Hawassa University
ሀዋሳ ዩኒቨርሲቲ

EXPLORING A BETTER FEATURE EXTRACTION METHOD FOR AMHARIC HATE SPEECH DETECTION

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

BY: YESUF MOHAMED YIMAM

HAWASSA UNIVERSITY, HAWASSA, ETHIOPIA

FEBRUARY, 2021

EXPLORING A BETTER FEATURE EXTRACTION METHOD FOR AMHARIC HATE SPEECH DETECTION

BY: YESUF MOHAMED YIMAM

ADVISOR: WONDWOSSEN MULUGETA(PHD)

A THESIS SUBMITTED TO THE

DEPARTMENT OF COMPUTER SCIENCE,

HAWASSA INSTITUTE OF TECHNOLOGY, SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

HAWASSA, ETHIOPIA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN COMPUTER SCIENCE

FEBRUARY, 2021

SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

EXAMINERS' APPROVAL SHEET

We, the undersigned, members of the Board of Examiners of the final open defense by Yesuf Mohamed Yimam have read and evaluated his thesis entitled "Exploring a better feature extraction method for Amharic hate speech detection", and examined the candidate. This is, therefore, to certify that the thesis has been accepted in partial fulfillment of the requirements for the degree of Masters Science in Computer Science.

_____	_____	_____
Name of the Chairperson	Signature	Date
Wendwossen Mulugeta (PhD)	_____	_____
Name of Advisor	Signature	Date
_____	_____	_____
Name of Internal Examiner	Signature	Date
_____	_____	_____
Name of External examiner	Signature	Date
_____	_____	_____
SGS Approval	Signature	Date

ACKNOWLEDGMENT

First of all, I would like to thank the creator of all creators (the almighty God) for giving me strength, patience, perseverance, starting from the beginning to the ending of this thesis work; and also, I would like to thank my mother who tried and succeeded on showing me the light of knowledge, without seeing the light of the world.

I would like to thank my advisor Dr. Wondwossen Mulugeta for his constructive support and guidance starting from teaching course till the completion of my thesis, and for giving me important lessons not only for my profession but also for my lifetime.

Last but not list I would like to thank my teachers who have shown me the light of knowledge, starting from the time my fingers held a pen, and my tongue and aye were first introduced to letters.

Declaration

I hereby declare that this MSc thesis is my original work and has not been presented for a degree in any other university, and all sources of material used for this thesis have been duly acknowledged.

Name: Yesuf Mohamed

Signature: _____

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

Name: Wendwossen Mulugeta (Ph.D.)

Signature: _____

Date of Submission: _____

Table of Contents

LIST OF ACRONYMS

NLP	Natural language processing
IR	Information Retrieval
SVM	Support Vector Machine
DT	Decision Tree
RF	Random Forest
NB	Naïve Bayes
CNN	Convolutional Neural Network
TF-IDF	Term Frequency – Inverse Document Frequency
BOW	Bag of Words
NFEM	New Feature Extraction Method
POS	Part-of-speech
NN	Noun
PR	Pronoun
VB	Verb
ADV	Adverb
PREP	Preposition
ADJ	Adjective
CONJ	Conjunction
Glove	Global vectors for word representation
HS	Hate Speech
Non-HS	Non Hate Speech
HTML	HyperText Markup Language
RFDT	Rotation Forest and Decision Trees
BLR	Binary Logistic Regression
Word2Vec	Word to Vector
GRU	Gated Recurrent Unit
RNN	Recurrent Neural Network
LR	Logistic Regression
LSTM	Long Short Term Memory

LIST OF TABLES

List of figures

List of equations

List of algorithms

List of sample codes

Abstract

Hate speech is a speech that causes people to be attacked, discriminated, and hated because of their personal and collective identities. When hate speech grows, it will cause death and displacement of peoples from their homes and properties. Social media has the ability of widely spreading hate speech. To solve this problem, various researchers have studied many ways to detect social media hate speeches that are spreading in international and local languages. Because the problem is so serious, it needs to be carefully studied and better addressed in a variety of solutions.

The previous studies detect a speech as hate speech, based on the frequency (occurrence) of a word in a given dataset; this means it does not consider the role of each word in a given sentence. The main purpose of this study is to design a method that can generate hate speech features from a given text by identifying the role of a word in a given sentence, so that hate speech can easily be distinguished from other forms of speech in a better way. To do this, various researches related to this study have been studied and reviewed.

This study created a new feature extraction method for Amharic hate speech detection. The created model needs a training and testing dataset, so that posts and comments, which are posted on 25 popular Facebook pages, have been collected to build the dataset.

Whether a speech is hateful or not, should be determined by the law that prohibits hate speech. So that, using different filtration methods, datasets that contain religious, ethnic, and hate words are collected and given to law experts, to annotate it manually. The law experts labeled 2590 datasets into three classes; Religion-hate, Ethnic-hate, and Non-hate. After dataset preparation, a new feature extraction method, which can distinguish hate speech from other speech, is developed.

The new feature extraction method and other feature extraction methods that are used in other related studies are implemented and computed with three machine learning classification algorithms: SVM, NB, and RF. The result in different evaluation metrics shows that the new feature extraction method performed better in all combinations of classification algorithms. By using 80% of 2590 labeled datasets as a training set and the rest as a test set, 96.2% average accuracy is achieved using the combination of SVM with the new feature extraction method.

Keywords: *Amharic hate speech, annotation, new feature extraction method, machine learning*

CHAPTER ONE

1. Introduction

1.1. Background

Speech is a way of expressing our idea and feeling so that other people can understand what we observe inside [1]. We can explain to others what we think verbally, textually, or graphically [2]. Speeches are many kinds: As there are speeches that teach, praise, and encourage peoples, there are also speeches that insult, offend, and incite peoples [3]. One of the speeches that incite humans on other peoples for destruction is hate speech, which is the focusing point of this study.

Some researchers [4], [5] agree that hate speeches are the same as terrorism, yes indeed, if hate speech is spread intentionally and then if it is the cause of destruction of property and loss of life which has no different meaning from terrorism. In a terrorism because the action is intentional someone or group will public that they are responsible, but in the case of hate speech, because people sometimes spread it ignorantly, the responsible person may not be available. The result of both can be destruction of properties, loss of human life and other injuries.

Many countries adopt different rules and laws to control hate speech, proclamation No. 1185/2020 [2] in Ethiopia defines hate speech as “a discourse that promotes hatred, discrimination or violence based on race, gender, ethnicity, religion, and disability on a particular person, and a specific group or community.” Then it describes a kind of punishment for those who practice this act. A person, who participated in prohibited acts described above, can punish up to three years in prison or a fine of up to 10,000 Birr. Because of performing the prohibited activities, if an attack is made or attempted on individuals or groups, the person who committed the prohibited acts could be punished with up to five years' imprisonment. Although the law [2] says this, identifying everyone who practices this activity is exhausting and made it by humans is not effective.

Once Facebook, LinkedIn, Twitter, and other social media came in and made all individuals the media's owner, billions of social media users can release their thoughts every second [6], so that it is difficult to control hate speech by human beings. By

recognizing this to classify this huge, constant flow, various languages, of social media posts as hateful or not, various researchers have conducted studies for different languages.

1.2. Motivation

Hate speech has been around for a long time, now social media is in the hands of each of us, and allowing all of us to say what we want is what makes the problem so difficult and new. Among the damages caused by hate speech in history, the 6 million Jews and 800,000 Rwandan genocide [7] is always remembered. Although what happened is serious, we can hold accountable for the cause of the problem, like a radio station that is responsible for the genocide of the Rwandans. But when it comes to social media, anyone can hide or reveal their identity and express what they want to the masses, which makes it difficult to find an accountable person and difficult to control them. Especially in Ethiopia because unemployment is expanding, most people spend their time on social media, so that if somebody spreads hate speech it has a high probability to reach the majority. Therefore, there is no guarantee that such historical events will not happen.

Innovations created for good are often used by bad peoples for bad purposes. Misael is a power generator for good thinkers and also it is a killing machine for negative thinkers. The same is true for the internet, even if Tim Burners Lee opened it by saying "anyone can say anything about any topic", we are watching when some people are spreading hate speech on it. Therefore, if we are not using it by correcting and controlling the bad aspects, as its benefits the risk is higher.

1.3. Statement of the problem

Before social media like Facebook and Twitter came, media such as newspapers, radio, and television were often broadcast by experienced and professional journalists [8]. Even if they broadcast hate speech, they are easy to be controlled because of their small size. But when social media arrives, the criteria for using it are reading, writing, and knowing how to use it. Profession or skill in the article to be written, and looking at the topic in different directions, are not mandatory so that any social media user can write his or her opinion on any topic without any reviewer [6] [9]. As a result, speeches that cause a person or groups to be discriminated against based on their personal and collective identities are widely circulated [6]. As a result of misinformation and hate speech circulated in social media in

Ethiopia, many lives have been lost and millions of people have been displaced from their homes and properties [10].

Although there are over 83 languages in Ethiopia, Amharic is the official language that serves as the federal language [11]. Therefore, a lot of Medias write in Amharic, and most of their airspace is covered by Amharic programs [12]. Thus, if a hate speech is distributed in Amharic it has a probability of reaching many peoples.

While there have been two previous research articles [13] [14] to identify Amharic hate speech, the method used for specialization does not consider the nature and the rule of the language. In the previous two research [13] [14], any article was considered as hate speech because if a word in the sentence has a high frequency, not understanding the concept or context of the article, and they have not used a satisfactory way to use the concept of the text for the detection. On the other hand, at the time of the study, because there was no hate speech law, the data collected and the results generated by the model are not acceptable under current law.

In previous works [13] [14], the used feature extraction method like TF-IDF,BOW works by counting the frequency of words in a given document, and taking words with high frequency. Language has its syntax and semantics but these feature extraction methods do not consider that. For example, if there is a training speech "I hate Abebe"," Abebe must die", "I will kill Abebe" the name Abebe occurs 3 times so that the name Abebe taken as a feature, the next time when a speech "Abebe is swimming" given to the model, it wrongly detects it as a hate speech. Also if a speech "I don't hate abebe" or "kill die I abebe "is given, it will detect it as hate speech because it only cares about the occurrence of top words not the syntax or semantics.

When training and testing datasets are converted using feature extraction methods like; TF-IDF, BoW, and word2vec, a large number of digits are used to represent every word in a given text. This consumes a large memory size. For example, if we use 1,000 features per dataset, and if our number of datasets is 3,000, it needs a large memory size to store 3,000,000 numbers.

The following research questions are raised to address the above statement of the problem.

- ✓ How can hate speech detection system is improved?

- ✓ What feature makes hate speech differ from other speech?
- ✓ How to build a better hate speech feature extraction method?
- ✓ How to reduce memory consumption, when converting to a vector?

1.4. Objective

1.4.1. General Objective

The main objective of this thesis is to study and create a better, more efficient, and effective feature extraction method for Amharic hate speech detection.

1.4.2. Specific Objective

The following specific objectives are extracted from the general objective

- ➢ To review papers that studied hate speech detection.
- ➢ To measure the performance of the method used in previous works.
- ➢ To investigate different approaches to hate speech detection.
- ➢ To collect and build Amharic hate speech datasets from social media
- ➢ To develop a better feature extraction method to detect Amharic hate speech by filling the observed gap of the reviewed papers.
- ➢ To measure the performance of the developed method by testing the efficiency and the effectiveness using different performance evaluation metrics
- ➢ To contribute a better hate speech detection model.

1.5. Scope and limitation

1.5.1. Scope

The scope of the study is creating a new feature extraction method for detecting Amharic hate speeches that are scripted in Ge'ez and identifying specifically what type of hate it is, whether it is ethnic hate or religious hate.

The new feature extraction method needs identifying of part of speeches that most of the time hold hate words and targeted entities. By studying the nature of hate speech, identifying supportive parts of speeches is one part of this study.

1.5.2. Limitation

The study does not include fake news detection, which needs a separate study, and also the study will not detect offensive and hate speech that does not meet the requirement of the hate speech definition of the law [2]. The study doesn't work for Latin scripted Amharic hate speeches.

The used datasets are collected between March/2020 and November/2020. Previous researchers have used more than 2590 datasets, and this is because at that time there was no hate speech law in Ethiopia therefore they collected more datasets without a restriction. Because of the current political awakening, the most spreading hatred type is political hate speech, but to expand the political space, the law [2] does not consider political hate as hate speech, this reduces the probability of collecting a large dataset.

1.6. Application of result

Social media companies, government, and everyone in Ethiopia, who uses or doesn't use social media, will benefit directly or indirectly from the result of this study. How they can benefit from the study is explained as follows.

- **Social media companies**: if they apply the outcome of this study, they will protect their customers from hate speech. Their preference will increase as their customer can use their media freely without insults and verbal abuse. On the other hand, they will be free from accountability and punishment, for example, Germany passes controversial law to fine social media over hate speech "*sites that do not remove hate posts could face fines of up to 50m euro (£44.3m). The law gives the networks 24 hours to act after they have been told about law-breaking material"* [15]. Although this law is in Germany, it will eventually come to Ethiopia. So the result of this study will protect social media companies from accountability and unnecessary expenses.
- **Social media users:** the other beneficiary of this study is social media end users who can use it freely without affecting their rights and without verbal warfare or insults.
- **Peoples who do not use social media:** The indirect beneficiaries of this study are peoples who do not use social media. As a result of hate speech in social media, if

violence, destruction of property, displacement, and genocide happened, group or individual even if they are not social media users, they may be harassed because of their religion, ethnicity, language, or other personal or collective identities. Therefore, if this study is implemented and hate speech is banned on social media, it will prevent these individuals or groups from being attacked by something they don't know.

- **Government:** as a result of hate speech spread on social media, various riots and conflicts can arise. So, to stabilize the country the government can take different actions including the deployment of soldiers to the conflict areas. With this, a budget can be wasted, and even human life can be lost. Additionally, with the focus on stabilizing the country, it will not be able to achieve other goals. Therefore, the implementation of the result of this study will eliminate these problems.

1.7. Thesis organization

The *first chapter* contains the background of the study, motivation, the problem to be solved, the research question, the general and specific objectives, the boundary of the study, and finally, the beneficiary of this study is included.

The *second chapter* is about related literatures, in this chapter different literatures, which discuss about hate speech are reviewed. In addition, disciplines used in the study area and different detection methodologies are reviewed.

The *third chapter* is reviewing related works, in this section different studies for local and international language hate speech detection are reviewed. Researche relevant to this study are properly reviewed, strong sides and gaps are also identified.

The *fourth chapter* is about the methodology used to achieve the objective of the study. Different dataset collection, dataset preparation, pre-processing, feature extraction, classification, and performance evaluation methodologies are discussed here.

The *fifth chapter* is about the proposed model to detect Amharic hate speeches. In this chapter the architecture of the proposed model is discussed, and how and what the proposed model will do is explained.

In *chapter five*, the implementation, used codes, the data source where the dataset is collected, labeled class types, used algorithms for classification, performance evaluation metrics, dataset annotation result, and performance evaluation result are presented.

Finally, *chapter seven* presents the conclusion of the study, recommendations, works to be done for the feature and contribution of this study.

CHAPTER TWO

2. Literature Review

In this chapter, to facilitate a better understanding of hate speech, relevant literatures to hate speech, and hate speech detection techniques are presented. Various disciplines, which are used in previous studies, are also presented.

2.1. Hate speech definitions

As Yonas [14] said there is no single agreed-upon definition for the term “hate speech”. When different organizations and individuals interpret hate speech, it is from their point of view so that it will not be the same. Although they all have different definitions, basically they are similar, even if the word they used and the way they express it is different, the message it conveys to a person is the same.

2.2. Hate speech and social media in Ethiopia

With the advent of social media, people have been able to express their thoughts freely at any time [16]. The number of social media users in Ethiopia is increasing day to day. as a result, various service providers are distributing their advertisements on various social media; Various artists are sharing their innovations and works, activists and politicians are distributing their ideas and messages to their followers, and the government also started disseminating information, laws, messages, and statements to citizens on social media by creating different social media accounts [17]. As a result, Ethiopia is home to an estimated 105 million people [18], different types of ideas will be hosted. Some people are seen expressing their views by opposing, insulting, and denigrating others, and by uttering hate speeches.

Particularly since 2018, with the advent of new political changes in the country, many people have been sharing various ideas on social media. Different activists are seen distributing different types of content to their followers [6] [19]. As a result, many messages and posts that show support, oppose, and identity-based insults are observed. With this in mind, the government has enacted a law that regulates hate speech; the law [2] prohibits anyone from spreading hate speech on social media and other print media. The

individual or group who committed this crime will be fined from money to imprisonment depending on the severity and simplicity of the crime.

2.3. Hate speech detection techniques

There are different researches, which are conducted to detect hate speeches in different languages. For detection, they used different feature extraction and classification techniques. In different researches, three disciplines are mainly used for hate speech detection. Those are information retrieval, natural language processing, and machine learning.

2.3.1. Information retrieval (IR)

After the coming of Internet and social media, different individuals and organizations started publishing and sharing a huge amount of information [20] which should be filtered, pre-processed, and extracted. So, the goal of Information retrieval is finding information or a document from a large number of unstructured data [21]. It is a technique of examining a collection of documents and returning a sorted document list relevant to the user requirements as expressed in the query [22]. For hate speech detection different researchers [14] [23] [24] [25] [26] used IR for crawling dataset (dataset building) and feature extraction.

Dataset preparation: social media has millions of users and these users generate a large amount of data from time to time [20]. So that it is necessary to filter what is useful for hate speech, so many researchers have used IR for extracting social media posts and comments to prepare a dataset.

