JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING

GRADUATE PROGRAM IN INFORMATION TECHNOLOGY

Statistical Topic Modeling for Afaan Oromo Document Clustering using Latent Dirichlet Allocation (LDA)

Fikadu Wayesa Gemeda

A THESIS SUBMITTED TO THE FACULTY OF COMPUTING IN PARTIAL FULFILLMENT FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION TECHNOLOGY

June 25, 2019

Jimma, Ethiopia

JIMMA UNIVERSITY

JIMMA INSTITUTE OF TECHNOLOGY

FACULTY OF COMPUTING

GRADUATE PROGRAM IN INFORMATION TECHNOLOGY

Fikadu Wayesa Gemeda

The work contained in this thesis entitled as "*Statistical Topic Modeling for Afaan Oromo Document Clustering using Latent Dirichlet Allocation (LDA)*" has not been previously submitted in partial fulfillment to meet requirements for the degree of Master of Science in Information Technology at this or any other higher institutions. To the best of my knowledge and belief, the thesis is my original work and contains no material previously published or written by another person except where due reference is made.

Signature: ________________________ Fikadu Wayesa Gemeda Date June 19, 2019

Approved by Advisors:

	Signature	**Date**
ADVISOR: Dr. Million Meshesha (PhD)	million	June 19, 2019
CO-ADVISOR: Kibret Zewde (MSc)	_______________	____________

Approved by the Examining Committee:

	Signature	**Date**
1. ___________________________________	__________	__________
2. ___________________________________	__________	__________

Abstract

The plenty of digital data poses challenge to understand and utilize the overwhelming amount of information. Manually reading large data for content analysis is inefficient. The amount of information available in digital form is getting double which is leading to the information overload almost in all languages. So, in Machine Learning Topic modeling has been accepted as a powerful technique for the comprehension of content analysis. In this study, we used Latent Dirichlet Allocation (LDA) based topic modeling for analyzing Afaan Oromo text documents to extract appropriate topic tags from the collection. It relies on unsupervised machine learning techniques to extract topics from document collection, by generating probabilistic word-topic and topic-document associations from the latent topics in the text using hidden random variables. We combined word embedding approach to capture semantic structure of words how they are semantically correlated to each other with LDA algorithm to improve the quality of extracted topics since the LDA suffers from the bag-of-model approach. We used a collection of 16 documents from 4 different categories (Health, Education, Sport and Weather condition). Clean text corpus and estimating parameter settings that generates interpretable topics are important prerequisites for acquiring a valid interpretation of topics from the algorithm. The experiments include all necessary steps of data collection, pre-processing, model fitting and an application of document exploring. We used Gibbs sampling to estimate the topic and word distributions. Experiments are carried out to confirm the topic extraction effect of this algorithm. The clustering of documents and exploration was done by the LDA based on the generated topics. Our study used three evaluation metrics; Perplexity to select better LDA model parameters, Topic Coherence is used to evaluate the coherence of topics and human judgment for topics interpretability. An average accuracy of **Perplexity score of -9.775** was estimated with number of 10 topics from different values of K topics (We used minimum K to 2 and maximum 20), Topic coherence of PMI with **52.5%** was scored, and overall human judgement of **66% of F Measure**.

Keywords: ***Topic Modeling, LDA, Statistical Modeling, Topic Extraction, Latent Topics, Big Data, Afaan Oromo.***

Acknowledgement

First and foremost, I would like to thank **God Almighty** for giving me the strength, knowledge, ability and opportunity to undertake this study. Without his blessings, this achievement would not have been possible.

I would like to express my sincere gratitude to my principal advisor **Dr. Million Meshesha** (PhD) for his patient guidance and continuous support throughout the work starting from the brainstorming of this idea to the completion of this study. I am grateful for his encouragement and helpful feedback in my work, and special thanks goes to my Co-Advisor **Kibret Zewde** for his valuable suggestions and help in this work.

I would like to thank my parents for their support and encouragement throughout my journey and their willingness to do everything by helping me in my aspects of life and in the successful completion of this degree.

My special thanks, appreciation and love to my wife Daditu Tefera for everything she did to help me to complete my study. I would like to express my gratitude to my special friends Zerihun Olana, Gizaw Tadele, Alemissa Endebu and Amanuel Aseffa.

Finally, I dedicate this thesis to my parents, especially to my secret counselor Mr. Getachew Kenei, my mother Shashitu Duguma and my brother Adugna Wayesa whose dream for me have resulted in this achievement and without their loving childhood and nurturing, this success would not have been possible without their unconditional love and support.

Contents

List of Figures

List of Tables

List of Acronyms

BBC	British Broadcasting Corporation
BoW	Bags of Words
FBC	Fana Broadcasting Corporation
GUI	Graphical User Interface
HTML	HyperText Markup Language
IR	Information Retrieval
LDA	Latent Dirichlet Allocation
NLP	Natural Language Processing
OBN	Oromia Broadcasting Network
PLSA	Probabilistic Latent Semantic Analysis
PMI	Point Mutual Information
POS	Part of Speech
TF*IDF	Term frequency –Inverse Document Frequency

CHAPTER ONE

INTRODUCTION

1.1. Background

We live in the world where ideas and feelings are registered as a tool of human communication. In the digital age, the rise of multimedia data such as text files, video, audio and animation are continuously generated every day and the amount of data is expanding and increasing alarmingly which can be the basis for numerous analyses.

As unstructured digital documents available is more, it is difficult to understand the core relevant information existed in it; that means, it become tedious and time consuming to search and identify the required information [1]. These large digital datasets content offer significant opportunities for researchers in finding methods and applications that automatically analyze the contents of a text in the field of Natural Language Processing (NLP), machine learning, data mining and Information Retrieval (IR) [1] [2] [3].

Grasping out quickly, the desired information in a collection of documents might be difficult by reading the whole content. To alleviate these problems, there is a need to apply tools and techniques that automatically analyze, organize, search and understand vast quantities of any textual information [4].

In order to extract hidden patterns and analyze large digital datasets consists of many attributes, we need tools for further analysis. Text mining or text analysis is one specific area of data mining. Any type of unstructured text can be used in text mining. There are several techniques within the area of text mining for analyzing text content. One of those is topic modeling [1] [5] [6]. This is a new area of research and one specifically designed for analysis of large datasets of digitized content [7].

It is a machine learning algorithm based on statistics that allow extraction of the most discussed topics in a given large collection of documents. It learns a set of latent variables called topics in a document where documents comprised of different of topics, topics are probability distribution over vocabulary of words where frequently co-occurring words [8]. The extraction of topics is based on the word co-occurrence patterns in a collection. The majority of topic models are probabilistic generative models in which documents arise from a generative process.

The study of Topic Modeling has become an important aspect of text analysis and widely used in text mining, machine learning, NLP and IR in recent years [9] [10]. As many studies suggested, it is an important tool in the applications like topic model-based IR for relevance judgment, document summarization, classification of texts, recommendation systems and information filtering [4].

Topic models allow for grouping documents in a corpus based on their corresponding theme that provide an abstract view about a set of subjects. It helps in discovering latent or hidden topics that are present across the collection, annotating documents according to these topics and by using these annotations to organize, search and summarize texts.

Documents can be classified, arranged and searched according to their subjects [11]. It is a form of unsupervised statistical machine learning techniques to identify patterns in a corpus [12] [13]. It helps to organize huge collection of documents and groups or clusters words across the corpus into topics by a process of similarity, where topics are described as a collection of words that frequently appear together in the same context that best represents the information in the collection [14].

One way to understand how topic modeling works is to imagine working through an article with a set of highlighters [15]. Suppose we are reading a newspaper and we have a set of colored highlighters in our hand. As we read through the article, we use the different color for highlighting the key words of themes within the document as we come across them. Then when we were done, we could copy out the words as grouped by the color we assigned them. The list of words is a topic, and each color represents different topic. This is the notion of topic modeling. Suppose we want to learn something about a document that is too big to read; it takes too much time and effort to browse the whole contents for human [16]. Why don’t we just throw all these documents to the automated system and see what interesting patterns (topics) it finds? An automated topic models facilitate understanding, organizing and summarizing huge text datasets. Same words can be interpreted in different meaning depending on their context. According to the notion of Topic modeling algorithms, documents are about more than one subject, but the majority of NLP algorithms and IR techniques implicitly assume that every document has just one topic [13] [17] [9].

The development of Topic Modeling for any language, to provide topic modeling tasks for any NLP application like filtering relevant information based on topic and visualization of the topics is important as an interactive user interface. So, this study aimed to make statistical topic modeling using Latent Dirichlet Allocation (LDA) Model for analyzing and organizing large document collections for Afaan Oromo text articles. As well, the document exploring system based on the extracted topics that allows human to navigate through relevant information on a large data collection is investigated.

LDA is a Bayesian network statistical approach [13] [18] that is effective on building latent topics by relating words contextually for huge documents like news articles, research paper abstracts. Given sufficiently long texts, these approaches are capable of identifying significant topics, based on the co-occurrence relationship among words. In this model, each document is sampled from a random mixture of topics and vocabularies are distributed across each topic.

The motivation of this work is to enable content analysis of Afaan Oromo textual data using Topic Modeling techniques specifically LDA. Most of the suggested Topic Models have been designed for many languages like English, European and Asian languages. As to the our knowledge there is no work done in topic modeling for Afaan Oromo language.

1.2. Statement of the Problem

In recent years, enormous amount of data is being generated every day from different blog, media and websites. The amount of information available in digital form is getting double which is leading to the information overload almost in all languages. This leads to the necessity of dimension reduction for effective use of the data as well as for effectively managing, searching, categorizing and exploring the textual data in well-organized format. As we tried to introduce in the background section, Topic Modeling comprehensively find out the themes covered in a given document for further analysis in the text analysis field. This technique has emerged as an effective method for discovering useful structure in collections. In this study, an attempt is made to extract hidden topics that can reflect the underlying meaning of text articles.

We can find lot of documents related to a subject, but in limited span of time human mind is not able to search all to get required information. Texts in any domain are written in detail and the readers must read the whole content to understand what it talks about. If the reader does not understand what the contents talking about, he/she needs to read again and again to extract the theme (topic) of the text. This consumes time and becomes very tedious one. Typical keyword-based operations of search applications count the number of occurrences of query terms in the documents and rank documents accordingly. But this requires the searcher to know the terminology used in the document that should be returned by the application.

Typically, when we search for information, we find as Keywords or via Links where we use the networked structure of web to traverse from page to page that shares similar or related content. Keyword based search is popular information retrieval scheme to discover relevant documents from the collection, but it loses understanding semantic meaning of documents. For example, some relevant documents may not contain the exact keywords specified by the user. This will reduce the number of documents to be filtered as the user requirement. This issue can be overcome by semantic search methods. As an alternative method it would be better to search and explore document collections of related themes rather than finding documents through keywords based only to make searching more effective. Hand coding the collection would be time consuming and would requires knowing the thematic structure of the documents before coding. Instead, we can use probabilistic topic models algorithms that analyze words in the documents to uncover the thematic structure of the collection without the requirement of any hand coding or labeling of documents prior analysis [11]. This study focuses on a small feature of concept search on document to address the problem of keyword-based searching through topics extracted from the documents. These models group word types in a collection of documents and the groups are referred to as topics where a word type can belong to more than one of these groups. Therefore, for each word type a probability distribution over all topics can be defined. Similarly, each document can be described by a probability distribution over all topics.

Applying topic models to concept search is used as implication regarding the underlying topic structure from document collection by categorizing the documents according to these topics. It can be adapted to many kinds of data such as collections of text documents (news articles, biomedical texts, etc.), images and social networks [12] [19]. It provides a way to group vocabulary from a corpus to form latent topics. For each document there is collection of topics. From vast collection which contain millions of documents and billions of tokens learning or extracting expressive topics is challenging; because, it needs to deal with large number of topics, scalable and efficient way of the computation.

The problem of automatic topic extraction and document categorization based on the theme is one big issue and many researches have been done in many languages like English, European languages (Portuguese, French) and Asian languages [1]. A popular approach to address this problem is text categorization that group the content of text into one or more separate topical categories. Numerous methods for text categorization have been introduced in the literature, and each one has its own costs and benefits. Afaan Oromo text readers are not exceptional to suffer from the mentioned problems. There are many domain areas in this language that produce large content of textual information in digital form and increasing highly from time to time, such as legal, news media agencies, government offices, etc. Almost all Media agency like Oromia Broadcasting networks (OBN), Fana Broadcasting Corporations (FBC), etc. publish their news items in digital form. These text items would be easy for understanding if they are clustered under their theme in order to make the searching and management of news easy. With the automatic topic modeling services that can potentially increase the users' browsing and reading time to get the theme of the text and browsing over the content is required.

Statistical topic modeling for Afaan Oromo texts released in digital form are desirable to employ a powerful computational approach to save time of readers. As far as the researcher knowledge is concerned, there is no attempt on topic modeling for Afaan Oromo. So, in this work, statistical topic modeling approach based on Latent Dirichlet Allocation (LDA) is proposed to extract topics from Afaan Oromo textual contents.

To this end, the following are the major research questions that are explored and answered in this work.

- How Topic Model is designed in content analysis using unsupervised machine Learning Model for a big data analysis in terms of organizing text collections?
- How to detect a topic by applying semantically enhanced feature word representation in LDA?
- To what extent LDA Topic Model generates set of interpretable topics structure and document clustering based on the generated topics in big data?

1.3. Objectives of the study

2.1.1. General Objective

The general objective of the study is to design and develop a statistical topic modeling for Afaan Oromo document collection to understand its content.

2.1.2. Specific Objectives

The following specific objectives are formulated so as to achieve the general objective of the study.

- ✓ To understand related works for identifying techniques and approaches in topic modeling.
- ✓ To prepare dataset for training and testing purpose.
- ✓ To develop a framework that extract topics from Afaan Oromo texts.
- ✓ To implement a semantic based LDA for extracting topic from large volumes of text.
- ✓ To Label the extracted topics
- ✓ To evaluate the performance of the designed topic model.
- ✓ To discuss findings, challenges, recommend further work and derive conclusion.

1.4. Methodology

To investigate the stated research problem and to achieve specific objectives, methods are applied by reviewing different related papers to solve the research gaps in the proposed system. Works that are more related to our approach are reviewed to identify and design a new architecture of our proposed model. The other methodology used in this study is the justification for the procedures like modeling the architecture and selecting the implementation tools, the data collection and data processing. We collected 16 documents from two websites FBC Afaan Oromo and BBC Afaan Oromoo for testing purpose. These documents fall into 4 categories; Health, Education, Sport and Weather Condition. Then implemented LDA algorithm in Java NetBeans for extracting Topics. The evaluation methods used for evaluating the performance of the system is considered. Perplexity is used to estimate the number of topics to be extracted from the corpus. Topic Coherence and Human judgment evaluation were used for the validity of the extracted topics.

2.1.3. Study design

Design science has been selected as the methodology used for this research work because it is best suited for the task of creating new process model by focusing on designing solutions to ideal problem through research process based on the activities of design (development). There are a number of stages involved in the process: awareness of the problem, suggestion, development, evaluation, and conclusion [20]. It is the design and investigation of components in context that interact with a problem. Design problems (prototyping) and knowledge creation would be the two parts of design science.

To define the step-by-step procedure in undertaking this study we used design science process model recommended by Roel [21]. The outcome of our synthesis is a process model consisting of seven activities in a nominal sequence, here described and presented below.

2.1.4. Problem identification and motivation

In this stage, the definition of meaningful and actionable statement of the problem should be established out to be addressed and guiding the research to the right direction. Before we got into the stated open problem (problem statement), we first gained an understanding of the area through reviewing many works done by different scholars. As we read many papers in the field, the major problem is information overloading and analyzing these huge collections within a given fraction of time. In any languages, information overloading is there since we are in the era of digital age. Afaan Oromo is an Oromia regional state official language. It is a media of instruction in education starting from lower class to higher institution. Many researches are conducting in this language and it is studied as master's level and even as PhD program. Generally, there are many sectors that are producing digital data in performing their operations. For instance, in Education, Health, metrology sector and government offices. As unstructured data is collected, if one reader needs to know something about its content in a fraction of time, topic modeling is a good tool for content analysis. This intrusion requires the identification and analysis of the problem. We defined the research problem and justify the proposed solution. The problem that this study addresses is there is no automatic topic modeler for Afaan Oromo text that can extract hidden topics in unstructured data.

2.1.5. Objectives of the solution

The creation of a solution requires defining the objectives as quantitative or qualitative based on the problem to be solved. The objective of this research is to develop an unsupervised statistical topic modeling for Afaan Oromo digital data

2.1.6. Data collection and analysis

The data used for the analysis purpose should be gathered. We collected the data based on the design science research process model. The dataset (corpus) was prepared manually as a sample to evaluate the proposed model. The corpus was organized by considering the structure of the language features and collected from websites of BBC and FBC published in Afaan Oromo. These datasets were pre-processed and the topics was learned by the algorithm employed as the required model.

2.1.7. Design and development

After the problem identification, proposing the solution and data were gathered, a model to a solution is created using prototyping. To do this, tools and the system architecture should be clearly defined. These tools are used for data collection and preparation, implementation of the proposed model algorithm and designing the model. Microsoft Visio Professionals is used for architecture designing. Java Programming Language (NetBeans 8.02) is used to process and implement the LDA algorithm that process topic modeling. The reason behind this software is because it includes very powerful packages help for generating an excellent output. The collected data for experimentation from different websites manually puts as TXT file and processed. Again, the techniques used to implement an appropriate algorithm for this approach would design and discussed for describing our works clearly.

2.1.8. Experimentation and demonstration

In this stage, the proposed model has been demonstrated with the dataset to judge the performance or how it is efficient according to the stated research problem. Methods used for evaluating the model have been discussed in this phase.

2.1.9. Evaluation

Observing and measuring how the proposed solution to the problem with respect to the objectives through relevant metrics and analysis techniques is required to evaluate the performance of the proposed model. The output of the proposed model was tested on texts and the testing evaluation is done for the extracted theme with the corresponding topics as well as the content of the text. Therefore, the performance measures used in the study are Perplexity, topic Coherence and human judgement. F-Measure is calculated from the human judgement. The model's performance is evaluated by human judgement to assure whether the model extracts the valid topic or not.

2.1.10. Communication

At the last, the obtained results have been presented both in theoretical oriented and reader oriented to summarize how this study has brought the development of topic modeling and to place in with context with future research.

1.5. Scope and Limitation of the study

The scope of this study is to develop a prototype that extract representative topics from the Afaan Oromo text by statistical technique through analyzing the contents. This work is limited to a textual document of Afaan Oromo corpus only. However, topic model can be applied to other data forms such as image, audio, video, and they are out of the scope of this study. The unstructured documents are provided to the Topic Modeler for topic finding from the collections. Our proposed model is based on, unsupervised approach to produce interpretations based on the understanding. For experiment, Afaan Oromo text documents are collected from four domain and text operation such as tokenization, normalization and stop word removal is performed. Then, the prototype group topics according to the pattern generated from the content analysis. through word frequency and co-occurrence analysis. So, the proposed topic modeler illustrates our approach by addressing the task of topic modeling in the text analysis to extract the topics. The study utilizes LDA with data represented as continuous bag of words model.

A comprehensive Topic Modeling system will involve a variety of natural language processing applications such as part of speech (POS) tagger, parser, stemmer and ontology specifically for topic labeling. However, it is challenging to integrate all these applications to the non-researched language like Afaan Oromo. Even though some of these applications have been developed by some different scholars, they are not freely available for integration with the system we proposed to develop. The absence of well-organized corpus for Afaan Oromo language may be a great constraint. The amount of corpus that was prepared for this study was small to evaluate the work.

1.6. Significance of the study

Topic modeling helps in understanding of what is going in a collection of millions of documents. It is an effective application in many areas where large number of documents collected, including information retrieval, visualization, Recommendation system, bio-informatics applications, document similarity and language modeling. They are successful use to understand scientific publication and in political texts. So, topic modeler for Afaan Oromo texts will be an input to the development of the language text and has significance to initiate further research in the area of document similarity, Recommendation system and Information Filtering for Afaan Oromo language. Also, it can help to initiate topic modeling for other local languages. This research proposes a mechanism to incorporate topic modeling techniques for generating more accurate topic models by representing words semantically using new approach of word embedding to use patterns to represent topics instead of individual words or Bag of words only as used in traditional topic models.

1.7. Thesis Structure

This study is organized in to six chapters including this part that provides a brief overview of the work. **Chapter two discusses** the literature review which covers the knowledge necessary to address the problems defined and states the related works with this work. **Chapter three** gives the general overview of Afaan Oromo Language. C**hapter four**, presents the proposed model by the current study that addresses the problems. In **chapter five**, experiments with collected datasets are conducted to verify the effectiveness of the proposed model on discovering topics and expressing topics with meaningful patterns. **Chapter six** summarizes the key findings and highlights the significant contributions in this study and also provides concluding remarks with stating further research work to be done in the future.

CHAPTER TWO

LITERATURE REVIEW

2.2. Overview

This chapter presents the background knowledge for this work, how document collections can be modeled using statistical approaches specially, LDA based Topic modeling in the process of topic extraction to address the research gap introduced in chapter one. Along with this clear statement, an understanding of how the topic modeling algorithms work is needed in order to use the tool successfully [13] [4] [22]. Some basic concepts and approaches of Topic Modeling would be reviewed in the next sections.

2.3. Modeling Document Collections

A fundamental problem in NLP is finding ways to represent large amounts of text in a compact way. One of the most common representations of a document collection is the bag-of-words and Vector Space Model (VSM) [23]. In a VSM document representation, terms are represented as the term-document matrix C. It is a WXD matrix that contains information about the occurrence (frequency) of W terms in D documents. Elements, c_{ij} in C represent the ith word in the jth document and are usually weighted by the raw frequency of terms in documents or using the tf*idf (term frequency – inverted document frequency) weighting.

A well-known problem with the term-document VSMs is the high dimensionality caused by the large number of unique terms that can exist in a corpus. High dimensional spaces are often also sparse, i.e. many documents contain only a few unique terms in a few documents. This makes it difficult to accurately compute the similarity between two documents or terms. In addition, VSMs do not fully captured semantic structure with polysemy. For example, the term *python* may occur in documents with the meaning as either *snakes or programming languages*, however there is only one instance of that vector which contains the co-occurrence of that word with all the documents in the collection. Statistical methods with word embedding can can solve this problem.

2.3.1. Notation

We introduce topic modeling notation following a similar approach as presented in [12] [24]:

- ✓ A term represents a unique word type of a fixed length vocabulary indexed by {1, …, W}. Each word is represented as unit-basis vector of length W that has a single element equal to one and all other elements equal to zero. The kth word in the vocabulary is represented by a vector w such that $w^k = 1$ and $w^i = 0$ for i! = k.
- ✓ A document of N words is represented as a sequence by d = (w_1, w_2, …, w_N), where wi is the ith word in the sequence. Note that this is also a bag-of words representation since the word sequence does not need to match the original word order of the document.
- ✓ A corpus is a collection of D Documents D = {d_1, d_2, …, d_D}.

Let's consider an example vocabulary, v = {be, not, or, to} with indices {1, 2, 3, 4}.

The word *be* can be represented as w^3 = (1, 0, 0, 0). The document d =”to be or not to be" will be represented as w_d = ($w_1^4, w_2^1, w_3^3, w_4^2, w_5^4, w_6^1$), using the notation described above. Note that any permutations of the word order in w_d do not have any effect and result in equal representations of the document because of the bags-of-word approach.

