The quality of the overall clustering solution as well as the quality of each cluster obtained by k-means clustering algorithm for 4 clusters is shown in Table 5.4.

4-way clustering: [1209 of 1209], **Entropy:** 0.235, **Purity:** 0.892							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	235	0.027	0.008	0.006	0.002	0.473	0.694
1	327	0.024	0.006	0.007	0.002	0.135	0.963
2	299	0.022	0.007	0.006	0.002	0.312	0.873
3	348	0.023	0.005	0.007	0.002	0.102	0.974

Table 5. 4: Clustering solution using k-means for 4 clusters.

As shown in Table 5.4, the average internal similarity (ISim) between the documents within a cluster is higher in cluster 0 than the other clusters. This shows that the documents in cluster 0 are more internally cohesive than documents in other clusters. From this we can conclude that the documents assigned to cluster 0 are more similar and deal with the same topics or issues than other clusters.

It is further observed from Table 5.4 that the average external similarity (ESim) of the documents in cluster 1 and 3 is higher than other clusters. This shows that the documents

in cluster 1 and 3 are less externally isolated from documents of other clusters and hence documents in cluster 1 and 3 are not well separated from documents of other clusters.

Table 5.4 also shows that cluster 0 with the highest value of entropy contains large number of documents from different classes. On the other hand, cluster 3 with the lowest value of entropy contains the majority of documents from one class and it is the best clustering solution. The value of purity for cluster 3 is higher which indicates that the majority of the documents assigned to cluster 3 are from a single class, namely, economy. The overall performance of the k-means algorithm for the 4 clustering solution yielded an entropy of 0.235, purity of 0.892 and an overall similarity of 0.024.

5.3.2 Experiment on Seven Clusters

Seven pre-defined classes: accident, culture and tourism, economy, education, health, law and justice and politics were considered in this experiment. The documents of the seven pre-defined classes are assigned to the 7 clusters as shown in Table 5.5.

Class Distribution							
Cluster	**Acc**	**CT**	**Eco**	**Educ**	**Health**	**Law**	**Politics**
0	3	1	7	3	283	1	2
1	4	0	5	302	2	0	2
2	3	243	11	1	0	2	13
3	159	5	78	1	6	30	4
4	10	1	340	1	4	3	14
5	0	5	5	12	2	157	104
6	0	8	0	1	2	4	313

Table 5. 5: Confusion matrix or class distribution of K-Means for the 7 clusters.

From Table 5.5, we can see that cluster 0, 1, 2, 3, 4 and 6 map most closely to health, education, culture and tourism, accident, economy and politics sections respectively, while cluster 5 corresponds best to law section although it contains large number of documents from other classes. From Table 5.5, we can also see that some of the documents from economy and politics sections are also misclassified to cluster 3 and 5 respectively.

The clustering results obtained from experiments using k-means clustering algorithm for the 7 clusters is shown in Table 5.6.

7-way clustering: [2157 of 2157], **Entropy:** 0.272, **Purity:** 0.833							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	300	0.032	0.010	0.006	0.002	0.157	0.943
1	315	0.029	0.008	0.006	0.002	0.116	0.959
2	273	0.026	0.009	0.005	0.002	0.249	0.890
3	283	0.024	0.008	0.005	0.002	0.591	0.562
4	373	0.024	0.006	0.006	0.002	0.218	0.912
5	285	0.023	0.006	0.006	0.002	0.517	0.551
6	328	0.019	0.007	0.003	0.001	0.122	0.954

Table 5. 6: Clustering solution using k-means for 7 clusters.

As shown in Table 5.6 the value of the average internal similarity (ISim) between the documents is higher in cluster 0 and this indicates that the documents in cluster 0 are more similar than documents in other clusters. On the other hand, cluster 6 with the smallest value of the average internal similarity (ISim) contains documents which are more dissimilar. Cluster 6 with the lowest value of the average external similarity (ESim) of the documents contains documents that are more dissimilar from documents in other clusters.

The highest value of entropy in cluster 3 indicates that it contains large number of documents from different pre-defined categories. On the other hand, Cluster 1 with smallest value of entropy contains less number of documents from different classes and it is the best cluster as compared to the other clusters from the 7 clustering solution. Cluster 1 with the highest value of purity also shows that it contains documents primarily from one class. The average entropy, purity and overall similarity values obtained by using k-means for the 7 clustering solution are 0.272, 0.833 and 0.025 respectively.

5.3.3 Experiment on Ten Clusters

In this section, three classes: science and technology, social affairs and sport were added on the previous seven classes. The descriptive and discriminating features for the 4 clustering solution are shown in Table 5.7.

Descriptive & Descriminating Features

Cluster 0	**Size:** 300	**ISim:** 0.035	**ESim:** 0.006					
Descriptive:	TEna	11.1%	weba	6.7%	Ec	3.9%	aey	3.9%
Descriminating:	TEna	7.4%	weba	5.2%	aey	2.8%	Ec	2.8%
Cluster 1	**Size:** 319	**ISim:** 0.031	**ESim:** 0.003					
Descriptive:	Wddr	8.2%	qenenisa	7.6%	yfTer	4.7%	sport	4.3%
Descriminating:	Wddr	4.9%	qenenisa	4.6%	yfTer	2.9%	sport	2.3%
Cluster 2	**Size:** 345	**ISim:** 0.032	**ESim:** 0.006					
Descriptive:	Aerso	5.9%	mrt	4.2%	mesno	3.5%	mrmr	3.0%
Descriminating:	Aerso	3.8%	mrt	2.8%	mesno	2.5%	mrmr	2.0%
Cluster 3	**Size:** 326	**ISim:** 0.031	**ESim:** 0.006					
Descriptive:	Tmhrt	11.0%	temari	6.3%	weyra	4.4%	memhr	3.9%
Descriminating:	Tmhrt	7.4%	temari	4.6%	weyra	3.5%	memhr	2.9%
Cluster 4	**Size:** 271	**ISim:** 0.030	**ESim:** 0.006					
Descriptive:	Menged	6.4%	meTeT	5.6%	Tge	4.6%	whe	4.5%
Descriminating:	Menged	4.7%	meTeT	3.8%	Tge	3.7%	whe	3.2%
Cluster 5	**Size:** 286	**ISim:** 0.029	**ESim:** 0.006					
Descriptive:	Buna	9.4%	bdr	4.8%	mahberat	3.1%	weTat	2.4%
Descriminating:	Buna	7.0%	bdr	3.4%	mahberat	2.0%	weTat	1.7%
Cluster 6	**Size:** 275	**ISim:** 0.027	**ESim:** 0.005					
Descriptive:	Hotel	8.4%	turist	3.5%	wbet	3.4%	bololo	3.3%
Descriminating:	Hotel	6.4%	turist	2.7%	wbet	2.4%	bololo	2.4%
Cluster 7	**Size:** 270	**ISim:** 0.023	**ESim:** 0.005					
Descriptive:	Aedega	5.1%	balehebt	3.6%	feqad	2.6%	gorf	2.6%
Descriminating:	Aedega	3.7%	balehebt	2.4%	feqad	1.9%	gorf	1.7%
Cluster 8	**Size:** 330	**ISim:** 0.019	**ESim:** 0.003					
Descriptive:	Taliya	6.2%	mals	5.3%	aekahidew	3.6%	temera	3.2%
Descriminating:	Taliya	3.3%	mals	3.2%	aekahidew	2.2%	temera	2.0%
Cluster 9	**Size:** 325	**ISim:** 0.022	**ESim:** 0.005					
Descriptive:	komiyunikExin	2.8%	elka	2.3%	aefe	2.3%	hg	1.4%
Descriminating:	komiyunikExin	2.3%	elka	1.7%	aefe	1.4%	fth	1.0%

Table 5. 7: Descriptive and discriminating features for 4 clustering solution.

As shown in Table 5.7, the descriptive and discriminating features of cluster 0 are ‘TEna’ ’weba’, ‘Ec’ and ‘aey’ which shows that cluster 0 is about health issues. Cluster 1, 4, and 6 are about sport, economy, and culture and tourism respectively. Cluster 5 is about social aspects. However, ‘buna’ and ‘bdr’ may also refer to economical issues. Cluster 2 is about economical issues, but it also refers to science and technology (‘mrmr’). Cluster 7 deals with both accident (‘adega’ and ‘gorf’) and economy (‘balehebt’ and ‘fegad’). Cluster 9 with features ‘hg’ and ‘fth’ seems about law and justice. But, it is somewhat difficult to interpret the features of cluster 8.

Table 5.8 shows the confusion matrix of the 10 cluster solution for the k-means algorithm.

Class Distribution										
Cluster	**Acc**	**CT**	**Eco**	**Educ**	**Health**	**Law**	**Politics**	**Science**	**Social**	**Sport**
0	5	0	3	2	275	0	1	2	10	2
1	1	3	1	0	0	0	2	27	1	284
2	3	0	23	4	8	0	5	293	9	0
3	3	0	6	302	0	0	2	4	9	0
4	10	3	209	0	6	0	8	15	19	1
5	14	6	118	1	3	4	3	13	119	5
6	2	236	11	0	0	2	9	4	10	1
7	141	0	71	0	3	41	4	0	10	0
8	0	8	1	2	2	4	310	1	1	1
9	0	7	3	10	2	146	108	2	46	1

Table 5. 8: Confusion matrix or class distribution of K-Means for 10 clusters.

As shown in Table 5.8, the documents found in economy, politics, science and social sections are distributed over different clusters. From the class distribution, we can also see that cluster 0, 1, 2, 3, 4, 6, 7 and 8 associate to a single pre-defined class. However, cluster 5 and 9 contain many of the documents from two predefined classes. Most of the documents from the economy and politics sections are misclassified to clusters 5 and 9 respectively.

Table 5.9 below depicts the clustering solution discovered by k-means clustering algorithm for 10 clusters.

10-way clustering: [3047 of 3047], **Entropy:** 0.327, **Purity:** 0.760							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	300	0.035	0.012	0.006	0.002	0.185	0.917
1	319	0.031	0.011	0.003	0.002	0.192	0.890
2	345	0.032	0.010	0.006	0.002	0.285	0.849
3	326	0.031	0.009	0.006	0.002	0.161	0.926
4	271	0.030	0.009	0.006	0.002	0.403	0.771
5	286	0.029	0.010	0.006	0.002	0.584	0.416
6	275	0.027	0.009	0.005	0.002	0.281	0.858
7	270	0.023	0.007	0.005	0.002	0.526	0.522
8	330	0.019	0.007	0.003	0.001	0.145	0.939
9	325	0.022	0.006	0.005	0.002	0.571	0.449

Table 5. 9: Clustering results of k-means for 10 clusters.

Table 5.9 shows that cluster 0 with the highest value of ISim is comprised of documents which are more internally cohesive than documents in other clusters. The inter-cluster similarity (ESim) of the documents in cluster 1 and 8 is lower than the other clusters. This shows that the documents in cluster 1 and 8 are dissimilar from the documents of the other clusters and hence cluster 1 and 8 are relatiely more dissimilar clusters from the rest of clusters.