TF-IDF (Term Frequency-Inverse Document Frequency): It is the statistical measure to evaluate the importance of the word in a given document. Term frequency is the counting of occurrence of a term in a document [27]. A term can occur more times in a long document than a short one so to get a frequency of a term, calculated as, the counted term frequency divided by the total number of terms in the document [28].

$$\text{Term frequency}(\text{TF}(t, D)) = \frac{\text{Number of times term t appears in the document}}{\text{Total number of terms in the document}} \text{ ----} \quad \text{Equation 2.1}$$

Inverse Document Frequency (IDF**)** measures the value of a term. Calculated by taking the total number of documents divided by the number of documents that contain the word, then calculating the logarithm of the result. [27] [29].

$$\text{IDF}(t, D) = log(\frac{total\ number\ of\ document}{number\ of\ document\ with\ term\ t\ in\ it}) \quad \text{----------} \quad \textit{Equation 2.2}$$

TF-IDF is calculated as the Term frequency of a term multiplied by the inverse document frequency of a document.

$$\text{TF-IDF}\ (t,D) = \text{TF}(t,D)*\text{IDF}(t,D) \quad \text{----------} \quad \textit{Equation 2.3}$$

Bag of words (BOW): is a representation of words in numbers using the frequency of a word itself. It is also a word-embedding model used to predict the probability of a word. In this model, a text is represented as the bag of its words. It is widely used for language modeling and document classification [30]. Because any information about the structure or order of words in the document is discarded It is called a "*bag*" of words, the concern of the model is only whether known words occur in the document or not. The steps to implement a bag of words is, first, vocabulary is constructed from all unique words by determining the unique vocabularies of the given document, then the vector of the word is built by counting the occurrence of the word in a given sentence or document. it is used for extracting features from texts for machine learning algorithms [31]. For hate speech detection BOW is used for measuring the importance of the word.

2.3.2. Natural Language Processing (NLP)

NLP is a subfield of linguistic, artificial intelligence, and computer science. It focuses on the interaction between computers and human language. More specifically, natural language processing is human language understanding, manipulation, analysis, and generation of a computer [32]. So to detect hate speech effectively and efficiently, natural language understanding is needed. In computer science studies different researchers applied NLP techniques for natural language understanding, prediction, and feature extraction [33].

N-gram: is a model used to assign a probability to a sentence or a sequence of items. We can think of N-gram as a sequence of N items. According to the application, the item can be words, letters, syllables, phonemes. If it is a sequence of two words, it is referred to as a 2-gram or bi-gram. For example, "how are", if it is a sequence of three words, it is referred to as a 3-gram or tri-gram [34]. For example, "how are you".

N-grams are used for a variety of different NLP tasks such as word prediction, part of speech tagging, and also for extracting features. So that for hate speech detection many researchers used it for extracting features. To know the upcoming word after "how are you" is "doing" using N-gram, we calculate the probability (P) of "doing" after "how are you" as follows.

$$P(\text{doing} \mid \text{how are you}) \quad \text{---------- Equation 2.4}$$

To get the result, the occurrence of "how are you doing" is counted and divided by the count(C) of "how are you" in the document [35].

$$\frac{C(\text{how are you doing})x^2}{C(\text{how are you})} \quad \text{---------- Equation 2.5}$$

Part-of-speech (POS): POS tagging is a process of labeling a word in a sentence with its appropriate part of speech [36]. Traditionally there are eight parts of speeches: noun (NN), pronoun (PR), verb (VB), adverb (ADV), preposition (PREP), adjective (ADJ), conjunction (CONJ), and article [37]. In NLP, part of speech tagging is widely used for word sense disambiguation and subsequent syntactic parsing. For hate speech detection it can be used to identify the role of the word in a sentence and to understand the context.

There are two approaches for POS tagging, the first one is training on human-annotated corpora and the second is Human crafted rules based on lexical and other linguistic knowledge [38]. Most of the local language hate speech detection researches [13] [14] are not used POS tagging for feature extraction like other feature extraction models. Because preparing a training corpus for under-resourced languages needs to be done manually and doing it manually takes more time.

Dictionary-based: the simple method in text mining is using a dictionary. In the dictionary-based method, a list of words is constructed and stored for text mining. Most of the time, this method is used to store and count unique words in the document. In hate

speech detection, a dictionary can be used to store hate and offensive words to cross-check the occurrence of hate words in a given sentence [39].

Skip-Gram: is a model for creating word embedding. Its main objective is to predict the context of a given target word. Contexts are predicted using immediate neighbors of the target word and retrieved using a sliding window of size N by capturing N-words to the right and N-words to the left of the target word [40].

Global vectors for word representation (Glove): is another model for creating word embedding based on matrix factorization techniques on the word-context matrix. First, a large matrix X is constructed, that contains the co-occurrence statistics of words, in the matrix, and each element Xij represents how often word i appears in the context of word *j*. then for each term frequency of the context word is counted. Less weight is given to more distant words from the term [41].

Word2vec: is developed by Tomas Mikolov, et al. at Google in 2013, it is a method of creating word embedding using a two-layer neural network. It takes text corpus as input and produces a set of vectors known as feature vectors as output. Word2vec is not a deep neural network but it converts text to a numerical form that deep neural network understands. Its main objective is to represent words that have a similar context in a similar embedding. It is a single algorithm but it is a combination of two techniques: CBOW (Continuous bag of words) and Skip-gram model [42].

2.3.3. Machine learning

Machine Learning is a sub-area of artificial intelligence, whereby the term refers to the ability of a computer system that finds a solution to a given problem independently by recognizing patterns from the given dataset [43]. Machine learning recognizes patterns based on datasets and existing algorithms, this enables machine learning to produce adequate solution concepts [44]. Therefore, on the basis of experience, artificial knowledge is generated. To have an accurate solution, the required data and algorithms must be fed in advance into the system. Once the required data, algorithm, and respective rules for recognition of patterns are fed into the system, the machine learning system can perform extraction, summarization, finding of relevant data, predicting values, calculating result of probabilities, processes optimization using recognized patterns [45].

2.3.3.1. Types of Machine Learning

In machine learning algorithms has two roles: On the one hand, they have a responsibility of recognizing patterns and on the other hand, they have a responsibility of generating a solution [45]. Machine learning can be divided into different categories as follows:

Supervised machine learning: Algorithm is trained using labeled data, this means on the input the desired output is fed together [44], for example, an image of a caw labeled "caw". The algorithm learns using a set of inputs with corresponding outputs by comparing the received output with the actual output [43]. Through machine learning methods like classification, regression, gradient boosting, and prediction supervised learning uses patterns to predict the values of the label on additional unlabeled data. The common applications of supervised learning are data mining, computer vision, speech recognition, spam detection, bioinformatics, cheminformatics, and market analysis [46]

Unsupervised learning: The unsupervised learning learns by exploring the data and find some pattern within [43]. In unsupervised learning, artificial intelligence learns without telling the right answer [44]. The system must figure out what is being shown by finding certain characteristics of the given data. Unsupervised learning works well on transactional data. Some Popular algorithms include k-means clustering, nearest-neighbor mapping, and singular value decomposition. These algorithms are also used to segment text topics, recommend items, and identify data outliers [47].

Semi-Supervised learning: it a combination of both supervised and unsupervised learning because it uses both labeled and unlabeled data for training [43]. Because labeled data is more expensive and takes more effort to acquire [48]. it uses a large amount of unlabeled data. This type of learning is useful when the cost of fully labeling a dataset is high. And it is applicable for prediction, classification, and regression [49].

Reinforcement learning: is often used for navigation, gaming, and robotics. In this learning, the system learns with trial and error using the three primary components: the agent (the learner or decision-maker), the environment (everything the agent interacts with), and actions (what the agent can do) [43]. The objective of reinforcement learning is choosing actions that maximize the expected result over a given amount of time. The agent

will reach the goal much faster by following a good policy. So the goal in reinforcement learning is to learn the best policy [50].

2.3.3.2. Machine learning algorithms

Support vector machine (SVM): This is the most popular supervised machine learning model used to classify the given dataset into some groups. SVM takes a labeled training dataset for each class, so that it is able to classify new unseen data. To run SVM, the given training and testing dataset must be converted into a vector of numbers. SVM uses a hyperplane, which is the decision boundary of two classes [51]. Classifying n-dimensional linearly separable dataset into two classes C1 and C2 based on the labels Li is the simplest task version of SVM [52]

For example, if we have two classes x, and y

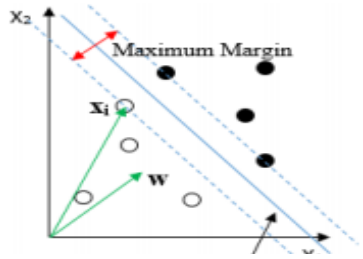

Figure 2.1 Support vector machine

The hyperplane is a line between classes used to separate n-dimensional linearly separable datasets if there exists at least one n−1 dimensional vector space [45]. Anything that falls on the above of the hyperplane will be classified into one class and anything fall below the hyperplane will be classified into another class. The gap between the linearly separable data and the hyperplane is called margin [53]. SVMs are originally designed for binary classification but it can be extended to multiclass classification using the divide and conquer method by breaking down the multiclass classification problem into a series of binary classification problems. So that the study goal is classifying more than two class Amharic hate speeches. This study selected One Vs. Rest SVM for multiclass classification. The task of the One Vs Rest approach is training a binary classifier for the dataset of each class [54] [55] [56].

Naïve Bayes: is a supervised machine learning algorithm primarily used for classification. It is a probabilistic classifier, learns a probability of the input with a certain feature

belonging to a particular group [57]. The algorithm is called “naïve” because the classifier assumes that the occurrence of certain features is independent of other features' value. Naïve Bayes is a probabilistic model that takes a vector of problem instances x = (x1, . . ., xn) represents the instances by n features. For classification, it assigns a probability for each of k possible outcomes or classes.

$$P(Ck|x) = \frac{P(Ck)p(x|Ck)}{P(x)} \quad \text{----------} \quad \textit{Equation 2.6}$$

Where, P(c|x) is the probability of data x to be classified in Class C, P(c) is the prior probability of class C, P(x|Ck) is likelihood which is the probability of x data predicted of class C, and P(x) is the prior probability of the data x [58].

Random forest: is also a supervised machine learning algorithm. It is a set of decision trees. In the tree, two types of randomness are integrated, the first randomness is each tree is built on a random sample from the original data. The second randomness is at each tree node, a subset of features are randomly selected to generate the best split [59], The decision trees are built using the bootstrap(bagging) method [43]. To interpret the result, it uses a majority vote in the case of classification.

2.4. Amharic language

Amharic language called “Amarigna” is the official language of the federal democratic republic of Ethiopia [11]. It is one of the two main and widely used languages of Ethiopia along with Afan Oromo. It is the southwest Semitic group of Afro-Asiatic language and is related to Ge`ez language, which Ethiopian Orthodox church uses it as a liturgical language [12], it is an official language of the country Ethiopia whose current population is estimated to be above 105 million [18]. In addition, it has become the second most-spoken Semitic language in the world next to Arabic [11].

Amharic uses the Ge`ez script for writing which has 33 basic characters, each of which has seven or eight forms depending on which vowel is to be pronounced in the syllable . Amharic is one of the languages in the world that have its indigenous scripts, even if it adopts its script from Ge`ez script [60].

2.4.1. Amharic Character Representation

Amharic uses Ge'ez script called Fidel (ፊደል) for writing. The script has 33 core characters; each character has seven or eight different forms and fewer having 12 or 13. Among those 33 characters, one is used as a vowel, which occurs in seven different forms. Some characters have the same pronunciation. For example, ሀ, ሐ, ሃ, ሓ, ኃ, and ኻ pronounced the same "Ha" [60].

Table 2.1 Amharic core characters

	ä/e	U	I	A	Ē	ə/ e	o	ua
H	ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ	ሇ
L	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ	ሏ
H	ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ	ሗ
M	መ	ሙ	ሚ	ማ	ሜ	ም	ሞ	ሟ
S	ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ	ሧ
R	ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ	ሯ
S	ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ	ሷ
Sh	ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ	ሿ
K	ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ	ቋ
B	በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ	ቧ
T	ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ	ቷ
Ch	ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ	ቿ
H	ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ	ኋ
N	ነ	ኑ	ኒ	ና	ኔ	ን	ኖ	ኗ
Gn	ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ	ኟ
A	አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ	-
K	ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ	ኳ
H	ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ	ዃ
W	ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ	ዏ
A	ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ	-
Z	ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ	ዟ
Zh	ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ	ዧ
Y	የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ	ዯ
D	ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ	ዷ
J	ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ	ጇ
G	ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ	ጓ
T	ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ	ጧ
Ch	ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ	ጯ

P	ጰ	ጱ	ጲ	ጳ	ጴ	ጵ	ጶ	ጷ
Ts	ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ	ጿ
Ts	ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ	ፇ
F	ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ	ፏ
P	ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ	ፗ

2.4.2. Punctuations

Amharic uses different punctuation marks to separate words, to show the end of a sentence, to show Interjection, to raise a question, and so on. Some of them are listed in the table below.

Table 2.2 some Amharic punctuation marks

።	፡	፣ or ፥	፤	…	፦
Full stop (አራት ነጥብ)	Word separator (ሁለት ነጥብ)	Comma (ነጠላ ሰረዝ)	Semicolon (ድርብ ሰረዝ)	So on (ወዘተ)	Preface Colon (ሁለት ነጥብ ከሠረዝ)

2.4.3. Character representation challenges for hate speech detection

The difficult challenge is the different types of people's writing style. For example, some people use the conjunction "እና "to connect two words. For example, "አበበ እና ከበደ", meaning: Abebe and Kebede, this is very easy to identify, but some people associate the conjunction with the word, for example, "አበበና ከበደ". If we make a stemmer or separator for such scripts and order it to stem or separate the words that end with "ና ", it can change "አበበና ከበደ" to "አበበ ከበደ" or "አበበ ና ከበደ" but it cannot work for words like ጎበና ሞጣ, ጀበና ገዛሁ, ኮረና ገባ, ጥገና ያስፈልጋል, ዋና ስራአስኪያጅ, ጀግና ነው, so this is a big challenge.

2.5. Detection model evaluation

In most researches [14] [24] [61] [62] machine learning performance evaluation metrics like confusion matrix, precision, recall, area under curve, ROC score, f1 score, and accuracy, are used for evaluating their model. Four important terms that must be known before proceeding to the presentation of evaluation metrics are:

- **True Positives(TP)**: are the case when the actual class is YES and the predicted class is also YES
- **True Negatives(TN)**: are the case when the actual class is NO and the predicted class is also NO
- **False Positives(FP)**: also called Type I error, are the case when the actual class is NO and the predicted class is YES
- **False Negatives(FN)**: also called Type II error, are the case when the actual class is YES and the predicted class is NO

2.5.1. Confusion matrix:

It is a machine learning model performance measurement mechanism, measures performance by counting the number of times an instance of class X is classified as class Y. It is a matrix representation of the prediction algorithm performance [63] [64].

Table 2.3 Confusion matrix for binary class

Actual Value		Negative	Positive
	Negative	True Negative (TN)	False Positive (FP)
	Positive	False Negativ (FN)	True Positive (TP)
	Predicted value		

Table 2.4 Confusion matrix for ternary class

Actual Value		Religious Hate (RH)	Ethnic Hate (EH)	Non-Hate (NH)
	Religious-Hate (RH)	True RH	False EH	False NH
	Ethnic Hate (EH)	False RH	True EH	False NH
	Non-Hate (NH)	False RH	False EH	True NH
	Predicted value			

2.5.2. Precision:

It is a popular metric used to evaluate the quality of a classification model. It shows the accuracy of the positive class [64]. It computes how likely positive class prediction is

correct. Calculated as, the number of correct positive results divided by the number of positive results predicted by the classifier [63].

$$Precision = \text{Value predictive Positive } \frac{TP}{(TP+FP)} \text{----------} \quad \text{Equation 2.7}$$

$$\text{Precision} = \text{Value predictive Negative } \frac{TN}{(TN+FN)} \text{----------} \quad \text{Equation 2.8}$$

2.5.3. Recall:

It measures the ratio of positive instances calculated as the number of correct positive results divided by the number of ***all*** relevant instances that should have been identified as positive [63] [64].

$$\text{Recall} = \frac{TP}{(TP+FN)} \text{----------} \quad \text{Equation 2.9}$$

2.5.4. F1 score

Is a popular metric that combines precision and recall as single metrics, used for models that need both precision and recall as evaluation metrics. It is the harmonic mean of precision and recall [63]. The resulting range is 0 to 1. It tells the number of instances classified correctly and the significant number of instances that are not missed. If the precision is high and the recall is lower, it gives an extremely accurate result but there will be a missing of a large number of instances that are difficult to classify [64]. Calculated as:

$$\text{F1 score} = \frac{2*\text{Precision}*\text{Recall}}{\text{precision}+\text{recall}} \text{----------} \quad \text{Equation 2.10}$$

2.5.5. Accuracy

It is the most common machine learning model evaluation metric for classification problems, it measures the total number of correct predictions. It is the success rate (percent) of instances we correctly classified from the whole given dataset [63] [64]. Calculated as:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \text{----------} \quad \text{Equation 2.11}$$

2.5.6. Cross-Validation

K-fold cross-validation is also used for evaluating the performance of the proposed model. In this evaluation model, the dataset is divided into k sup parts [64] [65], in this study case k is 5 then each subpart used as a testing dataset in turn, and the rest four subparts used as the training dataset. The dataset is divided into five subparts in each iteration 20% of the dataset used for testing the model. The following figure shows the role of each subpart of the dataset in each iteration.