2.4. Overview of Machine Learning

Machine learning has grown from the study of learning theory to pattern recognition for computational learning theory in Artificial Intelligence [5]. It is the most effective method used in the field of data analytics in order to predict something by building models and algorithms that can learn and make predictions on datasets. Model is created based on the results gathered from the training data [2] [5] [15] [25] [26]. Machine Learning involves different types, as shown in figure 2.1 below

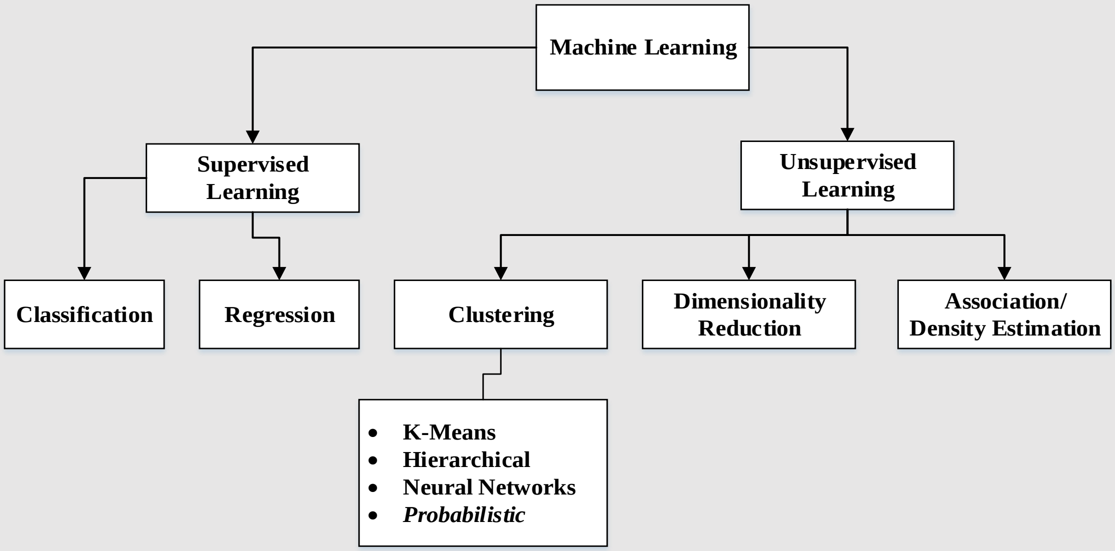


Figure 2. 1 Machine Learning Types [5]

In our work, we focused on one of the unsupervised machine learning approach probabilistic topic modeling using LDA that automatically organize and understand large contents of data. Many scholars have recently proposed topic models to learn topics from unstructured text.

2.4.1. Supervised Machine Learning

With the increasing availability of big data, the cost of manually coding documents for content analysis become unrealistic. One approach to automate this type of problem is supervised learning technique from machine learning. Purpura [27] and Hillard [28] describe the automated classification system used in the Congressional Bills Project, in which Support Vector Machines (SVMs) to classify legislative text into one of the 226 subtopics in the Policy Agendas Topics codebook. A similar approach has also been used to classify German online political news [29] [30].

Thus, similar to the human coding, the supervised learning method still has high pre-analysis cost since it used labelled training set [28] [27]. To alleviate this type of limitation, unsupervised learning approach is an option. One of the applications of unsupervised learning is Topic Modeling.

2.4.2. Statistical Topic Models

Statistical topic model has got potential benefit from recent advances in the fields of statistical modeling and machine learning to represent text by discovering latent topics in the large text collections. They are generative models that learn a set of latent variables called topics to model how words are generated in documents within a text corpus [13]. Examples of such models include probabilistic Latent Semantic Analysis (pLSA), Latent Dirichlet Allocation (LDA), and author-topic models. All of these models are derived from a basic language model, which models words and documents as joint probabilities of all of its word occurrence [31] [32] [33].

Latent Dirichlet Allocation or LDA [9] [12] [13] , one of the most popular unsupervised statistical algorithm in text modeling techniques that used to identify and describe thematic structure of text collections. The main assumption of LDA topic models is that; documents are generated by a mixture of topics while topics are probability distributions over words. The input is a set of documents and its output is a set of topics together with topic assignments to documents. Figure 2.2 shows an overview of a topic modeling pipeline including input and output.

In the input, each document is represented as a bag-of-words. Each document is often tokenized (split up) into words and words are normalized (converted to lower case) while word order is ignored. The only information relevant to the model is the number of times a word appears in each document. From figure 2.2 Document 1 contains information about Python (programming language) while document 2 contains information about pythons (snakes). The output of a topic model is a set of topics and a set of topic assignments for each document. Each topic is a probability distribution over all the unique words in the collection. Topics are often represented by the words with the highest probability in the topic (see the output in Figure 2.2). The term topic words refer to the set of n words with highest probability in a given topic. Words assigned high probability in some topics frequently appear together in documents and are likely to represent a coherent subject or theme. A document is represented as a probability distribution over topics with only a few topics assigned with high probability.

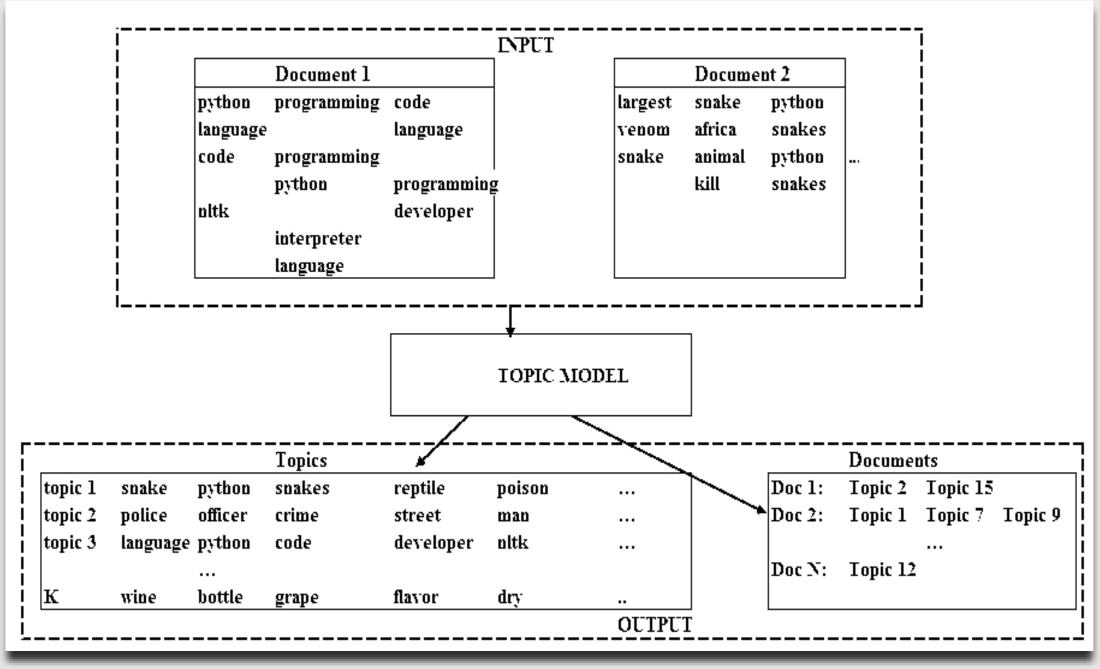


Figure 2. 2 Input and Output of a Topic Model [34]

In this study, a statistical topic modeling method is investigated for analyzing the context in large number of documents into topical patterns to understand the content in a short time. Topic models are generative models which can learn mixture of (unobserved) topics in an unsupervised way [12] [6] [35].

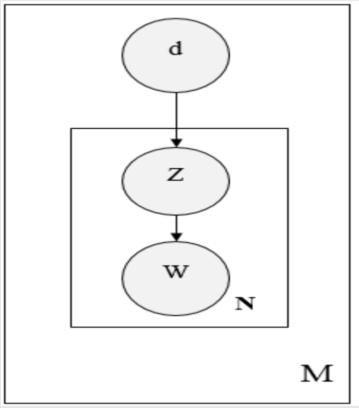


Figure 2 3 Plate Notation of Probabilistic Topic Modeling [22]

The probability of a word occurring in a document P(wi) can easily be derived using likelihood method [13]:

$$\text{P(wi)} = \frac{count\ of\ wi\ in\ dataset}{count\ of\ all\ words\ in\ dataset}\text{---}(2.1)$$

The probability of a document is determined by the product of the probability of each term in the document [13].

$$\mathbf{P(wi,\ldots,wN)}=\prod_{I=1}^{N} P(wi)\text{--}(2.2)$$

Thus, the joint probabilities of observing all terms are generated by the mixture process as follow [13]:

$$\boldsymbol{p(d,w) = p(d)\prod_{i=1}^{m} p(wi|d)}\text{--}(2.3)$$

Where

$$\boldsymbol{p(wi|d) = \sum_{z \varepsilon Z} p(wi|d)p(z|d)}$$

Latent Dirichlet Allocation (LDA) builds on pLSA [4] [16] [17] [35]. In pLSA, the probability for each topic is the same regardless of which document the words and the topics appear in. Similar to pLSA, LDA has one set of probabilities, denoted as ϕ, that model the probability of each term given the topic that generated it, and this probability is conditioned on a prior β. LDA then adds a second probability ϴ which denotes the probability distribution of the topics given the document [36] [37]. This probability is also conditioned on a prior, which is denoted by α. To better realize how to use a topic model in a given corpus, we first describe the basic ideas behind topic modeling by illustrating the key steps, including the bag of words (BoW), model training, and model output. We first assume that there are **N** documents, **V** words, and **K** topics in a corpus. Then, we discuss each component of this diagram in detail.

The Bag of Words (BoW)

In NLP, a document is represented by a BoW that is actually a word-document matrix. As shown in Table 2.1, there are four words (weather, science, school, and League) and four documents (d1–d4). Value w_{ij} in the matrix represents the frequency of word i in document j. For example, w3,1 = 1 means that the frequency of the word "school" in document d1 is 1.0. It is obvious that the number of words is fixed in a corpus, and the collection of these words constitutes a vocabulary and represented by the BoW. A BoW is a simplified representation of a corpus as the input of topic modeling [9]. After construction of the BoW, it serves as the input of the next step in topic modeling. Suppose there are N documents and V words in a corpus; thus, the BoW of this corpus is an N × V matrix. Moreover, the documents in a corpus are independent: there is no relation among the documents. The exchangeability of words and documents could be called the basic assumptions of a topic model. These assumptions are available in LDA.

Terms	Documents			
	D1	D2	D3	D4
Weather	2	0	3	0
Science	0	5	0	0
School	1	2	0	0
League	0	1	4	7

Table 1.1 An example of a BoW [9]

Model training

In a BoW, the dimensionality of word space may be enormous, and it reflects only the words of the original texts. The most important thing people expect to know about a document is the themes rather than words. The aim of topic modeling is to discover the themes that run through a corpus by analyzing the words of the original texts. We call these topics. Topics were discovered during model training. To have a better way of organizing the explosion of digital data collections, it requires the use of new techniques to automatically organize, search, index, and browse large collections [22]. The importance of topic modeling is to discover patterns of word-use and how to connect documents that share similar patterns and these documents are mixtures of topics, where a topic is a probability distribution over words [22] [38]. It also creates a new document by choosing a distribution over topics. After that, each word in that document could choose a topic at random depending on the distribution. In this study, following the automated content analysis approach, we introduce a word embedding approach that enables extraction of topics in topic models, which are guided by additional information associated with the text and designed to discover and analyze contents of Afaan Oromo text.

2.4.3. Generative Model

Probabilistic topic models attempt to capture latent structure in documents and easy to accommodate unlabeled data [13]. A Generative probabilistic model is a powerful way of learning data distribution using unsupervised learning. It assumes, there are a number of topics related to a collection of documents. Each document is assumed to be generated as follows: for each word in this document, choose a topic assignment and choose the word from the corresponding topic. Each word in a document is assumed to come from a hidden (latent) topic, and probabilistic topic models assign each word to the proper topic.

The two probability distributions, $p(z|d)$ and $p(w|z)$, are assumed to be multinomial distributions Thus, the topic distributions in all documents share the common Dirichlet prior α, and the word distributions of topics share the common Dirichlet prior β. Given the parameters α and β for document d, parameter θ_d of a multinomial distribution over K topics is constructed from Dirichlet distribution $Dir(\theta_d|\alpha)$. This allows the treatment of the problem of topic discovery as a parameter estimation problem. These parameters are an initial prior distribution about the distribution. They are called **hyper parameters** [12] [39]**.**

Similarly, for topic k, parameter βk of a multinomial distribution over V words is derived from Dirichlet distribution $Dir(\beta k|\eta)$. The multinomial dirichlet distribution is a convenient choice as a prior and can simplify the statistical inference in LDA [13].

In LDA the above step by step procedure can be formulated algorithmically as follows.

```
For each topic $k \epsilon \{1, \ldots, M\}$:
    Generate $\beta k \ \{kw\}^{V}_{w=1} \sim Dir\ (. \ |\eta)$
```

For each document d ϵ {1, …, N}:

Generate ϴd

So, in LDA, both topic distributions, over documents and over words have also correspondent priors, which are denoted usually with alpha and beta. Generative process is represented as a graphical model representation, to convey the idea more briefly. The way of representing multiple documents N total with multiple words per document of Li words in document i can be graphically represented as follows in Figure 2.5.

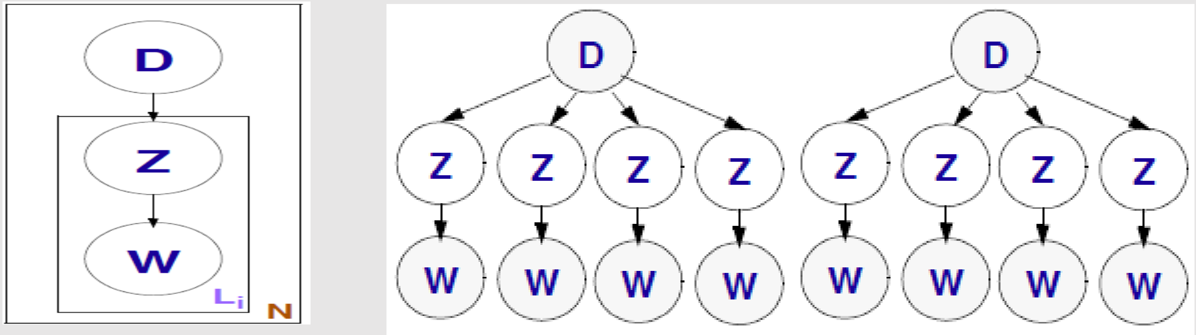


Figure 2 4 Plate Notation of generative model[[13]

The w_i approaches to Learning Conditional Probabilities is shown in figure 2.6 below.

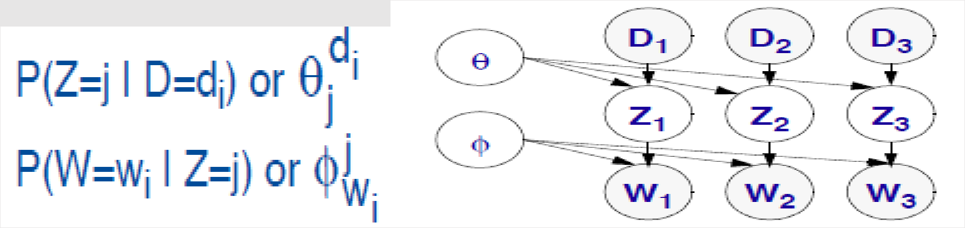
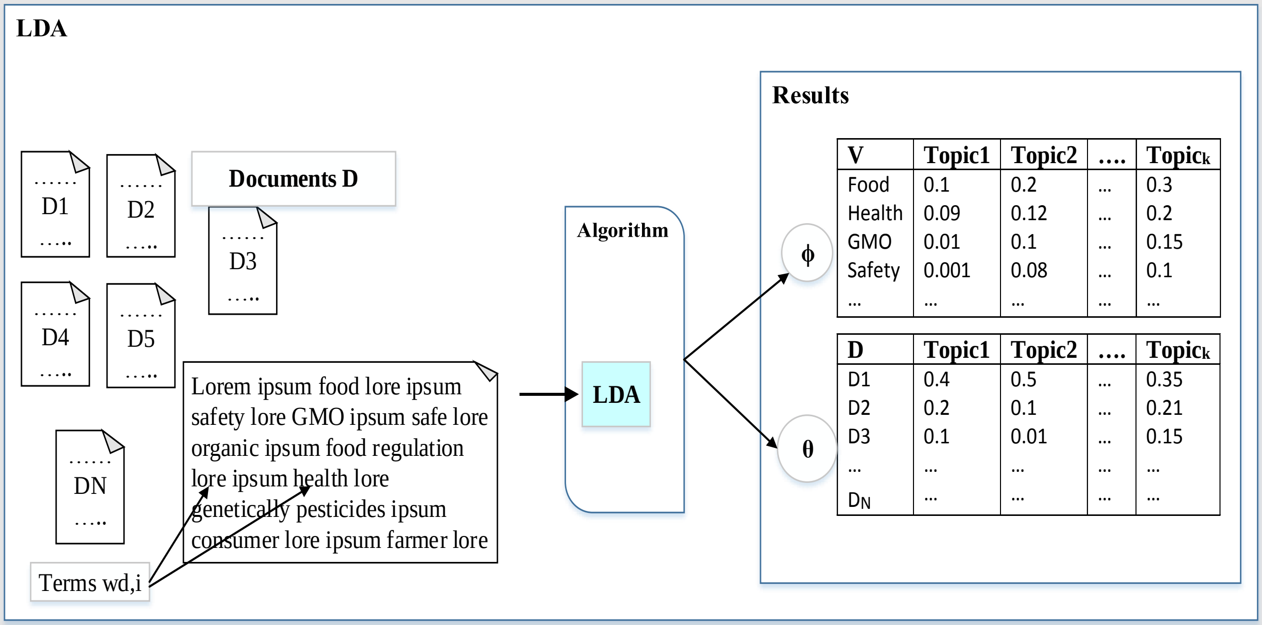

$P(Z=j \mid D=d_i)$ or $\theta_j^{d_i}$
$P(W=w_i \mid Z=j)$ or $\phi_{w_i}^{j}$
θ
ϕ
D1
D2
D3
Z1
Z2
Z3
W1
W2
W3

Figure 2. 5 Learning Conditional Probabilities [40]

2.5. Topic Modeling with Latent Dirichlet Allocation (LDA)

The aim of the LDA algorithm is to model a comprehensive representation of the corpus by inferring latent content variables, called *topics*. Since topics are hidden in the first place, no information about them is directly observable in the data. The LDA algorithm solves this problem by inferring topics from recurring patterns of word occurrence in documents [34]. Topics can be seen as factors that consist of sets of words, and documents incorporate such factors with different weights. Topic models draw on the notion of distributional semantics and particularly make use of the so-called *bag of words* assumption, i.e., the ordering of words within each document is ignored. To grasp the thematic structure of a document, it is sufficient to describe its distribution of words LDA relies on two matrices to define the latent topical structure: the word-topic assignment matrix ϕ and the document-topic assignment matrix θ (see Figure 2.7, right side). The word-topic assignment matrix ϕ has two dimensions, K and V, in which K is a numerical value defining the number of proposed topics in the model (which must be determined by the researcher), and V is the total number of words in the vocabulary of the corpus.



Lorem ipsum food lore ipsum safety lore GMO ipsum safe lore organic ipsum food regulation lore ipsum health lore genetically pesticides ipsum consumer lore ipsum farmer lore

V	Topic1	Topic2	….	Topic$_k$
Food	0.1	0.2	…	0.3
Health	0.09	0.12	…	0.2
GMO	0.01	0.1	…	0.15
Safety	0.001	0.08	…	0.1
…	…	…	…	…

D	Topic1	Topic2	….	Topic$_k$
D1	0.4	0.5	…	0.35
D2	0.2	0.1	…	0.21
D3	0.1	0.01	…	0.15
…	…	…	…	…
D$_N$	…	…	…	…

Figure 2. 7 LDA to a corpus [34]

Blei [12] has created a graphical model to represent how each variable relates to other variables. A probabilistic graphical model consists of a set of nodes, which are connected by edges. These graphical models are directed graphical model, like a Bayesian Network [41], that has links defined by arrows which imply directionality where the arrow expresses relationships between random variables, which are represented by nodes.

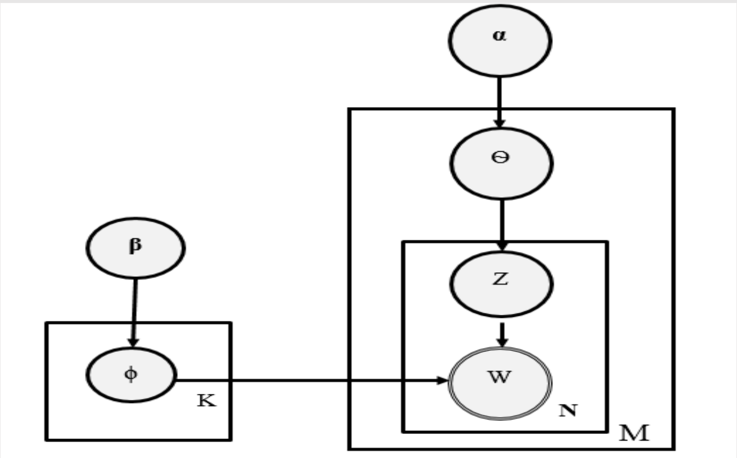


Figure 2. 8 Graphical model of the parameters of a dirichlet distribution [12]

Among the variables, the largest rectangle M denotes the number of documents or corpus, K denotes the number of hidden topics, N denotes the number of words in a document. The α and β are the document layer parameters of LDA, α denotes the relative strength of latent hidden topics in the document set and β denotes the probability distribution of all hidden topics. θ denotes the topic probability distribution for each document. ϕ denotes the word distribution for certain hidden topic, unilateral circle denotes hidden variables. Bi-circle denotes observable variables.

There is a Z value, for every word, in the document and in the corpus. Arrows indicate conditional dependencies between variables that provide great convenience for inferring the latent variables.

The computing formula of probability model is shown as [12]:

$$P(\Theta, Z, w|\alpha, \beta) = P(\Theta|\alpha) \prod_{i=1}^{N} P(Zn|\Theta)P(Wn|Zn, \beta) \qquad \text{-------------------------------------}(2.4)$$

The dirichlet distribution is an exponential family distribution over the simplex. The simplex is a space of positive vectors that sum to one. An exponential family distribution is a set of probability distributions based on a specific set of definitions [42] [43] [44].

When fitting an LDA model, the goal is to find the best set of latent variables that can explain the observed words in documents, assuming that the model actually generated the text collection. This involves inferring the probability distribution over words ϕ associated with each topic, the distribution over topics Θ for each document, and the topic responsible for generating each word.

The hyper parameters α and β are used as a prior to smooth the distribution over topics Θ and the distribution over words ϕ, respectively. These hyper parameters can be inferred from the observed data. Posterior inference can be conducted via standard statistical techniques such as Gibbs sampling [35], variational methods [45] and expectation-propagation [46].

Throughout this thesis, we focus on Gibbs sampling since it is easy to understand and to implement. By using this algorithm, it is possible to automatically assign a document as a mixture over latent topics or a topic is a distribution over words and they are learned through the help of statistical inference.