As Table 5.9 shows, cluster 8 with smaller value of entropy and with larger purity value is the best clustering solution which contains majority of documents from a single class, whereas cluster 5 with the highest value of entropy and with lowest purity value indicates that the clustering solution is bad. The entropy, purity and the overall similarity values obtained by k-means algorithm for the 10 clusters are 0.327, 0.76 and 0.028 respectively.

The visualization for each cluster is shown in Figure 5.1 which provides the number of constituent documents, internal Similarity, external similarity, and standard deviation.

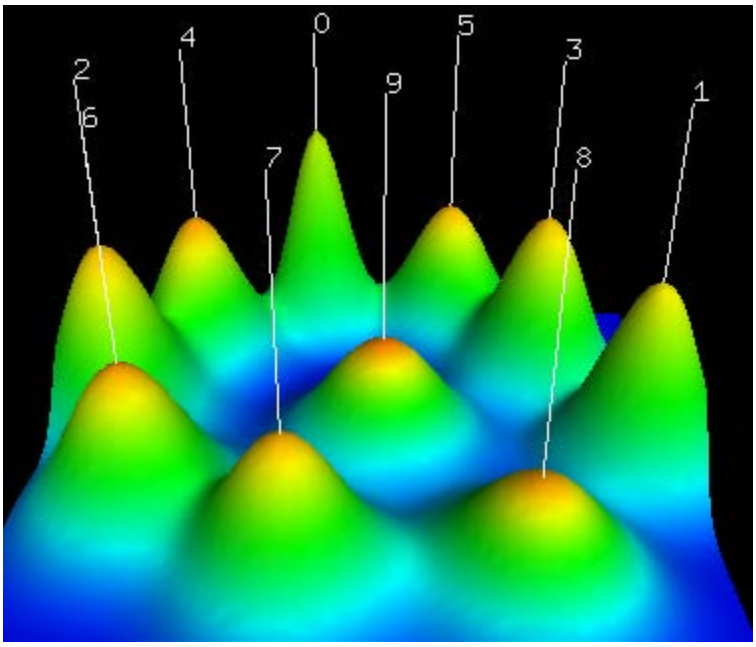

Figure 5. 1: Mountain visualization for 10 clusters obtained by k-means.

The ten peaks in the plane denoted by the numbers form 0 to 9 represent a single cluster in the clustering. Information about the corresponding cluster is represented by the peak's height, location on the plane, volume, and color.

The height of each peak on the plane is proportional to the internal similarity of the corresponding cluster. For instance, cluster 1 with the highest peak shows that the documents in cluster 1 are more internally cohesive than others, while cluster 8 and 9 with short peak contain dissimilar documents.

Clusters that are similar will have peaks that lie closely together, whereas more dissimilar clusters will be displayed with distant peaks. The volume of a peak is proportional to the number of documents (size) within the cluster. Finally, the color of a peak represents the internal standard deviation of the cluster's objects. Red represents low deviation, whereas blue represents high deviation.

5.4 Bisecting k-means Algorithm

To evaluate its performance at increasing number of clusters and documents, the bisecting k-means was also tested using four, seven and ten number of clusters on three different data sets.

5.4.1 Experiment on Four Clusters

Table 5.10 shows the class distribution obtained by bisecting k-means clustering algorithm over the four cluster data set.

Class Distribution				
Cluster	**Acc**	**CT**	**Eco**	**Educ**
0	165	6	63	2
1	8	1	3	313
2	1	255	29	2
3	5	1	351	4

Table 5. 10: Confusion matrix of bisecting K-Means over the 4 cluster's data set.

Table 5.10 indicates that Cluster 0, 1, 2 and 4 corresponds best to accident, education, culture and tourism, and economic sections respectively. However, some of the documents from the economic section are also misclassified to other clusters.

The result of the experimentation obtained by bisecting k-means over the four cluster data set is shown in Table 5.11.

4-way clustering: [1209 of 1209], **Entropy:** 0.240, **Purity:** 0.897							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	236	0.026	0.008	0.006	0.002	0.531	0.699
1	325	0.025	0.006	0.007	0.002	0.136	0.963
2	287	0.023	0.007	0.006	0.002	0.282	0.889
3	361	0.022	0.005	0.007	0.002	0.110	0.972

Table 5. 11: Clustering solution obtained by bisecting k-means for 4 clusters.

The result given in Table 5.11 above indicates that cluster 0 with the highest value of intra-cluster similarity (ISim) contains documents which are more internally cohesive than documents in other clusters. The value of the inter-cluster similarity (ESim) of the documents in cluster 0 and 2 is lower than cluster 1 and 3. This shows that the documents in cluster 0 and 2 are more externally isolated from the documents of other clusters.

As Table 5.11 shows, the entropy for cluster 0 is higher than others; implying that cluster 0 contains large number of documents from different sections, while cluster 3 with smaller value of entropy contains small number of documents different classes and it is the best cluster as compared to the others. The value of purity for cluster 3 is higher as compared with other clusters which also show that the cluster contains documents primarily from one class. The overall performance of the bisecting k-means for the 4 clustering solution achieved a value of 0.24, 0.897 and 0.024 entropy, purity and overall similarity respectively.

5.4.2 Experiment on Seven Clusters

In this experiment, seven clusters were considered. The confusion matrix of the bisecting k-means for the 7 cluster data set is shown in Table 5.12.

Class Distribution							
Cluster	**Acc**	**CT**	**Eco**	**Educ**	**Health**	**Law**	**Politics**
0	8	1	6	4	282	1	2
1	0	0	4	299	0	0	2
2	4	248	11	1	0	2	13
3	161	0	82	2	7	32	3
4	5	1	338	0	6	1	13
5	1	5	5	14	2	157	103
6	0	8	0	1	2	4	316

Table 5. 12: Confusion matrix of bisection K-Means for the 7 class's data set.

From Table 5.12, we can conclude that cluster 0, 1, 2, 3, 4 and 6 maps closely with health, education, culture and tourism, accident, economy and political section respectively. This is because most of the documents assigned to these clusters are from

those pre-defined classes. However, cluster 5 contains majority of documents from law and political sections. Table 5.12 also shows that some of the documents that belong to the pre-defined classes: economy and politics are distributed over the different clusters.

Table 5.13 shows the clustering result obtained by bisecting k-means clustering algorithm for the 7 clusters.

7-way clustering: [2157 of 2157], **Entropy:** 0.262, **Purity:** 0.835							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	304	0.032	0.010	0.006	0.002	0.190	0.928
1	305	0.030	0.008	0.006	0.002	0.056	0.980
2	279	0.025	0.008	0.005	0.002	0.253	0.889
3	287	0.024	0.008	0.005	0.002	0.565	0.561
4	364	0.025	0.006	0.006	0.002	0.178	0.929
5	287	0.022	0.006	0.006	0.002	0.535	0.547
6	331	0.019	0.007	0.003	0.001	0.121	0.955

Table 5. 13: Clustering solution of bisecting k-means for 7 clusters.

From Table 5.13, we can see that the documents in cluster 0 are more similar than documents in other clusters since the average internal similarity (ISim) between the documents is higher in cluster 0 as compared to the other clusters. Cluster 6 with the lowest value of the average external similarity (ESim) contains documents that are more separated from documents in other clusters.

Table 5.13 also shows that cluster 3 with the highest value of entropy contains large number of documents from different pre-defined classes. On the other hand, cluster 1 with smaller value of entropy contains homogeneous documents and it is the best cluster as compare to other clusters. Furthermore, cluster 1 with the highest value of purity shows that it contains majority of documents from a single pre-defined class. In general the overall clustering solution discovered by bisecting k-means for the 7 clusters achieved a value of 0.262, 0.835 and 0.025 entropy, purity and overall similarity respectively.

5.4.3 Experiment on Ten Clusters

In this section, ten pre-defined classes and the whole document collection were considered in the experiment. The set of documents assigned to each clusters from the predefined gold standard classes is shown in Table 5.14.

Class Distribution										
Cluster	**Acc**	**CT**	**Eco**	**Educ**	**Health**	**Law**	**Politics**	**Science**	**Social**	**Sport**
0	1	4	2	0	1	0	0	1	6	284
1	0	0	10	5	8	0	7	280	8	0
2	1	0	4	300	0	0	3	4	6	0
3	7	0	3	1	272	0	1	30	16	2
4	10	3	206	2	7	2	8	10	18	1
5	3	2	177	0	3	1	3	6	113	0
6	155	0	32	1	4	17	4	20	14	0
7	2	239	10	0	0	2	11	9	8	6
8	0	8	0	2	2	4	315	0	1	1
9	0	7	2	10	2	171	100	1	44	1

Table 5. 14: Class distribution of bisecting K-Means over the 10 clusters.

As shown in the clustering solution, most of the clusters contain documents assigned from a single pre-defined class. However, cluster 5 and 9 contain documents assigned from two pre-defined classes. Cluster 5 contains large number of documents assigned from economy and social sections, whereas cluster 9 is comprised of documents assigned from law and politics sections. From the class distribution, we can also see that most of the documents that belong to economy, politics, science and social sections are distributed over the different clusters as compared to the others.

The clustering solution obtained by bisecting k-means for 10 clusters over the whole data set is shown in Table 5.15.

10-way clustering: [3047 of 3047], **Entropy:** 0.313, **Purity:** 0.787							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	299	0.033	0.011	0.003	0.002	0.120	0.950
1	318	0.033	0.010	0.006	0.002	0.241	0.881
2	318	0.031	0.009	0.006	0.002	0.131	0.943
3	332	0.031	0.010	0.006	0.002	0.311	0.819
4	267	0.031	0.009	0.006	0.002	0.423	0.772
5	308	0.029	0.009	0.006	0.001	0.412	0.575
6	247	0.026	0.009	0.005	0.001	0.549	0.628
7	287	0.026	0.009	0.005	0.002	0.327	0.833
8	333	0.019	0.007	0.003	0.001	0.127	0.946
9	338	0.021	0.005	0.005	0.002	0.543	0.506

Table 5. 15: Clustering solution using bisecting k-means for 10 clusters.

According to the result given in Table 5.15 above, cluster 0 with high intra-cluster similarity (ISim) and low inter-cluster similarity (ESim) value is comprised of documents which are more similar to each other and dissimilar from the rest of documents in other clusters. This shows that cluster 0 is tighter and far away from the rest of the clusters.

The result given in Table 5.15 above also reveals that cluster 0 is the best clustering solution with smaller value of entropy and with larger purity value, while cluster 9 is the worst clustering solution with the highest and the lowest value of entropy and purity respectively. The overall clustering solution yielded by bisecting k-means matches the gold standard classes with 31.3%, of entropy and 78.7% of purity and it achieved 0.028 overall similarity values.

5.5 Agglomerative Hierarchical Clustering

The performances of the three agglomerative clustering functions: single link, complete link and average link were also tested at increasing number of clusters and documents. The experiments were done on four, seven and ten number of clusters over the corresponding pre-defined data sets and the results obtained is shown in Table 5.16.

	Single link			**Complete link**			**Average link**		
K	Entropy	Purity	Overall similarity	Entropy	Purity	Overall similarity	Entropy	Purity	Overall similarity
4	0.96	0.371	0.013	0.95	0.396	0.016	0.548	0.667	0.02
7	0.97	0.211	0.011	0.906	0.255	0.015	0.668	0.488	0.018
10	0.978	0.152	0.011	0.93	0.207	0.018	0.654	0.469	0.021

Table 5. 16: Performances of single link, complete link and average link in terms of entropy, purity and overall similarity measures for 4, 7 and 10 clusters over different data sets.