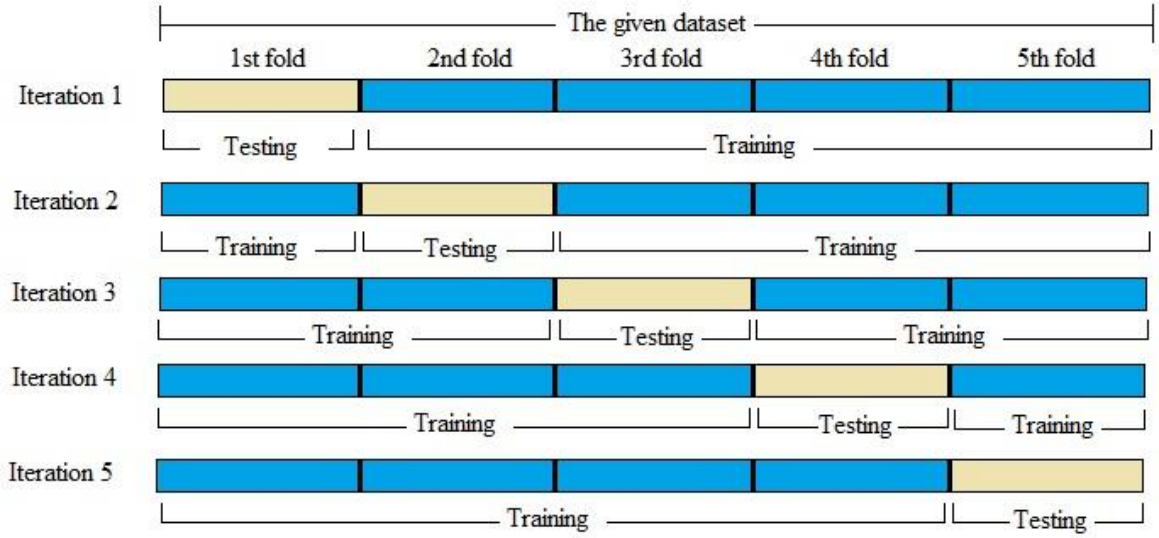

Figure 2.2 5-fold cross-validation

CHAPTER THREE

3. Related works

This chapter shows the review of different related researches to identify the gaps and to have a background knowledge. A lot of researches were done to detect hate speech in Amharic and international languages; like English and others. Classified and reviewed as follow:

3.1. Hate speech detection for Amharic language

In local languages two research was done on Amharic hate speech detection the first one is "Social network hate speech detection for Amharic language" [13], In this paper, the researcher collected Amharic posts and comments from different public Facebook pages. The collected datasets are pre-processed by removing null values, non-Amharic characters, punctuations, repetitions, elongations, and HTML tags. After that, discourse analysis was made by categorizing the dataset as political, ethnic, socio-economic, and religious context. Then the content analysis is also done by categorizing the dataset as hate or not-hate.

Features are extracted, using both Word2Vec and TF-IDF. After feature extraction 6120 annotated datasets were used for the experiment by taking 80% of the dataset as a training set and the remaining 20% as a test set. Both Naïve Bayes and Random Forest machine learning algorithms are tested for the classification but the better result (79%) was achieved by combining Naïve Bayes with Word2Vec.

On dataset preparation discourse analysis is made by identifying the context of each dataset as political, ethnic, socio-economic, and religious context but even if the discourse analysis is done in many categories, the classification is binary (Hate or Not-Hate). On other hand the dataset annotation is done by, 1 assistant professor from Amharic studies, 3 PhD, and 2 MSc students. But it should be annotated by law experts from a legal point of view. The used feature extraction methods don't consider the context of a word in a given sentence.

The second one is "Hate Speech Detection for Amharic Language on Social Media Using Machine Learning Techniques" [14], the speeches are classified into three classes as hate, offensive, and non-hate. TF-IDF, Word2Vec were used for feature extraction. For

classification, support vector machine (SVM), Random Forest (RF), and Naïve Bayes (NB) machine learning algorithms are used. Totally around 837077 Amharic posts and comments are collected using facepager, after filtering and annotating 5000 unique posts, and comments are used for clarification as Hate (HS), Offensive (OFS), and neither offensive nor hate (OK). In the study different models are tested and compared, finally by performing an accuracy of 75.39% SVM with word2vec showed slightly better performance than NB and RF models.

The strong side of this paper is, three machine learning algorithms are implemented with different feature extraction methods and comparison is made to identify the best combination of feature extraction method and classification algorithm. Finally, different evaluation metrics are used to measure the final model.

The weak side of the study is; the used feature extraction methods don't consider the context of a word in a given sentence. And also The hate speech law [2], , doesn't consider political hate as hate speech, but the study included political hate as hate speech.

3.2. Hate speech detection for English language

The study [62] used 14509 English tweets which are prepared by other studies. The datasets are classified into three classes 2,399 tweets as Hate, 4836 as Offensive, and the rest as Ok. Features are extracted using surface N-gram and Word skip-gram. For classification SVM supervised machine learning algorithm is used, additionally they tried to classify profanity speeches. Finally, they achieved 78% accuracy by combining the character 4-gram with SVM.

On the other hand, research [66] comes up with a different concept which is gathering the demographic information of the author and they used node representation to identify the relationship between the author and other authors. The general concept of this research is detecting hate speeches based on the demographic information of the authors. They used a dataset that is created by other researchers and labeled 12% of the dataset as racism, 19.4% as sexism, and 68.6% as none. Features are extracted using character N-gram, node2vec, and Author profile, the recurrent neural network is used for classification. The achieved result is 87.57% of the F1-score.

Thomas Davidson, et al [25]. Conducted a study to detect English language hate speeches in 2017 G.C. 25,000 tweets are used as a dataset and the majority are labeled as hate. For feature extraction, TF-IDF and N-gram are used. Five models naive Bayes, logistic regression, random forests, decision trees, and linear SVMs are tested. Finally, they showed logistic regression has better performance than others. The achieved result is an F1 score of 0.90, precision of 0.91, and recall of 0.90.

Muhammad Sajjad, et al. [67] used 12,000 tweets. 1302 tweets are labeled as sexist, 2901 tweets classified as racist, and the rest as neither. They used Glove, baseline features as the feature extraction method, and used CNN, SVM, Logistic Regression, Random forest as a classification mechanism. Finally, the best result they achieved is 0.984 of precision, 0.965 of recall, and 0.974 F1 by combining CNN + Glove + Baseline Features + LR.

In the first study [62], different feature extractions are implemented and compared, then, feature extraction that performed better is selected. The second study [66], came up with a different concept, which is the demographic information of the author is used as a feature.

The third [25] and the fourth [67] study, compared different machine learning algorithms and recommended the algorithm, which resulted in better performance.

The weak side of these papers are; the profession of the annotators is not mentioned, whether they are law experts or in another profession. Because knowing the annotators increases the reliability of the dataset. The feature extraction methods, which are used in all studies doesn't consider the context of a given text. The study [25], doesn't mention the number or percentage of datasets that are labeled as hate or not-hate.

3.3. Hate speech detection for Indonesian language

Ika Alfina, et al. [23] conducted a study to detect Indonesian language hate speeches. For the study 40,000 dataset was collected from Twitter, by removing duplication 1,100 tweets were labeled manually. The data was annotated by 30 volunteer college students who are selected from 5 different religions and races/ethnicity. After the annotation, the researchers take 713 tweets that had 100% agreement by annotators.

Among those 713 tweets, 260 tweets are labeled as hate speech (HS) and 453 tweets as non-hate speech (Non-HS). After that to overcome the unbalanced dataset negative effect, they decreased the Non-HS from 453 to 260 by selecting randomly, thus 520 datasets are selected for detection. They preprocessed the finally selected datasets, and then features are extracted using bag-of-words (BOW), word N-gram, character N-gram, and negative sentiment. For the sentiment analysis, a sentiment dictionary was used. Finally, 4 machine learning algorithms NB, SVM, RFDT, and BLR are used for classification and to evaluate the performance of each algorithm. The achieved top result is 93.5% of F-measure by combining word N-gram with RFDT.

On the other hand Junanda et al. [61] also conducted a study to detect Indonesian language hate speech by Junanda Patihullah and Edi Winarko. The study used 713 tweets, 260 tweets labeled as hate, and 453 labeled as not-hate. Features are extracted using Word2Vec, TF, and TF-IDF. By using supervised machine learning algorithms (SVM, NB, BLR, and RF) they made a comparison between feature extraction models listed above, thus the accuracy of Word2Vec is lower than TF, and TF-IDF. But by making the number of hidden layer neurons 200 92.96% accuracy was achieved by combining Word2Vec with a deep learning classification algorithm (gated recurrent unit). Finally, they showed that Word2Vec with GRU has a better performance than traditional supervised machine learning algorithms for the Indonesian language.

Both papers [23] [61], compared different combinations of feature extraction and machine learning classification algorithm, then identified the best performed one. In the first paper, the annotation is done by college students, not by law experts in the law point of view. In both papers, the used feature extraction method works based on the probability and the occurrence of a word (frequency), not by context understanding

3.4. Hate speech detection for Italian language

To detect the Italian language hate speech Fabio, research is conducted by Del Vigna, et al. [68]. They collected 17,567 from Italian Facebook public groups, newspapers, politicians, artists. By giving the crawled dataset to five annotators, they take the agreed 1,687 datasets as training and test set. For classification, they used SVM and RNN named long short term memory. They used different techniques to extract features, to extract lexical text features

they used word N-gram and to extract syntactic features they used part of speech tagging. Finally using SVM they achieved 80.60% accuracy.

Michele Corazza, et al [69], compared Italian language hate speech detection models. By implementing N-gram based neural networks, recurrent neural networks, and linear SVC they achieved a 0.68 F1 score using RNN. Both papers [68] [69], compared different combinations of feature extraction and machine learning classification algorithm, then identified the best performed one.

3.5. Hate speech detection for Hindi language

To detect hate speech written in both English and Hindi in the same sentence a study [70] is conducted. The used dataset contains 10,000 texts and divided into two equally then labeled the half as Hate and the rest non-hate. They made three experiment sessions on the first session by making the sliding window size 5 and the vector length 300 they created 10,000x300 vector features using doc2vec, fed it into machine learning algorithms such as SVM and Random Forest. Then Random Forest scored a better result than SVM which is 0.64 accuracy. In the second session, they used the same procedure but the 10,000x300 vector features are created using word2vec. In this session, SVM scored a better result than the random forest which is 0.7511 of accuracy. In the third session, features are extracted using character n-gram and achieved a better result (0.85 of accuracy) than the previous two sessions. The strong side of this paper is, the experiment is done in three sessions to select the best performance but the labeling of the dataset is not described whether it is approved by annotators or not.

3.6. Hate speech detection for Albanian language

To detect Albanian language hate speech [71] is studied. For the study, 4886 posts and comments are collected from Facebook by using developers.facebook.com/docs/graph-api. The dataset is analyzed by two annotators and they agreed and classified 2764 posts and comments as hate and the rest as non-hate. The selected datasets are preprocessed by normalizing and removing extra white spaces, punctuation marks, digits, emojis, and redundancies. For classification Support Vector Machine (SVM) is used. Finally, by

assigning 4000 datasets as a training set and 886 datasets as a test set, they achieved 0.61 precision, 0.57 recall, and 0.58 F-Score.

Strong side of this study is the way of the annotation; annotation is performed by two annotators and the dataset, which is commonly agreed on, is taken. This increases the reliability of the dataset.

The weak side is; The annotators agreed on 2764 posts and comments as hate, but they used 4000 datasets as hate for training and test set. Additionally, the feature extraction method used in this paper is not written clearly.

3.7. Hate speech detection for Portuguese language

For detecting Portuguese language hate speech, [24] is prepared. They used other researcher's datasets by exploring from the internet. For classification, two models are used, deep learning technique (CNN) and Logistic Regression, Linguistic inquiry with Word count. For training, they used 90% of the dataset and the rest 10% for testing. From downloaded 10,366 datasets, they randomly selected 1250 only. The downloaded datasets were annotated as intolerance, racism, sexism, xenophobia, homophobia, religious, cursing, and not offensive, but they selected the offensive and not offensive classes. On the implementation, they used two procedures, the first one is applying CNN with feature extraction methods (Word2vec and Glove), and the second one is Applying CNN alone. On the first procedure, they achieved 83.35% test accuracy.

The Strong side of this study is; two procedures of experiment are conducted to select the best combination. the weak side is; The feature extraction doesn't consider the contextual understanding of words in a given sentence.

3.8. Hate speech detection for Vietnamese language

To detect Vietnamese language hate speech, the study [26] is conducted. They used 20,345 human-labeled comments/posts for training and 5,086 for testing. They used three deep learning models text-CNN, Bi-GRU-CNN, Bi-GRU-LSTM-CNN finally they achieved a better result 70.576% F-1 score using Bi-GRU-LSTM-CNN. Three deep learning algorithms are implemented and selected the best performing one.

The issue seen as a weak side is, no information about the labeling of the dataset, whether it is approved by annotators or not.

3.9. Summary of related works

Table 3.1 Summary of related works

Title	Author	Class	Feature extraction method	Classification method	Result
Amharic Language					
Social network hate speech detection for Amharic language(2018) [13]	Zewdie Mossie and Jenq-Haur Wang	hate and Not-hate.	Word2Vec and TF-IDF	Naïve Bayes, Random Forest	(79%) using Naïve Bayes with Word2Vec
Hate Speech Detection for Amharic Language on Social Media Using Machine Learning Techniques(2019) [14]	Yonas Kenenisa Defar	hate, offensive, and non-hate	N-gram, TF-IDF, and Word2Vec	SVM, RF, and NB	75.39% using SVM with word2vec
English Language					
Detecting hate speech in social media(2017) [62]	Shervin Malmasi, et al.	Hate, Offensive, and Ok	surface N-gram and Word skip-gram	SVM	78% of accuracy
Author Profiling for Hate Speech Detection (2019) [66]	Pushkar Mishra, et al.	racism, sexism, and none	character N-gram, node2vec, and Author profile	RNN	87.57% of F1-score.
Automated Hate Speech Detection and the Problem of Offensive	Thomas Davidson, et al.	Hate, and not-hate	TF-IDF and N-gram	NB, LR, RF, DT, and linear	F1 score of 0.90, precision

Language(2017) [25]				SVM	0.91, and recall of 0.90 using LR
Hate Speech Detection using Fusion Approach(2019) [67]	Muhammad Sajjad, et al.	sexist, racist, and Neither	Glove, and baseline features	CNN, SVM, LR, and RF	0.974 F1 by combining CNN + Glove + Baseline Features + LR
Italian Language					
Hate me, hate me not: Hate speech detection on Facebook(2017) [68]	Fabio Del Vigna	Hate, and not-hate	N-gram and PoS tagging	SVM and RNN (long short term memory)	80.60% accuracy using SVM
Comparing Different Supervised Approaches to Hate Speech Detection(2018) [69]	Michele Corazza, et al.	Hate, and not-hate	N-gram	RNN and linear SVC	0.68 of F1 score using RNN
Indonesian Language					
Hate Speech Detection in the Indonesian Language: A Dataset and Preliminary Study(2017) [23]	Ika Alfina, et al.	HS, and Non-HS	bag-of-words, word N-gram, character N-gram, and negative sentiment	NB, SVM, RFDT, and BLR	93.5% of F-measure by combining word N-gram with RFDT.
Hate speech detection for Indonesia tweets using word embedding and	by Junanda Patihullah and Edi Winarko	hate, and not-hate	Word2Vec, TF, and TF-IDF	SVM, NB, BLR, and RF	92.96% accuracy by combining

Gated recurrent unit(2019) [61]					Word2Vec with GRU
Hindi Language					
Detection of Hate Speech Text in Hindi-English Code-mixed Data(2020) [70]	Sreelakshmi k., et al.	Hate, and non-hate	doc2vec, word2vec, and character n-gram	SVM-linear, SVM-RBF, and RF	0.85 of accuracy using character n-gram and SVM-RBF
Vietnamese Language					
Hate Speech Detection on Vietnamese Social Media Text using the Bi-GRU-LSTM-CNN Model(2019) [26]	Tin Van Huynh, et al.	Hate, Offensive, and Clean	Word2Vec	text-CNN, Bi-GRU-CNN, Bi-GRU-LSTM-CNN	70.576% F-1 score using Bi-GRU-LSTM-CNN
Albanian language					
Automatic hate speech detection in online contents using latent semantic analysis(2017) [71]	Xhemal Zenuni, et al.	hate and non-hate	Not mentioned	SVM	0.61 precision, 0.57 recall, and 0.58 F-Score
Portuguese language					
Hate-speech detection in Portuguese using CNN and psycho-linguistic dictionary(2019) [24]	Samuel Silva, et al.	Offensive and not offensive	Word2vec, and Glove	CNN, LR, and Linguistic inquiry with Word count	83.35% of accuracy using CNN

CHAPTER FOUR

4. Methodology

To conduct this study and to answer the specified research questions, different methodologies are used. In this chapter, the dataset building methods, feature extraction model creation and model evaluation techniques that help to achieve the research objective is discussed.

4.1. Literature review

To have a background knowledge and to identify so many gaps, a detailed review and critique have made on hate speech detection researches. Dataset preprocessing methods, feature extraction methods, and classification methods: like traditional machine learning and deep learning are reviewed.

4.2. Building dataset

To create a machine learning model for Amharic hate speech detection, collecting and building a dataset for training and testing is necessary. Because there is no relevant published Amharic hate speech dataset, it is necessary to gather and build a new dataset. The following methods are used for building the dataset.