LDA adopts the bag-of-words assumption, that does not take in consideration the order of the words in a document. The generative process for LDA is similar to pLSI, where the first step is to select the number of words that will appear in the document. In the second step of the generative process, a topic is randomly selected for each word position. Unlike pLSI, where the probability distribution of the topics is assumed to be the same for all documents, each document in LDA has its own topic distribution. Once the topic is generated, a word is randomly chosen from the word probability distribution for that topic.

This is the generative process of LDA, for a document w of the corpus D:

1. Choose $\theta \sim Dir(\alpha)$.
2. For each of the N words w_n:
    (a) Choose a topic $z_n \sim Mult(\theta)$
    (b) Choose a word w_n from $p(w_n|z_n, \beta)$, a multinomial probability distribution conditioned on the topic assignment z_n.

It takes as input a set of D documents, in which word tokens $\mathbf{\{w_d\}^D{}_{d=1}}$ are from a vocabulary of V unique word types. There is a need to understand what each variable represents, so that it is possible to understand the generative process. A word is the item in a vocabulary from 1 to V, a document consists of N words, w = (w1, w2, . . ., wN) and a corpus is a set of M documents, D = (w1, w2, . . ., wM). The probability density of the Dirichlet is shown in equation 2.7 below.

$P(\Theta|\alpha) = \frac{\sum_{K=1}^{K} \alpha i}{\prod_{K=1}^{K} \alpha i} * \theta i^{\alpha i-1}$ -- (2.5)

LDA imagines that there are K shared topics, each of which is a multinomial distribution over the vocabulary drawn from a Dirichlet distribution prior.

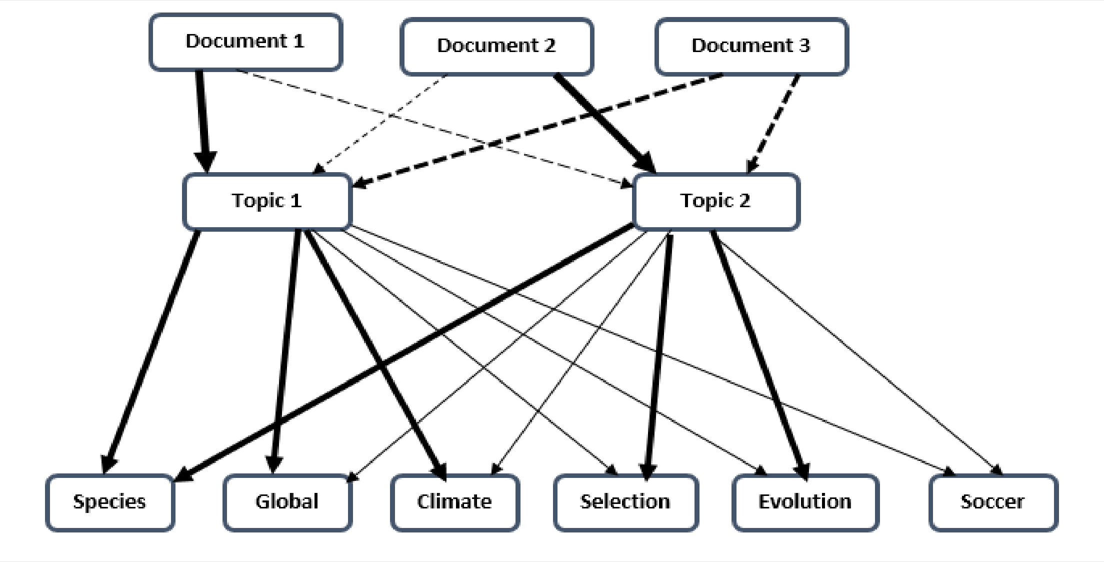


Figure 2 9 Topic modeling Probability Distribution [13]

There are two probability distributions used in topic modeling [13]. The first is being the probability distribution over topics. This is the group of topics that are most likely to be used in a specific document. An example of such is:

μ’1topic1 = 0.25

μ’1topic2 = 0.25

μ’1topic3 = 0.25

μ’1topic4 = 0.25

The second probability distribution is the probability distribution over words. These are the words most likely to be found in a specific topic. For example, in astronomy papers the probability distribution of one topic could be:

λ’1elliptical = 0.35 λ’1orbit = 0.25 24 λ’1satellite = 0.3

with probability distribution x = (x1, …, x_k) is

$p(x|\alpha) = \frac{\sum_{k=1}^{k} \alpha k}{\prod_{k=1}^{k} \alpha k} \prod_{k=1}^{k} x_k^{\alpha k-1}$ --(2.6)

Where x_i ∈(0,1) and $\sum_{i=1}^{k} xi = 1$

A Dirichlet distribution can also be parameterized by a concentration parameter α > 0 and a mean probability distribution p. This two-parameter Dirichlet distribution, denoted by Dirichlet (α ,p), is equivalent to Dirichlet (α) if we define $\alpha = \sum_{k=1}^{k} \alpha k$ and p=$\boldsymbol{\alpha}/\alpha$. When the mean distribution p is a uniform distribution, the Dirichlet distribution is called symmetric and often denoted by Dirichlet (α). Here Symmetric dirichlet distribution, where all of the elements making up the parameter vector α have the same value is used.

More specifically, the multinomial ϕk of topic k is a distribution over words

$p(\phi k|\beta) = \frac{\sum_{k=1}^{k} \alpha k}{\prod_{k=1}^{k} \alpha k} \prod_{k=1}^{k} x_k^{\alpha k-1}$ --(2.7)

Parameters are n number of trials and event probabilities pi, …, pk where $\sum pi = 1$. A Bayesian inference model is a method used to calculate the probability of an event occurring, given the observed data [15]. The more iterations done in this way, the more the model will accurately reflect the topics present in a corpus. The data is treated as observations arising from a generative probabilistic process that includes hidden variables. A generative probabilistic process is a process for randomly generating observable data. Posterior inference is where the hidden variables are estimated based on relevant background evidence. In the case of latent dirichlet allocation, topics are hidden variables and are inferred from the words in the documents [13]. An important point of LDA is that documents consist of multiple topics. LDA is used to decide which topics are being discussed in a specific document, based on the analysis of a set of documents already observed [19].

Specifically, LDA models a document as a probabilistic mixture of topics and treats each topic as a probability distribution over words. For the ith word in document d, denoted as $w_{d,\ i}$, the probability of $w_{d,\ i}$, P ($w_{d,\ i}$) is defined as:

$$\boldsymbol{P(w\ d, i) = \sum_{j=1}^{v} P(wd, i|Zd, i = Zj) * P(Zd, i = Zj)} \qquad \text{-------------------------------------(2.8)}$$

$\mathbf{Z_{d,i}}$ is the topic assignment for $w_{d,i}$, $Z_{d,i}$ = Zj means that the word $w_{d,i}$ is assigned to topic j and the V represents the total number of topics. Let $\boldsymbol{\phi_j}$ be the multinomial distribution over words for Zj, ϕj = (ϑj,1, ϑj,2, …, ϑj,n), $\sum_{k=1}^{n} \vartheta j, k = 1$. Θ_d refers to multinomial distribution over topics in document d. Θd = (ϑd,1, ϑd, 2, …, ϑd,v), $\sum_{j=1}^{v} \vartheta d, j = 1$. ϑ d,j indicates the proportion of topic j in document d. LDA is a generative model in which the only observed variable is $w_{d;i}$, while the others are all latent variables that need to be estimated. Blei et al. [12] [13] [47] introduce Dirichlet to the posterior probabilities ϕ_j and Θ_d, which contributes to optimize the distributions.

The process of generating a corpus is as follows

1. Randomly choose a distribution over topics
2. For each word in the document
    - b. randomly choose a topic from the distribution over topics
    - c. randomly choose a word from the corresponding topic

2.5.1.1. LDA Parameters

In unsupervised machine learning parameters selection for the model optimization is toughest challenges. These most critical hidden elements (hyper parameters) should be tuned to control the model in order to prevent overfitting (poor generalization). Hyper parameters are very specific to each model but there are common hyper parameters [47].

In many clustering algorithms finding the optimal value for the number of topics K is not trivial [15]. This is also the case for the Latent Dirichlet Allocation algorithm. The number of topics has great influence on the results of the clustering process, but often the evaluation of the results is subjective, difficult to be interpreted and time-consuming. This model parameter has to be set carefully, since based on it is the number of clusters. Indeed, low K values would lead LDA to be too coarse to be able to identify proper clusters, while too large K values would lead to a very complex model, difficult to be interpreted and difficult to be validated. Any way the K value depends on the dataset size [15]. In this study, an approach to find the most suitable value K would be explored.

2.6. Related Works

Statistical topic models, such as the Latent Dirichlet Allocation (LDA) [13] have been verified to be effective and widely applied in various text analysis tasks. Its notion is not new one and it is gaining increasingly considered in text mining communities. The nature of these topic models is that they are unsupervised and entirely probabilistic, therefore, they do not exploit any prior knowledge in the models. Recently, numerous works and schemes have used to extract hidden topics from text using this statistical algorithm for text analysis.

Latent Dirichlet Allocation (LDA) [12] [48] [49] is becoming a standard tool in topic modeling; it assumes that both word-topic and topic-document distributions and introduced in many works by applying to many domains. Griffiths and Steyvers [50] used LDA for capturing scientific topics in a collection of documents. The model was used for extracting author-topic model, to extract authorship information. In their model, the information about the authors of each document was used and each author was represented by a probability distribution over topics in addition to the statistics of words within documents. They applied the model to a collection of 1,700 NIPS conference papers and 160,000 CiteSeer abstracts. They used Gibbs sampling to estimate the topic and author distributions. Their work compares the performance with two other generative models for documents, which are special cases of the author-topic model: LDA (a topic model) and a simple author model in which each author is associated with a distribution over words rather than a distribution over topics. They showed topics recovered by the author topic model, and demonstrate applications to computing similarity between authors and entropy of author output.

Jonathan and David [51] developed the relational topic model, a model of documents and the links between them using LDA model and by the model they summarized a network of documents, to predict links between them, and predict words within them by derive efficient inference based on variational methods and evaluate the predictive performance of the model for large networks of scientific abstracts and web documents. From their result, their model shows a nearly 6% improvement in log likelihood for exponential model over baseline and 5% improvement over LDA + Regression.

Shi Jian-hong et al. [52] applied LDA topic model to Chinese micro blog topic and carried out better micro blog topic discoveries. Li Wen-bo et al. [53] used a labeled LDA topic model by adding text class information to the LDA topic model, which calculated the distribution of hidden topics in each class and raised the classification ability of the traditional LDA model.

The works of Abbas et al. [54] for Arabic language and Zrigui et al [55] proved that SVM outperforms TF-IDF by having the best values of precision and F-measure. Later, Abbas et al. [56] proposed another technique for topic identification named TR-Classifier based on triggers which are identified by using the Average Mutual Information where topics and documents are presented by triggers which are a set of words that have the highest degree of correlation. Then, based on the TR-distance, the similarity is calculated between triggers to identify the document's topic. As their result showed that, with the best choice of hyper parameters and stemming average accuracy of 65.8% result was obtained from LDA.

Zrigui et al [57] have proposed a new hybrid algorithm for Arabic topic identification named LDA-SVM. This algorithm is based on the combination of LDA and SVM. The LDA method is used to classify documents. Then the SVM method is employed to attach class label. The idea of this combination is to reduce the feature dimension by LDA before applying the SVM method. From their work they scored 88.1% of accuracy result.

Kelaiaia and Merouani [56] [58] proposed and employed LDA based topic identification for Arabic language documents. They showed 1.69% a consistently clear improvement of LDA over K-means on raw, cleaned, and stemmed forms of document collection with score result of 71.1%.

In addition, in [14] another work has been tried on Punjabi news items to extract topics found in the entire content of news. LDA modeling was also applied to the dataset to extract the themes. In their proposed models, they used the Labelled LDA topic modeling in calculating the probability of words given documents. As their report suggested that their work registered 67% of average accuracy of the result.

In fact, few works have been dealt on language other than Afaan Oromo on topic modeling and have been presented. Eyob and Dejene [59] proposed topic based Amharic text summarization using PLSA.

Our proposed approach is somewhat similar to a few previous works, particularly in terms of inferring topics [20][22]. Our work presented in this study differs from previous works, because none of the aforementioned works have used word embedding as their background knowledge in the topic models to capture the semantic structure.

For the Afaan Oromo language, there is a lack of research in the field of text mining that applies topic Modeling. So, in this work, we apply unsupervised LDA based topic models, that used generative process to extract topic in a corpus. LDA (unsupervised topic model) has been widely used in a variety of machine learning, natural language processing and information retrieval applications [18]. By using this algorithm, it is possible to describe a document as a mixture over latent topics, given that a topic is a distribution over words. The main goal is to automatically assign document collections with topic distributions, where a document may contain several topics that were learned with the help of statistical inference.

As we see from the previous works, they applied LDA under the bag-of-word model approach that does not consider any structure of the documents how words are correlated with each other. In the bag-of-word model, each word of the document is no order and independent of others. To enable LDA to produce the high-quality topics well-structured words should be feed into. So, in this study we incorporate the semantic representation using word embedding to represent co-occurred terms to provide context modeling.

CHAPTER THREE

AFAAN OROMO LANGUAGE

3.1. Overview

Afaan Oromo is one of the cushitic language family of Afro Asiatic languages. It has more than 40 million speakers. It is an official language of Oromia regional state [60] [61] [62]. With regard to the writing system, “Qubee” (a Latin-based alphabet) has been adopted and become the official script of Afaan Oromo since 1991 [63]. The language is also the media of instruction starting from elementary to university level and it is the language of public media, social issues, religion, political and technology. The language is rich in morphology [64]. The morphology used in this language need to be analyzed for basic text analysis before designing the proposed model.

3.2. Qubee Afaan Oromo Alphabets and Writing System

The writing system of the language is straightforward which is designed based on the Latin script called **Qubee** [64]. It consists of thirty-three basic letters, of which five are vowels, twenty-four are consonants, out of which seven are paired letters and fall together (a combination of two consonant characters such as ‘ch’). Alphabets are characterized by capital and small letters. Afaan Oromo vowels are represented by the five basic letters such as a, e, i, o, u. Vowels are sound makers as in English Language. Vowels in Afaan Oromo are can be short or long based on the sound made.

As in English Consonants, are few special combinations such as “ch” and “sh” (same sound as English), “dh” in Afaan Oromo is like an English "d" produced with the tongue curled back slightly and with the air drawn in so that a glottal stop is heard before the following vowel begins. Another Afaan Oromo consonant is “ph” made when with a smack of the lips toward the outside “ny” closely resembles the English sound of “gn”. We commonly use these few special combination letters to form words. For instance, ch used in barbaachisaa ‘important’, sh used in shamarree ‘girl’, dh use in dhadhaa ‘butter’, ph used in buuphaa ‘egg’, and ny used in nyaata ‘food’. Afaan Oromo has letters (consonants and vowels) shown in the following table 3.1.

		Bilabial/ Labiodental	**Alveolar/ Refroflex**	**Palatoalveolar /palatal**	**Velar/ Glottal**			
Stops	Voiceless	(P)	T	K	‘			
	Voiced	B	D	G				
	Ejective	Ph	X	Q				
	Implosive	Dh						
Affricates	Voiceless	Ch						
	Voiced	J						
	Ejective	Ch						
Fricatives	Voiceless	F	S	Sh	H			
	Voiced	(v)	-	Nasals		m	n	ny
Approximants		W	L	Y				
Flap/Trill		R						

	Front	Central	Back
High	i,ii	u,uu	
Mid	e,ee	o,oo	
Low	A	Aa	

Table 3.1 Afaan Oromo Alphabet (https://oromiaacademy.wordpress.com/oromo-language/afaan-oromo-basics)

3.3. Punctuation Marks in Afaan Oromo

Punctuations are placed in text to make meaning clear and reading easier in Afaan Oromo. The following are some of the most commonly used punctuation marks in Afaan Oromo [61]:

i. **Tuqaa Full stop (.):** is used at the end of a sentence and in abbreviations.

ii. **Mallattoo Gaaffii Question mark (?):** is used in interrogative or at the end of a direct question.

iii. **Raajeffannoo Exclamation mark (!):** is used at the end of command and exclamatory sentences.

iv. **Qoodduu Comma (,):** it is used to separate listing in a sentence or to separate the elements in a series.

v. **Tuqlamee colon (:):** the function of the colon is to separate and introduce lists, clauses, and quotations, along with several conventional uses.

3.4. Word and Sentence Boundaries

A word in Oromo cannot begin or end with a double consonant. The word for “sport” is converted to isportii. Three consonants cannot occur in a row in a word [64]. For this reason, certain suffixes may add an i to prevent this, as in arg (“see”) + na (1st per. plu. suffix) → argina (“we see”). Vowels cannot change without a break, either a consonant or apostrophe, between them. For example, “very” can be baa'ee, baayee, baa'yee, or baay'ee, and “to hear” can be dhaga'uu or dhagahuu. The apostrophe indicates that the vowels are produced independently and not as a diphthong [60] .

Space is used to separate words in the sentence as in English to show the end of one word; for example, “Caalaan Barataa Cimaa dha”. In this sentence the word “Caalaan”,” Barataa”, “Cimaa” and” dha” are separated from each other by white space character. Parenthesis, brackets, quotes can be used to show a word boundary. And again, sentence boundaries can be applied by using punctuations as in English language. A sentence may end with a period (.), a question mark (?), or an exclamation point (!) [61] [60].

Apostrophe mark (‘) in Afaan Oromo is used in writing to represent a glitch (called hudhaa) sound. In reading and writing system glitch plays an important role. For example, it is used to write the word in which most of the time two vowels appeared together like “ga’a” to mean (“enough”) with the exception of some words like “danda’a” to mean “can” which is identified from the sound created. Sometimes, apostrophe mark (‘) in Afaan Oromo can be used interchangeably with the spelling “h”. For instance, “ga’a”, “danda’a” can be interchanged by the spelling “h” like “gaha”, “dandaha” respectively still the senses of the words are not changed. So, in this work we replace all words comprised of apostrophe. The reason we replaced by ’h’ is while we process our corpus during the tokenization it consider as two words and it makes meaningless. For instance, the word ‘danda’a’ will be decomposed to “danda” and “a”.

3.5. Basic Sentence Structure

Afaan Oromo follows Subject-Object-Verb (SOV) format. But because it is a declined language (nouns change based role in a sentence).

Word order can be flexible, though verbs always come after their subjects and objects.

Example: Tolaan (Subject) saree (Object) ajjeese (Verb) to mean Tola killed the dog.

In addition to this, in Afaan Oromo the adjectives follow a noun or pronoun that they modify. For instance, Sangaa (Noun) diimaa (Adjective) to mean red ox.

3.6. Afaan Oromo Morphology

Morphology is a branch of linguistics that studies and describes how words are formed in a language [64]. Inflectional morphology is concerned with the inflectional changes in words where word stems are combined with grammatical markers for things like person, gender, number, tense, case and mode. Inflectional changes do not result in changes of parts of speech. On the other hand, derivational morphology deals with those changes that result in changing classes of words (changes in the part of speech). For instance, a noun or an adjective may be derived from a verb [65] [66].

3.6.1. Morphemes in Afaan Oromo

The smallest unit in a language is a morpheme [64]. Afaan Oromo comprise two categories of morphemes: free and bound morphemes. Free morpheme can stand as word on its own whereas bound morpheme does not occur as a word on its own [67]. Roots (stems) are bound as they cannot occur on their own. For example, “dhug-” (drink) and “beek-” (know), which are pronounceable only when other completing affixes are added to them [63]. An affix is also a bounded morpheme that cannot occur independently unless attached to the root [67]. Affixes are of three types – prefix, suffix, infix and circumfix. The prefix occurs at the beginning and suffix occurs at the end of a root whereas the infix occurs in between characters of the word. Circumfix occurs both at the beginning and end at the same time. In dhugaatii ‘drink’, for instance, -aatii is a suffix and dhug- is a stem. Moreover, an infix is a morpheme that is inserted within morpheme. As in the work of as stated by Debela [66] , discovered that Afaan Oromo lacks infix affixes. The morphological analyses of the language are majorly organized in nouns, verbs, adjectives, adverbs, functional words, and conjunctions.

3.6.1.1. Noun (Maqaa)

Nouns are used to identify class of people, places or things. They identify person, number, gender, and possession [64].

i. Gender

There are two gender system: feminine and masculine. Most nouns are not marked by gender affixes. Only a limited group of nouns differ by using different suffixes for the masculine and the feminine form. The **-ssa** affix for masculine and –**ttii** affix for feminine used in some words.

Ogeessa	expert (m.)	**ogeettii**	expert (f.)

ii. Number

Afaan Oromo has different suffixes to form the plural of a noun. In connection with numbers the plural suffix is very often considered unnecessary: miila isaa lamaaniin with his two leg(s).

– oota, –lee, -wwan, -een, -olii/ -olee and **–aan shows plural**.

-oota	saroota	dogs	-olii/-olee	gangoolii	mules
-lee	gaaffilee	questions	-een	fardeen	horses
-wwan	saawwan	cows	-aan	ilmaan	children

iii. Definiteness

In Afaan Oromo demonstrative pronouns are used to express definiteness.

kitaabni kun	*this/ the book (Subject)*
kitaaba kana	*this/ the book (Object)*
kitaaba sana	*that / the book (Object)*

Numerical can indicate indefiniteness like **tokko (**_one)_,

Example: **namni tokko** *one / a man*.

The suffix -**icha** (m.), -**ittii(**n**)**(f.) are definite articles.

Example: **jaarsichi**	*the old man (Subject*)	**jarsicha**	*the old man (Object)*
jaartittiin	*the old women (Subject)*	**jaartittii**	*the old lady (Object)*

iv. Derived noun forms

Afaan Oromo is very productive in word formation by different means. The most common word formation methods are derivational and compounding [67].

a. Derivation

Derivational suffixes are added to the root or stem of the word. From derived verbal stem and adjectives may be formed by means of derivational suffixes. The following suffixes play an important role in Afaan Oromo word derivation. They are -**eenya**, -**ina**, -**ummaa**, -**annoo**, -**ii**, -**ee**, -a, -**iinsa**, **-aa, -i(tii), -umsa**, -**oota**, -**aata**, and –**ooma**.

Examples:	**jabaa**	*strong*	**jabeenya**	*strength*
	jabina	*strength, hardiness*	**jabaa**	*intensive*
	jabummaa	*strength*	**jabaachuu**	*to be strong*
	jabaachisuu	*to make strong*	**jabeessuu**	*to make strong*
	jajabaachuu	*to be consoled*	**jabeefachuu**	*to make strong for one self*

b. Compound words

These are collective nouns that makes another noun and they represent titles.

abbaa caffee	*chairman of the legislative assembly*
abbaa gadaa	*traditional Oromo president*
abbaa duulaa	*traditional Oromo minister of war*
abbaa dubbii	*chief speaker of the caffee assembly*

3.6.1.2. Adjectives (Maq-ibsii)

An adjective is a word which describes or modifies a noun or pronoun [60]. A modifier is a word that limits, changes, or alters the meaning of another word. Unlike English adjectives are usually placed after the noun in Afaan Oromo.

For instance, in **Tolaan farda diimaa bite** "*Tola bought red horse"* the adjective **adii** comes after the noun **farda.** Moreover, in Afaan Oromo sometimes it is difficult to differentiate adjective from noun [68].