The three agglomerative hierarchical clustering algorithms produced clusters that are not similar to pre-defined classes. Moreover the total number of documents assigned to the clusters greatly varies since most of the documents from different pre-defined classes are assigned to a single cluster.

As observed from Table 5.16 above, among the results of agglomerative clustering algorithms, the average-link performed the best in terms of entropy, purity and overall similarity evaluation measures in all number of clusters. This is because in the average link, the similarity of two clusters is measured by considering all the documents in both clusters.

The quality of the overall clustering solution of both the single link and the complete link is very poor. The reason for the poor performance of single-link is that this algorithm assigns each document to the cluster of its nearest neighbor. However, any two

documents may share many of the same terms and be nearest neighbors without belonging to the same cluster. In addition, the performance of complete-link is also low since it is based on the assumption that all the documents in the cluster are very similar to each other.

Since the performance of the average linkage is better than the single link and complete link, the researcher run it over the whole data set many times to test whether the number of categories currently used by ENA matches with the number of categories revealed by clustering algorithms.

According to the experiment, the number of categories or clusters revealed by the average link is different from the number of categories currently used by ENA. According to the clustering solution produced by the average link, the whole data set which is pre-classified into 10 classes by ENA, is clustered into 15 clusters with better performance in terms of entropy, purity and overall similarity evaluation measures.

The average link achieved an entropy value of 0.654, purity value of 0.469 and an overall similarity value of 0.021 for 10 clusters, while for 15 clusters it achieved an entropy value of 0.555, purity value of 0.553 and an overall similarity value of 0.027. The clustering solutions obtained by the average link for 10 and 15 clusters are presented in appendix 9 and 10 respectively.

Once the correct number of clusters is identified, the bisecting k-means was used to determine the labeling of the discovered clusters. The cluster labels for the discovered clusters were assigned based on the descriptive and discriminating features of the clusters and the class labels provided by ENA (see the major and sub categories of News items in appendix 11). The clustering solution, the descriptive and discriminating features and the cluster labels for the 15 clusters obtained by bisecting k-means are also presented in appendix 12, 13 and 14 respectively.

5.6 Performance at Increasing Number of Clusters and Documents

The performances of k-means, bisecting k-means, single link, complete link and average link were tested for number of clusters equal to the number of pre-defined classes. To test their performances with increasing number of clusters and documents, the experiments were done on four, seven and ten clusters using different data sets. Figure 5.2 shows the performances of k-means, bisecting k-means, single link, complete link and average link in terms of entropy at increasing number of clusters and documents.

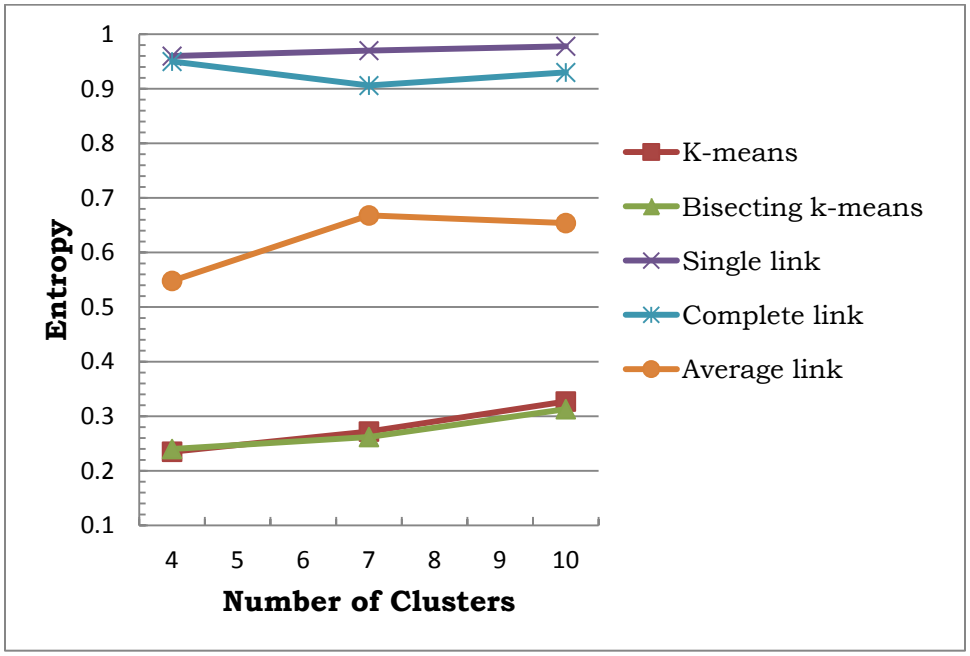

Figure 5. 2: Performances of k-means, bisecting k-means, single link, complete link and average link in terms of entropy at increasing number of clusters and documents.

As shown in Figure 5.2, the entropy values for k-means, bisecting k-means and single link are increased at increasing number of clusters and documents. The complete link achieved the highest entropy value on cluster 4 and the lowest entropy value on cluster 7, while the average link achieved the highest and the lowest entropy values over the 7 and 4 clustering solutions respectively.

Figure 5.3 displays the performances of k-means, bisecting k-means, single link, complete link and average link in terms of purity evaluation measure at increasing number of clusters and documents.

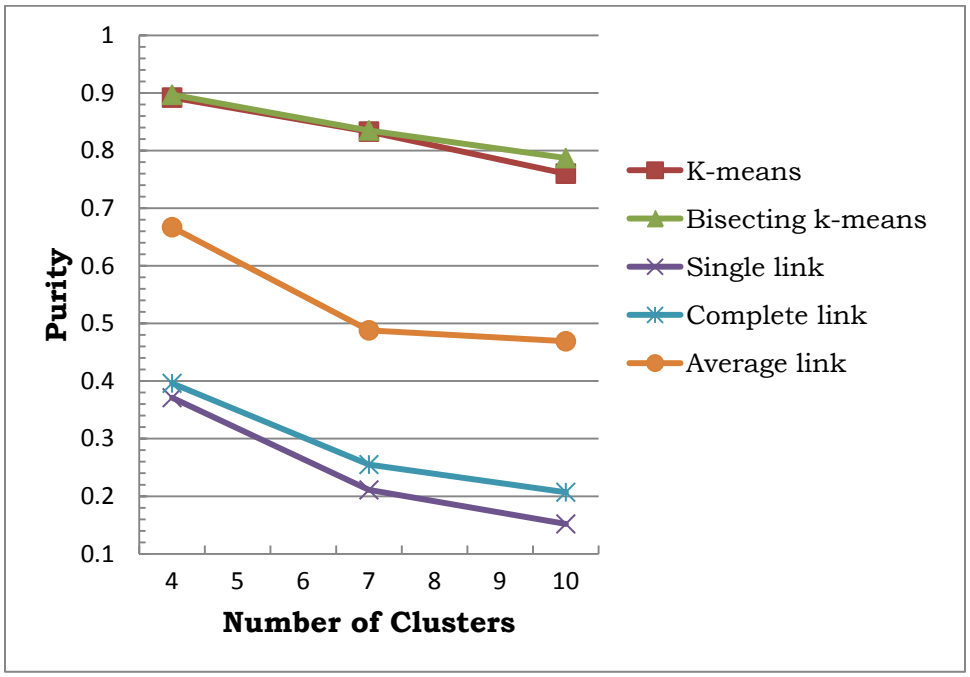

Figure 5. 3: Performances of k-means, bisecting k-means, single link, complete link and average link in terms purity at increasing number of clusters and documents.

Figure 5.3 shows that the value of purity for all clustering algorithms: k-means, bisecting k-means, single link, complete link and average link decrease as the number of clusters and documents increase. This indicates that the overall quality of the clustering solution do not match better with predefined classes at increasing number of clusters and documents. This is because the documents from a single predefined class are distributed to different clusters.

Figures 5.4 displays the performances of k-means, bisecting k-means, single link, complete link and average link in terms of overall similarity evaluation measure at increasing number of clusters and documents.

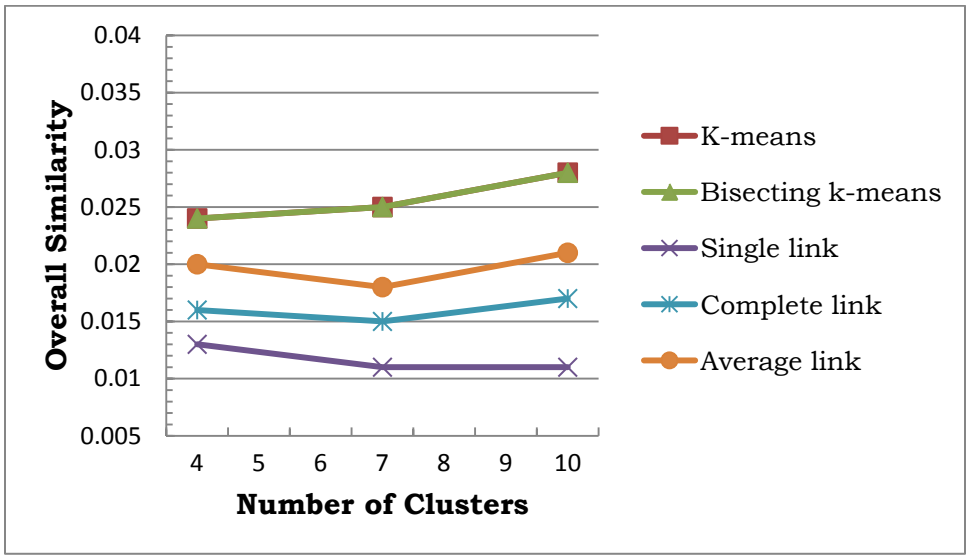

Figure 5. 4: Performances of k-means, bisecting k-means, single link, complete link and average link in terms of overall similarity at increasing number of clusters and documents.

Figure 5.4 shows that the value of the overall similarity of documents within a cluster increase as the number of clusters and documents increases for both k-means and bisecting k-means clustering algorithms. This indicates that the clustering solutions become more internally cohesive at increasing number of clusters and documents. On the other hand, the overall similarity values for single link, complete link and average link decrease from cluster 4 to cluster 7 and then it increase at cluster 10.

The results given in Figures 5.2, 5.3 and 5.4 above reveal that both k-means and bisecting k-means achieved the best clustering results in terms of entropy and purity over the 4 clustering solution. However, their performance in terms of overall similarity measure is poor. The performance on the 7 clustering solution is better than the 10 clustering solution in terms of entropy and purity. This shows that the performance of both k-means and bisecting k-means in terms of entropy and purity decrease at increasing number of

clusters and documents on different data sets. On the other hand, the overall similarity of documents within clusters increased as the number of clusters and documents increase.

From Figures 5.2, 5.3 and 5.4 above, we can also see that the value of purity for single link, complete link and average link decrease at increasing number of clusters and documents. The value of entropy for single link increases at increasing number of cluster and documents. However, the entropy value of complete link and average link and the overall similarity value of the three agglomerative functions do not depend at increasing number of clusters and documents.