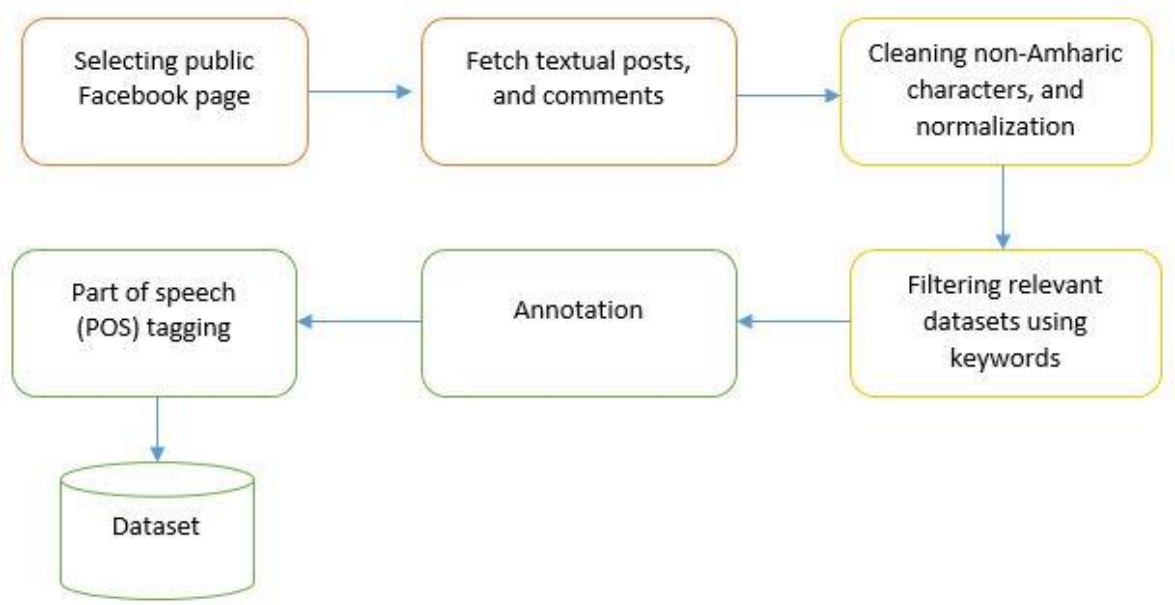

Figure 4.1 Used methods for building a dataset

4.2.1. Dataset collection

For this study, datasets are collected from selected Facebook public pages. Facebook is selected among other social media platforms because of its large number of users in Ethiopia when compared to other social media [72]. Ethiopian law [2] defines hate speech as "Speech that encourages discrimination or violence against an individual or group based on religion, race, disability or gender." Therefore, the data is collected from various Facebook pages that post, religious issues, gender issues, broadcast media pages that report about different nations and nationalities, government official pages which, many people react, artist, and activist pages that write about different issues, Facebook pages of institutions such as law, human rights, defense, and security, have been selected. After selecting these Facebook pages, sorting is performed based on their number of followers, finally the top 25 pages, which have higher number of followers are selected for building the final dataset.

On these Facebook pages, various news, government statements and messages, activities and situations of different nationalities, religious issues, and entertainment and educational articles are posted. Every selected Facebook page has more than 75,000 likes and followers and they constantly post information, with a large number of followers. Their followers also comment on what they think. Categorized as follows:

Security and defense agencies:

These institutions protect the peace and security of the people and post on their Facebook page about various crimes, threats, and issues related to peace and security. Their followers also express their feelings by commenting below the post. Therefore, we have selected these institutions' Facebook pages for data collection.

Government officials and offices:

These Facebook pages are created and controlled by senior government officials and offices. Used to send messages to their citizens, inform their citizens of various strategies and government positions, and to inform them about the current state of the country. These pages have a lot of followers, and when they post a message, people who are comfortable with the message write supportive words, and people who are not comfortable with the message write opposition in the comment section. The selected Facebook pages include the

prime minister and the Federal Government's top offices which are followed by many peoples.

Religious institutions, preachers, and teachers:

These Facebook pages are created by various religious institutions and teachers. They use it to teach, preach, and convey messages to their followers. People who follow the faith and those who do not follow it also comment on what they think. We collected data from these pages because different people's opinions are expressed in the comment section so it may affect the beliefs or beliefs of others.

Bloggers and activist:

These pages are created by various influential persons and they convey messages to their followers about various national and political issues. Because people write the comments they want, many types of ideas will be hosted. The selected pages have a lot of followers and quickly provide information to their followers.

Law enforcement agencies:

These agencies deal with legal issues, and they post their activities and messages on their Facebook page. The reason they are chosen is that they have so many followers, and it is allowed for the citizens to express their opposition and support on new rules and regulations in the comment section.

Other agencies:

As hate speech spreads, it affects human rights and the peace of citizens. These organizations have a lot of followers and they post various ideas and messages about human rights and peace. Their followers also comment on them.

When selecting the categorized Facebook pages mentioned above, we used the following parameters

1. A Facebook page, which can attract many followers from different angles in terms of religion, ethnicity, gender, etc.
2. A Facebook page, which posts a message usually in Amharic language.
3. A Facebook page, which delivers information to its followers continuously.
4. A Facebook page, which posts on urgent and most inclusive issues

Based on the above parameters more than 100 pages are nominated, then sorted by their number of followers finally, 25 pages that have more than 75,000 followers are selected, the collected posts and comments are created from March/2020 to November/2020 and various posts and comments have been fetched from all specified categories. Their details are described in Table as follows:

Table 4.1 detail of collected dataset and source

Categories	No	Page Name	created date	No of extracted data	No of filtered data	No of likes	No of Followers	Rate out of 5
News and Broadcasting agencies	1	Ethiopian Broadcasting Corporation	01-15/03/2020 And 01 - 11/11/2020	202796	29913	1,883,525	2,157,315	3.9
	2	FBC (Fana Broadcasting Corporate S.C.)	01-15/03/2020 And 01 - 11/11/2020	58602	9307	2,068,220	2,336,319	Not given
	3	Tigrai Media House	01-15/03/2020 And 01 - 11/11/2020	31727	2385	157,521	227,971	Not given
	4	VOA Amharic	01-15/03/2020 And 01 - 11/11/2020	7851	1010	999,012	1,096,824	Not given
	5	BBC News Amharic	01-15/03/2020 And 01 - 11/11/2020	13482	1790	572,292	651,956	Not given
	6	ESAT	01-15/03/2020 And 01 - 11/11/2020	6077	997	1,463,196	1,534,198	Not given
	7	DireTube	01-15/03/2020 And 01 - 11/11/2020	22081	3678	3,040,758	3,066,817	Not given

	8	Wazema Radio ዋዜማ ሬዲዮ	01-15/03/2020 And 01 - 11/11/2020	310	49	217,151	221,986	Not given
Security and defence agencies	9	Information network security(INSA)	16/10/2020 - 17/11/2020	240	38	75,443	79,712	4.2
	10	FDRE Defense Force - የኢፌዴሪ መከላከያ ሠራዊት	01-15/03/2020 And 01 - 11/11/2020	11207	1175	232,373	241,857	5
	11	Ethiopian Federal Police Commission - ህዝብ ግንኙነት	01 - 17/11/2020	4658	456	77,010	80,336	3.8
Government officials	12	Abiy Ahmed Ali	01-15/03/2020 And 01 - 11/11/2020	30684	6618	2,991,208	3,218,389	4.6
	13	Adanech Abiebie- አዳነች አቤቤ	01-15/03/2020 And 01 - 11/11/2020	331	54	172,373	192,593	Not given
Religious institutions and teachers	14	የኢኦተቤ መገናኛ ብዙኃን አገልግሎት ሥርጭት ድርጅት/ EOTC Broadcasting Service Agency/	01-15/03/2020 And 01 - 11/11/2020	974	111	98,971	106,477	5
	15	MARSIL TV WORLDWIDE	01-15/03/2020 And 01 - 11/11/2020	3200	65	288,394	384,032	4.9
	16	Mahibere Kidusan - ማኅበረ ቅዱሳን ዋናው ማዕከል	01-15/03/2020 And 01 -	710	119	405,740	429,093	Not given

			11/11/2020					
	17	Memeher Dr Zebene Lemma	01-15/03/2020 And 01 - 11/11/2020	383	32	322,883	345,953	Not given
	18	Ahmedin Jebel official - አህመዲን ጀበል	01-15/03/2020 And 01 - 11/11/2020	4837	174	604,975	665,488	Not given
Blogers and activists	19	Natnael Mekonnen	01-15/03/2020 And 01 - 11/11/2020	54259	7798	408,540	462,839	Not given
	20	Tamagne Beyene	01-15/03/2020 And 01 - 11/11/2020	3566	1392	468,940	537,035	Not given
	21	EthioTube	01-15/03/2020 And 01 - 11/11/2020	21759	2253	1,300,789	1,535,755	3.7
Government offices	22	Office of the Prime Minister-Ethiopia	01-15/03/2020 And 01 - 11/11/2020	3395	387	677,327	712,629	4.8
Law enforcement egencies	23	FDRE Attorney General የኢፌዲሪ ጠቅላይ ዐቃቤ ሕግ	01 - 17/11/2020	8957	1196	146,564	150,745	4.7
Other related agencies	24	Ethiopian Human Rights Commission	01 - 17/11/2020	1532	397	75,017	79,272	Not given
	25	Ministry of Peace የሰላም ሚኒስቴር	01 - 17/11/2020	2530	178	89,819	97,574	4.9
Total				478,231	69,337			

4.2.2. Dataset preparation

The dataset is collected blindly from various Facebook page posts and their comments, so it needs to be prepared in a way that suits the detection model. Therefore, various techniques have been applied to normalize and clear the collected data from various non-Amharic characters, symbols, punctuations, and emojis.

4.2.2.1. Filtering

When datasets are collected from the selected Facebook pages, it is not confirmed whether it contains hate speech or not, the criteria are just because it is an Amharic speech. So, a law expert is needed to annotate it by determining whether the statement is hateful or not. Because the collected data is so large, choosing hate speech from it, is very difficult and time consuming for the law experts. Therefore, the following techniques have been applied to easier the annotation.

Preparing a dictionary of words:

According to Ethiopian hate speech law [2], hate speech attacks a person based on religion, race, ethnicity, gender, and disability. This means, for a speech to be hateful, it must contain these words. Therefore, in the dictionary, different words are stored, which describe different religions, ethnicities, and races and also words that promote violence, discrimination and hatred by consulting language experts.

Applying dictionary words for filtering:

To make the annotation easier, checking the presence or absence of these words in the dataset, then select the data if the words are available in it, is a good method. Therefore, using the dictionary, a filtration method has been developed. After applying this method, the following amount of datasets are prepared for annotation.

Table 4.2 description of filtered dataset

Number	Categories	Number of Filtered datasets
1	Religious	14617
2	Ethnicities	93976
3	Races	2966

4	Genders	21493
5	Disabilities	1482
6	Hatred	29370
7	Attack	4888
	Total	**168792**

4.2.2.2. Annotation

Distinguishing hate speech from another form of speech should be based on the law [2] passed to control hate speech. So, lawyers who can manually label the speeches as hateful or not-hate is needed. Therefore, the filtered dataset is annotated by law experts. Instructions are given to annotators to label, "Religion-hate" if the speech is religious hate speech, "Ethnic-hate" if it is ethnic hate, and "None-hate" if it does not contain hate content.

4.3. Hate speech detection Modeling

4.3.1. Preprocessing

This method allows the collected posts and comments to be cleared of unnecessary characters. To make it convenient for feature extraction, different unnecessary non-Amharic characters are removed and normalization is performed. The following pre-processing methods are applied.

- Removing numbers and Non-Amharic letters
- Removing punctuations, special characters, symbols, emojis, and URLs
- Normalization

4.3.2. Feature extraction

Hate speech has some characteristics that distinguish it from other speeches, so identifying and knowing these features is very important to distinguish hate speech from other speech. In this study, a new feature extraction method is created using the following techniques.

Dictionary: In this study, a dictionary is used to check the presence or absence of properties and hate words in the discourse. If so, it used to answer the question "what are

they?” Promoting hatred and discrimination, or inciting violence? Also used to find out if there is a victim.

Part of speech (POS) tagging

This method tags part of speeches in each word to understand the message contextually that it wants to convey. To determine whether a speech is hateful, it is not enough just to find out the identity of the victim and the type of abuse, but also it is important to consider the semantically related terms of the sentence, as well as their contextual meaning. So that, to understand the context, the study used part of speech tagging. For this study, the eight main Amharic parts of speech with mixed parts of speeches are used, described below in the table.

Table 4.3 part of speech description

Part of speech	Tag name	Description	Example
Noun	NN	A name given to Person, places, animals, feeling, idea, things,	**አበበ** ተደብድቦ ይገደል
Pronoun	PR0N	A word that replaces a noun	**እሱ** ልጅ ያስጠላኛል
Adjective	ADJ	Gives information about a noun or a pronoun	**እረጅሙ** ልጅ
Verb	VB	Shows action or state in a sentence	አበበ ተደብድቦ **ይገደል**
Adverb	ADV	Gives information about a verb	አበበ **ተደብድቦ** ይገደል
Conjunction	CONJ	Used to connect ideas	ቀዩ **እና** እረጅሙ ልጅ
preposition	PREP	Shows relationships in time, places, and position	**ወደ** ጅማ
Interjection	INT	Used to show emotion or exclamation	**ኤጭ**, **ዋው**
Noun, conjunction	NN_CONJ	The mix of noun and conjunction	**አበበና** ከበደ
Preposition, noun	NN_PREP	The mix of noun and preposition	**የጅማ** ልጅ
Noun subject	NN_SUB	The action performer	**አበበ** ከበደን አይወደውም
Noun object	NN_OBJ	The action receiver	አበበ **ከበደን** አይወደውም

N-gram: is the most widely used probabilistic method for hate speech detection modeling, used to predict the next word after observing the previous N- 1 word. For this study, it is used for part of speech tagging.

TF- IDF: is a feature extraction method, it performs statistical measures to evaluate the importance of a word in a given document. For feature extraction, it counts the occurrence of each word in a given document then takes top words that have a high frequency. For this study, it is used to feature extraction by taking top frequent words and changing them to a vector.

BOW: Bag of words is also a frequency-based feature extraction method (more discussed in chapter two), used to extract features of hate speeches, which helps to make a comparison between the newly developed feature extraction method.

4.3.3. Classification

After features are extracted using the newly proposed feature extraction method and other methods, which are used in other studies, machine learning algorithms are needed for detecting hate speech automatically. So that to measure the effectiveness of the newly proposed feature extraction method and to compare it with other feature extraction methods, different machine learning algorithms are tested. These selected algorithms are recommended by other researchers and scored a better result on the background works. [14] [23] [61] [67] [62] [70] [25] [71] [13]

4.4. Evaluation

The performance of a model depends on the characteristics of the data to be classified. To measure the performance and to make a comparison between the selected supervised machine learning models, various empirical tests have been performed. On measuring the efficiency and effectiveness of the proposed Amharic hate speech detection model, different machine learning performance evaluation metrics like Confusion matrix, Recall, F1 score, and accuracy are used.

CHAPTER FIVE

5. Proposed model for Amharic hate speech detection

5.1. Introduction

This chapter explains the proposed method to detect hate speech, which is currently spreading on social media. First, the general overview of the proposed architecture is discussed then each module and components are described. how the dataset is collected, how the collected datasets are pre-processed, the way used to extract features of hate speech from the pre-processed data, the way used to classify as religion-hate, ethnic-hate or not-hate, the way the model is created to classify the newly coming texts, and used methods to evaluate the performance of the created model are discussed.

5.2. The proposed model

The architecture of the designed model contains seven main modules; dataset crawler, annotator, feature extractor, feature selector, context table, classifier, and result evaluator. Inside each module, there are sub-components. To make it clear, the discussion started by showing the architecture of the proposed model. Then the following points are described, how and what each component works, the contribution of each component to the next component, and how components interact with each other along with the resource needed.

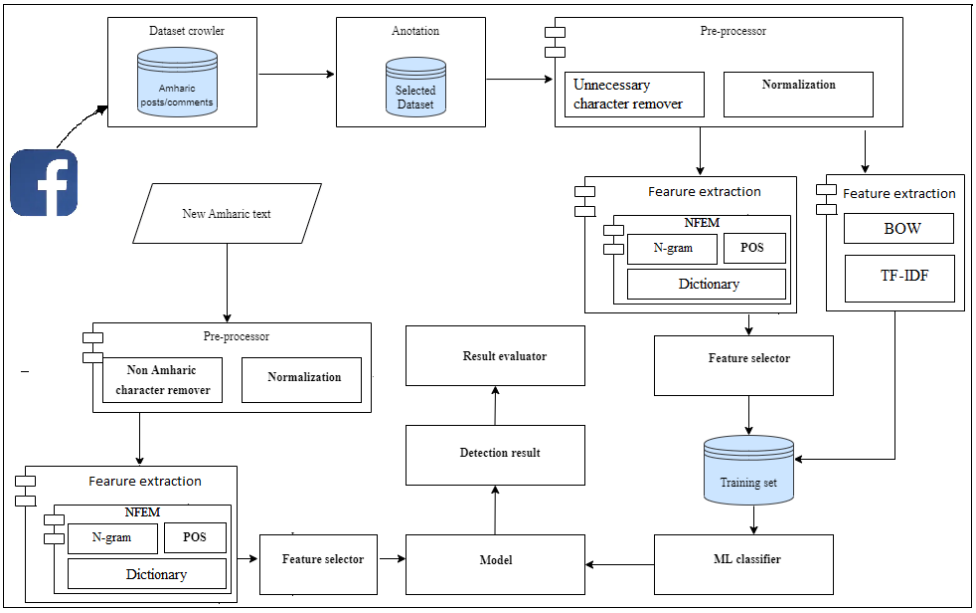

Figure 5.1 Architecture of Amharic Hate speech detection

5.2.1. Dataset crawler

When compared with offline, most of the time Hate speech is widely spreading online on the Internet, specifically on social media. So, collecting Amharic posts and comments from social media for training data is very helpful. Our model is concerned with not only classifying Amharic texts as hate or not-hate but also includes identification of what type of hate it is; religious hate, or ethnic hate, So the collected dataset should be from people of different genders, political ideologies, activists, nationalities, and working in different sectors. Therefore, by selecting different Facebook pages, which are created and controlled by various artists, politicians, government officials, activists, and religious peoples, their posts and comments are collected using the crawler.