Example: **dhugaa** *truth, reality, true, right* **dhugaa keeti** *your truth/ you are right* (truth served as noun) **obboleessi hiriyaa dhugaati** *brother is the friend for truth / brother is a true friend* (true served as adjective)

i. Gender

In Afaan Oromo adjectives are inflected for gender. We can divide adjectives into four groups with respect to gender marking. These are: In the first group the masculine form terminates in –**aa**, and the feminine form in **–oo**.

Example: **guddaa (***m***.)**	**nama guddaa**	*a big man*
guddoo(*f*.)	**nama guddoo**	*a big woman*

In the second group the masculine form terminates in –**aa**, the feminine form in – **tuu** (with different assimilations).

Example: **dheeraa**(*m.)*	**nama dheeraa**	*a tall man*
dheertuu(*f.)*	*i***ntal dheertuu**	*a tall girl*

Third Adjectives that terminate in –**eessa** or **–(a)acha** have a feminine form in –**eettii** or –**aattii**.

Example: **dureessa** (*m.)*	**nama dureessa**	*a rich man*
dureettii (*f.)*	**niitii dureettii**	*a rich woman*

Fourth, Adjectives whose masculine form terminates in a long vowel other than **–aa** as in short vowel **–a** are not differentiated with respect to their gender.

collee(*m*.) **farda collee**	*an active horse*
collee(*f.)* **gaangee collee**	*an active mule*

ii. Number

Most of the adjectives form the plural by reduplication of the first syllable masculine and feminine adjectives differ in plural as they do in singular [68]:

Example: Singular Plural

guddaa(*m***.)**	**guguddaa(***m.)*	
guddoo(*f***.)**	**guguddoo(***f.)*	
xinnaa(*m***.)**	**xixinnaa(***m***.)**	
xinnoo	**xixinnoo**	
pl.f. **lageewwan guguddoo**		*big rivers*
pl.m. **qubeewwan guguddaa fi xixiqqaa**		*big and small letters*

In other way, there are neutral adjectives.

Sg.m	dheeraa	jabaa
Sg.f	dheertuu (	jabduu
Pl.m	dhedheeraa	jajjabaa
Pl.f	dhedheertuu	jajjabduu
Gender neutral	dhedheertuu/jajjaboo	

3.6.1.3. Verbs

Verbs are words that denote an action, occurrence, or state of existence [68]. In Afaan Oromo base (stem) verbs and four derived verbs are there. Moreover, verbs in Afaan Oromo are inflected for gender, person, number and tenses.

i. Derived stems

According to Rabira [61], there are four derived stems, the formation of which is still productive in Afaan Oromo such as: Autobenefactive (AS), Passive (PS), Causative (CS) and Intensive (IS).

a. Autobenefactive

In Afaan Oromo this is formed by adding **-(a) adh**, **-(a) ach** or **-(a)at** or sometimes **-edh**, **-ech** *or* –**et** to the verb root.

Example: **bitachuu** *to buy for oneself* the root verb in this case is **bit-**

The conjugation of a middle verb is irregular in the third person singular masculine of the present and past (-**dh** in the stem changes to **-t**) and in the singular imperative (the suffix is **-u** rather than **–i)**.

Examples: **bit-** *buy* **bitadh-** *buy for oneself*

Infinitive and participles are always formed with **-(a)ch**, while the imperative forms have -(**a)(a) dh** instead of ***-(a)ch.***

Infinitive	imperative	sg. Imperative pl.	English
argachuu	**argadhu**	**argadhaa**	*to find / get*

b. Passive

It is formed by adding **-am** to the verb root. The resulting stem is conjugated regularly.

Example: **beek-** *know* **beekam**- *be known*

c. Causative

The Afaan Oromo causative of a verb corresponds to English expressions such as 'cause ', ‘make ‘, ‘let ‘. With intransitive verbs, it has a transitiviving function. It is formed by adding **-s, -sis, or -siis** to the verb root example:

deemuu to go **deemsisuu** to cause to go

A second causative of an intransitive verb would create a real causative.

Agarsiisuu to show

Sg. 1. p. agarsiisa	Sg. 2.p. agarsiifta
Sg. 3.p.m. agarsiisa	Sg. 3.p. f. agarsiifti
Pl. 1.p. agarsiifna	pl. 2.p. agarsiifti
pl. 3.p. agarsiisu	

d. Intensive

Formed by duplication of the initial consonant and the following vowel, geminating the consonant.

Example: **waamuu** to call, invite **wawwaamuu** to call intensively

ii. Simple tenses

Simple tenses in Afaan Oromo tell us when the action happens as in English language.

a. Infinite forms

Infinitive is an uninflected form of the verb. In Afaan Oromo infinitive form of verbs terminates in -uu.

Examples: **arguu** *to see* **deemuu** *to* go

On the other hand, the infinitive forms of Auto-benefactive verbs terminate in **-chuu**.

Example: **jiraachuu** *to live* **bitachuu** *to buy for oneself*

Participle/ gerund

Participle is a non-finite form of the verb whereas a gerund is a noun formed from a verb (in English the '-ing' form of a verb when used as a noun). In Afaan Oromo a participle is formed by adding -**aa** to the verb stem [61].

Example: **deemaa** *going* **jiraachaa** *living*

According to the meaning of the verb these forms may serve as agent nouns.

barsiisaa *teacher* **gaafatamaa** *responsible person*

For these agent nouns feminine forms are used according to the pattern of feminine adjective formation.

barsiiftuu *teacher* **gaafatamtuu** *responsible person*

On the other hand, a gerund is formed by adding **-naan** to the verb stem.

deemnaan *after having gone* **nyaannaan** *after having eaten*

b. Imperative

Imperative singular of base stems and all derived stems beside autobenefactive stems is formed by means of the suffix -**i**.

Example: **deemi**! *go!* **argi**! *look!*

The imperative singular of autobenefactive stems is formed by means of the suffix -**u**.

Example: **jiraadhu**! *live!*

Imperative plural of all stems is formed by means of -*aa*.

Example: **deemaa**! *go!* **argaa!** *see!*

Negative imperatives are formed by means of **-(i)in** for singular and **-(i)inaa** for plural.

Example: **Daandii balaaf nama saaxilu irra hin deemiinaa.** *Don't go in danger way*.

c. Finite forms

The Afaan Oromo language uses different conjugations for the verbs in main clauses and in subordinated clauses for actions in present or near future. The first-person singular is differentiated from the third person masculine by means of an -*n* that normally is suffixed to the word preceding the verb.

i. Present tense main clause conjugation

The present tense main clause conjugation is characterized by the vowel -**a**:

deemuu *to go*

sg. 1.p. **deema** 2.p. **deemta** 3.p.m **deema** 3. p. f **deemti** pI. l.p. **deemna**

2.p. and polite form **deemtu/deemtan(i)**

3.p. and polite form **deemu/deeman(i)**

Examples: **Isaan gara mana amantaa deemu.** *They go to church*

ii. Past tense conjugation

The past tense conjugation is characterized by the vowel ***-e:***

Deemuu *to go*

sg. 1.p. **deeme** 2.p **deemte** 3.p.m **deeme** 3.p.f **deemte** pI. l.p. **deemne**

2.p. and polite form **deemtani**

3.p. and polite form **deemani**

Example: **Caaltuun gara mana yaalaa deemte.** *Chaltu went to the hospital*

iii. Subordinate conjugation

The subordinate conjugation is used in affirmative subordinated clauses and in connection with the particle **akka**. Beside this the subordinate conjugation is used to negate present tense actions.

Deemuu *to go*

sg. l.p **akkan deemu** 2. p. **akka deemtu**

3.p.m. **akka deemu** 3. p.f. **akka deemtu**

pI. l.p. **akka deemnu** 2. p. and polite form **akka deemtani**

3.p.and polite form **akka deemani**

Examples: **Akkan yaadutti barattootni dhufaniiru.** *As I thought, students have come.*

iv. Contemporary verb conjugation

The contemporary verb conjugation is used only in connection with the temporal conjunction **-odoo,-otoo,-osoo,-otuu** or **-utuu** that being connected with this conjugation means '*while'*.

*Example: "***Osoon si waamuu maaliif deemta ?"** . *"While I was calling you (pI.) why do you go?".*

v. Jussive

To form the jussive in Afaan Oromo the particle **haa** has to be used in connection with the subordinate conjugation. Example:

Isaan haa nyaatani *they shall to eat*

vi. Negation

Present tense main clause actions are negated by means of the negative particle **hin in Afaan oromo.**

Example: **Caalaa hin jiru.** *Chala is not present.*

Present tense actions in subordinated clauses are negated by means of the negative particle **hin** and a suffix –**ne, -tu, -tan, -an** that is used for all persons. Past tense actions are negated in the same way.

Example: *Isaan hin dhufan.* *They don't come.*

Verb derivation

Some Afaan Oromo verbs are derived from nouns or adjectives by means of an affix **-oom**. These verbs usually express the process of reaching the state or quality that is expressed by the corresponding noun or adjective. From these process verbs causative and autobenefactive stems may be formed.

Examples: **danuu** *much, many, a lot* **guraacha** *black*
danoomuu *to become much* **gurraachomuu** *to become black*

Causative verbs, however, can also be derived directly from adjectives or nouns by suffixing a causative affix -**eess** to the stem of the noun or adjective,

Example: **danuu** *much* **daneessuu** *to increase, multiply*

Another means to derive process verbs from adjectives in Afaan Oromo is to form an autobenefactive stem,

Example: **Adii** *white* **addaachuu** *to become white*

Compound verbs

Compound verbs can be formed by means of pre-/postpositions, pronouns and adverbs in Afaan Oromo such as **ol** *above,* **gad** *below,* **wal, waliin, walitti, wajjin** *together,* **keessa** *in, jala under.* They precede different verbs and express a broad variety of meanings [66].

Example: **gadi dhiisuu** *to let go of* **gaddhiisuu** *to let* go *of*

Compound verbs can also be formed with **jechuu** or **gochuu**.

Example: *cal* **jechuu** (*to be quiet, silent) cal* **gochuu** (*to make quiet silent)*

'To be' and 'to have'

Afaan Oromo has different means to express *'to be'*. One of them are copulas, other means are the verbs **ta'uu**, **jiruu** and **turuu** [68].

The morphemes *(-)* **dha** and *(-)* **ti** (suffixed or used as independent words) serve as affirmative copulas as well as the vowel **-i** that is added to nouns terminating in a consonant. The copula **dha** is used only after nouns terminating in a long vowel.

Negative copula is **miti**, irrespective of the termination of the noun.

Examples: Present tense: **Atis jabaa dha.** *You are strong, too.*

Nouns terminating in a short vowel do not take any copula.

Example: **Isheen durba.** *She is a girl.*

Nouns and pronouns terminating in a consonant are combined with the copula.

Example: **Kuni bisbaani**. *This is water.*

In all utterances related to possession only the copula -*ti* may be used.

Example: **Hojiin hundee guddinaa ti!** *Work is the basis of development.*

Present progressive: **Inni dhufaa jira.** *He is coming*

Past tense:

In Afaan Oromo **ture, -iiru, jiru, -tti, -eera** expresses past

Ani dhufeen ture. *I was come*

3.6.1.4. Adverbs

Adverbs have the function to express different adverbial relations such as relations of time, place, and manner or measure. Some examples of adverbs of time:

amma *now*

booda *later*

Some examples of adverbs of place:

achi(tti) *there* **ala** *outside*

Some examples of adverbs of manner:

saffisaan *quickly* **sirritti** c*orrectly*

Some examples of adverbs of measure: **baay'ee, danuu** *much , many , very*

duwwaa *only, empty*

3.6.1.5. Pre-, post- and Para- positions

Afaan Oromo language uses prepositions, postpositions and para-positions. As other language Afaan Oromo is rich in prepositions and postpositions; links a noun to an action (e.g. go from there) or to another noun (e.g. the pen on the table). Preposition comes before noun and postposition comes after noun. Postpositions can be grouped into suffixed and independent words [68].

Suffixed postpositions include the following.

-tti *in, at, to*

-rra/irra	*on*
-rraa/irraa	*out of, from*

The postposition –**tti** is used to show the locative. The postposition **-rraa/irra** may be used to express a meaning similar to ablative.

Example: **Adaamaatti yoom deebina?** *When shall we go back to Adama*?

Gammachuun sireerra ciise. *Gemachu lay down on bed.*

Postposition as independent words include the following.

ala	*outside*	**wajjiin**	*with, together with*
bira	*beside*	**teellaa**	*behind*

Example: **Namoota nu bira jiraniis hin jeeqnu.** *We don't hurt people who are with us.*

Prepositions

akka	*like, according to*	**hanga/hamma**	*until, up to*
gara	*to, in the direction of*	**karaa**	*along, the way of, through*

The prepositions **gara**, **hanga**, and **waa'ee/waayee** are still treated as nouns and therefore are used in a genitive construction with other noun they belong to, expression: the direction to, the matter of, etc.

Example: **Namni akka isaa hin jiru.** *Nobody is like him.*

Para-positions

Gara… tti	to
Gara… tiin	from the direction of

Example: Caalaan gara manaatti gale. *Chala went to home.*

3.6.1.6. Conjunctions

The main task of conjunctions is to be a syntactical formative element that establishes grammatical and logical relation between the coordinated constituents [68].

i. Independent Conjunctions

They join words, phrases or clauses in particular sentence.

1. **Coordinating**

Example: **garuu** *but*

Hoolaan garuu rooba hin sodaattu. *But the sheep is not afraid of rain.*

2. **subordinating**

Example: **akka** *that, as if, as whether*

Maaliif akka yaada dhuunfaa yookaan yaada haqaa akka ta'e adda baasii barreessi

ii. **Suffixed Conjunctions**

They can be formed by suffixing to the root word.

Example: **–f/ -fi/ -dhaaf** *and, that, in order to, because, for*

Loon horsiisuuf bittee?

Did you buy the cattle for breeding?

iii. **Conjunction with one, two or more parts.**

They can be formed made up of two single conjunctions that are used after each other in order to give more detailed information about the logical relation or to intensify it.

Example: **akkam akka** *how, that*

Dura namni tokko beekumsa mammaaksaa akkam akka jabeeffatu ilaaluu nu barbaachisa. *At first, we have to see how a person extends the knowledge of proverbs*

iv. Conjunctions with several segments

These are stable, stereotyped constructions the first segment of which has to be followed by a certain second segment:

Example: **–s… -s,** *as well as*

Jechoota hudhaa wajjiiniis, hudhaa malees karaa lamaan barreeffaman

Words with glottal stop as well as without glottal stop are written in two ways.

In general, the morphological complexity of Afaan Oromo languages increases the load on professionals working in the field of natural language processing (NLP).

For the purpose of topic modeling, in any NLP, the variants of words in morpheme should be manipulated so that they can be counted with term frequency and removing stop words that do not carry important information in the document like prepositions, conjunctions.

3.7. Challenges of Topic modeling

The Topic modeling is under various challenges to get the better accuracy and to provide with better quality topics. And those challenges are under research to be solved by many researchers using different algorithms. The major challenge is the ambiguity of the language; that means, the capability of being understood in two or more possible sense. One word or phrase may have multiple meanings those can lead to ambiguity problem. We described major types of ambiguity in Topic Modeling as follows.

1. **Lexical ambiguity or word sense disambiguation (WSD):** The same word may indicate multiple meanings depending on it placed and used in a sentence. Homonomy and synonymy are common ambiguity in NLP. With homonomy, the word is the same but the meaning is different through the evolution of the words through history. For example, consider two different meanings for the same word "***bank***":

Chala jogged along the ***bank*** of the river.	Geographical description
CBE is the oldest ***bank*** in Ethiopia.	Specific financial institution

Synonymy refers to different words that have similar meanings, sometimes interchangeable meanings. For example, in these two sentences, "large" and "big" are used interchangeably:

The fish very **large**.

The fish very **big**.

In Afaan Oromo, these both challenges can occur and it may provide difficulty in topic interpretation. For Instance, consider the word "teessoo".

Teessoo kee natti himi (Tell me your address)	address/where someone live
Teessoo kanarra taa'I (Have a seat on this chair)	Stool/Chair

Afaan Oromo is rich in synonymy. In both sentences "qoricha" and "dawaa" are same in meaning.

Qoricha mataa bowwuu naaf ajaje. (He ordered headache tablet)

Dawaa liqimsaan jira (I am swallowing the tablets)

Both terms refer to **drug**

2. **Abbreviations and Acronym:** Jargon words like “btw” for by the way, and “ppl” for people do not easily interpreted since they are not in the dictionary and they affect the results quality. In Afaan Oromo terms like, ***kkf*** (etc.), ***wkf***(etc.), fkn (for example), g/g (center), bg (very good) can be exist in the corpus and if they are extracted as the representative of the topic, they may make difficulty in the topic labeling.
3. **Spelling Variations and Misspelling**- Some letters in Afaan Oromo plays a very important role. Words make a major difference in phonetics and make a major difference in the way of writing and its spelling. Misspellings are sometimes obvious and easily corrected, though sometimes the intended word is not clear except from the context. Spelling variants make difficulty for the analyst again in topic labeling. Some variants are happened due to location. For instance, beektota and beeytota (can be wriiten in Hararge) both means “wise”.
4. **Keyword selection:** Before a topic model is estimated on the corpus, the filtration techniques is applied on the original data to remove undesirable components. Cleaning and preprocessing affect the input vocabulary and the documents included in the modeling process. Some words remain in the text affect the result. Too big unique keywords, may be an extremely challenging problem.
5. **Compound Words:** Two or more words together, but the overall meaning of the compound word may not reflect the meaning of its component words. “Airport”, “crosswalk,”,” bookworm”, and “deadline” are few English examples of compound words examples. For instance, the English “deadline” refers to the final acceptable time to receive something. It has nothing to do with death or a line.
6. **Data Sparsity:** Data sparsity is more of an issue problem in machine learning that influence the accuracy of the system because the co-occurrence matrix will have a lot of zeros in it. This is a bad thing because it is very CPU- and memory-inefficient.
7. **Evaluation Metrics:**

There are other challenges that diminish the quality of topics to be extracted in LDA topic model.

8. **Model Parameter Selection:** Three model parameters must be selected (K, α, and β), which affect the substantial importance for the resulting good topic model and a priori defined distribution of the target variables, ϕ and θ. Thus, the selection of appropriate prior parameters and the number of topics is crucial to retrieve models that adequately reflect the data and can be meaningfully interpreted. Thus far, there is no statistical standard procedure to guide this selection; thus, this remains one of the most complicated tasks in the application of LDA topic modeling. Our proposal suggests a two-step approach: In the first step, the prior parameters are adjusted along the mean intrinsic coherence of the model, i.e., a metric focused on the interpretability to find appropriate candidate models with different numbers for the K proposed topics. In the second step, a qualitative investigation of these candidates follows, which aims to match the models' results with the theoretical concept under study.
9. **Model Reliability and Interpretability Evaluation:** Since there is no label in unsupervised learning, it is difficult to get a reasonably objective measure of how accurate the algorithm is. Metrics that drive decision making model tuning is defined. Due to both random initialization and statistical inference, the results from topic models are not entirely deterministic. To evaluate reliability metric random initialization is a weakness in the LDA architecture. To make LDA topic modeling more accessible and to ensure compliance with standards the above-mentioned challenges need to be addressed carefully.

CHAPTER FOUR

THE PROPOSED DESIGN

4.1. Overview

This chapter explains the proposed system architecture to solve the defined problem. It defines the structure and behavior of each individual component of the system. In this work, we propose a semantic based probabilistic model using LDA model for Afaan Oromo texts.

4.2. The proposed Architecture

To achieve the objective of the study a architecture and the algorithm should be implemented based on the designed architecture. The general overall architecture of the LDA based topic model with its contents is depicted in Figure 4.1. The proposed Topic Modeler for Afaan Oromo aimed to extract the theme of a given text collection using generative LDA algorithm where words are arranged into groups with similar meaning. To get quality topics different data cleaning techniques have applied to the corpus; data collection, pre-processing and fitting the model. Final phase predicts the topic based on probability for a word in the collection.

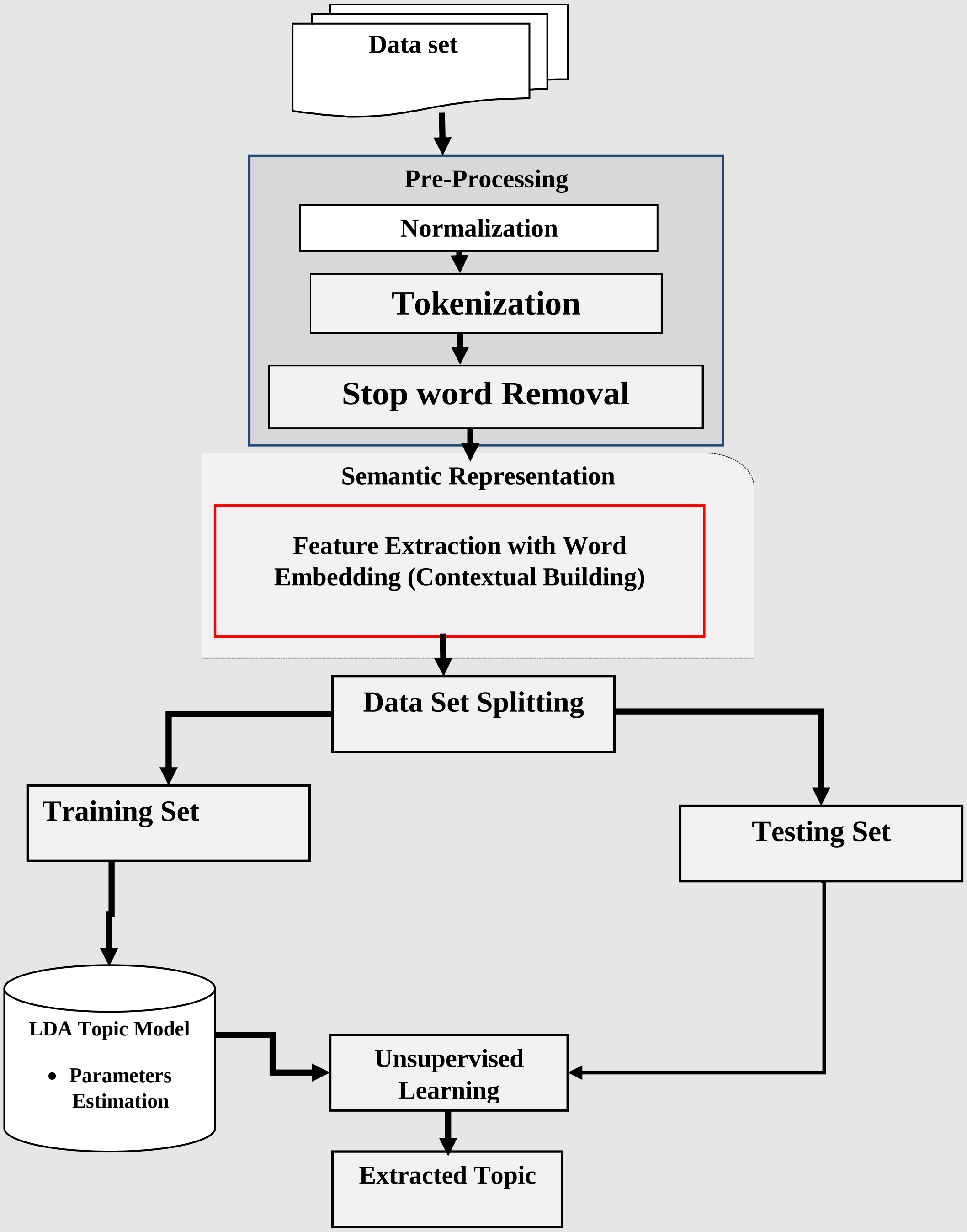


Figure 4. 1 The Proposed Architecture

4.2.1. Data Collection

Documents are collected manually from FBC and BBC Afaan Oromo News collections and stored in TXT file format for processing. We accessed the two sites and collected the documents by going through their categories. We selected four categories only for testing purpose. The selected categories include Education, Health, Sport and Weather. Each category consists of 4 instances, with total of 16 documents. These instances consist of different sentence length and words, as shown in table 4.1.