5.7 Performance at Increasing Number of Clusters

The performances of k-means, bisecting k-means, single link, complete link and average link were also tested at increasing number of clusters using the same data set. Figures 5.5, 5.6 and 5.7 show the quality of the clustering solution in terms of entropy, purity and overall similarity evaluation measures over the 10 clusters data set.

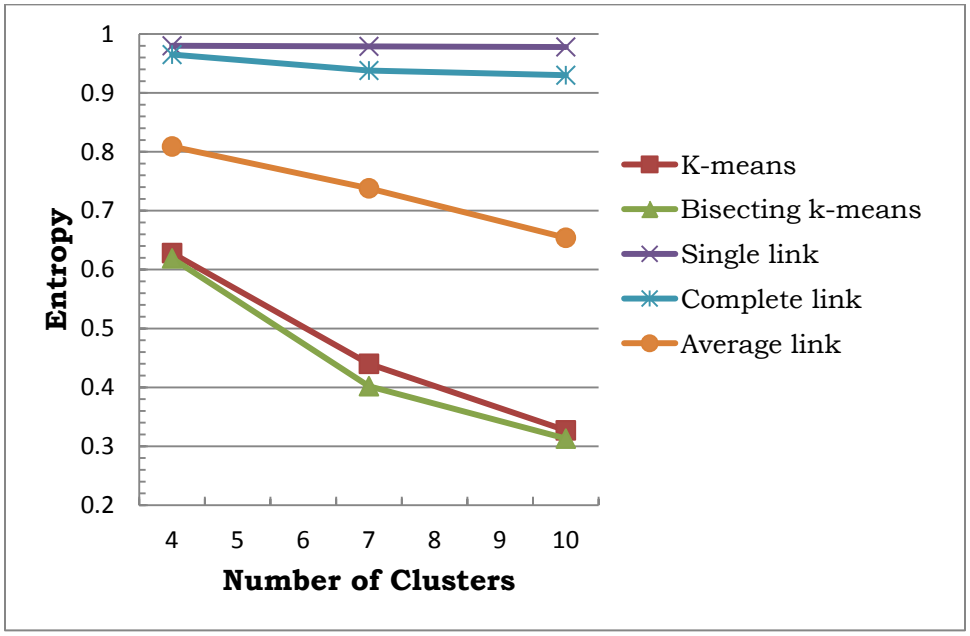

Figure 5. 5: Performances of k-means, bisecting k-means, single link, complete link and average link in terms of entropy at increasing number of clusters over the 10 cluster dataset.

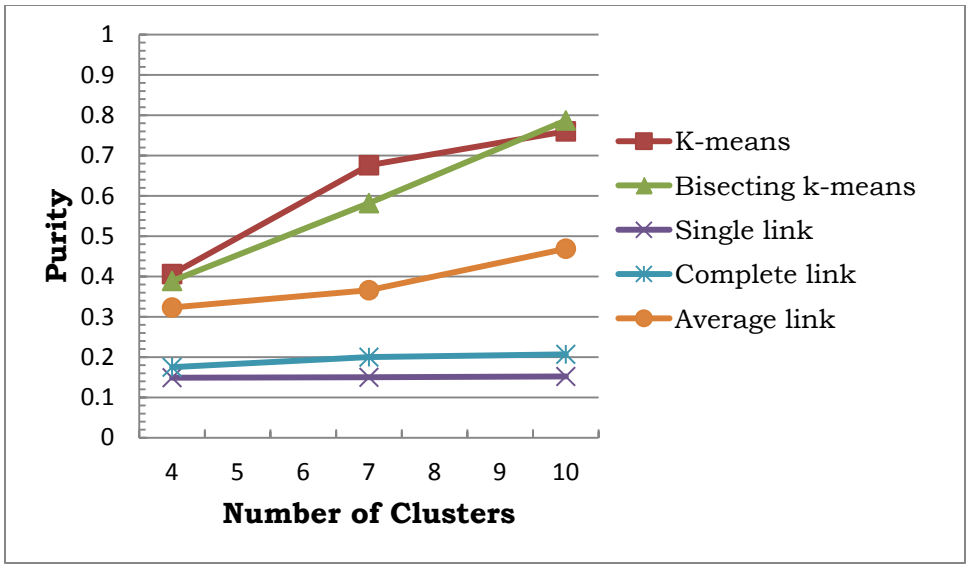

Figure 5. 6: Performances of k-means, bisecting k-means, single link, complete link and average link in terms of purity at increasing number of clusters over the 10 cluster dataset.

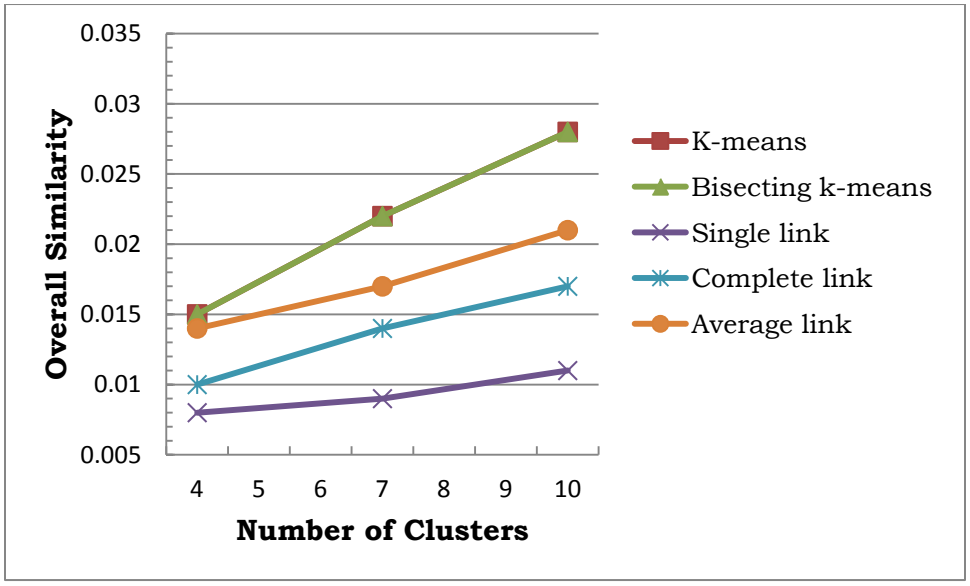

Figure 5. 7: Performances of k-means, bisecting k-means, single link, complete link and average link in terms of overall similarity measure at increasing number of clusters over the 10 cluster dataset.

As shown in Figures 5.5, 5.6 and 5.7, the values of entropy for all the clustering algorithms: k-means, bisecting k-means, single link, complete link and average link

decrease at increasing number of clusters, while the value of purity and overall similarity of these clustering algorithms increase as the number of clusters increase. This shows that as the number of clusters increase, the clustering solutions become more internally cohesive and externally isolated. Furthermore, the clustering results match better with the predefined classes.

5.8 Comparison of Clustering Algorithms

The performances of document clustering algorithms: k-means, bisecting k-means and average link were compared for the 4, 7 and 10 clustering solutions using entropy, purity and overall similarity evaluation metrics over the different pre-defined data sets. Table 5.17 shows the comparison of the clustering results obtained by k-means, bisecting k-means and the average link for the 4, 7 and 10 clusters.

Clustering Algorithms	K	Entropy	Purity	Overall Similarity
K-means	4	0.235	0.892	0.024
	7	0.272	0.833	0.025
	10	0.327	0.76	0.028
Bisecting K-means	4	0.24	0.897	0.024
	7	0.262	0.835	0.025
	10	0.313	0.787	0.028
Average link	4	0.548	0.667	0.02
	7	0.688	0.488	0.018
	10	0.654	0.469	0.021

Table 5. 17: Comparison of entropy, purity, and overall similarity values for k-means, bisecting k-means and average-link for 4, 7 and 10 clusters over the corresponding data sets.

As shown in Table 5.17 above, the bisecting k-means achieved the highest performance in terms of purity in all number of clusters, while k-means achieved the best entropy value in the 4 clustering solution. The performances of k-means and bisecting k-means are similar in terms of the overall similarity measure in all number of clusters. The k-

means algorithm performed better than the agglomerative clustering algorithms in terms of entropy, purity and overall similarity measures. The performances of agglomerative criterion functions are poor as compared to both k-means and bisecting k-means clustering algorithms in all evaluation metrics.

CHAPERT SIX

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

In this study, the potential application of unsupervised learning techniques for the classification of Amharic text News documents was explored and is both feasible and crucial. The effect of the number of clusters and the size of documents used on the performance and efficiency of clustering algorithms was tested and compared using different data sets. Moreover, the performances of these clustering algorithms were also tested at increasing number of clusters using the same data set. The agreement between the number of predefined classes and the number of clusters discovered by the agglomerative clustering algorithm was also tested for 10 clusters over the whole document collection.

Based on the experiments done in this thesis, the following concluding remarks were made.

- As the number of clusters and documents increase, the clustering solutions produced by k-means and bisecting k-means become more internally cohesive and externally isolated. However, the clustering results do not match better with the pre-defined classes and requires relatively high computational requirements. Moreover; the purity values of single link, complete link and average link decrease. According to the results obtained, it was difficult to determine the entropy and the overall similarity values of the three agglomerative approaches at increasing number of clusters and documents.

- All the clustering algorithms: k-means, bisecting k-means, single link, complete link and average link achieved better clustering quality as the number of clusters increases with the same data set. The clustering solutions became more internally cohesive, externally isolated and match better with the pre-defined classes.

- The performances of k-means and bisecting k-means are similar in terms of the overall similarity measure in all number of clusters and they produced similar clustering solutions. However, the results of the findings indicate that the bisecting k-means produced better clustering solutions consistently according to the entropy and purity evaluation measures.

- The results also shows that k-means and bisecting k-means clustering algorithms consistently produced clusters that are most similar to pre-defined classes at different data sets. Moreover, both k-means and bisecting k-means clustering algorithms produced clusters relatively with similar cluster size (number of documents), while the agglomerative hierarchical clustering algorithms generally produced clusters that are not similar to pre-defined classes and clusters with unbalanced cluster size (number of documents).

- Agglomerative hierarchical clustering algorithms produced low quality results as compared to the k-means and the bisecting k-means clustering algorithms. Among the agglomerative clustering algorithms, the average link achieved the best performance as compared to single link and complete link in all evaluation measures.

- According to the clustering results, there is a mismatch between the news categories provided by ENA and the categories or clusters discovered by clustering algorithms.

In general, the findings of this research show that, for Amharic text document clustering, k-means and bisecting k-means are more appropriate than the agglomerative hierarchical clustering algorithms both in terms of relatively low time requirements and the quality of the clusters produced in both the internal and external evaluation measures.

6.2 Recommendations

This study shows the potential application of unsupervised machine learning techniques to the analysis of textual Amharic documents is both feasible and crucial. However, recommendations for further research are forwarded to improve the performance of document classification and to explore all algorithms and applications of unsupervised document classification especially for local languages. Thus, the recommendations forwarded are organized as follows.

- As to the researcher's knowledge, there is no standard corpus open for researchers to apply different machine learning approaches. Researchers can devote much time on their work and explore more if standard corpus is prepared for Amharic classification experiments like 'Reuters-21578' for English.