Pseudo code: 4.1 Crawling dataset from Facebook
Input: (URL of the Facebook page)
Output: (dataset)
Begin: Step 1: Send a request to the selected Facebook pages Step 2: Copy and save the post Step 3: Extract links in the page Step 4: Send a request to the comment link Step 5: While (! end of comment page) Extract comments and save Step 5: Remove HTML tags
Return Page post and comments
END

Algorithm 5.1 dataset crawling

5.2.2. Annotation

After datasets are collected from different Facebook pages, the useful texts must be identified and annotated as hate or not-hate for training and testing data. But this must be done not by us, but by lawyers. As known in Ethiopia law is enacted to control social media hate speech and false information. So to decide whether a speech is hateful or not, it must be from the law point of view. Therefore, to make our data reliable and to produce

accurate results, after we collect the dataset from various Facebook pages, we give it to the lawyers to annotate it manually. This stage has two steps: the first step is done by the researcher and the second step is by the lawyers. In social media people express their different feelings on different issues, giving all collected posts and comments to the lawyers for annotation is wasting the time of lawyers because selecting hate and non-hate speeches from more than 400,000 datasets is a boring and time-consuming task. So primarily the researcher created a dictionary that contains a list of religious words, ethnic names, then posts and comments that contain those dictionary words are filtered for annotation. After that, the second stage is the task of the lawyers.

Pseudo code: 4.2 Selecting datasets for annotation
Input: (all dataset, list of properties(identities) and hate words in a dictionary)
Output: (selected dataset)
Begin: for i in dataset if datatataset[i] contains property then select dataset[i]
Return selected dataset
END

Algorithm 5.2 filtering dataset

5.2.3. Pre-processor

The dataset is collected from social media, so that it contains different numbers, non-Amharic characters, punctuations, and other unnecessary contents. Therefore, to have a cleaned dataset, unnecessary contents must be removed, and the same phoneme in different characters must be normalized. In this step, the unnecessary content is removed from the collected dataset, and texts which have the same meaning but are written in different forms are normalized.

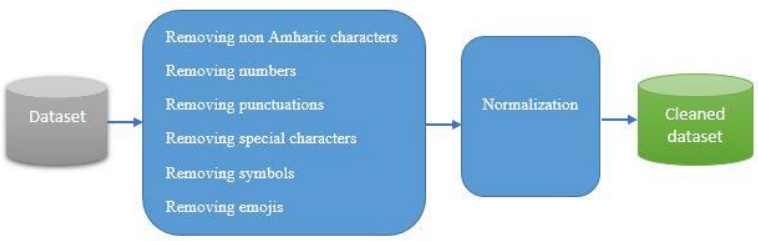

Figure 5.2 preprocessing steps

Unnecessary character remover: On social media, people express their thoughts not only in text but also in various images and symbols. They also write Amharic by mixing it with English and write Amharic in Latin script. Additionally, for a post in Amharic, they comment in English, Afan Oromoo, and other languages. So that to have a pure Amharic text the cleaner removes texts, special characters, punctuations, images, emojis, and other symbols except for Gee'z scripts.

Pseudo code: 4.3 cleaning dataset

Input: (dataset)

Output: (cleaned dataset)

Begin:

for i in dataset

if datataset[i] contains special_characters[~!$#@%^&*]

then remove special_characters

if datataset[i] contains symbols [፡።\"'()+,-./:;<=>?[\]^_`{|}~፡፡፣፤፥፦¡]

then remove symbols

if datataset[i] latin_words[A-Z|a-z]

then remove latin_words

if datataset[i] contains numbers [0-9| ፩|፪|፫|፬|፭|፮|፯|፰|፱|፲|፳|፴|፵|፶|፷|፸|፹|፺|፻|፼]

then remove numbers

if datataset[i] contains emojis[👊|👩‍🏭|😂🖐|😎|💀👉|😉|💪|💚]

then remove emojis

Return cleaned dataset

END

Algorithm 5.3 text cleaner

Normalizer: As we know Amharic is a language that uses two and more characters for the same phoneme, for example, all ሀ, ሃ,ሐ,ሓ,ኀ, and ኃ are pronounced the same "ha". For human beings, it is easy to identify that they have the same pronunciation but for computers, they are completely different. So to resolve these issues normalization is performed.

Pseudo code: 4.4 normalizing dataset

Input: (dataset)

Output: (normalized dataset)

Begin:

forms=8

haletaw = [ሀ, ሁ, ሂ, ሃ, ሄ, ህ, ሆ, ኀ]

hameru = [ሐ, ሑ, ሒ, ሓ, ሔ, ሕ, ሖ, ኋ]

bizuhan = [ኀ, ኁ, ኂ, ኃ, ኄ, ኅ, ኆ, ኋ]

esatu = [ሰ, ሱ, ሲ, ሳ, ሴ, ስ, ሶ, ሷ]

nigusu = [ሠ, ሡ, ሢ, ሣ, ሤ, ሥ, ሦ, ሧ]

tsaddik = [ጸ, ጹ, ጺ, ጻ, ጼ, ጽ, ጾ, ጿ]

tsehay = [ፀ, ፁ, ፂ, ፃ, ፄ, ፅ, ፆ, ፇ]

adam = [አ, ኡ, ኢ, ኣ, ኤ, እ, ኦ]

begeu = [ዐ, ዑ, ዒ, ዓ, ዔ, ዕ, ዖ]

ha = [ሃ, ሐ, ሀ, ሓ, ኀ, ኃ, ኻ]

for i=0 to forms

  if hameru[i] or bizuhan[i] in dataset

    then replace by haletaw[i]

  if nigusu[i] in dataset

    then replace by esatu[i]

  if begeu[i] in dataset

    then replace by adam[i]

  if tsehay[i] in dataset

    then replace by tsadik[i]

for i=0 to forms-1

  if begeu[i] in dataset

    then replace by adam[i]

  if ha[i] in dataset

    then replace by haletaw[0]

Return normalized dataset

END

Algorithm 5.4 Normalizer

5.2.4. Feature extractor

In this section, we discuss about metrics that make hate speech differ from other speeches. To extract necessary features of hate speech, the following methods are proposed.

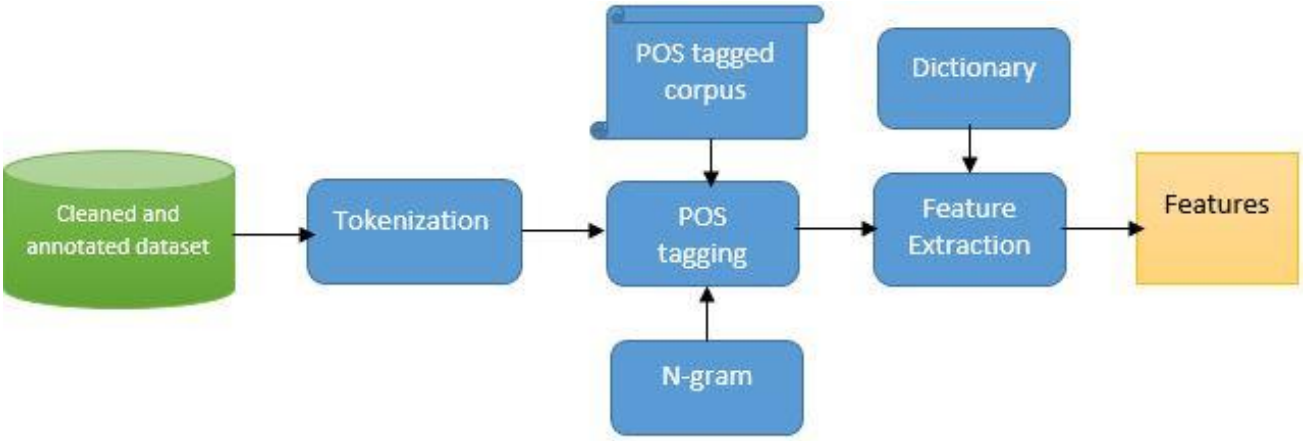

Figure 5.3 feature extraction steps

In this stage, it is necessary to identify features that make hate speech different from other speeches. Languages have their syntax and semantics, the word order that Amharic uses is subject, object, verb (SOV). For example, in the sentence "አበበ እንጀራ በላ", አበበ is a subject, እንጀራ is an object and በላ is a verb. Hate speech and any other speeches are not different by their word order or syntax, for example, a hate speech "ኦሮሞ የአማራ ጠላት ስለሆነ እጠላዋለሁ" and a love speech "ኦሮሞ የአማራ ሙከታ ስለሆነ እወደዋለሁ" are syntactically the same. If we look at the part of speech of the above two examples, it is the same. Therefore, we can't identify hate speeches from other speeches by their syntax. To identify hate speeches, it is mandatory to know the semantics of a sentence and the meaning of each word in its context.

According to the hate speech law passed in Ethiopia [2] "hate speech is a discourse that promotes hatred, discrimination or violence based on race, gender, ethnicity, religion, disability, on a particular person, and a specific group or community." So that It is important to know and store in a dictionary, who these nations are, who these peoples are, what are religions? What are the races? Types of disabilities? And Words Promoting Discrimination and Hate. After that, if the dictionary word occurred in the sentence or in the speech, then we extract and select the features. For example, if there is a word "ይጨፍጨፉ/yechefchefu" (meaning; let them massacre) in the speech, it has to be checked

who is supposed to be killed? Are trees being destroyed? Or are peoples being killed based on their identities and attitudes? And also for example, if there is a religious word "Muslim" then the context should be clarified in the speech, what the speaker says about Muslims.

In natural language processing, to identify the meaning of a word in the context, we have to consider the left and the right neighboring words by setting the window size to some number. So that to understand the context of those words, neighboring words are used by setting the window size to some number.

5.2.4.1. Dictionary

As tried to show in this research, hate speech often includes the name of the hated entity. This means, the hated race, religion, or other identity is mentioned in the text, it also includes words that denigrate, demean, hate, discriminate, or incite violence on the identity mentioned above. Therefore, it is important to identify and put into a dictionary, these ethnic, religious, and other types of identities mentioned in the law [2] including words that promote violence, discrimination, and hatred. Using the dictionary helps to verify the occurrence of these identities mentioned above in the speech. Getting these personal and collective identities using the dictionary, helps to determine whether or not the person or groups have experienced hate speech in the speech. If so it helps to determine what kind of hate it is, encouraging discrimination? Exposing to violence? And so on.

5.2.4.2. N-Gram

On social media, people express their opinions in different ways, some briefly describe what they mean in a few words, and others elaborate and write extensively. When a speech is considered hateful, it does not mean that all the words in it are hate speech. The author may include hate speech at the beginning, middle, or end of the article. An article can talk about love and other things and then hate in the middle. Or, conversely, it may begin with hatred and end with love or other conversations. So, it is important to identify only the part where hate is expressed. Processing the entire article to get the hate speech out of the long article is a wastage of time and computing resources. Therefore, when words listed in the dictionary occur in the speech, it is necessary to take the left and right words using N-gram. This helps to take the necessary sentence from it by ignoring the others.

Pseudo code: 4.4 getting left and right neighbor words

Input: (dataset)

Output: hate speech sentence

Begin:

```
for i in dataset
    for j in dictionary
        for word in j
         count_left=8
         count_right=0
         match=word
              if match exist in dataset[i]
                   while(count_left>0)
                       left[count_left]=word
                       count_left=count_left-1
                       swap(count_left)
                   while(!count_right=8)
                       right[count_right]=word
                       count_right= count_right + 1
    Reduced = left+match+right
```

Return Reduced dataset

END

Algorithm 5.5 Reducer

5.2.4.3. Part of speech (POS) tagging

According to Baye [37], the Amharic language has eight main parts of speech: Noun (“ስም”), Pronoun (“ተዉላጠ ስም”), Verb (“ግስ”), Adverb (“ተዉሳከግስ”), Conjunction (“መስተጻምር”), Adjective (“ቅጽል”), Preposition (“መስተዋድድ”) and, Interjection (“ቃለ አጋኖ”). These parts of speech are sorted in the sentence, according to the syntax. Using the part of speech listed above, we can find out who the talk is about and what message the speaker is conveying.

To identify hate speech in a conversation, first of all, it is important to know the entity that the speech is talking about then after the entity is identified, what is said about it? To do

this it is important to know the part of speech of each word in the sentence. For example, if there is a hate speech “አበበ ሞገደል አለበት” meaning “Abebe must be killed” and if we look at the parts of speech, “አበበ “is a Noun or an entity who is being talked about and “ሞገደል አለበት” is an adverb and verb or the action taken against the entity. Often the entities are included in the noun section and actions are included in the verb part of the speech.

In the dataset collected for hate speech detection, sometimes it is difficult to identify the entity who the sentence is talking about. Because the dataset is collected from different posts and comments, the entity is often not mentioned in the comment. For example, if someone posts "አበበን እወደዋለሁ" meaning “I like Abebe”, this will be stored in the dataset as one data. Next, if someone else comments “እኔ ግን አልወደዉም” meaning "but I don't like him", this also stored as another data in the dataset. In the first sentence, because it is mentioned that the beloved body is Abebe, the entity can be found easily. But in the second sentence, the phrase "I don't like him" Raises the question "Who?”. But, there is no answer to this question in the sentence because the speech is talking about some unloved thing without revealing who he is. So in such case, it is important to skip by knowing that there is an unloved entity called x or go to the main post and find out who that entity is.

When it comes to the action/verb of a speech, According to Baye [37], in Amharic language part of speeches verbs are actions like: “በላሁ”/I ate, “ተቀመጠ”/he sat down, “መጣሽ”/you came. But when we see the English language verb, it contains not only actions but also states. For example, in the speech "እኔ ተማሪ ነኝ" meaning “I am a student” there is no action, but a state. The speech is about being a student. So the word taken as a verb is am. In this study to facilitate the identification and classification of hate speech contextually, the part of speech of the collected datasets is tagged by language experts.

According to Baye [37], It is a combination of nouns and verbs that characterize things and actions in Amharic sentences, the role of the other parts of speeches is accompanying. So, to identify hate speech, we need to focus on these two parts of speeches. Once a Noun is found in a sentence, the words that come before to express or specify the Noun tagged as an adjective. The words that came before the verb to describe how the action took place, tagged as an adverb. Next, the word that expresses the action is tagged as a verb, and if the speech describes not an action but a situation, the word that describes the situation is

tagged as a verb. For example, if there is a speech,”ጥቁር ረጅም ሰዉ በጣም እፈራለሁ” meaning "I'm very scared of a tall black man." if we look at these words one by one separately, “black” is a Noun because it is a type of color, "tall" is also a Noun because it is a type of height and “Man” is also a Noun. But "Very" is not a Noun because it does not stand alone or represent things so it is an adverb or action describer. The word "scared" refers to action, so we take it as a verb. But if we look at their context as a sentence The Noun “red” and “tall” are entered to describe the noun and to answer the question “what kind of person?” so tagged in the adjective part of speech section. The word "very" answers the question "How scared are you?" and describes the action. So, it is tagged as an adverb.

Mixed tag types are also included, this means that it is one word and has two parts of speech. For example, the action in the example above is “ፈራሁ” meaning "I'm afraid." Although it speaks of the act of fear, but it has the pronoun “I”. If the word is "እኀፈራለኀ" meaning "we afraid", it will have the pronoun "we". Therefore, for such words, a mixed part of speech “V_PR” called “VERB_PRONOUN” is included. Generally, part of speech tagging is very important to know the role of a word in the given text contextually.

5.2.5. Feature selector

It is known that Amharic has eight main parts of speeches. To extract hate speech features, deeply look at these parts of speech, and knowing their role in the speech is very necessary. To identify hate speech in a speech contextually, it is necessary to know which part of speech is better useful. In this section, we will look at the part of speeches that is useful for hate speech feature selection. The discussion below answers the question; which part of speech has a better role for hate speech detection.

- **Interjection**: The use of interjection in Amharic speech is to express a feeling, but hate speech cannot be detected using Wow, Ouch… words only. Because a speech to be hate speech, it must contain the name of the hated person or group with the type of hatred, discrimination, or violence.
- **Preposition and Conjunction:** these parts of speech are used for connecting words and describing the situations, but not used for identifying whether the speech is hateful or not.

- **Noun and Pronoun:** these parts of speeches used to identify the targeted person or group, after identifying the type of hatred, discrimination, or violence, it used to identify the hated entity.

According to Baye [37], most Amharic sentences are a combination of nouns and verbs, which are manifestations of things and actions. The other parts of speeches accompanies the two words. So while these two parts of speeches are useful for identifying hate speech, we also need to add part of speech (Adjective and Adverb), that gives us information about these two parts of speeches. In one sentence, hate speech can be expressed in the following three ways

1. **Adjective:** when the noun is expressed inappropriately. For example, if the speech is "አሸባሪዉ ብልጽግና ፓርቲ" meaning "the terrorist prosperity party" the adverb "አሸባሪዉ" expressed the entity "prosperity party" hatefully. If we change the adjective "አሸባሪዉ" by "ተወዳጁ" meaning "the lovable" the speech will be normal speech. So that adjective has a role in detecting hate speech.
2. **Verb:** when the action or the state contains a hate word. For example, if the speech is "አማርኛ ተናጋሪዎችን እጠላለሁ" meaning "I hate Amharic speakers", the speech that makes this sentence hateful is, the verb "እጠላለሁ" meaning "I hate". Let us take another example "ትግሬ ሌባ ነዉ" meaning "Tigre is a thief" the verb in this example is "ነዉ" which shows a state, being a thief or not. But, if we change the verb by the word"አይደለም" the speech will change from hate speech to normal speech. So that a verb can play a great role in making a sentence hateful or hateless.
3. **Adverb:** when the adverb contains hate words. For example, "አበበ ገዳይ ነዉ" meaning "Abebe is a killer", the example above is about being or not being a killer, the adverb" ገዳይ" tells what is to be, then the verb" ነዉ" decides whether he is a killer or not. If we change the adverb by the word "ተወዳጅ" meaning "lovable" the speech can't be hate speech because the sentence will be changed to "Abebe is lovable". Therefore, an adverb has a great role in making speeches hateful or hateless.