News type	Sentences	Words
Education	350	3357
Health	256	2328
Sport	790	2348
Weather	420	5470
Total	**1816**	**13,503**

Table 4.1: distribution of data collected for training and testing

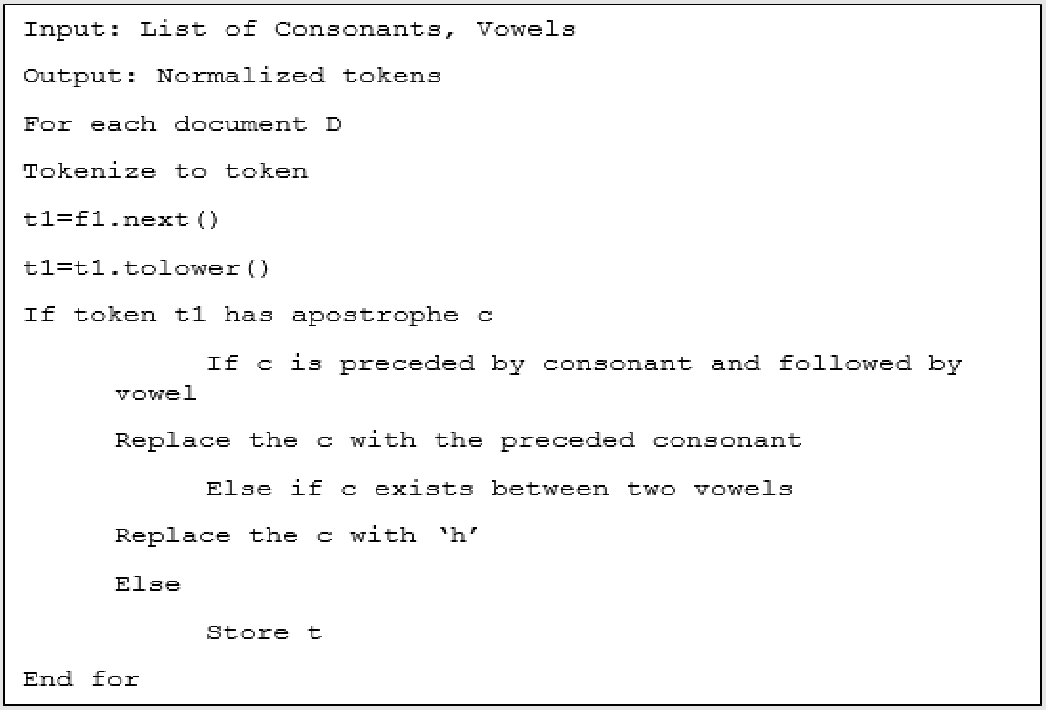
4.2.2. Text Pre-Processing

In pre-processing stage, the collected news articles are passed through normalization, tokenization and stop words removal. This step would be performed to acquire the most representative words which is then converted to vector matrix to represent the document in LDA.

4.2.2.1. Normalization

All words were converted to lower case. Numbers and punctuations (**“. , ; : ” ? !**) were removed from the dataset. Words in Afaan Oromo compose apostrophe to make sound for a given word; for instance, ‘many’ English word would be ‘baay’ee’. During tokenization this word can be considered as two strings since we used Java standard regular expression. So, to avoid this, we little bit change the structure of such words. If a character before an apostrophe is consonant, we make that character double to omit that apostrophe. In addition, if the character before the apostrophe is vowel, we replace the apostrophe position with affricate voiceless character **‘h’** before the actual tokenization process is going on. So, the word **‘baay’ee’** becomes **‘baayyee’** because the character **‘y’** before the apostrophe is consonant so we doubled the **‘yy’**. The word

‘boba’aa’ (fuel in English) would be **‘bobahaa’** since the character **‘a’** before apostrophe is vowel. Figure 4.1 presents pseudocode to normalize the data.

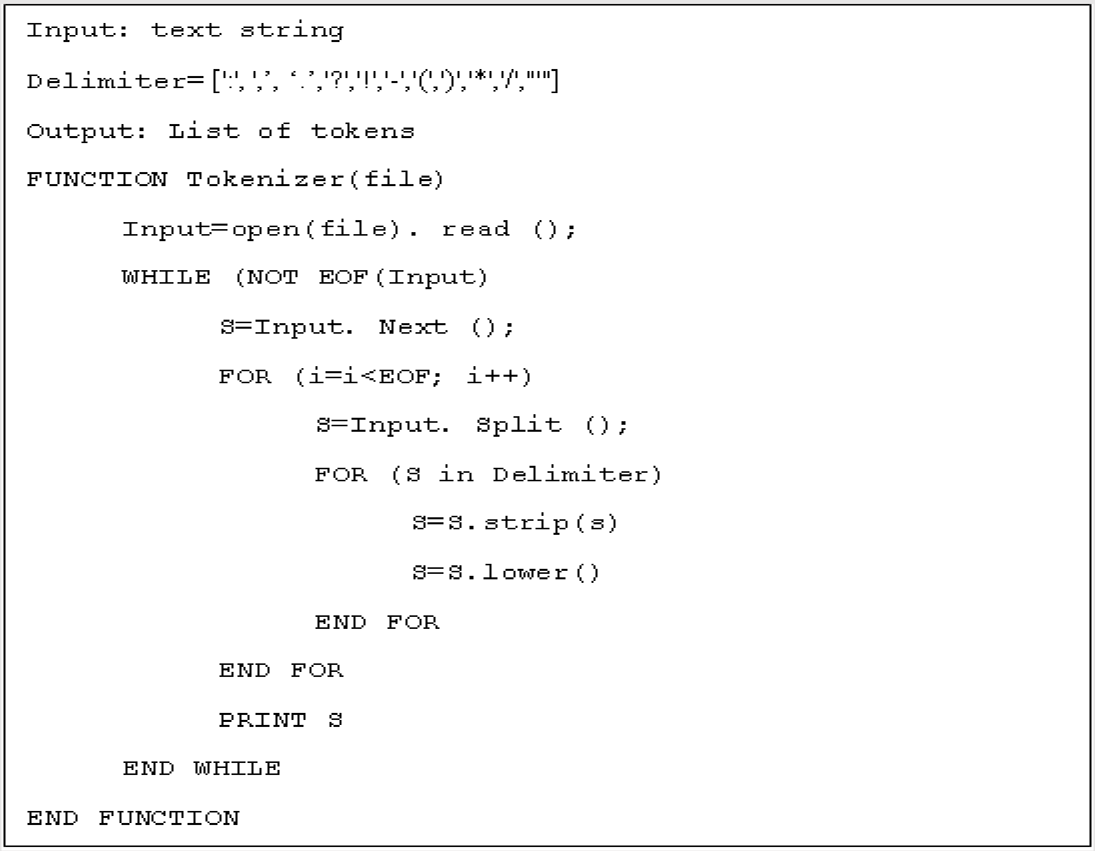


```
Input: List of Consonants, Vowels
Output: Normalized tokens
For each document D
Tokenize to token
t1=f1.next()
t1=t1.tolower()
If token t1 has apostrophe c
          If c is preceded by consonant and followed by
     vowel
     Replace the c with the preceded consonant
          Else if c exists between two vowels
     Replace the c with ‘h’
     Else
          Store t
End for
```

Figure 4. 2 Pseudocode for Afaan Oromo text Normalization

4.2.2.2. Tokenization

First, the whole document collection as an input would be tokenized into individual tokens. This is done to remove unwanted part of reviews in dataset. The list of tokens becomes input for further processing in text mining. Strings were broken into tokens with set of delimiters to distinguish numbers and punctuations.



```
Input: text string
Delimiter=['.',',', ‘.’,'?','!','-','(',')','*','/',""]
Output: List of tokens
FUNCTION Tokenizer(file)
      Input=open(file). read ();
      WHILE (NOT EOF(Input)
            S=Input. Next ();
            FOR (i=i<EOF; i++)
                  S=Input. Split ();
                  FOR (S in Delimiter)
                        S=S.strip(s)
                        S=S.lower()
                  END FOR
            END FOR
            PRINT S
      END WHILE
END FUNCTION
```

Figure 4. 3 Pseudocode for tokenization of Afaan Oromo texts

4.2.2.3. Stop words Removal

Common words that are not useful for content analysis were removed from the corpus since they do not carry the semantic meanings. These words since they overpower the rest of the corpus and they are not important for content analysis [32]. Terms which appear 3 times or less than in documents were removed. In fact, Stop-word lists can be domain-dependent, we identified common words that appeared in a dataset. We collected list of stopwords from different works and we identified **315** words that assumed as common words in our case.

Stemming is used to reduce the inflectional and derivational forms of words to a common base form. Stemming is generally not useful for topic modeling because the topics become difficult to interpret because word’s morphological roots can make a topic’s words in unrecognizable. This makes ambiguity of topic interpretability in topic labelling. Since the model group related words together simply by context, and stemming often produces words that are difficult for users to interpret. So, we did not apply this task.

4.2.3. Building Training and Test Set

We split the corpus into a training data and test data based on percentage. For each document we split 90% as training and 10% as test set. The stored the vocabulary words represented as count matrix features with word embedding approach that fed to the Unsupervised LDA function.

```
test_docs, train_docs = [], []
for doc in docs:
    test_docs.append(np.array(doc[0:len(doc)]*0.1))
    train_docs.append(np.array(doc[len(doc)]*0.9))
test_docs, train_docs = np.array(test_docs),np.array(train_docs)
```

Figure 4. 4 Algorithm of Splitting Corpus

4.2.4. Semantic Representation

Text documents are represented in NLP as a bag-of-words that represented as a fixed-length vector with length equal to the vocabulary size. This model representation doesn’t consider the semantic relation between words [14]. Neighbor words in a document are useful for figuring out semantic relatedness. This is very good approach for LDA algorithm to capture the quality topics during learning to predict the context of words. We can handle this through word embedding to improve model’s performance. There are many forms of word embedding. The most popular are word2vec, GoVe and FastText. From this we used word2vec to train our model by considering semantic structure. Word2vec is highly popular word embedding model, developed by Mikolov et al. There are two main architecture of word2vec; Continuous Bag-of-word (CBOW) and Continuous Skip gram. In this work, CBOW is used. This model uses the target word to predict window of context words. We used context with two window size.

Words are represented as fixed-length vectors or embeddings for capturing semantic structure of documents. Words that occur in the same context are represented by vectors in close proximity to each other.

Then we interpreted data in a more general space, with fewer dimensions, to create an interesting the representation using word embedding. Words in a given document are represented as discrete atomic symbols in the form of a vector using word embedding representation with relative similarity correlate with the semantic model of words which provides that different text objects are connected by semantic links. Similar words tend to have similar vectors. This work focused to integrates word embedding into the LDA model using continuous bag of words model through Word2Vec approach that works well in large dataset. LDA is not based on context rather it based on BOW.

The first step of the process is to create a binary vector, V with the corpus vocabulary, where each Vi represents a word of the corpus.

V= [fayyaa mana barumsa ogeessa barataa eegumsa yaalaa]

Using this vector, each of the given sentences are represented, by turning on and off the corresponding index of each word. For example, the sentence “Mana barumsaa deemna” (Sentence1), “Fayyaa keenyaaf eegumsa gochuu qabna” (Sentence2) and “Ogeessa fayyaa mana yaaalaa” (Sentence3) are equal to:

Sentence1= [0 1 1 0 0 0 0]

Sentence2= [1 0 0 0 0 1 1]

Sentence3= [1 1 0 1 0 0 1]

For simplicity we represent using binary values for each word, but we can represent using the times of occurrence in the given sentence.

We used continuous bag-of-words (cbow) model approach to provide more effective accuracy of sentences. For example, these models from the third sentence “Ogeessa fayyaa mana yaaalaa” would create the following smaller ones for training, resulting in a more flexible dataset:

Continuous Bag-of-words with bi-gram: -

‘Ogeessa fayyaa ‘fayyaa mana’,’ mana yaaalaa’

We count how many times the bigram co-occurs with each other. Therefore, we can capture the weights/features learnt from each word that represent an efficient method for learning high-quality distributed vector representations of syntactic and semantic word relationships.

4.3. The LDA Model Training

4.3.1. Parameters Estimation

In LDA parameters to improve our topics generated from the input text and the topic is used as a feature based on which we cluster our data according to their semantics. The parameter selection process for the LDA model can be provided by Gibbs sampling under fixed hyper-parameter. Table 4.1 shows the parameters used for the LDA model:

Variables	Description
A	Dirichlet prior on the per document topic distributions
B	Dirichlet prior on the per topic word distributions
K	Number of topics to be generated
minTokenCount	Minimum token count (if a word appears less than minTokencount ignore
numTokens	Number of tokens of dataset
numDocs	Number of documents

Table 4. 1 LDA Adjustable Parameters

a. Parameter α

A document can have multiple topics, so we need the dirichlet distribution which models this relation. The α controls the shape of document–topic distribution; where a large α leads to documents containing many topics. In contrast, small values of α and result in sparse distributions: documents containing a small number of topics.

b. Parameter β

Words can belong to multiple topics, so we need β Dirichlet to model this. The β parameter describes the concentration for topics' distributions over terms. The low β values produce topics that are well described just by few words, while high values will create topics composed by a mixture of most of the words.

Generally, the typical standard value of α used in practice is 50/K where K is number of topics (50 words per topic on the uniform distribution) and the value of β is 0.1 or 200/W, where W is number of words in vocabulary [50]. Anyway, this can be varied based on the size of corpus.

The algorithm of our proposed model use α=2/K and β =0.1 the standard default value to learn the topics existed in the collection using proposed LDA algorithm.

α controls the shape of the document–topic distribution Θ for each document d in D, β controls the shape of the topic–word distribution ϕ respectively. A large α leads to documents containing many topics, and a large η leads to topics with many words.

c. Parameter *K*

The number of topic K is the major parameter for the LDA, is one of the main targets of this research. Indeed, the more accurate the number of topics given to the model; better clustering results given by the LDA. In this study we will propose a new approach to identify the most suitable K values, taking into account the level of semantic similarity of the topics inferred by the LDA algorithm. As reported in [23], the research has not yielded to easy way to choose proper values for K beyond a major iterative approach. The followed approach is still iterative, as all the approaches known so far in literature: this means that in the framework, several LDA models with different values for the K parameter will be created. Though, the goal of the research is to find the optimal K values evaluating not only probabilistic quality metrics.

In LDA, each document viewed as a mixture of various topics where each document is considered to have a set of topics that are assigned based on automatic detection of term co-occurrence. A lexical word may occur in several topics with a different probability, however, with a different typical set of neighboring words in each topic [12] [13] [15] [35] [69].

The core idea of the proposed model is to discover the inner relations among all associated terms within each topic. The model tries to construct new topical datasets and generate new representations, in order to extract and highlight the semantics of topic representations. This shows the meaning of extracted topics.

The LDA model learns the unobserved groups from similar groups of data, where the words are assigned to particular topics from documents. A representation is usually defined as a set of related terms or words. In LDA the idea of the topics representations starts from the knowledge of frequent pattern mining [19]. It plays an essential role in many data mining tasks directed toward finding interesting patterns in datasets [70] [11].

We believe that the related terms representations are more meaningful and more accurately represent topics than word-based representations. After generating the patterns, we have to find the meaningful term that represent the category. In order to discover semantically meaningful patterns to represent topics and documents, two steps are proposed. First, construct a new transactional dataset from the LDA results of the document collection D. Second, generate pattern-based representations from the transactional dataset to represent the meaning of topics.

4.3.2. Implementation of LDA Algorithm

The Java programming language was used to read the data from the files and to remove special characters and stop-words. The LDA takes document as input and based on the assumption that a document contains a mixture number of K underlying different topics and decomposed into two low rank matrices (Document – topic probability matrix and topic - word probability matrix). The document is generated by these topics with different probabilities.

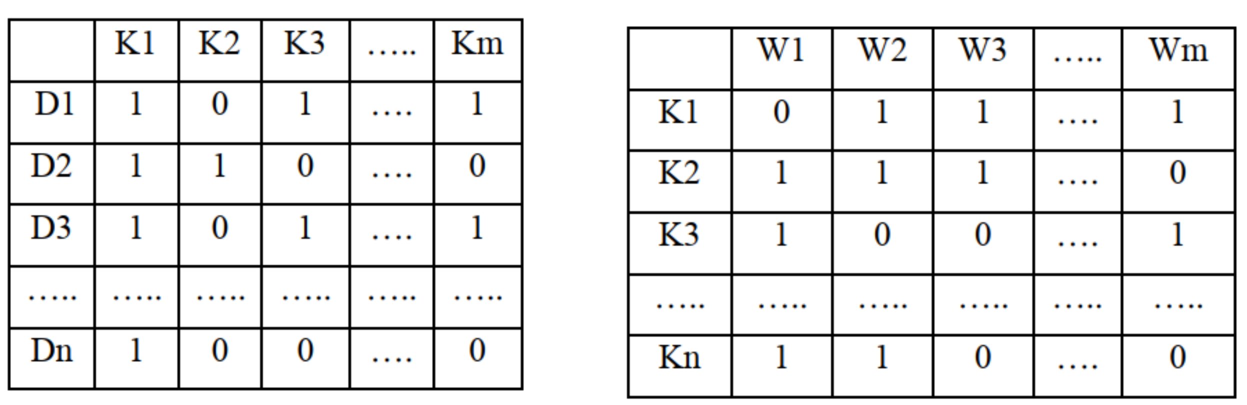
LDA is able to find out the topics and their relative proportions, which are distributed as a Latent Dirichlet random variable. Those topics then generate words based on their probability distribution [13]. It forms models in unsupervised mode, i.e., does not need labeled training data. The algorithm's performance can be managed though assumptions on the word and topic distributions. In this work using CBOW we represent our corpus as a document-term matrix in vector space.

The following matrix shows a corpus of N documents, D1, D2, D3 … Dn and vocabulary size of M words W1, W2 ... Wm. The value of i,j cell gives the frequency count of word Wj in Document Di.

	w1	w2	w3	…..	wn
D1	0	2	1	….	3
D2	1	4	0	….	0
D3	0	2	3	….	1
…..	…..	…..	…..	…..	…..
Dn	1	1	3	….	0

Table 4.2 Document Term Matrix

Document-Term Matrix converted into two lower dimensional matrices – M1 and M2.
M1 is a document-topics matrix and M2 is a topic – terms matrix with dimensions (N, K) and (K, M) respectively, where N is the number of documents, K is the number of topics and M is the vocabulary size.



	K1	K2	K3	…..	Km
D1	1	0	1	….	1
D2	1	1	0	….	0
D3	1	0	1	….	1
…..	…..	…..	…..	…..	…..
Dn	1	0	0	….	0

	W1	W2	W3	…..	Wm
K1	0	1	1	….	1
K2	1	1	1	….	0
K3	1	0	0	….	1
…..	…..	…..	…..	…..	…..
Kn	1	1	0	….	0

Table 4. 3 Document Topic Matrix and Topic word Distribution

The two matrices provide topic word and document topic distributions, and the distribution needs to be improved using sampling techniques in LDA by iterating through each word ‘w’ for each document ‘d’ and tries to adjust the current topic – word assignment with a new assignment. A new topic “k” is assigned to word ‘w’ with a probability P which is a product of two probabilities, p1 and p2 calculated for each topic.

P1 – p (topic t / document d) = the proportion of words in document d that are currently assigned to topic t.

P2 – p (word w / topic t) = the proportion of assignments to topic t over all documents that come from this word w.

The current topic – word assignment is updated with a new topic with the probability, product of p1 and p2. In this step, the model assumes that all the existing word – topic assignments except the current word are correct. Essentially this is the probability that topic t generated word w, so it makes sense to adjust the current word's topic with new probability. After a number of iterations, a steady state is achieved where the document topic and topic term distributions are fairly good. We used probabilistic LDA generative topic model approach that generates mixtures of topics based on word frequency from sets of documents. Topics Y and documents X Jointly P (Y, X) where the topics Y with highest joint probability given X has the highest conditional probability. That means when we put with equation:

$$p(Y|X = \max P(Y,X) ---------------------------- (4.1)$$

In generative process, assume we have X words and Y topics in a given document, as shown below:

X1=	Barattoonni	mana	barumsa	fayyaa	ispoortii	atleetiksii	kubbaa	miilaa	qilleensaa	dhibee	fayyaa	midhamaa
Y1=	0	0	0	1	2	2	2	2	0	3	1	1

	Topic 0	Topic 1	Topic 2	Topic 3
Document i	4	3	4	1

Table 4. 4 Word Topic Counters

Now, we can get matrix that shows the overall counter of words versus topics in the whole collection as shown below in table 4.5.

	Topic 0	Topic 1	Topic 2	Topic 3
Barattoonni	3	32	12	2
Mana	7	23	1	0
Barumsaa	10	5	4	1
Fayyaa	13	8	2	5
Ispoortii	6	11	16	2
Atleetiksii	1	0	7	23
Kubbaa	0	2	8	6
Miilaa	0	0	5	4
Qilleensaa	3	1	1	3
Dhibee	2	7	9	2
Miidhame	4	3	2	4

Table 4. 5 Sample Overall Word Topic distribution

From this we can decide the topic distribution for each document by using the following equation 4.2:

$$p(y|Yi) = \frac{count(y,Yi)--}{count(Yi)} \quad \text{---(4.2)}$$

Example: p(y=0|Y1) =4/12=0.33

And we can decide word distribution for each topic by applying equation 4.3 below:

$$p(x|y) = \frac{count(x,y)}{count(y)}) \quad \text{--(4.3)}$$

P(x=fayyaa|y=1) =2/3= 0.67

The algorithm calculates two probability values: **P (word | topics)** and **P (topics | documents) from the cluster assignments**. These values are calculated based on an initial random assignment, after which they are repeated for each word in each document, to decide their topic assignment. In an iterative procedure, these probabilities are calculated multiple times, until the convergence of the algorithm. To clearly understand this idea in Sampling in Topic Models we Sample one y $_{i,j}$ at a time:

X1=	Barattoonni	mana	barumsa	fayyaa	ispoortii	atleetiksii	kubbaa	miilaa	qilleensaa	dhibee	fayyaa	midhamaa
Y1=	0	0	0	1	2	2	2	2	0	3	1	1

{0,1,2,3} = {4/12, 3/12, 4/12, 1/12}

For instance, if we do not know the word ‘kubbaa’, (ball) to which topic it corresponds we multiply topic probabilities by word given topic probabilities.