- The news specific stop word lists used in this research may not be helpful in other areas or domains of applications such as research papers. Therefore, an exhaustive and standard stop word list for Amharic language should be developed. In addition, due to the various problems of the Amharic writing system, it was difficult to develop a full fledged Amharic document preprocessing system for Amharic language. Developing a full fledged Amharic document preprocessing system might increase the performances of different machine learning approaches in text classification and clustering.

- The bag of words representation approach which describes each document with its most significant terms was used in this thesis. However, future researchers may consider different document representation approaches such as phrase based and ontology based representations to select index or representative terms.

- The choice of cluster labels is another issue that needs further investigation. In this study, the most descriptive and discriminating features or terms of clusters are listed. However, in some cases, these terms belong to the same noun phrase, such

as the name of a person or place and can not describe the cluster labels correctly. This suggests that the phrase based approach might be more appropriate than bag of words representation for cluster labeling.

- Unsupervised classification can also works hand-in-hand with the supervised learning techniques when the number of categories is not known in advance. Hence, future researchers can explore the use of clustering to select the documents that are best representatives of a category to train the classifier.

- Feature researchers can also compare the performance of unsupervised learning approaches with the supervised approaches using the same preprocessing techniques, evaluation methods and document collections.

- Currently, few researches were conducted on automatic Amharic news classification and the results of the researches are promising. However, ENA still uses manual classification of News. So, it is better for the agency to review the different research works and to start the implementation of automatic classification of news.

- There is also a mismatch between the news categories provided by ENA and the clusters discovered by automatic clustering algorithms. Hence, it is better for the agency to revisit it news categories based on these findings.

- A number of researches were done on Amharic text document classification. However, as to the knowledge of the researcher, all the previous studies were conducted using Amharic text news item only. Future researchers can also explore document classification techniques to various real world problems such as classification and clustering of general documents. Moreover, document classification and clustering techniques can also be extended to other local languages if huge collection of documents is available.

REFERENCES

1. Aha, D. (1995) Machine learning: An annotated Bibliography for the 1995 AI and Statistics tutorial on machine learning. *In Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*, Fort Lauderdale, Florida, USA.
2. Alemu, K. (2010) Hierarchical Amharic text news classification: Master's Thesis. Department of Information Science, Addis Ababa University, Ethiopia.
3. Amine, A., Elberrichi, Z. and Simonet, M. (2010) Evaluation of Text Clustering Methods Using WordNet: *The International Arab Journal of Information Technology*, Vol. 7, no. 4.
4. Atelach, A. (2002) Automatic Sentence Parsing for Amharic Text an Experiment Using Probabilistic Context Free Grammars: Master's Thesis. Addis Ababa University, Ethiopia.
5. Baeza-Yates, R. and Ribeiro-Neto, B. (1999) *Modern Information Retrieval*: Addison- Wesley: New York.
6. Bender, M., Bowen, J., Cooper, R. and Ferguson, C. (1976) *Language in Ethiopia:* Oxford University Press: London.
7. Beil, F., Ester, M. and Xu, X. (2002) Frequent term-based text clustering: *In KDD '02: Proc. of the 8th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 436–442, New York, NY, USA. ACM. ISBN 1-58113-567-X.
8. Berkhin, P. (2002) Survey of Clustering Data Mining Techniques. Research paper, Accessed on December 20, 2010. Web site: http://www.accrue.com/products/researchpapers.html
9. Blumberg, R. and Atre, S. (2003) Automatic classification: moving to the mainstream. Accessed on December 20, 2010. Web site: www.soquelgroup.com/articles/dmreview0403 classification.pdf.
10. Bradley, P. and Fayyad, U. (1998) Refining Initial Points for K-Means Clustering: *Proceedings of the Fifteenth International Conference on Machine Learning ICML98*, PP. 91-99. Morgan Kaufmann, San Francisco.
11. Carrasco, A. (2007) Unsupervised Classification of Text Documents: Master thesis, Scientific Computing, university of puerto rico, mayagüez campus.

12. Census Summery of Ethiopia, (2007). The Federal Democratic Republic of Ethiopia Central Statistics Agency. Accessed on March 02, 2011. Website: http://www.csa.gov.et/census-summary-final-report.pdf
13. Chen, Y., Qin, B., Liu, T., Liu, Y. and Li, S. (2010) The Comparison of SOM and K-means for Text Clustering: *Computer and Information Science*, V.3, no.3, pp. 268-274.
14. Daniel, Y. (1996) Frequently Asked Questions about SERA: Accessed on March 02, 2011. Web site: http://www.abyssiniacybergateway.net/fidel/sera-faq.txt.
15. Eiring, H. (2002) The evolving information overload, *information management formal journal*, Vol. 36, no. 2002, pp. 20-24.
16. Fayyad, U. and Piatetsky, S. (1996) *From Data Mining to Knowledge Discovery: An Overview*, Advances in Knowledge Discovery and Data Mining, ISBN 978-0-262-56097-9, MIT Press, Cambridge, Mass, pp.1–34.
17. Jain, A. Murty, M. and Flynn, P. (1999) Data clustering: A review. *ACM Computing Surveys*, Vol. 31, no. 3, pp. 264–323.
18. Jain, A. (2008) Data Clustering: 50 Years Beyond K-Means, Department of Computer Science & Engineering, Michigan State University, USA.
19. Hammouda, K. (2001) *Web Mining: Identifying Document Structure for Web Document Clustering:* Master's Thesis, Department of Systems Design Engineering, University of Waterloo, Ontario, Canada.
20. Han, J. and Kamber, M. (2001) *Data Mining: Concepts and Techniques:* Morgan Kaufmann Publishers.
21. Ho, T. (1999) Fast Identification of Stop Words for Font Learning and Keyword Spotting: *In Proceedings of the Fifth International Conference on Document Analysis and Recognition*, (PP. 333-336), IEEE Computer Society.
22. Hotho, A., Nurnberger, A., Paab, G. and Ais, F. (2005) A Brief Survey of Text Mining: Knowledge Discovery Group, School of Computer Science, University of Kassel.

23. Karanikas, H., Koundourakis, G., Kopanakis, I., Mavroudakis, T. and Pelekis, N. (2002) Discovering market trends in the biotechnology industry: *Int. J. Business Intelligence and Data Mining*, Vol. 2, no. 2.
24. Karypis, G. (2003) CLUTO: A Clustering Toolkit (Release 2.1.1): Technical Report, #02-017, Department of Computer Science, University of Minnesota.
25. Khan, A., Baharudin, B., lee, H. and Khan, K. (2009) A Review of Machine Learning Algorithms for Text-Documents Classification: *Journal of Advances in Information Technology*, vol. 1, no. 1
26. Ko, Y. and Seo, J. (2000) Automatic Text Categorization by Unsupervised Learning: Department of Computer Science, Sogang University, Korea.
27. Koller, D. and Sahami, M. (1997) Hierarchically classifying documents using very few words: *Proceedings of the 14th International Conference on Machine Learning (ML)*, PP. 170-178. Stanford University, United Kingdom.
28. Kumar, V., Tan, P. and Steinbach, M. (2005) *Introduction to Data Mining*, Addison-Wesley.
29. Krishnakumar, A. (2006) Text Categorization: Building a KNN Classifier for the Reuters-21578 Collection. Accessed on February 22, 2011. Web site: http://en.scientificcommons.org/42606011.
30. Liu, H. and Motoda H., (1998) *Feature Extraction, constraction and selection: A Data Mining Perspective: Boston, Massachusetts (MA*): Kluwer Academic Publishers.
31. Manning, C., Raghavan, P. and Schutze, H. (2009) *An Introduction to Information Retrieval*: Cambridge university press, Cambridge, England.
32. Massey, L. (2004) Evaluating and Comparing Text Clustering Results: Royal Military College, Canada.
33. Michie, D. (1991) Methodologies from machine learning in data analysis and software. *The Computer Journal*, Vol. 34, no.5, pp. 59-565.
34. Nega, A. and Willett, P. (2002) Stemming of Amharic Words for Information Retrieval: *Literary and Linguistic Computing*, Vol. 17, no.1, pp. 1-17.

35. Nigam, K., Mccallum, A., Thrun, S. and Mitchel, T. (2000) Text Classification from Labeled and Unlabeled Documents using EM: *Machine learning*, 39, pp. 103-134.
36. Nicholas, O., Andrews, A. and Edward, A. (2007) Recent Developments in Document Clustering, Department of Computer Science, Virginia Tech.
37. Ozgur, A. (2004) Supervised and unsupervised machine learning techniques for text document categorization: Master thesis, Computer Engineering, Bogazifici University, Turkey.
38. Rasmussen, M. and Karypis, G. (2004) gCLUTO – An Interactive Clustering, Visualization, and Analysis System: CSE/UMN Technical Report. University of Minnesota, Department of Computer Science and Engineering,
39. Rasmussen, E. (1992) *Clustering Algorithms: In Data Structures and Algorithm.* Prentice Hall PTR.
40. Rennie, D. (2001) Improving Multi-Class Text Classification with Naive Bayes: Masters Thesis. Massachusetts Institute of Technology.
41. Rosell, M. (2006) Introduction to Information Retrieval and Text Clustering: KTH School of Computer Science and Communication, Stockholm, Sweden.
42. Rosell, M. (2009) Text Clustering Exploration, Swedish Text Representation and Clustering Results Unraveled: Doctoral Thesis, Stockholm, Sweden.
43. Sebastiani, F. (2002) Machine Learning in Automated Text Categorization: Computer *Journal of ACM Computing Surveys*, Vol. 34, no. 1, pp. 1-47.
44. Simon, H. (1983) *Machine Learning: An artificial intelligence approach.* Morgan Kaufmann, San Mateo, CA.
45. Steinbach, M., Karypis G., and Kumar, V. (2000) A Comparison of Document Clustering Techniques: In *KDD Workshop on Text Mining*, Boston, MA, USA.
46. Surafel, T. (2003) Automatic categorization of Amharic news text: A machine learning approach. Masters Thesis. Department of Information Science, Addis Ababa University, Ethiopia.
47. Tessema, M., Meron, S. and Teshome, K. (2009) The Need for Amharic WordNet: Computer Science Department, Addis Ababa University and Ministry of Finance and Economic Development, Ethiopia.