5.2.6. Context table

In this section, we discuss how features are selected and stored in the context table. After understanding the context of the sentence, features are extracted using the newly developed feature extraction method. Then features of each sentence are stored in the table in separate rows. As discussed above, hate speech occurs in the sentence in three forms. The first one is if the noun is the name of religions, races, ethnicity, gender, and disability and the adjective is hateful. For example, if there is a speech "ጨፍጫፊዉ አማራ" the noun is "አማራ" which is an ethnic name, and "ጨፍጫፊዉ" meaning "the killer" is an adjective that makes the speech hateful. So that the adjective value of the context table for this sentence will be 1. In this way, the features are stored in the table. The following algorithm is created for the extraction of features and storing the value in the table.

Pseudo code: 4.5 feature extractor

Input: (Part of speech(POS) tagged dataset)

Output: Feature table

Begin:

```
for sentence in taged_dataset
  for i to length_of(tagged_dataset)
    token=tokenize(tagged_dataset[i])
    length_of_each_dataset=length_of(token)
    for j to length_of_each_dataset
////////////////////////////// Condition 1 ////////////////////////////////////////////////////////
      if(part_of_speech(token[i]) ==”ADJ” && token[i] in list_of_hatered)
        for k=j to length_of_each_dataset
          if((part_of_speech(token[i]) ==”ADJ” | “CONJ”)
            continue
          elseif (part_of_speech(token[i]) ==”NN” | “NN_OBJ” | “NN_SUB” | “NN_CONJ”)
          and (token[i] in list_of_ethnicities |list_of_races | list_of_religions | list_of_genders)
            table[i][1]=1
            break
          else:
             break
```

```
////////////////////////// Condition 2 ////////////////////////////////////////////////
    elseif(part_of_speech(token[i])==”VB” | “VB_CONJ” && token[i] in list_of_ attack)
      for k=j to 0
       if (part_of_speech(token[i]) ==”NN” | “NN_OBJ” | “NN_SUB” | “NN_CONJ”)
       and (token[i] in list_of_ethnicities |list_of_races | list_of_religions | list_of_genders)
         table[i][2]=1
         break
       else
          break
/////////////////////Condition 3/////////////////////////////////////////////////
    elseif(part_of_speech(token[i])==”VB” | “VB_CONJ” && token[i] in list_of_ hatred)
      for k=j to 0
       if (part_of_speech(token[i]) ==”NN” | “NN_OBJ” | “NN_SUB” | “NN_CONJ”)
       and (token[i] in list_of_ethnicities |list_of_races | list_of_religions | list_of_genders)
         table[i][3]=1
       elseif (part_of_speech(token[i]) ==”NN” | “NN_OBJ” | “NN_SUB” | “NN_CONJ”)
       and (token[i] not in list_of_ethnicities |list_of_races | list_of_religions |
       list_of_genders)
         break
       else
         continue
      for k=j to length_of_each_dataset
       if((part_of_speech(token[i]) ==”VB”  && token[i] in list_of_positive_state)
         table[i][4]=1
         break
       elseif((part_of_speech(token[i]) ==”VB”  && token[i] in list_of_ negative _state)
         table[i][5]=1
```

Return Feature value of a given text

END

Algorithm 5.6 new feature extraction method

5.2.7. Classifier

Classification is performed using three supervised machine learning models: support vector machine (SVM), naïve Bayes (NB), and random forest (RF) which is discussed in chapter four. During the classification, a vector of features are given to those three machine learning models. So to explore a better combination of classifier and feature extraction methods, 9 experiments are done by testing each machine learning model with three feature extraction methods. The feature extraction methods are TF-IDF, Bag of Words (BOW), and the newly developed feature extraction method NFEM. TF-IDF and BOW are used to compare with the newly developed feature extraction method and to identify the better feature extraction mechanism. So that the study shows that, it comes up with a better feature extraction method.

5.2.8. Result evaluator

The result evaluator measures the performance of the model using different machine learning evaluation metrics. For evaluation different machine learning model evaluation metrics like precision, recall, F1-score, accuracy, k-fold cross-validation are used, which is discussed in chapter four.

CHAPTER SIX

6. Implementation and Result

Using the methodology discussed in chapter four and the proposed model presented in chapter five, implementation is performed to show the efficiency and effectiveness of the proposed Amharic hate speech detection feature extraction method. This chapter presents how the model is implemented and the achieved result of the experiment. Generally, the presented sub-sections in this chapter are development environment, dataset description, preprocessing implementation, feature extraction implementation: including the newly developed feature extraction method, machine learning algorithm implementation, model evaluation, and result of each subsection.

6.1. Implementation

This study follows a design science research methodology so that it is necessary to design and implement a model for the specified problem. This section shows the steps and procedures of the implementation of the proposed Amharic hate speech detection model.

6.1.1. Implementation environment

This study used python as a programming language and anaconda navigator as a tool for implementation, testing, and developing a prototype. Python is selected because it contains several packages for natural language processing, and the familiarity of the researcher with python programming language is considered. Python contains several packages that help for preprocessing, feature extraction, classification, and evaluation of the model. The following tools and packages are used for experimenting and evaluating the result of the proposed model.

Table 6.1 tools used for implementation

No	Tools	Version	Purpose
1	Anaconda navigator	1.10.0	To launch, manage, and update different development tools. And to install and upgrade python packages.
2	Jupyter notebook	6.0.1	To build a model by preprocessing the dataset,

			applying feature extraction methods, implementing classification algorithms, and evaluating the model using the python programming language
3	Microsoft excel	2013	To hold the collected Facebook dataset in an organized manner and to make it ready for the next step by removing duplications and sorting. And also used for annotation.

Table 6.2 packages used for implementation

No	Packages	Version	Purpose
1	Natural language toolkit (NLTK)	3.4.5	Used for tokenizing words, reading tagged corpora, and tagging part of speech.
2	Regular expression(RE)	2.2.1	Used for preprocessing like replacing and removing unwanted characters and emojis.
3	Pandas	0.25.1	To read the excel dataset and to work with it.
4	Numpy	1.16.5	To convert a vector of dataset into array
5	Sklearn	0.21.3	To implement the classification algorithms and feature extraction method, evaluate the performance of the model.
6	Matplotlib	3.1.1	To visualize the performance of the model and the distribution of the dataset in pictorial format
7	Python	3.7.4	Used as a programing language for development
8	Tkinter	8.6.10	To develop a graphical user interface(GUI) based application for the final model

6.1.2. Data source

The dataset is collected from 25 different Facebook pages. These pages are created and used by news and broadcast agencies, security and defense agencies, government officials, religious institutions and preachers, law enforcement agencies, agencies which work for peace and human right, government officials, artists, activists, and popular persons, to

share their messages to their followers. The selected pages have more than 75,000 followers and like with more than 3.5 rates and most of the time they use Amharic language for writing their posts. Primarily 487,231 datasets are collected from these pages blindly without identifying whether the message is hateful or not.

As discussed in chapter five, hate speech must contain at least one hateful words that promote hatred, discrimination, or attack, and the hated person or group. So that a dictionary is created which contains the list of religious words, list of ethnic group and races, list of disabilities, list of genders, list of words that promote hatred, discrimination, and attack. By writing the following python code dataset, which contains words in the dictionary, are filtered and categorized in a folder. So that this makes the annotation process easy.

```
for i in files:
    name_only=os.path.splitext(i)[0]  #removing the extension
    #creating folder and naming the category
    directory=os.path.join("C:/Users/dellll/Desktop/ds/excel1/",name_only)
    if not os.path.exists(directory):
        os.mkdir(directory)

    PATH = ("C:/Users/dellll/Desktop/ds/excel1/"+i)
    dataset = pandas.read_excel(PATH, sheet_name='Sheet1') #reading the dataset
    for r in criteria:
        in_it=dataset['message'].str.contains(r,na=False) #selecting datasets that contains the criteria
        filtered=dataset.loc[in_it]
        filtered.to_excel("C:/Users/dellll/Desktop/ds/excel1/"+name_only+"/"+r+'.xlsx') #writing the selecting dataset to excel
```

Sample code 6.1 filtering the dataset

After the dataset is collected from 25 different Facebook pages, it is stored separately in excel file format by giving it the page name as a file name. The above code creates a folder for each excel file by taking the file name as a folder name. then, by using the dictionary words, each file collected from the selected Facebook page classified into specific groups like ethnic: አማራ, ኦሮሞ, ጉራጌ,… religious: ክርስቲያን, ሙስሊም,…. , gender: ወንድ, ሴት, and disability: አይነ-ስዉር, መስማት የተሳናቸዉ.... Finally stored in the created folder. This helps to count the dataset by group and type.

The dataset contains the following three attributes

- **Message:** It is a post, or a comment given to the post
- **Created_time:** Creation date_and_time of the message
- **Object_id**: the given number of a message, to identify uniquely.

- **Parent id**: the id given to the comments to identify, for which post the comment given
- **Label**: the class of the message assigned by the annotator.

6.1.3. Defining classes

For classification and detection, it is necessary to know the class types, meaning, and differences. Even if Ethiopian hate speech law [2] considers gender and disability hate as hate speech, due to lack of dataset this study doesn't include it in the class., the datasets are classified into the following three classes:

- ✓ **Religious-hate**: this is a speech that contains words that promote attack, discrimination, or hatred on a person or groups based on his/her/their religion.
- ✓ **Ethnic-hate**: this is a speech that contains words that promote attack, discrimination, or hatred on a person or groups based on his/her/their race or ethnicity.
- ✓ **Non-hate**: this is a speech that does not contain words that promote attack, discrimination, or hatred based on personal and collective identities

6.1.4. Annotation

The filtered dataset with the annotation instruction is given to the law experts to categorize the speeches into their classes. After filtering the collected dataset 69,337 datasets are given for two annotators.

6.1.5. Pre-processing implementation

To prepare the dataset for training and testing, unnecessary content has to be cleaned and repeated characters have to be normalized. First, the annotated excel format datasets are loaded using pandas python package. To load the dataset, the path that shows the location of a file, and the sheet name is given to the pandas read_excel library as an argument, two variables named "labels" and "post_and_comments" are created to hold the class name and the text as shown in the sample code below.

```
PATH = "C:/Users/dellll/Desktop/labeled dataset.xlsx"
filtered_dataset = pandas.read_excel(PATH, sheet_name='Sheet1')
filtered_dataset = shuffle(filtered_dataset)
labels = filtered_dataset.iloc[:, 0]
posts_and_comments = filtered_dataset.iloc[:, 1]
```

Sample code 6.2 Loading the dataset

After loading the dataset, using regular expression (re) package, the post and comments are cleaned. The cleaning includes removing Latin alphabets, numbers, emojis, punctuations, and extra white spaces using the following function.

```
def cleaning_text(posts_and_comments):
    ### Removing latin alphabets
    posts_and_comments = [re.sub(r"[A-Za-z]", "", str(post)) for post in posts_and_comments]
    ### Removing Numbers
    posts_and_comments = [re.sub("[0-9]", "", str(post)) for post in posts_and_comments]
    ### Removing emojis
    posts_and_comments = [re.sub("♥|💐|👨|👍|🌺|😊|😡|😊|👇|😰|😟|❗|😕|😠|💰|🍒|😔|😞|✅|👈|😑|👌|😈|😕|🔫
    ### Remove Punctuations
    punctuations = "፨!\"#$%&'()*+,-./:;<=>?@[\]^_`{|}~:¤፣÷¡"
    posts_and_comments = [str(post).translate(str.maketrans('', '', punctuations)) for post in posts_and_comments]
    ### Remove extra white space
    posts_and_comments = [(" ".join([ single for single in str(post).split()])) for post in posts_and_comments]
    return posts_and_comments
```

Sample code 6.3 Cleaning text

Regular expression (re) package is used also to normalize characters which has the same phoneme using the following python sample code.

```
def normalization(posts_and_comments)
    posts_and_comments = [re.sub("ሐ|ሃ|ሓ|ኀ", "ሀ", str(post))for post in posts_and_comments]
    posts_and_comments = [re.sub("ሑ|ኁ",     "ሁ", str(post))for post in posts_and_comments]
    posts_and_comments = [re.sub("ሒ|ኂ",     "ሂ", str(post))for post in posts_and_comments]
    posts_and_comments = [re.sub("ሔ|ኄ",     "ሄ", str(post))for post in posts_and_comments]
    posts_and_comments = [re.sub("ሕ|ኅ",     "ህ", str(post))for post in posts_and_comments]
    posts_and_comments = [re.sub("ኅ|ኆ",     "ሆ", str(post))for post in posts_and_comments]

    posts_and_comments = [re.sub("ሠ","ሰ", str(post))for post in posts_and_comments]
    posts_and_comments = [re.sub("ሡ","ሱ", str(post))for post in posts_and_comments]
```

Sample code 6.4 Normalization

6.1.6. Feature extraction implementation

As discussed in chapter four and five, this study proposed a new feature extraction method for Amharic hate speech detection, to build a better model, and to achieve a better result in detection and classification. To compare and contrast the new feature extraction method with other methods, the previous feature extraction methods, which are used by other related studies, are also implemented. The first task of the proposed feature extraction method is tagging the part of speech of each word in the post and comments using tagged corpora.

```
for i in range(len(posts_and_comments)):
    words=nltk.word_tokenize(posts_and_comments[i])
    tagging = bigram_tagger.tag(words)
```

Sample code 6.5 Part of speech(POS) tagging

After tagging, each word in the text is checked for its occurrence in the dictionary, and its part of speech is identified to know the role of each word in the sentence contextually. Primarily, the method checks the occurrence of the entity (the hated person or group) using the created dictionary and part of speech, then checks the adjectives that give information about a noun whether it is a hate word or not. The second and the third task of this method is checking the occurrence of hate words as a verb or as an adverb as discussed in chapter five. Finally, features are extracted and saved in excel format for later use. The figure below shows a sample python code of the new feature extraction method implementation.

```
for j in range(leng):
    if((q[j][0] in list_of_ethnicities) or (q[j][0] in list_of_races) or (q[j][0] in list_of_religions) or (q[j][0] in list_of_g
        m.cell(row=b+2, column=3).value=1
        if((q[j][0] in list_of_ethnicities) or (q[j][0] in list_of_races) or (q[j][0] in list_of_informal_names)):
            m.cell(row=b+2, column=4).value=2
        elif((q[j][0] in list_of_religions)):
            m.cell(row=b+2, column=4).value=1
        elif(q[j][0] in list_of_genders):
            m.cell(row=b+2, column=4).value=3
    elif(((q[j][0] in list_of_hatred) or (q[j][0] in list_of_discrimination) or (q[j][0] in list_of_attack))):
        m.cell(row=b+2, column=5).value=1
    if((q[j][1]=='ADJ') and (q[j][0] in list_of_hatred )):
        for x in range(j,leng):
            if((q[x][1]=='ADJ') or (q[x][1]=='CONJ')):
                continue
            if(((q[x][1]=='NN') or (q[x][1]=='NN_OBJ') or (q[x][1]=='NN_SUB') or (q[x][1]=='NN_CONJ')) and ((q[x][0] in list_of_
                m.cell(row=b+2, column=6).value=1
            else:
                break
    if(((q[j][1]=='VB') or (q[j][1]=='VB_CONJ')) and (q[j][0] in list_of_attack)):
        for y in range(j,1,-1):
            if(((q[y-1][1]=='NN') or (q[y-1][1]=='NN_OBJ') or (q[y-1][1]=='NN_SUB') or (q[y-1][1]=='NN_CONJ')) and ((q[y-1][0] i
                m.cell(row=b+2, column=7).value=1
            elif(((q[y-1][1]=='NN') or (q[y-1][1]=='NN_OBJ') or (q[y-1][1]=='NN_SUB') or (q[y-1][1]=='NN_CONJ')) and ((q[y-1][0]
                break
            else:
                continue
    if(((q[j][1]=='ADV') or (q[j][1]=='ADV_CONJ')) and (q[j][0] in list_of_hatred)):
        for z in range(j,1,-1):
            if(((q[z-1][1]=='NN') or (q[z-1][1]=='NN_OBJ') or (q[z-1][1]=='NN_SUB') or (q[z-1][1]=='NN_CONJ')) and ((q[z-1][0] i
```

Sample code 6.6 Proposed feature extraction method

Other feature extraction methods that are used in related works are also implemented by importing the sklearn python package to compare with the new feature extraction method.

```
from sklearn.feature_extraction.text import CountVectorizer
to_array=np.asarray(posts_and_comments)
count_vectorizer = CountVectorizer(max_features=1000)
vectorized_ds = count_vectorizer.fit_transform(to_array).toarray()
```

Sample code 6.7 Bag of words(BOW) vectorizer

```
from sklearn.feature_extraction.text import TfidfVectorizer
to_array=np.asarray(posts_and_comments)
count_vectorizer = TfidfVectorizer(max_features=1000)
vectorized_ds = count_vectorizer.fit_transform(to_array).toarray()
```

Sample code 6.8 TF-IDF vectorizer

There is no published hate speech dataset so that the dataset is collected from scratch. After collection, annotation, and part of speech tagging is another task. Due to the limitation of time and budget, the number of the collected dataset is not good enough. Therefore, 80% of the datasets are used for training, and the remaining 20% used for testing as follows.

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(vectorized_ds, y, test_size=0.2, random_state=4)
```

Sample code 6.9 Splitting the dataset

6.1.7. Classification algorithm implementation

Three supervised machine learning algorithms are selected for classification. These algorithms are selected by considering their good performance in the related works [13] [14] [62] [25] [68] [70] [71]. The implementation is performed by importing the necessary python packages like sklearn. Then, the training and testing datasets are given to those selected supervised machine learning classification algorithms. The reason for implementing three classification algorithms is to make a comparison between them and to select the best one for building the final model. The sample code is shown in the following figure.