So, to show how it works, we make it easy and step through a concrete example. Sample documents and topics are shown in the above. Let’s say our hyper parameters are alpha = 0.5, beta = 0.01, topics = 2, and iterations = 1.

$tn = \frac{current\ z+\beta}{current\ z+1+K*\beta} * \frac{current\ z+\alpha}{current\ z+1+N*\alpha}$--- (4.4)

Where K is the number of topics and V is vocabulary.

$p(wi = Topic\ i|\theta) = \frac{t\ i}{\sum_{i=0}^{T-1} ti}$ --(4.5)

Recall that when we iterate through each word in each document, we don’t assign its current topic assignment and reassign the word to a new topic. The topic we reassign the word to is based on the probabilities below.

P(document "likes" the topic)×P(topic "likes" the word w′)

$P(z_i = j \mid z_{-i}, w_i, d_i) = \frac{(C_{di,j}^{DT}+\alpha)}{\sum_{t=1}^{T} C_{di,t}^{DT}+T\alpha} * \frac{(C_{wi,j}^{WT}+\beta)}{\sum_{w=1}^{W} C_{di,j}^{DT}+w\beta}$ --------------------------------------(4.6)

Where,

Starting from the left side of the equal sign:

- P(zi=j): The probability that token i is assigned to topic j.
- z−i: Represents topic assignments of all other tokens.
- wi: Word (index) of the ith token.
- di: Document containing the ith token.

For the right side of the equal sign:

- C(WT): Word-topic matrix.
- $\sum_{w=1}^{w} \frac{CWT}{wj}$: Total number of tokens (words) in each topic in document i.
- C(DT): Document-topic matrix.
- $\sum_{t=1}^{T} \frac{CDT}{di,t}$: Total number of words in document i assigned to topic t.
- β: Parameter that sets the topic distribution for the words, the higher the more spread out the words will be across the specified number of topics (K).
- α: Parameter that sets the topic distribution for the documents, the higher the more spread out the documents will be across the specified number of topics (K) or - smoothing parameter (hyper parameter - make sure probability is never 0).
- W: Total number of words in the set of documents.
- T: Number of topics, equivalent of the K we defined earlier.

Or

$$\frac{ni,k+\alpha}{Ni-1+K\alpha} * \frac{mw\prime,k+\beta}{\sum_{w\epsilon\beta} mw,k+V\beta} \text{---(4.7)}$$

- ni,k - number of word assignments to topic k in document i
- α - smoothing parameter (hyper parameter - make sure probability is never 0)
- Ni - number of words in document i and −1- don't count the current word you're on
- K - total number of topics
- mw′,k - number of assignments, corpus wide, of word w′ to topic k
- β - smoothing parameter (hyper parameter - make sure probability is never 0)

- ∑w∈Vmw,k - sum over all words in vocabulary currently assigned to topic k
- V size of vocabulary i.e. number of distinct words corpus wide

4.3.2.1. Gibbs Sampling LDA Implementation

Implementing an LDA Gibbs sampler involves setting up the count variables, randomly initializing them, and then running a loop over the desired number of iterations where on each loop a topic is sampled for each word instance in the corpus. The only required count variables include nd, k, the number of words assigned to topic k in document d; and nk, w, the number of times word w is assigned to topic k. However, for simplicity we keep a running count of nk, the total number of times any word is assigned to topic k.

Variables such as a representation of the corpus, w, we need an array z which contain the current topic assignment for each of the N words in the corpus. We can do this by decrementing the counts associated with the current assignment because the topic assignments in LDA are exchangeable; i.e., the joint probability distribution is invariant to permutation. We then calculate the un-normalized probability of each topic assignment. This discrete distribution is then sampled from and the chosen topic is set in the z array and the counts are then incremented. The step-by-step procedure for LDA is depicted in Figure 4.3 for the full LDAA combination is a weighted sum whose weighting proportion coefficient sum to one. A word w is generated from a combination of topics z. In such a mixture model, the probability that a word w instantiates term t is:

$$p(w = t) = \sum_{k=1}^{K} p(w = t|z = k)p(z = k) \text{ --(4.8)}$$

Where $\sum_{k=1}^{K} p(z = k) = 1$; where each mixture component p(w=t|z=k) is a multinomial distribution over terms in our case the unigram model that corresponds to one of the latent topics z=k of the text corpus.

The algorithm of LDA assumes that new documents are created in the following way.

1. Determine the number of words in the corpus and K number of topics
2. Choose a topic mixture for the document over a fixed set of K topics i.e. 20 % topic 1, 30% topic 2 and 50 % topic 3

3. Generate the words in the document by:
    i. First pick a topic based on the document’s multinomial distribution
    ii. Next pick a word based on the topic’s multinomial distribution

```
Input: words w є document d

Output: Topic assignments z and counts nd,k; nk,w and
nk→Topic probability distribution for each word in a
document
--------------------------------------------------------
Initialize parameters
      K Number of Topics
      β Vocabulary matrix
      α Topic distribution
FUNCTION doc2word_Transformation ()
Begin
      Randomly initialize Z and increment counter
      Foreach iteration Do
            For i=0 to N-1 DO
                  Word=w[i]
                  Topic=z[i]
                  Nd,topic-=1;Nword,topic-=1; ntopic-=1
                  For k=0 to k-1 DO
                        $P(z=k|*)=(nd,k+\alpha k)(nk,w+\beta w)/(nk+\beta w)$
      End   //End inner for loop
                  Topic=sample from p(z|*)
                  Z[i]=topic
                  Nd,topic+=1;nword,topic+=1;ntopic+=1
            END   //End outer for loop
      END   //End foreach
      Return z, nd,k;nk,w;nk
End //End Begin
```

Figure 4.3 Pseudo code for LDA Gibbs Sampling

4.4. Finding optimal number of topics

In LDA, the number of topics K and hyper parameters are the initial inputs required for fitting the model. K is specified by the user. The optimal number of topics, K, must be enough to generate interpretable categories that have not been over-aggregated and enough to be usable at all [71]. If K is too low, the topics will be over-aggregated. Setting K too high cause user fatigue and overload human effort, making the results uninterpretable [16].

Selecting the number of topics is regarded as one of the most challenging issues in topic modeling, since there is really no agreed-upon formula exists to predict the optimal number of topics. Instead, the best choice of K largely depends on the task and the size of data set. The optimal number of topics can be derived by probability (perplexity) or human judgment. Perplexity is a probability-based estimate of how well a model will fit a sample. It measures the effectiveness of a given set of parameters (calculated using the training set data) on a set of unknown data [67]. Alternatively, expert human judgment is effective for guessing the number of topics [71].

In order to find the optimal number of topics to properly configure the LDA modeling process for a corpus we used perplexity. We trained the model using different values for K, based on range as minimum and maximum number of topics. We then, calculated perplexity for each K value and score the result. The K Value with the lowest perplexity was considered as the estimated optimal K.

4.5. Topic Labelling

In LDA based topic model, the algorithm generates a user specified number of topics that hold words relevant to topics, without knowing the exact name of topic. This has to be specified manually by the user to set topics name. Word-based multinomial distribution is used to represent topics based on the statistics without considering the semantic interpretability of the topics.

Normally words with high probability of a topic tend to suggest the meaning of the topic, but single words have the problems of polysemy and synonymy. We proposed a feature intended to capture the best topic word, to label a topic from the ranked terms semantically. The multinomial learned topics are generated by the LDA model and provided to humans as a first-order output based on the word's probability.

In this work we keep separate list file that can be assigned after the topics are extracted. We selected a topic word that is best label for a given topic, as a means of enhancing the topic interpretation. First, a set of candidate terms are generated for a specific topic based on a probabilistic ranking, which indicates how well a term can characterize a topic. We can choose a top-ranked term for labelling a topic by mapping the learned topics to the pre-defined label set to make topics more readily interpretable.

The K topics are probability distribution over terms with every term having finite probability in every topic. Terms wi are presented in descending order of P(wi|tj) for the topic tj. The top-N terms reasonable setting of number of topics, and usually provide sufficient information about the dataset to determine the subject area and interpretation of a topic, and distinguish one topic from another. Naturally, not all topics are equally coherent, however, and the lower the topic coherence, the more difficult the label selection task becomes.

To decide the topics category conditionally, we used word counts from the entire collection of articles. Conditional probability is defined as:

$$p(wi|wj) = \frac{p(wi,wj)}{p(wj)})\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..\ (4.9)$$

where i = j and p (wi, wj) is the probability of observing both wi and wj in the same sliding window, and P(wi) is the overall probability of word wi in the corpus.

```
Foreach I in topics
    Select top two words from topic i
        Foreach d in documents
            Compute the PMI score of the words in d
            Store the highest PMI with document ID
        Refer DocID Label from the golden label file
```

Figure 4. 5 Pseudocode for Topic Labelling

This pseudo code depicts how to label the extracted topic based on the word occurrence for human reading. The topic is labelled based on the semantic coherence of words co-occur with each other. We used labelled reference corpus based on the domain concepts whether the topic is referring the same or different domain. This needs some procedure; segmentation, probability calculation based on the referenced corpus and deriving mean average with respect to documents. For example, from 10 topics, to judge the label of Topic 1, we propose topic coherence measurement that selects the first two words of topics and compute the PMI of the co-occurrence of the two words in each document using equation 4.9. We store the document that has highest coherence value from all documents. Let document 4 has a high coherence score, we refer the label of that document and finally, we assign a label to Topic 1. If the label of document 4 is Barnoota (Education) we assign Topic as Education.

CHAPTER FIVE

EXPERIMENT AND EVALUATION

5.1. Overview

In this chapter we present an experiment and evaluation methods conducted to perform the topic modeling on the selected and prepared dataset to explore the theme of the articles and create new knowledge on the text. We have conducted experiments to evaluate the performance of the proposed topic modeling methods. The experiments have been carried out using LDA algorithm to identify the topics for the articles. We used the GUI developed in java to execute the LDA algorithm. The experiments have been carried out using LDA to identify the topics for various articles of Afaan Oromo.

5.2. Experimental Setup

To implement the prototype of the model of LDA, Java programming language (NetBeans 8.0.2) is used. Afaan Oromo text articles that are collected manually from two sources; BBC Afaan Oromo and FBC Afaan Oromo websites. We store our dataset in a folder as a text file and we browsed the folder through the user interface for processing. The model is tested on 16 news texts consists of four (4) categories [Barnoota (Education category), Fayyaa (Health Category), Haala_Qilleensaa (Weather Category) and Ispoortii (Sport Category)]. This individual category constitutes 4 documents. This corpus contained 1816 sentences. Pre-processing tasks have been carried out to the dataset. Each sentence has tokenized into words, words are normalized and stop words are removed for adding accuracy. We used 13,503 tokens and 13,278 unique words.

When the processed output is found the prototype stores the output into a new folder in the form of HTML files for the document clustering. And users see the output in the specified folder. For empirical analysis, LDA topic modeling algorithm is used to extract topics from a given collection of documents in unsupervised mode without need labeled training data. Document contains a mixture of N different topics as assumption and these topics are extracted with different proportions from this document using LDA as a Latent Dirichlet random variable [12] [15].

In this work, settings for the LDA models have been used: As a first step to determine the optimal K for an LDA model, we conducted validation with candidate numbers of topics to estimate a K latent topic in a corpus of documents. We made an experiment within topic range, in increment of one, setting a lower limit 2 and an upper limit of 20, as more than 20 topics is judged to be impractical to manually analyze. In second-round we conducted human coding task, human judges were asked to evaluate: (1) the quality of topic words, and (2) coherence between the topic words and the top related documents.

The possible range of K values, [Kmin, Kmax], has been set to [2,20]. Indeed, 2 is the lowest possible number of clusters to divide the corpus and 20 has been considered as upper bound. Since we used Gibbs sampler, the α value and the β value should be greater than or equal to 0 [9]. For this study, the value set for this parameter is α= 2/K, as proposed in the literature [13] [49], and the value set for β to 0.1, as proposed in the literature by Griffiths et al. [72] [73].

The Gibbs sampling [74] [75] learns the distributions, which requires a given pre-set of parameters such as α, which is the parameter of the Dirichlet prior on the per-document topic distributions and β, which is the parameter of the Dirichlet prior on the per-topic word distribution. The only required input provided by the user is the documents and fixed topic number N.

For the model, in addition to the mentioned parameters required; K, the number of topics and the maximum number of iterations is mandatory. As we observed from different literatures average iteration number within the model has set to 200 to converge. The iteration is a tradeoff between the time taken to complete sampling and the quality of the topic model and the best number depends on what we are looking for in the model. We used this value in this work. The output of LDA contains two parts. Part one is N topics with a list of words and count numbers for each word. Part two, for each document, indicates which topics it might belong to and the relative probability.

We developed the Java implementation of LDA to learn the topic and topic-word distributions and finally generate the topics for the categories. The algorithm’s performance can be managed though assumptions on the word and topic distributions. We provide a pre-set of parameters α, which is the parameter of the Dirichlet prior on the per-document topic distributions and β, which is the parameter of the Dirichlet prior on the per-topic word distribution, documents as input and fixed number of topics.

Generally, for the processing procedure using LDA with default parameters; α = 0.1, β = 0.01 and topic number N. The next step is input data file, preprocessing and Running the LDA algorithm to get the latent structure behind the text and print out the result is followed.

5.2.1. Implementation procedure

The model has been implemented using java programming language. The operation of Topic Modeling has been carried out by different steps. The whole procedure taken in the experiments is depicted in Figure 5.1. The first step is dataset preparation and preprocessing. Then in the step of topic generation, we utilize the sampling-based to generate LDA topic models. The number of topics K = 20, the number of iterations of Gibbs sampling is 200, the hyper parameters of LDA α = 2/K = 0.1; β = 0:01 in this experiment. In the last step we construct the topical representations, and to generate the frequent term-based topic representations using the proposed method.

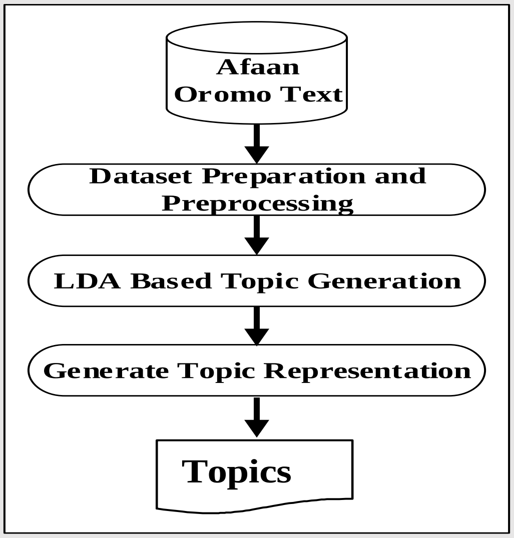

Afaan Oromo Text
Dataset Preparation and Preprocessing
LDA Based Topic Generation
Generate Topic Representation
Topics

Figure 5. 1 steps for generating topics

5.2.2. Evaluation Metrics

The evaluation of unsupervised nature of methods can be challenging task since there is no labeled ground truth data to compare with. The quality of Topic model tends to be evaluated to determine the performance. However, other ways of evaluation are also used in the literature [46]. Standard Information Retrieval metrics such as precision and recall cannot be directly applied to this approach because they necessarily depend on the notion of true classes for a corpus. One can evaluate clustering methods, but accuracy is not an applicable criterion [71].

No standard universal evaluation method is present for all the topic models. Hence, there is a need for finding the most suitable evaluation methods [16]. Topic Models can be evaluated by the Topic coherence, human judgment evaluation and Perplexity for each topic generated with top words [46]. In this study we selected the first two metrics for the generated topics and perplexity is used for selecting the suggested optimal number of topics from the corpus. For the evaluation the extracted topics we used Topic coherence and Topic coherence Human judgment.

Perplexity

Topic models can be evaluated based on a standard quantitative method using perplexity. It is described as how well the model predicts a sample word likelihood averaged over the documents using word occurrences in a topic.

Mimno et al. [46] defines perplexity as" Algebraically equivalent to the inverse of the geometric mean per-word likelihood." And calculated for a test corpus Dtest by calculating the natural exponent of the mean log-likelihood of the corpus words, as shown below.

$$Perplexity(D\ test) = \exp\left\{\sum_{d=1}^{M} \log \frac{p(wd)}{\sum_{d=1}^{M} Nd}\right\}\text{-------------------------------------}(5.1)$$

Or

$$\text{Perplexity}=2^{\text{LL}} \qquad \text{\#tokens---}(5.2)$$

The lower the perplexity score the better generalization ability. To evaluate our model, we used perplexity for estimating the number of topics *k* for model evaluation for our corpus since model assumed K parameter is required. Computing log perplexity returns negative value because of its logarithm of a number [76] [71].

In our work, we evaluate our model based on the interpretability quality criteria through human judgment and Topic Coherence measurement metrics. In human judgement, the inspection of each resulting topic is manually recorded to tell whether the topic has meaning or not. Quantitatively evaluation of topic model is still open problem [16] [45] metrics task that try to assess a model's quality in order to get the best performance since the need depends on wat we want to do with the model.

Topic Coherence

To evaluate the model automatically from the generated topics from the corpus using the latent variable, topic coherence is applied to the top N words from the topic. Since Topics are not guaranteed to be well interpretable, coherence measures distinguish between good and bad topics [45].

We applied the topic coherence to the extracted topics with top 10 words using equation 5.1. The pointwise mutual information (PMI) score as our topic coherence metric. For a topic z, given the top-ranked T words, namely, w1, w2, …, wT, the PMI score of the topic z can be calculated as follows:

$$PMI - Score(Z) = \sum_{1\leq i\leq j\leq T} \log\frac{p(wi,wj)}{p(wi)p(wj)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(5.3)}$$

Where p(wi,wj) represents the probability that words wi and wj co-occur and p(wi)=$\sum w\, p(wi, w)$. The average PMI score of word pairs in each topic is computed. From the average score of all the topics is computed.

Topic Coherence Human Judgement

Representative words are extracted for each topic based on their probability ranking from the corpus without any annotations and they are not guaranteed to be well interpretable. Coherence measures distinguish between good and bad topics by humans. The validity and interpretability of the generated topics with top N words is applied by human [77].

In our case, we limit top N words to 10. When human judges are used to interpret topic model outcomes, the most effective approach is careful reading of the underlying texts and of the model output by domain experts. The testing is done by manual to the obtained output theme from the model [77]. The quality of the topic-word distributions produced by our model is examined using precision, recall and F measure based on the manual analysis of the generated topics.

The performance measures used in the study are from the manual analysis:

Precision: Precision measures the ratio of relevant output instances to the total instances obtained from the output [8], as shown below.

$$P = \frac{no.of\ correctly\ identified\ topics}{all\ topics\ given\ by\ topic\ model} * 100 \tag{5.4}$$

Recall: Recall is the ratio of relevant output instances to the total instances in the data set [8], as shown below.

$$R = \frac{no.of\ correct\ topics\ obtained\ by\ the\ human}{total\ no.of\ topics\ that\ have\ been\ used\ for\ testing} * 100 \tag{5.5}$$

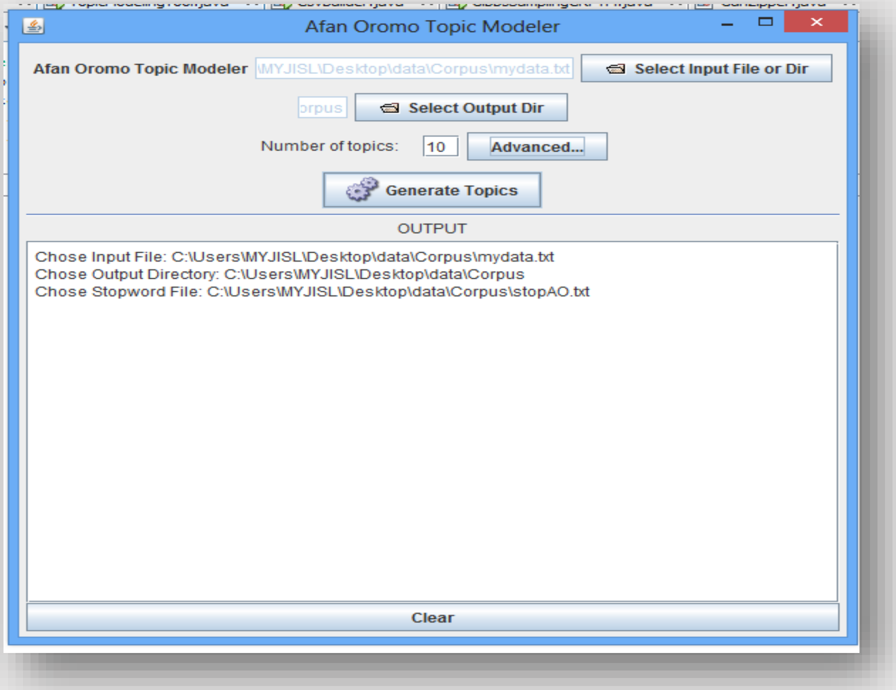
F-measure: The parameter F- measure gives the overall performance of the topic model [8], as shown below.

$$F = \frac{2PR}{R+P} * 100 \tag{5.6}$$

5.3. Experimental Results

This section discusses the quality of the model presented. To get a senseful generated topic there are prerequisite to be followed. The first one is, the data quality to be fed. This shows the strengths of the models and identify the roots of their performance. The results we obtained in our study is analyzed as follows to show what achieved and how much the problems stated are solved by our model. Topic model take a set of documents as an input. Then, the desired parameters; assumed number of topics (k) that are to be extracted from the corpus, iteration, β and α. Then the model learns the k-topics from the corpus. The number of topics should depend to some degree on the size of the collection

The model in this work uses small Afaan Oromo sample of 16 articles after 200 iterations of Gibbs sampling, with K = 10 topics that produce fine-grained results, and Dirichlet hyper-parameters β = 0.01 and α = 2 / K. When the fitted model is given, learning of topics from the dataset is followed. The figure 5.2 is the prototype developed for the model.