48. Thangamani, M. and Thangaraj, P. (2010) Integrated Clustering and Feature Selection Scheme for Text Documents: *Journal of Computer Science* 6 (5): 536-541.
49. Veeramachaneni, S., Sona, D. and Avesani, P. (2005) Hierarchical Dirichlet model for document Classification: *Proceedings of the 2^{2nd} International Conference on Machine Learning*, Bonn, Germany.
50. Wapedia (2009). **አማርኛ**. Accessed on March 02, 2011. Web site: http://wapedia.mobi/am/.
51. Wilbur, J. and Sirotkin, K. (1992) The automatic identification of stop words: *Journal of Information Science,* 18, pp. 45-55.
52. Worku, K. (2009) Amharic text news classification: A neural network approach. Masters Thesis. Department of Information Science, Addis Ababa University, Ethiopia.
53. Yang, K. (2004) Literature review of dissertation: Accessed on December 18, 2010 from http://www.ils.unc.edu/yangk/dissertation/litrevcontent.html
54. Yang, Y. and Pedersen, P. (1997) A Comparative Study on Feature Selection in Text Categorization: The Fourteenth International Conference on Machine Learning (ICML). pages 412-420, Nashville, TN.
55. Yohannes, A. (2007) *Amharic news text classification using SVM approach*: Masters Thesis, Department of Information Science, Addis Ababa University, Ethiopia.
56. Zaghloul, W. (2005) Text classification: neural networks vs. support vector machines, journal of Industrial Management & Data Systems, Vol. 109, No. 5 pp. 708- 717, ISSN: 0263-5577, Emerald Group Publishing Limited.
57. Zelalem, S. (2001) *Automatic Amharic news text classification: Statistical approach*, Masters Thesis. Department of Information Science, Addis Ababa University, Ethiopia.
58. Zeleke, A. (2010) *Phrase based Amharic text news classification*: Masters Thesis, Department of Information Science, Addis Ababa University, Ethiopia.
59. Zhao, Y. and Karypis, G. (2002) Evaluation of hierarchical clustering algorithms for document datasets. *In Proceedings of International Conference on Information and Knowledge Management* (*CIKM*), pp. 515–524.

APPENDIX

Appendix 1: Interview questions for the ICT Coordinator of ENA

1. How the news are created in ENA and distributed for users?
2. What kind of technology do ENA use for managing the news? For what purpose the use is/are?
3. How the news items are classified into their respective categories? Is it manually or automated?

Appendix 2: Amharic characters (‘Fidel’) (Zelalem, 2001)

Order							Labialized				
1st	2nd	3rd	4th	5th	6th	7th					
ሀ ha	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			ሏ $l^{w}a$		
ሐ ha	ሑ hu	ሒ hi	ሓ ha	ሔ he	ሕ h	ሖ ho					
መ mä	ሙ mu	ሚ mi	ማ ma	ሜ me	ም m	ሞ mo			ሟ $m^{w}a$		
ሠ sä	ሡ su	ሢ si	ሣ sa	ሤ se	ሥ s	ሦ so					
ረ rä	ሩ ru	ሪ ri	ራ ra	ሬ re	ር r	ሮ ro			ሯ $r^{w}a$		
ሰ sä	ሱ su	ሲ si	ሳ sa	ሴ se	ስ s	ሶ so			ሷ $s^{w}a$		
ሸ šä	ሹ šu	ሺ ši	ሻ ša	ሼ še	ሽ š	ሾ šo			ሿ $š^{w}a$		
ቀ qä	ቁ qu	ቂ qi	ቃ qa	ቄ qe	ቅ q	ቆ qo	ቈ $q^{w}ä$	ቍ $q^{w}i$	ቋ $q^{w}a$	ቌ | $q^{w}e$	ቍ $q^{w}ə$
በ bä	ቡ bu	ቢ bi	ባ ba	ቤ be	ብ b	ቦ bo			ቧ $b^{w}a$		
ተ tä	ቱ tu	ቲ ti	ታ ta	ቴ te	ት t	ቶ to			ቷ $t^{w}a$		
ቸ čä	ቹču	ቺ či	ቻ ča	ቼ če	ች č	ቾ čo			ቿ $č^{w}a$		
ኀ hä	ኁ hu	ኂ hi	ኃ ha	ኄ he	ኅ h	ኆ ho	ኈ $h^{w}ä$	ኊ $h^{w}i$	ኋ $h^{w}a$	ኌ $h^{w}e$	ኍ $h^{w}ə$
ነ nä	ኑ nu	ኒ ni	ና na	ኔ ne	ን n	ኖ no			ኗ $n^{w}a$		
ኘ ñä	ኙ ñu	ኚ ñi	ኛ ña	ኜ ñe	ኝ ñ	ኞ ño			ኟ $ñ^{w}a$		
አ a	ኡ u	ኢ i	ኣ a	ኤ e	እ ə	ኦ o					
ወ wä	ዉ wu	ዊ wi	ዋ wa	ዌ we	ው w	ዎ wo					
ዐ a	ዑ u	ዒ i	ዓ a	ዔ e	ዕ ə	ዖ o					
ከ kä	ኩku	ኪ ki	ካ ka	ኬ ke	ክ k	ኮ ko	ኰ $k^{w}ä$	ኲ $k^{w}i$	ኳ $k^{w}a$	ኴ $k^{w}e$	ኵ $k^{w}ə$
ኸ hä	ኹhu	ኺhi	ኻha	ኼhe	ኽ h	ኾ ho					
ዘ zä	ዙ zu	ዚ zi	ዛ za	ዜ ze	ዝ z	ዞ zo			ዟ $z^{w}a$		
ዠžä	ዠžu	ዢži	ዣža	ዤže	ዥ ž	ዦ žo					
የ yä	ዩ yu	ዪyi	ያ ya	ዬ ye	ይ y	ዮ yo					
ገ gä	ጉ gu	ጊ gi	ጋ ga	ጌ ge	ግ g	ጎ go	ጐ $g^{w}ä$	ጒ $g^{w}i$	ጓ $g^{w}a$	ጔ $g^{w}e$	ጕ $g^{w}ə$
ጀ dä	ጁdu	ጂ di	ጃ da	ጄde	ጅ d	ጆ do			ጇ $d^{w}a$		
ጘğä	ጘğu	ጚği	ጛğa	ጜğe	ጝğ	ጞğo					
ጠṭä	ጡṭu	ጢṭi	ጣṭa	ጤṭe	ጥṭ	ጦṭo			ጧ $ṭ^{w}a$		
ጨćä	ጨću	ጨći	ጫća	ጨće	ጭ ć	ጮćo			ጯ $ć^{w}a$		
ጸ şä	ጹ şu	ጺ şi	ጻ şa	ጼ şe	ጽ ş	ጾ şo			ጿ $ş^{w}a$		
ፀ şä	ፁ şu	ፂ şi	ፃ şa	ፄ şe	ፅ ş	ፆ şo					
ፈ pä	ፉ pu	ፊ pi	ፋ pa	ፌ pe	ፍ p	ፎ po					
ፈ fä	ፉ fu	ፊ fi	ፋ fa	ፌ fe	ፍ f	ፎ fo			ፏ $f^{w}a$		
ፐ pä	ፑ pu	ፒ pi	ፓ pa	ፔ pe	ፕ p	ፖ po					
ቨ vä	ቩ vu	ቪ vi	ቫ va	ቬ ve	ቭ v	ቮ vo					

Appendix 3: Amharic punctuation marks (Atelach, 2002)

No.	Punctuation mark	Symbol	Purpose
1	The four dots or double colon	::	Mark end of a sentence
2	Colon	:	Separate words in a sentence: not common
3	White space		Separate words in a sentence: current practice
4	Question mark	?	Placed at the end of questions
5	Exclamation mark	!	Used at the end of sentences that show exclamation
6	Comma	፣	Used like comma
7	Semi-colon	፤	Used like semi-column
8	Three dots	...	For deliberate omission of words, phrases, or sentences
9	Quotation marks	‹‹ ››	Used at the beginning and at the end of quoted word, phrase, etc.
10	Parenthesis	( )	To enclose elaboration
11	Stroke	/	Separate date, month, etc.
12	Mocking mark	፥	Placed at the end of mocking sentence

Appendix 4: Amharic numbers (Zelalem, 2001)

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

Appendix 5: Lists of abbreviations and their expanded form.

Abbreviation	Expanded Form	Abbreviation	Expanded Form
መ/ቤት	መሥሪያ ቤት	ወ/ሮ	ወይዘሮ
ሚ/ሩ	ሚኒስትሩ	ዓ/ም	ዓመተ ምህረት
ም/	ምክትል	ዓ/ዓ	ዓመተ ዓለም
ም/ቤት	ምክር ቤት	ዶ/ር	ዶክተር
ሠ/ፌዴሬሽን	ሠራተኛ ፌዴሬሽን	ጀ/ል	ጀኔራል
ሻ/	ሻምበል	ገ/	ገብረ
ቤ/መ	ቤተመንግስት	ጠ/ሚ	ጠቅላይ ሚኒስትር
ቤ/ክ	ቤተክርስትያን	ጠ/ሚኒስትር	ጠቅላይ ሚኒስትር
ተ/	ተክለ	ጠ/ሚ/ቢሮ	ጠቅላይ ሚኒስትር ቢሮ
ኃ/	ኃይለ	ጽ/ቤት	ጽህፈት ቤት
አ/አ	አዲስ አባባ	ፍ/ቤት	ፍርድ ቤት
ኮ/ል	ኮለኔል	ፕ/ር	ፕሮፌሰር
ወ/	ወልደ	ፕ/ት	ፕሬዚዳንት
ወ/ሪት	ወይዘሪት		

The abbreviation lists are collected from http://nlp.amharic.org/resources/lexical/word lists/abbreviations/Lingua-AM-Abbreviate-19990516.txt.

Appendix 6: SERA transliteration table.

ሀ	he	ዠ	Ze	ቩ	vu	ፉ	fu
ለ	le	የ	ye	ቱ	tu	ፐ	pu
ሐ	He	ደ	de	ቹ	cu	ሂ	hi
መ	me	ጀ	je	ኍ	hu	ሊ	li
ሠ	se	ገ	ge	ኑ	nu	ሒ	Hi
ረ	re	ጠ	Te	ኙ	Nu	ሚ	mi
ሰ	se	ጨ	Ce	ኡ	`u	ሢ	si
ሸ	xe	ጰ	Pe	ኩ	ku	ሪ	ri
ቀ	qe	ጸ	Se	ኹ	`ku	ሲ	si
በ	be	ፀ	Se	ዉ	wu	ሺ	xi
ቨ	ve	ፈ	fe	ዑ	`u	ቂ	qi
ተ	te	ፐ	pe	ዙ	zu	ቢ	bi
ቸ	ce	ሁ	hu	ዧ	Zu	ቪ	vi
ኀ	he	ሉ	lu	ዩ	yu	ቲ	ti
ነ	ne	ሑ	Hu	ዱ	du	ቺ	ci
ኘ	Ne	ሙ	mu	ጁ	ju	ጊ	hi
አ	`a	ሡ	su	ጉ	gu	ኒ	ni
ከ	ke	ሩ	ru	ጡ	Tu	ኚ	Ni
ኸ	`ke	ሱ	su	ጩ	Cu	ኢ	`i
ወ	we	ሹ	xu	ጱ	Pu	ኪ	ki
ዐ	`e	ቁ	qu	ጹ	Su	ኺ	`ki
ዘ	ze	ቡ	bu	ፁ	Su	ዊ	wi