```
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train, y_train)
y_predict=classifier.predict(X_test)
```

Sample code 6.10 naïve bayes classifier implementation

```
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators=1000, random_state=0)
classifier.fit(X_train, y_train)
y_predict=classifier.predict(X_test)
```

Sample code 6.11 Random forest classifier implementation

```
from sklearn.svm import SVC
classifier = SVC(kernel='linear')
classifier.fit(X_train, y_train)
y_predict=classifier.predict(X_test)
```

Sample code 6.12 Support vector machine classifier implementation

6.1.8. Performance evaluator implementation

To measure the performance of the proposed model, different machine learning model evaluation metrics are implemented. For the implementation, several python packages are imported and used. The models are measured in terms of confusion matrix, precision, recall, f1-score, accuracy, and k-fold cross-validation as discussed in chapter four. The following figure shows the sample code of the implementation.

```
from sklearn.metrics import classification_report,confusion_matrix, accuracy_score
from sklearn.metrics import precision_score,recall_score
from sklearn.model_selection import cross_val_score
from sklearn.metrics import f1_score
conf_mat=confusion_matrix(y_test, y_predict)
report=classification_report(y_test, y_predict)
accuracy=accuracy_score(y_test, y_predict)
cross_val=cross_val_score(classifier,X_train, y_train,cv=5)
```

Sample code 6.13 Performance evaluator implementation

6.2. Result and discussion

To measure the performance of the created feature extraction method and to identify the best fit, implementation is performed with three machine learning classifiers. Other two feature extraction methods are also implemented to compare with the new feature extraction method. This section presents the outcome of each machine learning algorithm when combined with each feature extraction method. So that this allows to make a comparison between models and to select the better model. Also, the results from data collection to model evaluation are presented.

6.2.1. Dataset filtering and annotation result

From 25 public Facebook pages 487,231 posts and comments are extracted blindly without ensuring whether it is hate speech or not. Using a list of words contains hate words, and personal and collective identity store in a dictionary, a filtering method is developed. Using

the developed filtering method 69,337 datasets are extracted from the whole dataset. The following figure shows data distribution after filtration.

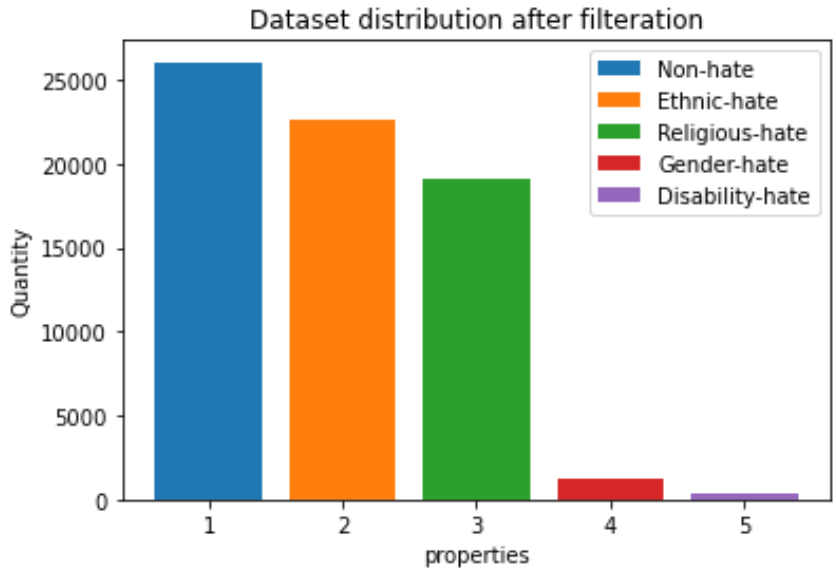

Figure 6.1 dataset distribution after filtration

The filtered dataset with annotation instruction is given to two law experts. Then they annotated manually based on the law [2]enacted in Ethiopia. The following table shows the annotation result of the filtered dataset.

Table 6.3 Annotation Report

No	Class	Annotator 1	Annotator2
1	Religion-hate	320	563
2	Ethnic-Hate	410	468
3	Gender-hate	84	60
4	Disability-hate	17	32
5	Non-hate	489	340
Sub-total		1320	1463
Total		**2783**	

To prevent unbalanced dataset effect, the three classes which have a top number of dataset are taken. Totally, 2590 labeled datasets are used. Among those 2590 datasets, 883 are labeled as Religion-hate, 878 are labeled as Ethnic-hate, and the rest 829 labeled as Non-hate.

6.2.2. Hate speech detection model evaluation result

After a feature is extracted using both the new feature extraction method and the previous method, features are given to the selected machine learning algorithms to perform the classification. After classification is performed, performance is evaluated using the following metrics.

6.2.2.1. Confusion matrix

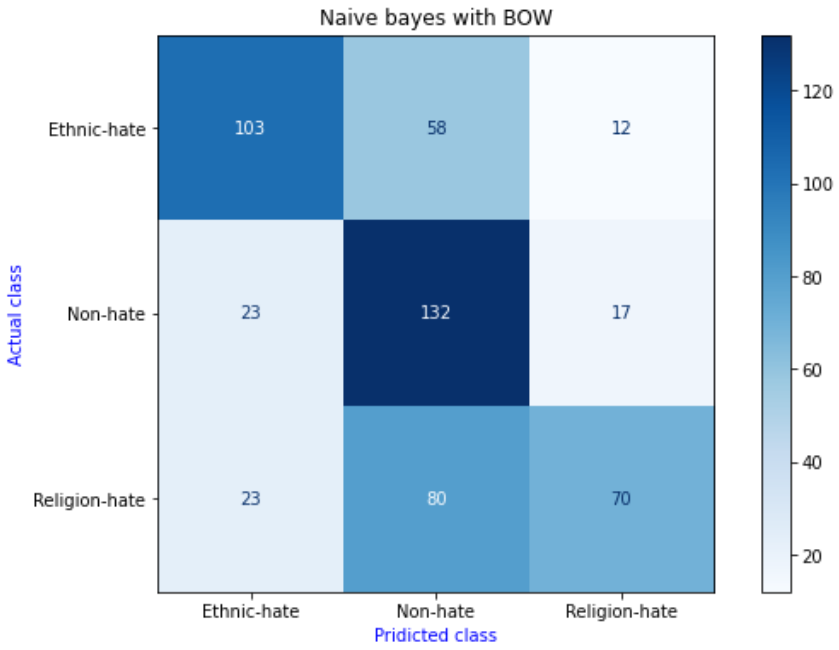

Figure 6.2 Confusion matrix of naive Bayes classifier using BOW

The above *figure 6.2* shows the confusion matrix of naïve Bayes classifier when combined with (BoW) feature extraction method. From 173 ethnic-hate datasets, 59.5% are correctly classified and the rest wrongly classified as Non-hate and Religion hate. Out of 172 Non-hate datasets 76.7% datasets are classified correctly and the remaining 23.3% datasets are wrongly classified to other classes. For testing the Religion-hate class 173 datasets are given then 40.4% datasets are classified correctly and 59.6% are wrongly classified to other classes. The report shows that Non-hate datasets are classified better than other classes. Generally, 58.8% of datasets are correctly classified from all given training sets.

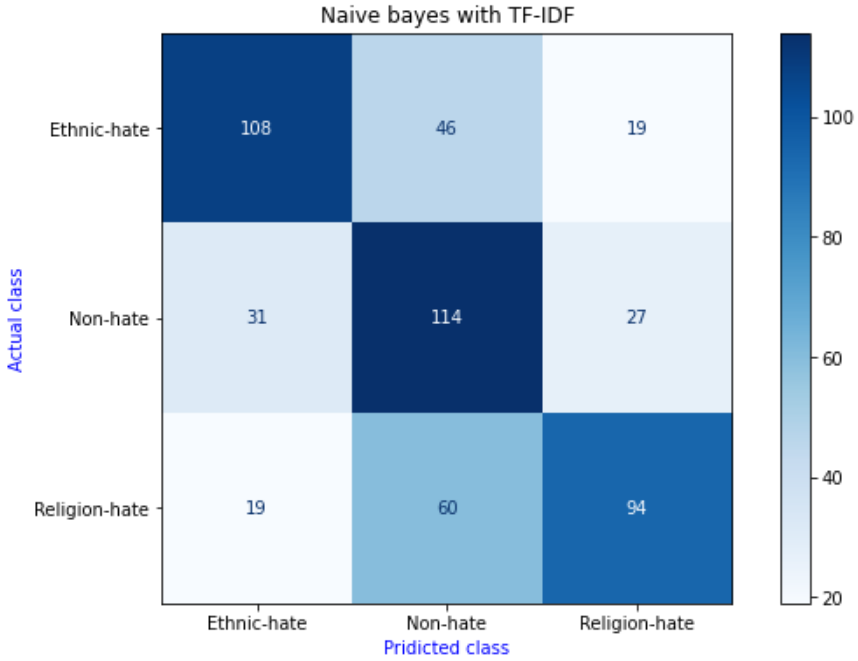

Figure 6.3 Confusion matrix of naive Bayes classifier using TF-IDF

The above figure 6.3 shows the confusion matrix of Naïve Bayes classifier when TF-IDF is used as a feature extraction method. From 173 ethnic-hate datasets, 62.4% are correctly classified as Ethnic-hate and the rest are wrongly classified to Non-hate and Religion-hate classes. Out of 172 Non-hate datasets 66.2% datasets are classified correctly and the remaining 33.8% datasets are wrongly classified to the rest classes. For classifying the Religion-hate class 173 datasets are given then 54.3% are classified correctly and 29.2% of the dataset wrongly classified to other classes. This report shows that Non-hate datasets are classified better than other classes. Generally, 61% of datasets are correctly classified from all given training sets. When compared with BOW feature extraction method, this showed a 2.2% higher result.

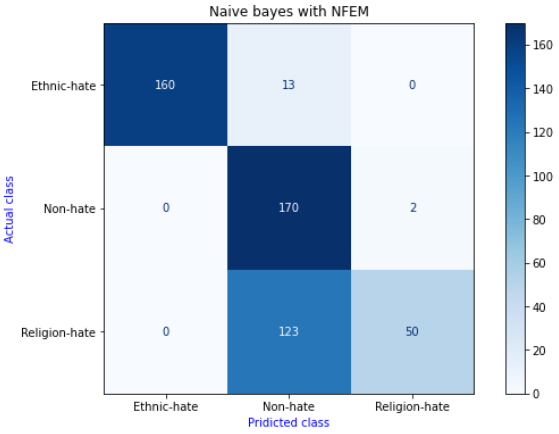

Figure 6.4 Confusion matrix of naive Bayes classifier using NFEM

The above *figure 6.4* shows the confusion matrix of Naïve Bayes classifier when using NFEM as a feature extraction method. From 173 ethnic-hate datasets, 92.4% are correctly classified and the rest wrongly classified to other classes. Out of 172 Non-hate datasets, 98.8% of datasets are classified correctly. From 173 given Religion-hate datasets, 71% are classified correctly and only 29% of the dataset wrongly classified to other classes.

As seen in the report, in this feature extraction method, non-hate datasets are classified better than other classes. Generally, 73.3% of datasets are correctly classified from all given training sets. This shows, in this classifier, the NFEM feature extraction method performs better than the other two feature extraction methods.

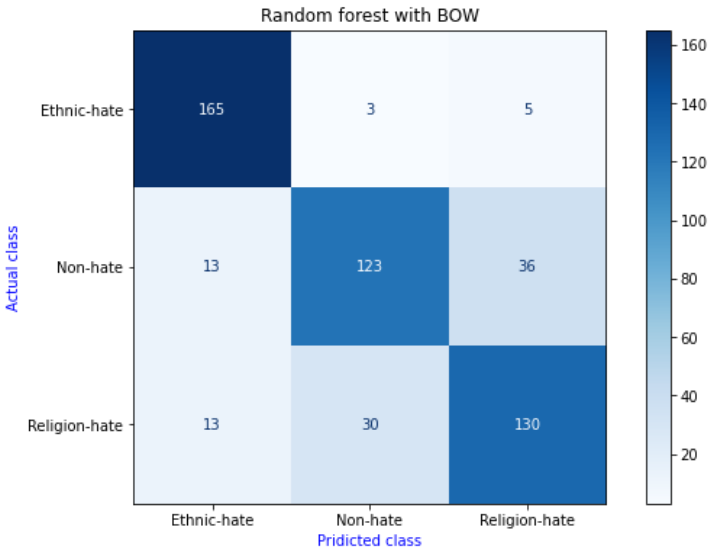

Figure 6.5 Confusion matrix of random forest classifier using BOW

Figure 6.5 shows the confusion matrix of random forest classifier with bag of words (BoW) feature extraction method. Out of 173 ethnic-hate datasets, 95.3% datasets are correctly classified and the rest 4.7% are wrongly classified as Non-hate and Religion hate. Out of 172 Non-hate datasets, 71.5% of datasets are classified correctly and the remaining datasets are wrongly classified to other classes. For testing the Religion-hate class 173 datasets are given, then 75.1% of datasets are classified correctly and 24.9 are wrongly classified to other classes.

The report shows that, in this combination, Ethnic-hate datasets are classified better than other classes. Generally, 80.6% of the given datasets are correctly classified to the desired classes. BoW feature extraction method showed better performance on this classifier than Naïve Bayes.

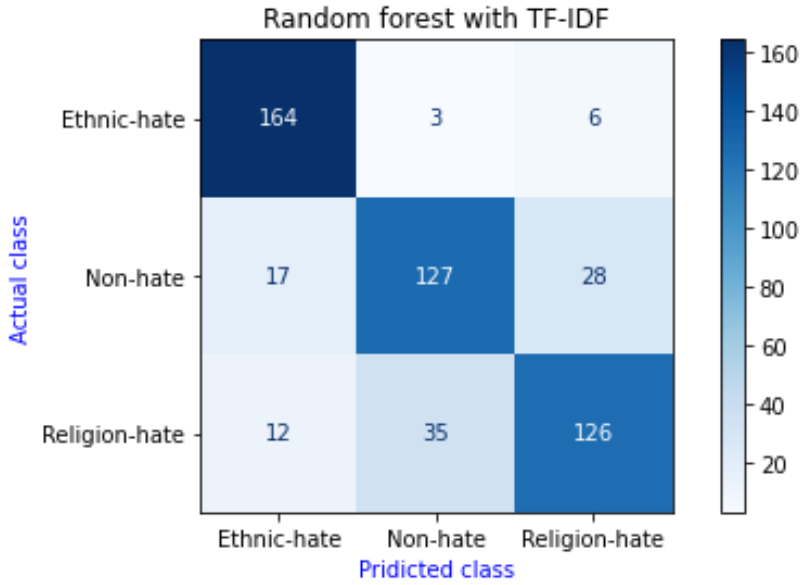

Figure 6.6 Confusion matrix of random forest classifier using TF-IDF

The above Figure 6.6 shows the confusion matrix of random forest classifier when using TF-IDF as a feature extraction method. From 173 ethnic-hate datasets, 94.7% are correctly classified and the rest 5.3% of the datasets are wrongly classified to other classes. Out of 172 Non-hate datasets, 73.8% of datasets are classified correctly and the remaining 26.8% of datasets are wrongly classified into religion-hate and ethnic-hate classes. For classifying the Religion-hate class 173 datasets are given then 72.8% classified correctly and the rest datasets are wrongly classified to other classes. Generally, 80.5% of all training datasets are correctly classified. TF-IDF performed better when combined with random forest classifier than Naïve Bayes.

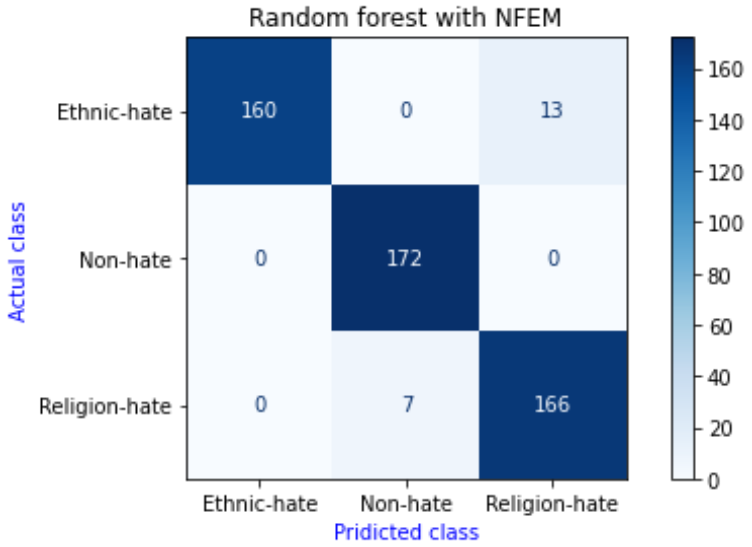

Figure 6.7 Confusion matrix of random forest classifier using NFEM

The above *Figure 6.7* shows the confusion matrix of random forest classifier when using NFEM as a feature extraction method. For testing 173 Ethnic-hate datasets are given then 92.4% are correctly classified, and the rest wrongly classified to Non-hate and Religion hate classes. Out of 172 Non-hate datasets, 100% of datasets are classified correctly. For classifying the Religion-hate class 173 datasets are given then 95.9% of the datasets are classified correctly and only 4.1% of the datasets are wrongly classified to ethnic-hate and non-hate classes. As seen on the confusion matrix report, non-hate datasets are classified better than other classes which is 100%. Generally, 96.1% of the given datasets are correctly classified. This shows, NFEM feature extraction method performs better than other tested feature extraction methods with random forest classifier and also with Naïve Bayes.

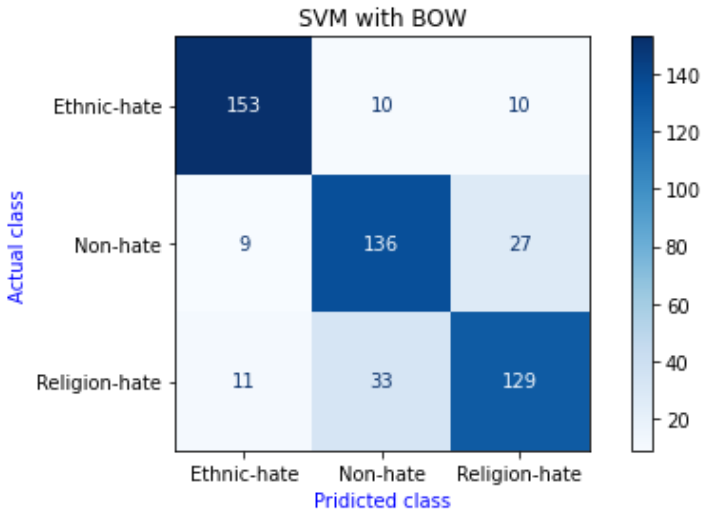

Figure 6.8 Confusion matrix of SVM classifier using BOW

The above figure 6.8 shows the confusion matrix of SVM classifier when using bag of words (BoW) as a feature extraction method. From the given 173 ethnic-hate datasets, 88.4% of datasets are correctly classified, and the rest classified wrongly as Non-hate and Religion hate. Out of 172 Non-hate datasets, 79% of the datasets are classified correctly and the remaining 21% of the datasets are wrongly classified to other classes. For testing the Religion-hate class 172 datasets are given then 74.5 % datasets are classified correctly and 25.5% of the datasets are wrongly classified to other classes. The report shows that the Ethnic-hate datasets are classified better than other classes. Generally, 80.6% of the given training sets are correctly classified. BoW feature extraction method showed better performance on SVM classifier than other implemented two classifiers.

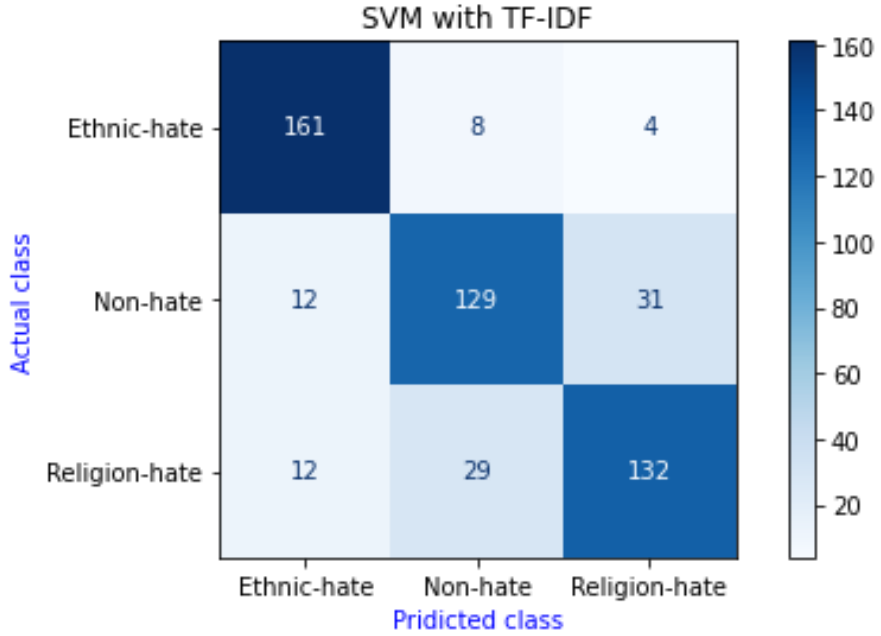

Figure 6.9 Confusion matrix of SVM classifier using TF-IDF

The above *figure 6.9* shows the confusion matrix of the SVM classifier when TF-IDF is used as a feature extraction method. From the given 173 ethnic-hate datasets, 93% of datasets are correctly classified and the rest wrongly classified into Non-hate and Religion-hate classes. Out of 172 Non-hate datasets, 75% of the datasets are classified correctly and the remaining 25% datasets are wrongly classified to other classes. For classifying the Religion-hate class 173 datasets are given then 76.3% classified correctly and the remaining classified wrongly to other classes. Generally, 81.4% of all given training datasets are correctly classified. TF-IDF performed better in this classifier than other implemented classifiers.

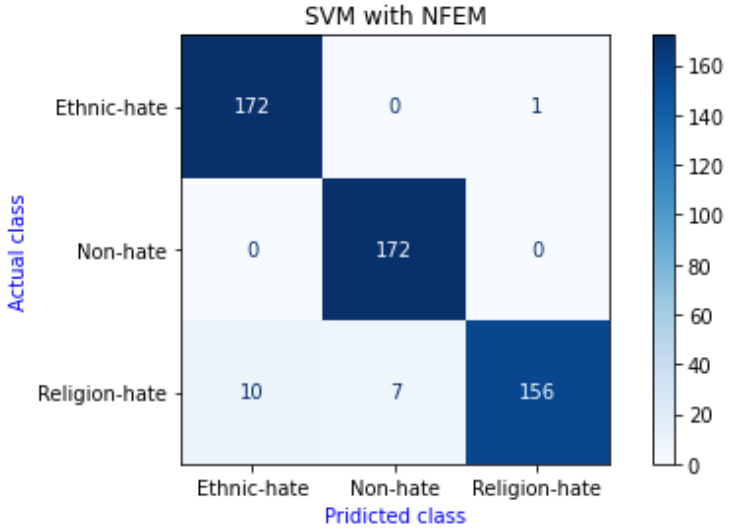

Figure 6.10 Confusion matrix of SVM classifier using NFEM