Afan Oromo Topic Modeler \MYJISL\Desktop\data\Corpus\mydata.txt

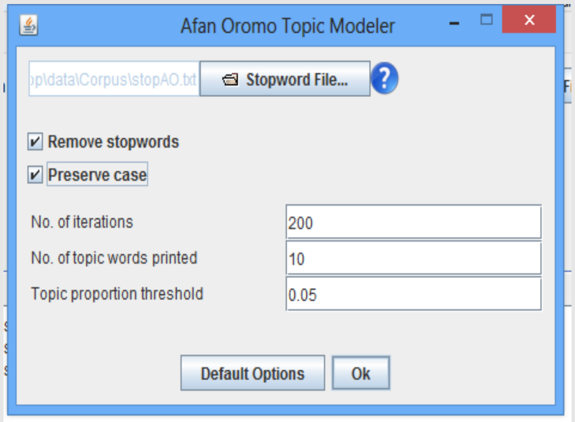
ɔrpus

Number of topics: 10

```
Chose Input File: C:\Users\MYJISL\Desktop\data\Corpus\mydata.txt
Chose Output Directory: C:\Users\MYJISL\Desktop\data\Corpus
Chose Stopword File: C:\Users\MYJISL\Desktop\data\Corpus\stopAO.txt
```

Figure 5. 2 Afaan Oromo Topic Modeler Main Interface

This is the main form where our options are select input file, select output file and number of topics. We load our dataset as input file by clicking the button "***Select Input File Button***" and select a folder that stores our output through "***Select Output Button***". After loading the corpus, folder that stores our result and suggested number of topics pre-processing step followed. To estimate the suggested optimal number of K topics we used perplexity. See Section 4.4 and section 5.2.2. After that, we have to make preprocessing of our corpus. Tokenization is the first task, then punctuation marks removed and all tokens changed to lowercase. Then, stopwords or common words are removed. We stored list of Afaan Oromo stopwords in separate file and we load the file that contains list of stopwords through clicking the "**Advanced" Button**. When we click that button, the form depicted in figure 5.3 is displayed.



p\data\Corpus\stopAO.txt

☑ Remove stopwords

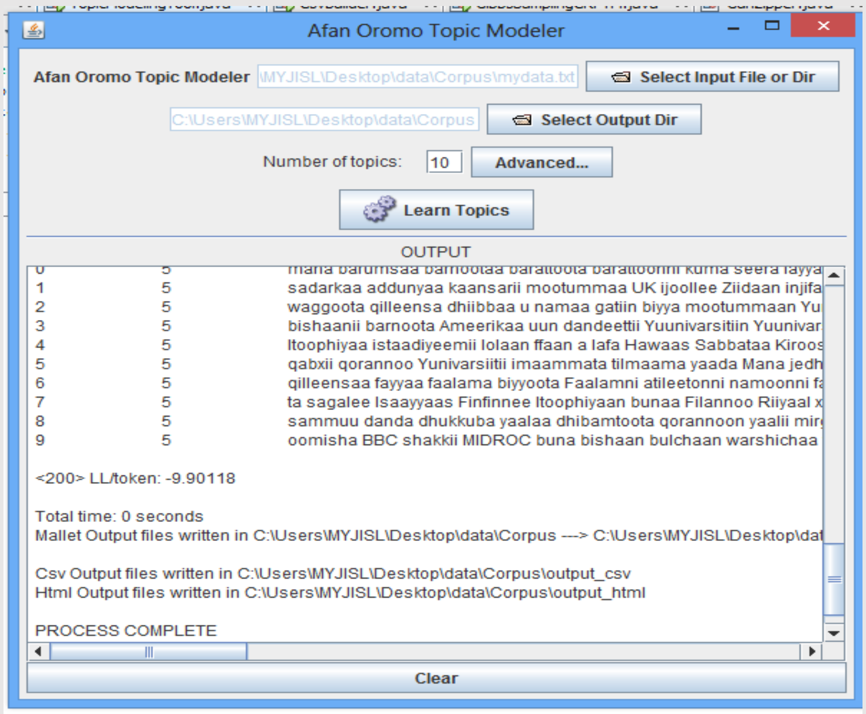
☑ Preserve case

No. of iterations	200
No. of topic words printed	10
Topic proportion threshold	0.05

Figure 5. 3 Advanced Options

Now we load the stop words file by clicking "***Stopwords File***" button. After selecting the directory path, we check the checkbox "***Remove Stopwords***" to remove the stopwords from the corpus. And we change the tokens to lowercase through selecting the "***Preserve Case***" button. After doing this, we provided the number of iterations used to evaluate the samples through maximum iterations, number of top words to be printed per each topic. This word become sorted according to their probability distribution and lastly, we provided the topic proportion threshold. This discards words that have equal or less than this threshold. This is necessary to consider the quality of the topic. After feeding all this information, ***"OK"*** button take us to the main form as follows and we can learn the topics. In this model, the maximum number of topics set was 10. The number of topics increase as the size of the dataset become larger.

When selecting the number of iterations to run, the maximum number of iterations possible was the goal. The 200 iterations to produce a maximum number of 10 topics is required.



Afan Oromo Topic Modeler \MYJISL\Desktop\data\Corpus\mydata.txt

C:\Users\MYJISL\Desktop\data\Corpus

Number of topics: 10

```
0               5               mana barumsaa barnootaa barattoota barattoonni kuma seera fayya
1               5               sadarkaa addunyaa kaansarii mootummaa UK ijoollee Ziidaan injifa
2               5               waggoota qilleensa dhiibbaa u namaa gatiin biyya mootummaan Yu
3               5               bishaanii barnoota Ameerikaa uun dandeettii Yuunivarsitiin Yuunivar
4               5               Itoophiyaa istaadiyeemii lolaan ffaan a lafa Hawaas Sabbataa Kiroos
5               5               qabxii qorannoo Yunivarsiitii imaammata tilmaama yaada Mana jedh
6               5               qilleensaa fayyaa faalama biyyoota Faalamni atileetonni namoonni fa
7               5               ta sagalee Isaayyaas Finfinnee Itoophiyaan bunaa Filannoo Riiyaal x
8               5               sammuu danda dhukkuba yaalaa dhibamtoota qorannoon yaalii mir
9               5               oomisha BBC shakkii MIDROC buna bishaan bulchaan warshichaa

<200> LL/token: -9.90118

Total time: 0 seconds
Mallet Output files written in C:\Users\MYJISL\Desktop\data\Corpus ---> C:\Users\MYJISL\Desktop\dat

Csv Output files written in C:\Users\MYJISL\Desktop\data\Corpus\output_csv
Html Output files written in C:\Users\MYJISL\Desktop\data\Corpus\output_html

PROCESS COMPLETE
```

Figure 5. 4 Sample Output

Figure 5.4 shows the extracted 10 topics. After, we evaluate the performance of final output result by the metrics to measure the performance of the model. To evaluate the model fitting parameters should be provided. From the parameters, estimating the suggested number of K topic is one. The suggested number of K topics is done through perplexity. The set the following parameters for the model; α = 2/K, β = 0.01 and K=10. We gained the assumed K=10 from input topic number N = 2 to 20. We registered all log likelihood that obtained from the perplexity for the minimum and maximum K interval. See section 5.2.2. In our case, 2 is minimum K and 20 is maximum K. The lower perplexity value is the better assumed number of K topics. We obtained the lower result at K=10 it is the better topics have enough words to tell information about the documents. The Log likelihood was calculated at each iteration of 5 intervals. So, in the experiment, we set K = 10 and tuning parameter α and β by setting α = 2/K while β = 0.01. α is a dirichlet prior on the per-document topic distributions and β is a dirichlet prior on the per-topic word distributions. For example, LDA (10, 0.1, 0.01) means that we applied LDA with 10 topics, α = 0.1 and β = 0.01.

After providing the parameters the model produce output. For instance, from Table 5.1, Global topic 1 is about Barnoota (Education). Topic 2 is about Fayyaa (Health). Topic 3 is about Faalama Qilleensa (pollution). Topic 4 is about ispoortii (Sport).

Topic 1	**Topic 2**	**Topic 3**	**Topic 4**	**Topic 5**	**Topic 6**	**Topic 7**
Mana	Sammuu	Faalama	Itoophiyaa	Dhiibbaa	Atleetiksii	Oomisha
Barnootaa	Dhukkuba	Qilleensaa	istaadiyeemii	Lolaan	Dheeraa	Faalmni
Barattoota	Dhibee	Fayyaa	Fiigicha	Yaalaa	Isaayyaas	Bishaanii
Barattoonni	Kaansarii	sadarkaa	dorgommii	Lafa	Ijoollee	Shakkii
Barumsaa	yaalii	Addunyaa	Itoophiyaan	Hawaas	Riiyaal	MIDROC
Kuma	Qabxii	Qorannoo	xumuraaf	Nama	Tapha	Bishaan
Dandeettii	dhibamtoota	Biyyoota	Kirooshiyaan	Buna	Jiraattonni	Barattoonni
Seera	Qorannoon	Qilleensa	shaakala	Bunaa	Filannoo	Bulchaan
Fayyadamuu	Nageenya	Faalamni	sahaatii	Sabbataa	Kilabii	Warshichaa
Barnoota	Imaammata	atileetonni	Lama	Miidhaa	Jimmaa	Warqii

Table 5. 1 Topic Words Distributions

Table 5.1 shows a list of 7 topics with top 10 terms. The words are arranged in the order of highest probability of words distribution to topics. For the manual analysis of the extracted topics, we confirmed that the LDA results are able to identify and reveal relevant information. For example, from our dataset the LDA with (10, 0.1, 0.01), topic 1: “mana”, “barnoota”, “barattoota”, “barumsaa” give us an idea of ‘Barnoota’ (Education). In topic 2, the outliers are “sammuu”, “dhukkuba”, “dhibee”, “kaansarii”, “yaalii” lead us to Fayyaa (Health). In topic 3, “faalama”, “qilleensaa”, “fayyaa”, “sadarkaa” indicate the Faalama Qilleensaa (Air pollution). Not every word in a topic can be justified as to carry the semantic structure of topics. Topics are appeared with their keywords as their proportions contributed in the topic. Table 5.2 below shows how the topics are contributed in each document.

DocId	Top-topics	Topic contributions to Documents												
1	1	0.71	6	0.089	4	0.065								
2	1	0.403	6	0.202	7	0.121	5	0.097	4	0.089	3	0.065		
3	2	0.471	1	0.178	3	0.138	6	0.103	5	0.069				
4	7	0.312	1	0.201	5	0.134	6	0.114	4	0.109	2	0.089		
5	2	0.396	3	0.173	5	0.115	1	0.108	6	0.101	4	0.101		
6	4	0.327	3	0.245	6	0.163	5	0.102	7	0.082	2	0.061		
7	2	0.252	6	0.215	5	0.178	1	0.131	4	0.093	7	0.075	3	0.056
8	7	0.354	2	0.177	1	0.125	6	0.115	3	0.104	4	0.083		
9	4	0.647	3	0.141	2	0.082	5	0.059						
10	4	0.718	3	0.068	6	0.06								
11	6	0.355	2	0.211	3	0.184	4	0.132	5	0.066				
12	6	0.654	3	0.102	4	0.079	7	0.071						
13	5	0.4	3	0.179	2	0.105	1	0.084	7	0.079	4	0.079	6	0.074
14	5	0.655	4	0.084	7	0.059	6	0.059	2	0.059				
15	3	0.531	2	0.166	5	0.094	4	0.067						
16	7	0.544	5	0.133	3	0.095	6	0.076	1	0.057	4	0.051		

Table 5. 2 Document Topics Distributions

To illustrate the above table 5.2, the first document, **doc 1** is 71% **topic 1**, 8.9% of **topic 6**, and 6.5% of **topic 4**. For example, document 1 could be an Education (Barnoota) article consisting of 71% and 8.9% sports (Ispoortii). The same is true in other documents. Once we produced an output, different variety of ways to further analyze and explore the data are there like a list of the weightings of each topic for each document. This allows for a more comprehensive understanding of the themes present in each text. For each document to what extent the topics contributed is specified. This distribution for all topics makes a links between the documents and this is helpful for the document exploration (see section 5.4).

The performance results we obtained in this study according to our metrics would be discussed. To estimate the suggested number of topics we registered the following output using Equation 5.1 as described in table 5.3 below.

Number of Latent Topics	Perplexity
2	-2.53
3	-4.91
4	-5.97
5	-6.5
6	-7.95
7	-9.72
8	-9.12
9	-9.74
10	-9.81
11	-5.77
12	-3.45
13	-2.33
14	-2.22
15	-3.44
16	-5.88
17	-4.22
18	-4.12
19	-6.67
20	-3.54

Table 5. 3 Perplexity of the model

From the table 5.3 the results show that the perplexity decreases as the number of topics increases from topic 2 to 10, at topic 7 to 10 it registered some related constant perplexity value with lowest one. So, we inferred some meaningful semantic structure of topics between 7 to 10. Thereby, 10 latent number of topics is selected since this is lowest obtained perplexity value as we can see from the table above. So instead of this, the optimal number of topics were selected by calculating the differences in perplexities between each number of topics from 7 to 10. The number of topics, in an observed pair with the least perplexity difference, was considered as the optimal number of topics. The differences between the topic perplexities for the selected range of topics has been presented in Table 5.4. From this table, it can be observed that the difference in the perplexities is less when the number of topics is from 7 to 10. Hence, we considered 10 to be the optimal number of topics generated by the model as we can see Figure 5.5.

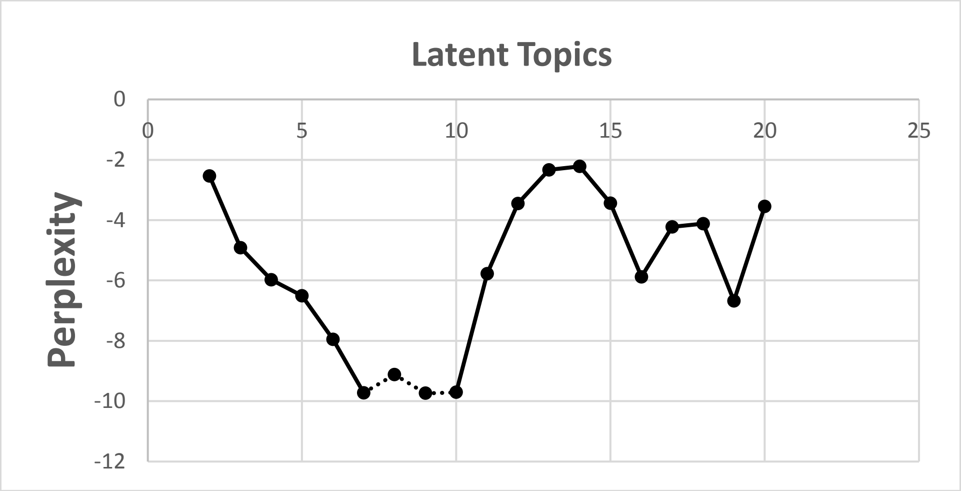


Figure 5. 5 The Perplexity of LDA evaluated on the Corpus

# of Topics	Average Perplexity
(2,3)	-3.72
(3,4)	-5.44
(4,5)	-6.235
(5,6)	-7.225
(6,7)	-8.835
(7,8)	-9.42
(8,9)	-9.43
(9,10)	-9.775

Table 5. 4 Perplexity difference among the topics

Table 5.4 shows the coherence score for the number of topics from 2 to 10. From topic 2 to topic 10 the coherence score decreases and increases from topic 11 to 20 so that we discard all perplexity value of this range. Now, choosing the number of topics would be with lowest perplexity from all average that is at 9 and 10.

We evaluate our Model considering:

- Is our model able to find coherent and meaningful topics?
- Is our model able to infer the topic distribution of a document when the document contains words which were previously unseen?

Our model demonstrated that embedding word tokens into the model can achieve coherent and interpretable topics based on the estimated topics. Topic coherence based on perplexity gives a good picture to take better decision on estimating number of topics to be extracted. In the selected range [2,20] we got lowest perplexity **-9.775**

We used PMI for the ranked ten words in each topic. Candidates for the best words were selected by choosing the top-1, top-2 and top-3 ranked words. We calculated the conditional likelihood of the words occurring with each other as Equation 5.3. For instance, Topic 1 **{mana barnootaa Barattootaa battoonni barumsaa}** → [**{mana, barnootaa}, {mana,barattoota},{mana,barumsaa},{barnoota,barattoota}**, etc] the confirmation measure of the topic coherence was calculated from. In this case, the conditional likelihood is obtained for the frequently co-occur words in the topic. We used two words conditional probability i.e. p(wi,wj)=number of documents that contain words wi and wj per total number of documents.

The nature of subjects influences the choice of words. Though the choice of top-m words for a topic affects its interpretability we just used top-10 words of a topic. That is the choice of m for different topics may be different and not constant. We found that the top six action recommendation under some topic as the action in the ground truth. PMI is computed for each pair of topic's words. We got average PMI of **0.525% or 52.5 %**. This only to quantify how much the extracted topics is interpretable.

We experiment with a simple ranking method based on the component scores. We label 10 topics based on the order words probability distribution to topics. To predict which words that annotators tend to select as most representative list of ten words, we formulate ranking a task, and treat the top-4 ranked terms as the best words, and compare that chosen words with the label set identified.

In the topics, words that are semantically coherent in our corpus have been captured by an LDA with referenced labelled corpus according to the domain category based on the defined kinds of keywords in the domain for human readability. We put a thresh hold of 0.2 (the words that contributed less than 20% of average mean in a topic is considered as difficult to label otherwise the co-occurred word can be referenced to label). See section 4.5.

For example, given a list of words mana, barumsaa, barnoota, barattoota, barataa, which is an Education topic, the first four words could be the most representative word. This is because it is natural to think about the Education after seeing the words mana, barumsaa and barnoota individually. A good candidate for best word could be the word that has high average conditional probability given each of the other words. Therefore, a good candidate, wi, might also correlate with high P(wj|wi) using equation 4.9 and Equation 5.3.

Based on this the task, we applied topic’s pair word selection that has highest probability and compute PMI, for best label for a given topic. In our experiment, we had 16 documents with 4 labels (ground truth). As our result shows, from 10 topics almost more than 5 of them perfectly matched with accurate labels.

The performance measure of the learnt topics is also done using human judgment evaluation metric. Human judgment is a way to evaluate the performance of topic models [46]. We selected nine participants (N=9) to rate the coherence of each topic and they were presented with top 10 term sets, each of which characterize a topic. We selected 3 masters students, 3 Afaan Oromo expert and 3 Teachers. Participants were asked to read the given topic first and then rate the two choice with their own standards. They asked to judge topic on two scale; Relevant (topic is coherent and interpretable if can easily assign predefined categories) and Irrelevant (words appear random and unrelated to each other). In addition to showing several examples of relevant and irrelevant topics, we instructed participant to decide whether the topic was to some extent coherent, meaningful, interpretable and easy to label. For our purposes, the usefulness of a topic can be thought of as whether one could categorize to one of the four topical areas (Education, Health, Sport and Weather condition) as particular to describe a topic.

We first judge our own judgment that means we set it as golden judgment and we compared the golden judgement to the judgement made by the participants. Each participant was asked to judge 10 topics and the average response for each topic was calculated to measure the model as recall, precision, F-measure. If a given topic can falls under one of categories (Education, Health, Sport and Weather condition), participants put relevant, otherwise irrelevant. Table 5.5 shows the rating of 10 topics and human judgment.

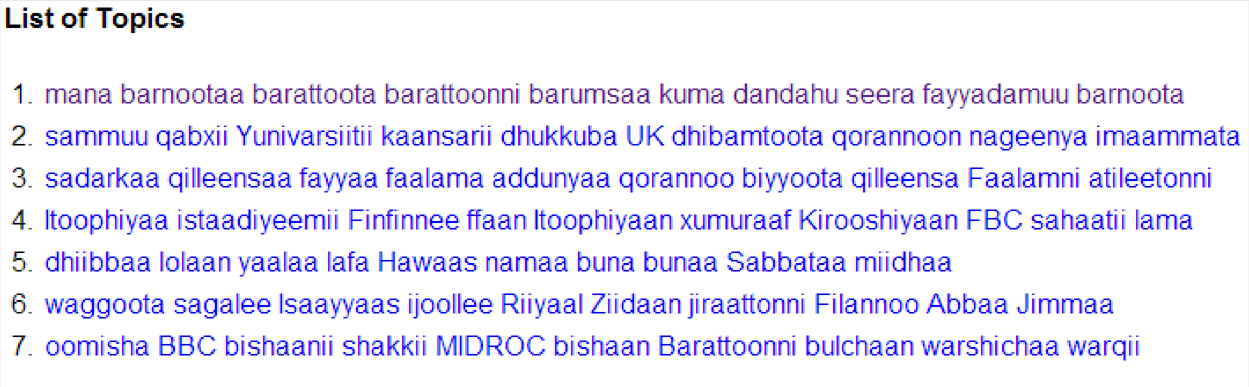
	u1	u2	u3	u4	u5	u6	u7	u8	u9	Average Recall	Average Precision
Topic1	R	R	I	R	I	R	R	R	R		
Topic2	R	R	R	R	I	R	R	R	R		
Topic3	I	R	R	R	I	I	R	R	R		
Topic4	R	R	R	R	I	I	R	R	R		
Topic5	R	R	R	R	R	R	I	R	I		
Topic6	R	I	R	R	R	R	R	R	R		
Topic7	R	R	R	R	R	R	R	R	R		
Topic8	I	R	R	R	R	R	R	R	R		
Topic9	I	R	R	R	R	R	R	R	R		
Topic10	R	R	R	R	R	R	R	R	I		
Recall	0.7	0.9	0.6	0.8	0.7	0.7	0.9	0.6	0.8	**0.744**	
Precision	0.57	0.6	0.7	0.75	0.43	0.57	0.67	0.5	0.62		**0.605**
F-Measure											**0.66**
R-Relevant **I-Irrelevant** **U-User**											

Table 5. 5 Overall human ratings

Table 5.5 show that the model has an **average** score of **74.4% Recall, 60.5% precision and 66% F-measure** based on the human judgment. It is calculated based on Equation 5.2, 5.3 and 5.4. The challenge here is to judge a topic as relevant, human understanding with similar topic was varied. Everyone held a different opinion on a topic. Even with the same people, we can have a different outcome in a different time frame. It depends on many factors, word relatedness, and personal educations. This method is limited with the size of the experiment and may have different results in different people. But the idea of the method and the way to evaluate may be useful for inspiring other researchers. In our experimentation results we discuss and evaluate the performance of our model using three methods; Perplexity for estimating the optimal number of topics, automatic topic coherence and Human judgment. The accuracy result would be different according to the number of topics that user would extract.

5.4. Topic Based Document Exploration

After training the model, it follows three steps: document clustering, calculating the similarity by comparing the topic proportion and assembling HTML files. We combine the clustering results and document similarity to assemble the document exploring system for our dataset. The system is based on HTML webpages and divided into two layers: topic page and document page. The first layer is the index page including K topics and its topic terms. This enables user to choose one of the topics and start exploring by clicking a certain topic, to look at the related documents of that topic. For each document in our data set we identify the topic index for which the probability is the largest, i.e., the main topic. Grouping by the topic index, counting, and sorting results in the counts of documents per topics. Inside the output folder there are two further folders: output.csv and output.html. opening output.html first, we find an html called **alltopics** and two further folders: Docs and Topics. Topics contains 10 html files, because this is specified in the before. Topic1.html contains clusters of terms.