ዒ	`i	ባ	ba	ፉ	fa	ዜ	zE
ዚ	zi	ቫ	va	ፓ	pa	ዧ	ZE
ዣ	Zi	ታ	ta	ሄ	hE	ዬ	yE
ዩ	yi	ቻ	ca	ሌ	lE	ዶ	dE
ዲ	di	ኃ	ha	ሐ	HE	ጀ	jE
ጂ	ji	ና	na	ሜ	mE	ገ	gE
ጊ	gi	ኛ	Na	ሤ	sE	ጤ	TE
ጢ	Ti	ኣ	`a	ሬ	rE	ጩ	CE
ጪ	Ci	ካ	ka	ሴ	sE	ጴ	PE
ጲ	Pi	ኻ	`ka	ሼ	xE	ጼ	SE
ጺ	Si	ዋ	wa	ቄ	qE	ፄ	SE
ፂ	Si	ዓ	`a	ቤ	bE	ፌ	fE
ፊ	fi	ዛ	za	ቬ	vE	ፔ	pE
ፒ	pi	ዣ	Za	ቴ	tE	ህ	h
ሃ	ha	ያ	ya	ቼ	cE	ል	l
ላ	la	ዳ	da	ኈ	hE	ሕ	H
ሓ	Ha	ጃ	ja	ኔ	nE	ም	m
ማ	ma	ጋ	ga	ኜ	NE	ሥ	s
ሣ	sa	ጣ	Ta	ኤ	`E	ር	r
ራ	ra	ጫ	Ca	ኬ	kE	ስ	s
ሳ	sa	ጳ	Pa	ኼ	`kE	ሽ	x
ሻ	xa	ጻ	Sa	ዌ	wE	ቅ	q
ቃ	qa	ፃ	Sa	ዔ	`E	ብ	b

ቭ	v	ፕ	p	ፖ	Zo	ቿ	cWa
ት	t	ሆ	ho	ዮ	yo	ኈ	hWe
ች	c	ሎ	lo	ዶ	do	ኗ	nWa
ኀ	h	ሐ	Ho	ጆ	jo	ኟ	NWa
ን	n	ሞ	mo	ጎ	go	ኰ	kWe
ኝ	N	ሦ	so	ጠ	To	ዟ	zWa
እ	`I	ሮ	ro	ጮ	Co	ጿ	ZWa
ክ	k	ሶ	so	ጰ	Po	ዷ	dWa
ኽ	`k	ሾ	xo	ጸ	So	ጇ	jWa
ው	w	ቆ	qo	ፆ	So	ጐ	gWe
ዕ	`I	ቦ	bo	ፎ	fo	ጧ	TWa
ዝ	z	ቮ	vo	ፖ	po	ቋ	CWa
ዥ	Z	ቶ	to	ሏ	lWa	ፏ	PWa
ይ	y	ቾ	co	ሗ	HWa	ፗ	SWa
ድ	d	ኖ	`ho	ሟ	mWa	ፏ	fWa
ጅ	j	ኖ	no	ሧ	sWa	ፐ	pWa
ግ	g	ኯ	No	ሯ	rWa	ቈ	qWu
ጥ	T	ኦ	`o	ሷ	sWa	ኈ	hWu
ጭ	C	ኮ	ko	ኗ	xWa	ኰ	kWu
ጵ	P	ኾ	`ko	ቈ	qWe	ጐ	gWu
ጽ	S	ዎ	wo	ቧ	bWa	ቊ	qWi
ፅ	S	ዖ	`o	ቫ	vWa	ኈ	hWi
ፍ	f	ዘ	zo	ቷ	tWa	ኰ	kWi

ጒ	gWi
ቋ	qWa
ኋ	hWa
ኳ	kWa
ጓ	gWa
ቌ	qWE
ኌ	hWE
ኴ	kWE
ጔ	gWE
ኧ	ea
።	.
::	.
፣	,
፤	;
፥	:
፦	:-
፧	?
?	?
/	/
:	:
.	.
"	"
-	-
#	#

$	$
%	%
*	*
(	(
)	)
–	–
+	+
=	=
\	\
{	{
}	}
<	<
>	>
~	~
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

Appendix 7: Amharic characters with the same sound and their transliterations.

Characters with the same sound	Translated to
ሀ, ሃ, ሐ, ሓ and ኃ	he
ሁ, ሑ and ኁ	hu
ሂ, ሒ and ኂ	hi
ሄ, ሔ and ኄ	hE
ህ, ሕ and ኅ	h
ሆ, ሖ and ኆ	ho
ሰ and ሠ	se
ሱ and ሡ	su
ሲ and ሢ	si
ሳ and ሣ	sa
ሴ and ሤ	sE
ስ and ሥ	s
ሶ and ሦ	so
ሸ and ሼ	xe
ሺ and ሽ	xi
አ, ኣ, ዐ, ዓ	ae
ኡ and ዑ	u
ኢ and ዒ	i
ኤ and ዔ	E
እ and ዕ	I
ኦ and ዖ	o
ጸ and ፀ	Se
ጹ and ፁ	Su
ጺ and ፂ	Si
ጻ and ፃ	Sa
ጼ and ፄ	SE
ጽ and ፅ	S
ጾ and ፆ	So

Appendix 8: Lists of affixes removed from Amharic words

Prefixes
ለ
ስለ
በ
በየ
እንደ
እንደየ
እየ
ከ
ወደ
ወደየ
የ

Suffixes	
ም	አቻችን
ምና	አቻችንም
ና	ው
ንም	ዎቻቸው
ንና	ዎቻቸውን
እና	ቻቻቸውንም
ኡ	ዎች
አች	ዎቻችን
አችም	ዎችን
አችን	

Appendix 9: Clustering solution obtained by average link for 10 clusters.

10-way clustering: [3047 of 3047], **Entropy:** 0.654, **Purity:** 0.469							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	15	0.416	0.045	0.001	0.000	0.477	0.333
1	287	0.021	0.008	0.003	0.002	0.285	0.864
2	289	0.032	0.012	0.003	0.002	0.218	0.903
3	8	0.205	0.024	0.004	0.001	0.649	0.375
4	276	0.024	0.010	0.005	0.002	0.456	0.750
5	28	0.093	0.028	0.003	0.002	0.503	0.643
6	23	0.094	0.028	0.004	0.002	0.612	0.609
7	1788	0.011	0.003	0.004	0.001	0.856	0.237
8	166	0.033	0.011	0.005	0.002	0.441	0.759
9	167	0.027	0.009	0.005	0.002	0.462	0.737

Appendix 10: Clustering solution obtained by average link for 15 clusters.

15-way clustering: [3047 of 3047], **Entropy:** 0.555, **Purity:** 0.553							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	15	0.416	0.045	0.001	0.000	0.477	0.333
1	287	0.021	0.008	0.003	0.002	0.285	0.864
2	267	0.025	0.010	0.005	0.002	0.428	0.772
3	8	0.205	0.024	0.004	0.001	0.649	0.375
4	281	0.033	0.012	0.003	0.002	0.223	0.900
5	28	0.093	0.028	0.003	0.002	0.503	0.643
6	23	0.094	0.028	0.004	0.002	0.612	0.609
7	162	0.035	0.010	0.005	0.002	0.432	0.765
8	136	0.034	0.010	0.005	0.002	0.329	0.816
9	9	0.145	0.022	0.005	0.001	0.569	0.333
10	8	0.551	0.076	0.002	0.000	0.000	1.000
11	31	0.069	0.019	0.005	0.002	0.698	0.387
12	4	0.298	0.025	0.004	0.001	0.452	0.500
13	864	0.015	0.005	0.006	0.002	0.764	0.319
14	924	0.015	0.005	0.005	0.002	0.638	0.434

Appendix 11: Major and sub categories of news items.

No.	Major Class	Subclasses
1	አደጋ Accident	ሰውሰራሽ አደጋ (Man made accidents)
		የተፈጥሮ አደጋ (natural accidents)
		አደጋን መከላከል (accident protection)
2	ባህል እና ቱሪዝም Culture and Tourism	ሀይማኖታዊ ጉባየዎች (religious conferences)
		ሀይማኖታዊና ብሄራዊ ባህላት (religious & national holidays)
		ጎጅና ልማዳዊ ድርጊቶች (Taboos)
		ጎብኝዎች (visitors)
		ታሪክ (history)
		ኪነ-ጥበብ (art)
		ቅርሶች (heritages)
		ብሄርብሄረሰቦችና ህዝቦች(NNP)
		የቱሪዝም ልማት (tourism development)
3	ኢኮኖሚ Economy	ማእድንና ኢነርጂ (Mining and energy)
		ማይክሮ-ኢንተርፕራይዝ(micro-enterprise)
		ባንክና ኢንሹራንስ (Banking and insurance)
		ንግድ(Trade and commercial)
		አጠቃላይ የኢኮኖሚ እድገት(GDP)
		ዕርዳታና የልማት ትብብር (Development and Aid cooperation)
		ኢንቨስትመንት (investment)
		ግብርናና ገጠር ልማት(Agriculture and rural development)
		መሰረታዊ ልማት(basic infrastructures)
		የዉሀ ሀብት ልማት (Water resources)
		የኢንዱስትሪ ልማት(Industry development

4	**ትምህርት** Education	**ሁለተኛ ደረጃ ትምህርት**(secondary school)
		ከፍተኛ ደረጃ ትምህርት (higher institutions)
		መደበኛ ያልሆነ ትምህርት (informal school)
		መዋዕለ ህፃናት(kindergartens)
		ሴቶችና ትምህርት (women's education)
		ተከታታይና የርቀት ትምህርት (distance & continuing education)
		የመጀመሪያ ደረጃ ትምህርት (primary level education)
		የመምህራንና የተማሪዎች ጉባዬ (teachers & students forum)
		ነፃ የትምህርት ዕድል (free scholarship)
		የቴክኒክና ሙያ ትምህርት (technical & vocational education)
		የትምህርት ሽፋን (education coverage)
		የትምህርት መገናኛ ዘዴዎች (educational communication systems)
		የትምህርት መሳሪያዎች (educational materials)
		የትምህርት ተቓማት ግንባታ (educational institution development)
5	**የዉጭ ግንኙነት፣መከላከያ እና ደህንነት** Foreign relation, Defense and Security	**አለም አቀፍና አህጉራዊ ክንዋኔ** (international and continental activities)
		ሽብርተኝነት (terrorism)
		ዲፕሎማሲያዊ ግንኙነት (diplomatic relations)
		ወታደራዊ ስልጠናና ማዕረግ (MiliterialTrainig and status)
		ወታደራዊ ተልዕኮ (militarilial missions)
		የሀገር ደህንነት (National security)
		የዉጭ ግንኙነቶችና ዉይይቶች (Foreign relations)
		ዜግነትና ስደተኞች (citizenship and e/immigration)
6	**ጤና**	**ባህላዊ ህክምና** (traditional)

	Health	**በሽታና ህክምና** (Disease treatment)
		በሽታን መከላከል (Disease protection)
		ሌሎች በሽታዎች(other diseases)
		መድሀኒቶችና አደገኛ ዕፆች(Drugs and Pharmatituicals)
		ወባ፣ ቲቪ እና ኤች አይቪ(Malaria, TB & HIV)
		የጤና ባለሞያዎች (health professionals)
		የጤና ተቓማት(health center development)
		የጤና አገልግሎቶች (health services)
		የህፃናትና የዕናቶች ጤና (children's and maternity health)
		የህክምና መሳሪያዎች(Medical materials)
7	**ህግ እና ፍትህ** Law and Justice	**ህገመንግስታዊ ጉዳዮች**(Constitutional affairs)
		ሙስና(corruption)
		ብሄርብሄረሰቦችና ፍትህ(nations and nationalities & justice)
		የፍትህ አካላት (Justical and legal bodies)
		የወንጀል ጉዳዮች (crime affairs)
		ዘር ማጥፋት(genocide)
8	**ፖለቲካ** Politics	**ዲሞክራሲና መልካም አስተዳደር** (democracy and good governance)
		ብሄራዊ ፖለቲካ (national politics)
		ምርጫ (election)
		ሰላምና መረጋጋት(peace & stabilization)
		ሰብዓዊና ዲሞክራሲያዊ መብቶች (human & d/rights)
		ዉይይቶች፣ ዉሳኔዎችና አዋጆች(discussions, decisions and proclamations)
		አለም አቀፍ ፖለቲካ (international politics)
		የፖለቲካ ሹመት (political delegation)
		የፖለቲካ ፓርቲዎች(political parties)
9	**ሳይንስ እና ቴክኖሎጂ**	**መገናኛ ብዙሀን** (mass media)

	Science & Technology	**ምርምርና ጥናት**(research and dissertations)
		ኢንፎርሜሽን ቴሌኮሙኒኬሽን ቴክኖሎጂ(ICT)
		የፈጠራ ስራዎች(creative works)
10	**ማህበራዊ** Social	**የሴቶች ጉዳይ** (Women Affairs)
		ስራ አጥነት(Unemployment)
		ሰብዓዊ እርዳታ(Social aid)
		ስርዓተ ፆታ(Sex)
		ጋብቻና ፍች(Marriage & Divorce)
		እድር(Idir)
		አሰሪና ሰራተኛ(Employer and Worker)
		አረጋዉያን(old persons)
		የህፃናትና ወጣቶች ጉዳይ(kids & youths affairs)
		የሙያና ህዝባዊ ማህበራት(professional & public corporations)
		የአካል ጉዳተኞች (physical disabled persons
11	**ስፖርት** Sport	**ባህላዊ ስፖርት**(traditional sport)
		ቦክስ(boxing)
		ዘመናዊ ስፖርቶች(modern sports)
		የፌደሬን አካላት (federation bodies)
		እግር ኳስ (football/soccer)
		አትሌቲክስ (athletics)
12	**የአካባቢ ጥበቃእና የአየር ሁኔታ** Weather & Environmental Preservation	**በርሀማነት**(Desert)
		የደን ልማት (forest development)
		የዱር እንስሳት ጥበቃና እንክብካቤ (wild animal protection)
		የአካባቢ ብክለት (environmental pollution)
		የአየር ትንበያ (weather forecasting)

Appendix 12: Clustering solution obtained by bisecting k-means for 15 clusters.

15-way clustering: [3047 of 3047], **Entropy:** 0.309, **Purity:** 0.791							
Cluster	Size	ISim	ISdev	ESim	ESdev	Entrpy	Purity
0	158	0.050	0.018	0.006	0.002	0.262	0.861
1	177	0.049	0.018	0.006	0.002	0.136	0.944
2	159	0.046	0.019	0.005	0.002	0.568	0.491
3	165	0.046	0.018	0.006	0.002	0.482	0.679
4	179	0.046	0.015	0.006	0.002	0.191	0.911
5	174	0.043	0.014	0.007	0.002	0.415	0.707
6	178	0.042	0.012	0.006	0.002	0.172	0.916
7	294	0.034	0.011	0.003	0.002	0.080	0.969
8	209	0.037	0.011	0.006	0.001	0.399	0.603
9	160	0.035	0.012	0.005	0.001	0.326	0.825
10	243	0.033	0.008	0.006	0.002	0.442	0.716
11	242	0.030	0.011	0.005	0.002	0.183	0.917
12	174	0.029	0.009	0.004	0.002	0.377	0.764
13	221	0.028	0.007	0.006	0.002	0.610	0.502
14	314	0.020	0.007	0.003	0.001	0.183	0.911

Appendix 13: Descriptive & discriminating features obtained by bisecting-means for 15 clusters.

Descriptive & Descriminating Features

Cluster 0	**Size:** 158	**ISim:** 0.050	**ESim:** 0.006					
Descriptive:	weyra	12.2%	memhr	5.1%	mimeTu	5.0%	zendero	4.6%
Descriminating:	weyra	8.9%	mimeTu	3.3%	zendero	3.3%	memhr	3.1%
Cluster 1	**Size:** 177	**ISim:** 0.049	**ESim:** 0.006					
Descriptive:	TEna	18.8%	weba	13.8%	bexita	2.9%	kEla	2.6%
Descriminating:	TEna	11.9%	weba	10.1%	bexita	1.8%	kEla	1.6%
Cluster 2	**Size:** 159	**ISim:** 0.046	**ESim:** 0.005					
Descriptive:	buna	20.9%	altaye	4.0%	waga	3.5%	mahberat	3.0%
Descriminating:	buna	14.2%	altaye	2.7%	waga	2.2%	mahberat	1.6%
Cluster 3	**Size:** 165	**ISim:** 0.046	**ESim:** 0.006					
Descriptive:	Ec	9.2%	aey	8.7%	Eds	6.3%	vi	6.0%
Descriminating:	Ec	6.6%	aey	6.2%	Eds	4.5%	vi	4.3%
Cluster 4	**Size:** 179	**ISim:** 0.046	**ESim:** 0.006					
Descriptive:	Mrt	5.7%	mrmr	5.3%	zer	4.9%	sebl	3.4%
Descriminating:	Mrt	3.3%	mrmr	3.3%	zer	3.2%	mefTer	2.3%
Cluster 5	**Size:** 174	**ISim:** 0.043	**ESim:** 0.007					
Descriptive:	mesno	9.1%	aerso	6.9%	manabat	3.5%	aeder	3.2%
Descriminating:	mesno	7.0%	aerso	3.9%	manabat	2.9%	dolo	2.5%
Cluster 6	**Size:** 178	**ISim:** 0.042	**ESim:** 0.006					
Descriptive:	tmhrt	14.8%	temari	6.5%	memariya	4.6%	aendeNa	4.3%
Descriminating:	tmhrt	8.9%	temari	4.5%	aendeNa	2.9%	memariya	2.7%
Cluster 7	**Size:** 294	**ISim:** 0.034	**ESim:** 0.003					
Descriptive:	wddr	8.6%	qenenisa	8.0%	yfTer	4.9%	sport	4.5%
Descriminating:	wddr	5.1%	qenenisa	4.8%	yfTer	3.0%	sport	2.4%
Cluster 8	**Size:** 209	**ISim:** 0.037	**ESim:** 0.006					
Descriptive:	balehebt	6.4%	bdr	4.4%	investment	3.5%	weTat	3.4%
Descriminating:	balehebt	4.5%	bdr	2.8%	weTat	2.5%	investment	2.4%
Cluster 9	**Size:** 160	**ISim:** 0.035	**ESim:** 0.005					
Descriptive:	aedega	10.2%	gorf	5.0%	gudat	2.8%	Irdata	2.2%
Descriminating:	aedega	6.8%	gorf	3.1%	gudat	1.6%	mulat	1.4%
Cluster 10	**Size:** 243	**ISim:** 0.033	**ESim:** 0.006					
Descriptive:	Whe	7.4%	menged	6.8%	Tge	5.2%	tElEkomuikExin	3.2%
Descriminating:	menged	4.6%	whe	4.5%	Tge	3.9%	tElEkomuikExin	2.4%
Cluster 11	**Size:** 242	**ISim:** 0.030	**ESim:** 0.005					
Descriptive:	hotEl	9.9%	turist	4.1%	bololo	4.0%	wbet	2.7%
Descriminating:	hotEl	7.3%	turist	3.1%	bololo	2.8%	wbet	1.7%
Cluster 12	**Size:** 174	**ISim:** 0.029	**ESim:** 0.004					
Descriptive:	polis	8.7%	komiyunikExin	8.6%	hg	4.2%	wenjel	2.4%
Descriminating:	komiyunikExin	6.1%	polis	5.4%	hg	2.6%	wenjel	1.7%
Cluster 13	**Size:** 221	**ISim:** 0.028	**ESim:** 0.006					
Descriptive:	Elka	3.8%	aefe	2.3%	masfe	1.7%	rIse	1.4%
Descriminating:	Elka	2.9%	masfe	1.3%	aefe	1.3%	teketatay	0.9%
Cluster 14	**Size:** 314	**ISim:** 0.020	**ESim:** 0.003					
Descriptive:	Taliya	6.0%	Telo	5.4%	aekahidew	3.8%	temeraC	3.4%
Descriminating:	Telo	3.1%	Taliya	3.1%	aekahidew	2.3%	temeraC	2.1%

Appendix 14: Cluster labels for the 15 clusters.

Cluster ID	Cluster labels
0	Teachers?
1	Health
2	Social affairs and trade
3	HIV AIDS
4	Science and Technology
5	Agriculture
6	Education
7	Sport
8	Investment and Finance
9	Accident
10	Basic infrastructure developments
11	Culture and Tourism
12	law and criminal affairs
13	General?
14	Political Parties and Election?

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ACRONYMS

AI	Artificial Intelligence
ASCII	American Standard Code for Information Interchange
CLUTO	CLUstering TOolkit
DF	Document Frequency
ENA	Ethiopian News Agency
gCLUTO	graphical CLUstering TOolkit
ICT	Information and Communication Technology
IG	Information Gain
IDF	Inverse Document Frequency
IR	Information Retrieval
KDT	Knowledge discovery in Text
ML	Machine Learning
NLP	Natural language processing
SERA	System for Ethiopic Representation in ASCII
TC	Text Classification
TF	Term Frequency
TF*IDF	Term Frequency by Inverse Document Frequency
TM	Text mining
WWW	World Wide Web
ISim	Internal Similarity
ESim	External Similarity

ABSTRACT

With the advancement of technology and proliferation of computers in the country, the amount of Amharic news items produced is increasing which becomes a difficult task for news agencies to organize such huge collection of news items manually. To solve this problem, few researches were conducted using supervised approaches which rely on expensive, cumbersome and error prone labeling of training data. Hence, the aim of this study is to explore the application of unsupervised learning approaches for Amharic document clustering with low cost and best quality of clustering solution.

The methodology adopted in this study has three phases: document collection, document preprocessing and cluster analysis. Document preprocessing techniques such as tokenization, normalization, stop word and numbers removal, stemming, term weighting and dimension reduction were done on Amharic News documents collected from Ethiopian News Agency (ENA). The vector space model was used to represent Amharic documents using the Term Frequency by Inverse Document Frequency (TF*IDF) term weighting approach. The performances of k-means, bisecting k-means, single link, complete link and average link were evaluated.

At increasing number of clusters and documents, k-means and bisecting k-means produced more internally cohesive and externally isolated clusters. But, the clustering results do not match better with the pre-defined classes. All algorithms: k-means, bisecting k-means, single link, complete link and average link achieved better clustering quality as the number of clusters increases using the same data set. Moreover, there is a mismatch between the news categories provide by ENA and the clusters discovered by clustering algorithms.

For Amharic text document clustering, both k-means and bisecting k-means produced better results as compared to the agglomerative hierarchical clustering techniques both in terms of time requirement and the quality of the clusters produced.