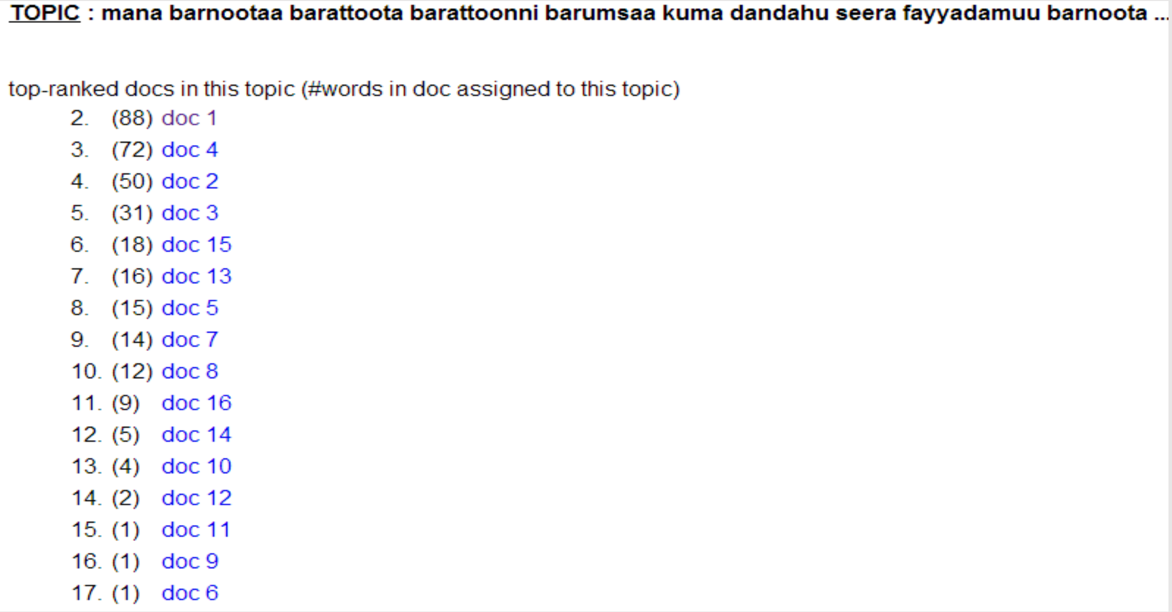


List of Topics

1. mana barnootaa barattoota barattoonni barumsaa kuma dandahu seera fayyadamuu barnoota
2. sammuu qabxii Yunivarsiitii kaansarii dhukkuba UK dhibamtoota qorannoon nageenya imaammata
3. sadarkaa qilleensaa fayyaa faalama addunyaa qorannoo biyyoota qilleensa Faalamni atileetonni
4. Itoophiyaa istaadiyeemii Finfinnee ffaan Itoophiyaan xumuraaf Kirooshiyaan FBC sahaatii lama
5. dhiibbaa lolaan yaalaa lafa Hawaas namaa buna bunaa Sabbataa miidhaa
6. waggoota sagalee Isaayyaas ijoollee Riiyaal Ziidaan jiraattonni Filannoo Abbaa Jimmaa
7. oomisha BBC bishaanii shakkii MIDROC bishaan Barattoonni bulchaan warshichaa warqii

Figure 5. 6 Topic List

By clicking certain topic, the related documents and it enables user to explore documents through topics and documents according to a topic. This means, it can automatically discover topics that documents contain.

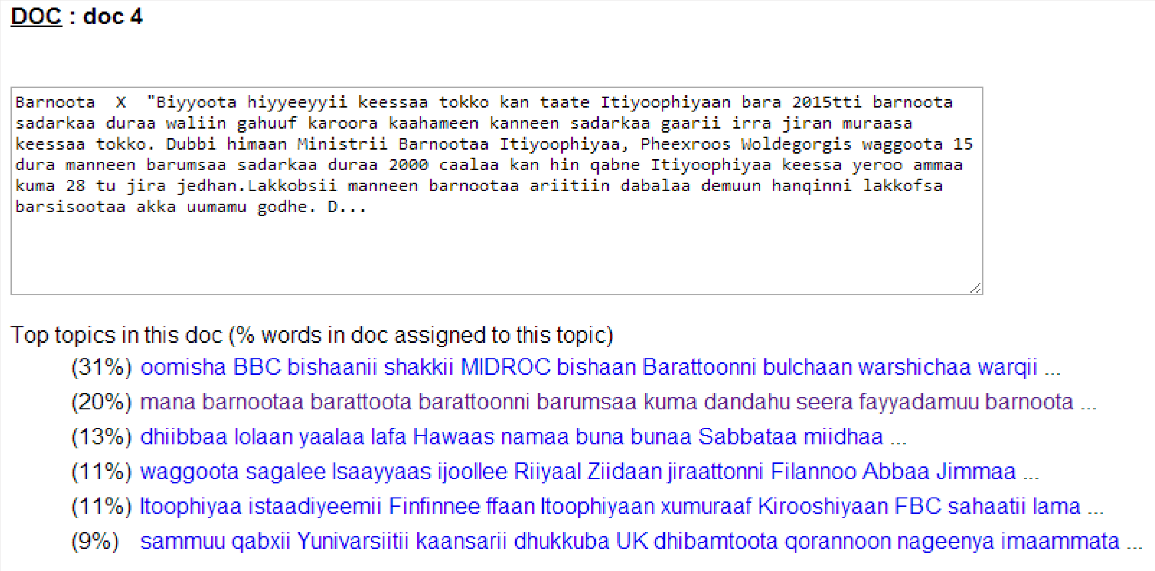


TOPIC : mana barnootaa barattoota barattoonni barumsaa kuma dandahu seera fayyadamuu barnoota ...

```
top-ranked docs in this topic (#words in doc assigned to this topic)
        2.   (88) doc 1
        3.   (72) doc 4
        4.   (50) doc 2
        5.   (31) doc 3
        6.   (18) doc 15
        7.   (16) doc 13
        8.   (15) doc 5
        9.   (14) doc 7
        10.  (12) doc 8
        11.  (9)  doc 16
        12.  (5)  doc 14
        13.  (4)  doc 10
        14.  (2)  doc 12
        15.  (1)  doc 11
        16.  (1)  doc 9
        17.  (1)  doc 6
```

Figure 5. 7 Top Ranked Documents Related to Topic 1

If we click document 4 from the above list, we get how a specific topic contributed in the document. It shows us for this specific document topic 7 holds 31%, topic 1 20%, topic 5 13%, topic 6 11% and so on.



DOC : doc 4

```
Barnoota  X  "Biyyoota hiyyeeyyii keessaa tokko kan taate Itiyoophiyaan bara 2015tti barnoota
sadarkaa duraa waliin gahuuf karoora kaahameen kanneen sadarkaa gaarii irra jiran muraasa
keessaa tokko. Dubbi himaan Ministrii Barnootaa Itiyoophiyaa, Pheexroos Woldegorgis waggoota 15
dura manneen barumsaa sadarkaa duraa 2000 caalaa kan hin qabne Itiyoophiyaa keessa yeroo ammaa
kuma 28 tu jira jedhan.Lakkobsii manneen barnootaa ariitiin dabalaa demuun hanqinni lakkofsa
barsisootaa akka uumamu godhe. D...
```

Top topics in this doc (% words in doc assigned to this topic)

(31%) oomisha BBC bishaanii shakkii MIDROC bishaan Barattoonni bulchaan warshichaa warqii ...

(20%) mana barnootaa barattoota barattoonni barumsaa kuma dandahu seera fayyadamuu barnoota ...

(13%) dhiibbaa lolaan yaalaa lafa Hawaas namaa buna bunaa Sabbataa miidhaa ...

(11%) waggoota sagalee Isaayyaas ijoollee Riiyaal Ziidaan jiraattonni Filannoo Abbaa Jimmaa ...

(11%) Itoophiyaa istaadiyeemii Finfinnee ffaan Itoophiyaan xumuraaf Kirooshiyaan FBC sahaatii lama ...

(9%) sammuu qabxii Yunivarsiitii kaansarii dhukkuba UK dhibamtoota qorannoon nageenya imaammata ...

Figure 5. 8 Topic Distribution of certain Document

5.5. Discussion

We noted that, topic modeling is a valuable method for analyzing large collection of documents by extracting latent topics that enable to understand, organize huge data. In this work we developed a Semantic based LDA topic modeling to analyze Afaan Oromo corpus. The experiments were performed on 16 docs, belonging to the four categories.

We removed abbreviations, punctuations, stop words and numeric data in the documents that affects the learnt topic interpretations. The model is incorporated with semantic approach to capture more meaningful Topics using word2vec approach. Estimated parameter settings were identified as prior to extract topics. The assumed number of K topics was set to 10, α=2/K and β=0.01.

The result of our experiment in this work contains two way of experimentation. The first one is evaluating the model at which K topics it scored lowest perplexity and experiments done on the extracted topics. The experiments done on judging the quality of topic is done in two ways; automatic topic coherence and human judgement.

According to the experiments results, we analyzed that our model performs the optimal number of perplexities with -9.775, for automatic topic coherence calculated by PMI is 0.525% and the overall human judgement F-Measure of 0.66%.

We cannot conclude these results are the optimal performance score because the performance is changed based on the parameters. This means that since we were estimating the parameters because there is no defined way to know the fixed parameters in topic modeling. Even the quality of our dataset can affect our performance score.

Human judgment has been used to evaluate topic modeling outputs, to judge the interpretability of topics even though, it is labor and time intensive. The model is trained with the estimated K topics that obtained from the perplexity measure for each iteration and human judges would manually assess the quality of topics. The approach is to compare the LDA-generated topic assignments with human-generated topic assignments. Once a model is trained, the model can automatically assign topics to new documents by holding the model parameters constant. Human judges can assign each of the documents to one of the topics, then the human generated topic list can be compared with topic-assignment results from the trained LDA model. The level of agreement indicates the final model's reliability and gauges the interpretability of LDA-generated topics.

CHAPTER SIX

CONCLUSION AND FUTURE WORK

6.1. Conclusion

The information overload has led to problem as the growth of digital information in a recent decade. Text analytics for such data presents many new challenges for research and development, and has also gained interest including analysis of large sets of text documents. Statistical language models are a solution to this problem. The statistical techniques are faster in implementation and they worked efficiently with larger documents.

So, in this thesis, statistical Topic model for Afaan Oromo language is proposed. The statistical techniques in semantic analyses of the textual units has touched in this work. This study presented topic model that combine semantic structure using LDA for extraction of quality topics in a collection. The investigation of how semantic approach can be applied in LDA to alleviate the problem of BOW has dealt in this work. The LDA algorithm developed in this work is based on word embeddings to handle semantic features. We used the co-occurrence model with a 2-word window context to handle the semantic.

In order to evaluate the performance of the proposed Topic modeler scheme, an extensive experiment is conducted. Before the performance of extracted topic is done, we found the optimal number of topics through perplexity estimation metric since the performance of the whole scheme is based on this. After the number of K topic is estimated we conducted the evaluation of topics as whether they are good or bad topics. Automatic topic coherence and human judgement were used for this purpose.

Moreover, based on our experimentation, the proposed model performed lowest perplexity for 10 number of topics from range of minimum topic 2 and maximum topic 20. Based on the extracted topics we made evaluation as topic coherence calculated from PMI with 0.525 and human judgement as F-Measure of 0.66. Generally evaluating our model was very hard since unlabeled data is used.

6.2. Future Work

Based on the finding and concluding remarks of the study, the following research directions are recommended. In the current study there is small size dataset prepared for training and testing LDA model. By its nature LDA is good algorithm for analyzing a larger dataset to achieve better results in Topic modeling, there is a need to prepare large and standard dataset for evaluating our approach.

We also, plan to test other topic modeling algorithms in order to find the most appropriate one for Afaan Oromo corpus. The application of topics-based search is efficient than keywords based So, our further study would be applying topic-based search to information retrieval.

Conducting deeper grammatical analysis focusing on Part of Speech Tagging (POS) is left for future work since Nouns are more representative of the topics than frequencies of features to acquire good topics. Again, this work can be extended by applying knowledge-based approaches like using Ontology and rule-based approach. This work also can be a further study in the comparison of these approaches; Knowledge based versus statistical approach or Rule based versus Statistical approach.

Finally, applying topic modeling to other local language like Amharic, Tigrigna and others language is recommended as further study.

References

[1] J. Dean, "Big Data, Data Mining, and Machine Learning : Value Creation for Business Leaders and Practitioners," 2014.

[2] J. Grimmer, "We are all social scientists now: How big data, machine learning,and causal inference work together," *PS: Political Science & Politics,* pp. 80-85, 2015.

[3] S. K. H. GirishMaskeri, "Mining Business Topics in Source Code using Latent Dirichlet Allocation," *ACM,* 2008.

[4] I. Biro, "Document classification with latent dirichlet allocation,” Ph.D.dissertation," 2009.

[5] M. J. &. M. W. Zaki, "Data Mining and Analysis: Fundamental Concepts and Algorithms," *Cambridge University Press,* 2014.

[6] J. J. D. S. L. L. Jianguang Duy, "Topic Modeling with Document Relative Similarities," *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence,* 2015.

[7] S. Y. Y. R. Z. Z. Z. X. Yang, "Topic Modeling on Short Texts withCrowdsourcing".

[8] V. M. S. Suganya C, "Statistical topic Modeling for News Articles," *International Journal of Engineering Trends and Technology (IJETT),* vol. Volume 31, Number 5- January 2016.

[9] L. T. W. D. S. Y. a. W. Z. Lin Liu, "An overview of topic modeling and its current applications in bioinformatics," SpringerPlus, 2016.

[10] k. H. Bettina Grun, "Topicmodels: An R Package for Fitting Topic Model," *Journal of Statistical Software,* vol. Vol. 40 No. 13., 2011.

[11] D. Z. W. a. J. N. W. L. M. E. P. G. a. A. S. Clint P. George, "A Machine Learning based Topic Exploration and Categorization on Surveys," *International Conference on Machine Learning and Applications,* 2012.

[12] A. Y. N. M. I. J. David M. Blei, "Latent Dirichlet Allocation," *Journal of Machine Learning Research,* vol. 3, pp. 993-1022, 2003.

[13] C. L. D. D. Blei D, "Probabilistic topic models," *IEEE Signal Process Mag,* 2010.

[14] L. Q. Minglai Shao, "Text Similarity Computing Based on LDA Topic Model and Word Co-occurrence," in *2nd International Conference on Software Engineering, Knowledge Engineering and Information Engineering*, China, 2014.

[15] J. Dickman, "Topic Modeling Explained: LDA to Bayesian Inference," *Tech Talk,* 2014.

[16] J. B.-G. S. G. C. W. a. D. M. Jonathan Chang, "Reading Tea Leaves:How Humans Interpret Topic Models," *in Advances in neural information processing systems,* p. 288–296, 2009.

[17] L. H. F. a. O. B. OunasAsfari, "Ontological Topic Modeling to Extract Twitter users' Topics of Interest," *International Conference on Information Technology and Applications,* vol. 8, 2013.

[18] M. T. N. B. M. V. M. P. R. a. S. C. Nitin Sukhija, "Topic Modeling and Visualization for Big Data in Social Sciences," *Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress,* 2016.

[19] S. P. R.-Z. M. G. T. Steyvers M, "Probabilistic author-topic models for information discovery," *Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* p. 306–15, 2004.

[20] Dr.John, "Prototyping in Design Science Research," website, 2010.

[21] R. J. Wieringa, Design Science Methodology for Information Systems and Software Engineering, New York , 2014.

[22] K. A. Rubayyi Alghamdi, "A Survey of Topic Modeling in Text Mining," *International Journal of Advanced Computer Science and Applications,* Vols. Vol. 6, No. 1, 2015.

[23] S. C. D. S. T. L. T. K. F. G. W. a. H. Deerwester, "Indexing by latent semantic analysis," *JASIS,* vol. 6, pp. 381-407, 1990.

[24] J. C. a. W. Z. W. Zhao, "Best Practices in Building Topic Models withLDAfor Mining Regulatory Textual Documents," *CDER,* 9th November, 2015.

[25] M. Z.-Y. C. T.-S. L. S. Y. H. e. a. Mao X-L, "SSHLDA: a semi-supervised hierarchical topic model," *In Proceedings of the 2012 joint conference on empirical,* 2012.

[26] M. M. J. S. J. D. a. Y. M. Sarah ElShal, "Topic modeling of biomedical text," *IEEE International Conference on Bioinformatics and Biomedicine (BIBM),* 2016 .

[27] S. a. H. D. Purpura, "Automated classification of congressional legislation.," *In Proceedings of the 2006 international conference on Digital government research,* no. Digital Government Society of North America, pp. 219-225, 2006.

[28] D. P. S. a. W. J. Hillard, "Computer-assisted topic classification for mixed-methods social science research," *Journal of Information Technology and Politics,* vol. 4, pp. 31-46, 2008.

[29] M. Scharkow, "Thematic content analysis using supervised machine learning:An empirical evaluation using german online news.," *Quality & Quantity,* vol. 2, pp. 761-773, 2013.

[30] S. D. E. v. d. B. A. a. M. M. Verberne, "Automatic thematic Classification of election Manifestos," *Information Processing & Management,* vol. 4, pp. 554-567, 2014.

[31] T. L. a. R. M. S. David R. H. Miller, "A hidden markov model information retrieval system," *In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR ,* pp. 214-221, 1999.

[32] J. M. P. a. W. B. Croft., "A language modeling approach to information retrieval," *In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval,* pp. 275-281, 1998.

[33] C. H. T. H. R. P. a. V. S. Papadimitriou, "Latent semantic indexing: A probabilistic analysis," *In Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems,* pp. 159-168, 1998.

[34] A. W. P. M. G. W. A. N. A. K. B. Daniel Maier, "Applying LDA topic modeling in communication research: Toward a valid and reliable methodology, Communication Methods and Measures," *Communication Methods and Measures,* 2018.

[35] T. Hofmann, "Probabilistic latent semantic analysis.," *In Proceedings of Uncertainity in Artificial Intelligence,* pp. 289-296, 1999.

[36] D. M. B. a. J. D. McAuliffe, "Supervised topic models".

[37]]. A. K. a. H. Merouani, "Clustering with probabilistic topic models on arabic texts," *in Modeling Approaches and Algorithms for Advanced Computer Applications, ser. Studies in Computational Intelligence,A. Amine, A. M. Otmane, and L. Bellatreche, Eds. Springer International Publishing,* vol. vol. 488, p. pp. 65–74, 2013.

[38] W. Z. W. &. C. J. J. Zhao, "Topic modeling for cluster analysis of large biological and medical datasets," *BMC bioinformatics,* 2014.

[39] N. d. F. A. D. a. M. J. C. Andrieu, "An introduction to mcmc for machine learning," *Machine Learning,* pp. 5-43, 2003.

[40] C. D. M. a. S. D. Daniel Ramage, "Partially Labeled Topic Models for Interpretable Text Mining," 2011.

[41] F. V. Jensen, An introduction to Bayesian networks, London: UCL press, 1996.

[42] E. B. Andersen, "Sufficiency and exponential families for discrete sample spaces," *Journal of the american statistical association,* pp. 1248-1255, 1970.

[43] D. M. a. A. M. H. M. Wallach, "Rethinking LDA : Why Priors Matter," *in Advances in Neural Information Processing Systems,* vol. vol. 22, p. 1973–1981, 2009.

[44] M. W. P. S. a. Y. W. T. A. Asuncion, "On Smoothing and Inference for Topic Models," *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence,* p. 27–34, may 2012.

[45] S. S. a. M. Spruit, "Examining Topic Coherence Scores Using Latent Dirichlet Allocation," *in The 4th*

IEEE International Conference on Data Science and Advanced Analytics, p. 165–174, 2017.

[46] H. M. M. I. S. R. a. M. D. Wallach, "Evaluation Methods for Topic Models," *in ICML 09 Proceedings of the 26th Annual International Conference on Machine Learning,* p. 1105–1112, 2009.

[47] T. M. a. J. Lafferty, "Expectation-propagation for the generative aspect model," *In Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence,* 2002.

[48] Q. M. e. al, "Automatic labeling of multinomial topic models," *in Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* vol. 1, no. San Jose, California, USA, 2015.

[49] D. A. a. G. M. K. Christidis, "Exploring Customer Preferences with Probabilistic Topics Models.," 2014.

[50] G. T. a. S. M, "Finding Scientific Topics," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 101, p. 5228–5235, 2004.

[51] J. C. a. D. M. Blei, "Relational Topic Models for Document Networks".

[52] C. X.-s. W. W.-x. Shi Jian-hong, "Discovering topic from microblog based on hidden topics analysis," *Application Research of Computers,* vol. 31(3), pp. 700-704, 2014.

[53] S. L. Z. D.-k. Li Wen-bo, "Text classification based onLabled-LDA model," *Chinese Journal of Computers,* vol. 31(4), pp. 620-627, 2008.

[54] M. A. a. D. Berkani, "A Topic Identification Task for Modern Standard Arabic," *In Proceedings of The 10th Wseas InternationalConference On Computers,* pp. 1145-1149, 2006.

[55] M. W. Beck, "Average dissertation and thesis length," *https://github. com/fawda123/diss,* 2014.

[56] A. A. a. M. Abbod, "Enhanced Topic Identification Algorithm forArabic Corpora," *UKSIM-AMSS International Conference on Modellingand Simulation,* pp. 90-94, 2015.

[57] R. A. M. M. a. M. M. M. Zrigui, "Arabic text classificationframework based on latent dirichlet allocation," *Journal of Computing and Information Technology,* vol. 3, pp. 125-140, 2012.

[58] A. K. a. H. Merouani, "Clustering with Probabilistic Topic Models on Arabic Texts," *In Modeling Approaches and Algorithms for AdvancedComputer Applications,* pp. 65-74, 2013.

[59] E. D. Y. a. D. Ejigu, "Topic-based Amharic Text Summarization with Probabilisic Latent Semantic Analysis," *ACM,* 2012.

[60] I. Bedane, "The Origin of Afaan Oromo:Mother Language," 2015.

[61] G. Rabirra, "SeerLuga Afaan Oromoo," *Finfinnee Oromiyaa Press,* 2014.

[62] "Ethiopia‟s population now 76 million," *Report, Census http://ethiopolitics.com/news,* 2008.

[63] C. A. O. J.-1. Tullu Guya, "Gumii Qormaata Afaan Oromootiin Komishinii," *Aadaa fi Turizimii Oromiyaa, Finfinnee,* 2003.

[64] " “Caasluga Afaan Oromoo Jildi I”,Komishinii Aadaaf Turizmii Oromiyaa,Finfinnee, Ethiopia,," *Gumii Qormaata Afaan Oromoo,* pp. 105-220, 1995.

[65] M. M. Getachew Mamo Wegari, "Parts of Speech Tagging for Afaan Oromo," *International Journal of Advanced Computer Science and Applications,Special Issue on Artificial Intelligence.*

[66] E. A. Debela Tesfaye, "Designing a Rule Based Stemmer for Afaan Oromo Text," *International Journal of Computational Linguistics (IJCL),* vol. I, no. 2.

[67] D. J. a. J. H. Martin, Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, 1999.

[68] M. C.G., "A grammatical sketch of Written Oromo," vol. 3, 2001.

[69] S. M, "Probabilistic Topic Models. Latent Semantic Analysis:A Road to Meaning," 2007.

[70] C. C. P. S. a. M. S. David Newman, "Analyzing Entities and Topics in News Articles using Statistical Topic Models," 2011.

[71] I. M. R. S. a. D. M. Hanna M. Wallach, "Evaluation Methods for Topic Models," *in Proceedings of the 26th Annual International Conference on Machine Learning, New York, NY, USA,* p. 1105–1112, 2009.

[72] T. G. M. S. a. P. S. Michal Rosen-Zvi, "The Author-Topic Model for Authors and Documents," *Proceeding UAI '04 Proceedings of the 20th conference on Uncertainty in artificial intelligence,* pp. Pages 487-494, 2004 .

[73] P. F. a. W. N. Ralf Krestel, "Latent Dirichlet Allocation for Tag Recommendation," *ACM,* 2009.

[74] Z. G. T. S. J. a. L. K. S. M. I. Jordan, "An introduction to variational methods for graphical models," *In Proceedings of the NATO Advanced Study Institute on Learning in graphical models,* p. 105–161, 1998.

[75] K. M. M. B. L. C. M. C. M. H. a. R. D. R. Quinn, "How to analyze political attention with minimal assumptions and costs," *American Journal of Political Science,* vol. 1, pp. 209-228, 2010.

[76] J. Risch, Detecting Twitter topics using Latent Dirichlet Allocation, 2016.

[77] B.-W. O. I. L. a. G. S. C. Muhammad Omar, "LDA Topics: Representation and Evaluation," *Journal of Information Science,* 2015.