Figure 6.10 shows the confusion matrix of SVM classifier when NFEM is used as a feature extraction method. From the given 173 ethnic-hate datasets, 99.4% are correctly classified and the rest wrongly classified into Non-hate and Religion-hate classes. Out of 172 Non-hate datasets, all of them are classified correctly. For classifying the Religion-hate class 173 datasets are given then 90.1% classified correctly and only 9.9% of the datasets are wrongly classified to wrong classes. Generally, 96.5% of datasets are correctly classified from all given training sets. The following table and figure shows the summarized comparison between feature extraction methods.

Table 6.4 Comparison of feature extraction methods based on confusion matrix result

ML Classifiers	Previous methods		New Method (NFEM)	Superiority of NFEM
	Name	Result	Result	
NB	TF-IDF	61%	73.3%	12.3%
	BoW	58.8%		14.5%
RF	TF-IDF	80.5%	96.1%	15.6%
	BoW	80.6%		15.5%
SVM	TF-IDF	81.4%	96.5%	15.1%
	BoW	80.6%		15.9%

In all implemented classifiers, NFEM performed better than BOW and TF-IDF, and showed much better result on SVM classifier. On NB classifier, NFEM performed 14.5% higher result than BoW and 12.3% than TF-IDF. On RF classifier, it performed 15.5% higher result than BoW and 15.6% than TF-IDF. Using SVM classifier, it performed 15.9% higher result than BoW and 15.1% than TF-IDF. This shows the newly developed feature extraction method is more efficient and effective.

6.2.2.2. Precision, recall, F1-score, and accuracy

For evaluating and comparing the implemented models, different evaluation metrics are applied. The following table 6.5 shows the performance evaluation result of each model.

Table 6.5 Performance evaluation using precision, recall, F1-score, and accuracy

Classifier	Feature extraction	Precision	Recall	F1-score	Accuracy
RF	TF-IDF	0.80	0.80	0.80	0.805
	BOW	0.80	0.81	0.80	0.806
	NFEM	0.96	0.96	0.96	0.961
SVM	TF-IDF	0.81	0.81	0.81	0.814
	BOW	0.81	0.81	0.81	0.806
	NFEM	**0.97**	**0.97**	**0.96**	**0.965**
NB	TF-IDF	0.62	0.61	0.61	0.610
	BOW	0.63	0.59	0.58	0.588
	NFEM	0.84	0.73	0.71	0.733

When comparing feature extraction methods result, as shown in table 6.5 The new feature extraction method performed better in all of the classifiers, and a much better result is achieved on SVM classifier, which is a 0.965 accuracy score. The following figure shows the graphical representation of each feature extraction methods performance in NB, RF, and SVM classifiers.

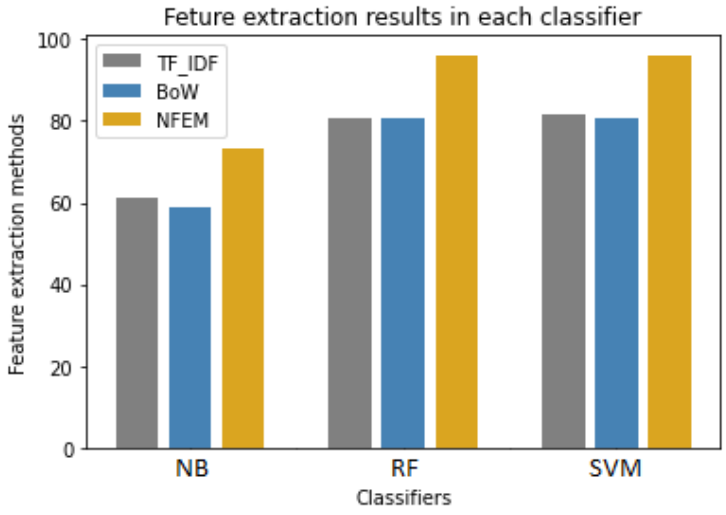

Figure 6.11 Comparison between feature extraction methods

6.2.2.3. *K-fold cross-validation*

The other evaluation metrics used is k fold cross-validation. The datasets are divided into five equal parts then each part in turn used for testing the model, and the rest four parts used for training the model.

Table 6.6 Classification performance evaluation using 5 fold CV

Classifier	Feature extraction	5 fold cross-validation					
		1st fold	2nd fold	3rd fold	4th fold	5th fold	Average
RF	TF-IDF	0.763	0.775	0.789	0.806	0.770	0.781
	BOW	0.749	0.795	0.816	0.792	0.763	0.783
	NFEM	0.963	0.961	0.968	0.956	0.956	0.961
SVM	TF-IDF	0.754	0.768	0.811	0.782	0.758	0.775
	BOW	0.759	0.790	0.782	0.775	0.758	0.773
	NFEM	**0.959**	**0.966**	**0.971**	**0.956**	**0.958**	**0.962**
NB	TF-IDF	0.590	0.607	0.594	0.625	0.603	0.604
	BOW	0.592	0.60	0.60	0.620	0.608	0.606
	NFEM	0.715	0.722	0.722	0.719	0.727	0.721

Table 6.6 shows the result of each classification model with different feature extraction methods and when the testing dataset is iterating from 1st fold to 5th fold. For comparison, the average value is calculated. The average shows, in every classifier the new feature extraction method performed better than other feature extraction methods. So here also the SVM classifier with NFEM has performed 0.962 average score, which is a better result than others.

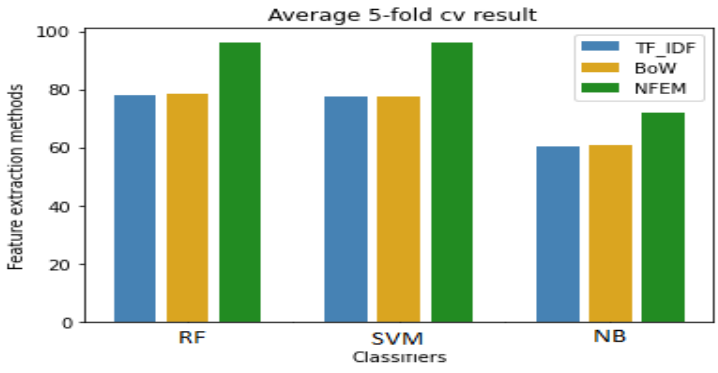

Figure 6.12 Average result of 5-fold Cross Validation

When compared with other related works, Zewdie Mussie [13] achieved 79% of accuracy by combining NB with word2vec, but the proposed model showed a 17.5% higher result. Yonas Kenenisa [14] used SVM with Word2Vec and scored 75.39%, here also the proposed model has a 21.11% superiority. The difference between some other related works are summarized in the table below.

Table 6.7 Comparison between the proposed model and related works

No	Title	Used methods	Achieved Result	The new result	Difference
1	Social network hate speech detection for Amharic language(2018) [13]	Naïve Bayes with Word2Vec	79%	96.5%	17.5%
2	Hate Speech Detection for Amharic Language on Social Media Using Machine Learning Techniques(2019) [14]	SVM with word2vec	75.39%		21.11%
3	Detecting hate speech in social media(2017) [62]	SVM with Word skip-gram	78%		18.5%
4	Hate me, hate me not: Hate speech detection on Facebook(2017) [68]	N-gram and PoS tagging	80.6%		15.9%

CHAPTER SEVEN

7. Conclusion, Recommendation, Future work, and Contribution

This chapter presents the conclusion derived from the conduct of this study, which is exploring a better feature extraction method for Amharic hate speech detection. It also provides recommendations, and feature works are suggested for those who are studying or have a plan to work in this research area.

7.1. Conclusion

Hate speech is a speech that promotes hatred, discrimination, attack on a person or a group of persons based on their personal and collective identities. Currently hate speech is widely spreading on social media, it means that it increases the ability to reach to peoples. This study stands for creating a better feature extraction method for Amharic hate speech detection. To solve the stated problem, it is necessary to know what hate speech is, although hate speech has different definitions, using the definition of the hate speech law leads to consensus.

To have a background knowledge, different literatures and related works are reviewed and criticized. In the background works the researchers used different frequency-based feature extraction methods but to have an improved result, contextual understanding of words in a given text is necessary, that is what this study did. Hate speech has characteristics that distinguish it from other speech, hate speech should contain the name of the victim as a noun, and hate-word that promotes hatred, discrimination, aggression as an adjective, verb, or adverb.

To conduct this study, datasets are collected from various public Facebook pages. the collected datasets are preprocessed and manually annotated by law experts into three classes; Ethnic-hate, Religion-hate, and Non-hate. Then different feature extraction methods including the proposed feature extraction method are tested by combining with three machine learning algorithms. Finally, the result gained in each combination of feature extraction and machine learning algorithm are evaluated using different machine learning performance evaluation metrics. Using frequency-based feature extraction methods consumes a large memory size because features are represented using up to thousands of digits per one dataset, but the proposed method represents features by 8 digits per dataset.

As evidenced by the result of all classification algorithms, the new feature extraction method, which is proposed in this study, is effective. This feature extraction method showed more than 12% superiority in all classification algorithms when compared with other tested feature extraction methods. On other hand Yonas [14] and Shervin Malmasi, et al, [62] used SVM classifier for their final model, this study also used SVM to build the final model but the result shows that the proposed model is 15 to 18 percent higher. This shows, giving a better feature to machine learning classifiers produces a better result.

7.2. Recommendation

Due to the spread of hate speech on social media, hate speech has become a negative aspect of social media. So that it is recommended that social media companies to involve in the implementation of such research ideas to protect their customers from different verbal abuse.

On social media, speeches that make peoples vulnerable to attack based on their political views, profession, and language, are widely spreading. for example, “ሹፌሮች አጭበርባሪ ስለሆኑ ለባልነት አይመጥኑም“, meaning "Drivers are not suitable for marriage because they are fraudulent", “ የብልፅግና ደጋፊዎችን መጨፍጨፍ ነበር “ meaning “It is good to massacre of supporters of prosperity party”, “እንደ ኦሮሞኛ ተናጋሪዎች ይችን ሀገር የበጠበጠ የለም” meaning "No one has disturbed this country like Afan Oromo speakers". But the law [2] specifies the definition of hate speech, by saying "Speech that encourages discrimination or violence against an individual or group based on **ethnicity**, **religion**, **race**, **gender or disability**.” So that to prevent human right violations in all directions, it is recommended that the legislator to re-examine the law [2].

7.3. Future works

From the experience gained during this study and from the observed gap during reviewing different related works, for those who are working in this area, for those who have the interest to work on this area for the future and as this study progresses in the future, the work to be added in the future is suggested as follows

- On social media, people write words by combining without space consciously or unconsciously. For example, "ይሄሰውዬይደብረኛል". This is difficult for tagging part of speech and to understand the role of a word in a sentence. So that a method to separate such words must be included for the future.
- As discussed in the methodology, a dictionary is used to create a new feature extraction. In this dictionary, there are different religious words, ethnic names, gender names, and hate words. So that, a Morphology analyzer is needed to avoid duplications of the same root words in the dictionary like "ክርስቲያን, ክርስቲያኑ, ክርስቲያኗ, ክርስቲያኖች, ክርስቲያኖቹ". Therefore, this must be taken into consideration for future work.
- On the other hand, some people write the Amharic speech in the Latin alphabet so the language is still Amharic and Amharic hate speech is still spreading, therefore, there should be a mechanism to resolve it together for the future

7.4. Contribution

After reviewing different related literatures and identifying gaps, the study contributed the following works for the science.

- ✓ By discussing with language experts part of speeches that most of the time hate speech is occurring are identified, this will help for other researchers in other professions.
- ✓ A new effective and efficient hate speech feature extraction method is developed.
- ✓ Memory consumption for feature extraction is reduced
- ✓ By making a comparison between different models, the best model is selected.
- ✓ Using the created feature extraction method Hate speech detection model is developed
- ✓ Part of speech tagged hate speech datasets are prepared

References

[1] Bela Banathy, Patrick M. Jenlink, Dialogue as a Means of Collective Communication, New York: Kluwer Academic/Plenum Publishers, 2005.

[2] h. o. p. r. Federal democratic republic of Ethiopia, "Hate Speech and Disinformation Prevention and Suppression Proclamation," *Proclamation No. 1185 /2020,* 2020.

[3] D. Moss, "Free Speech, Love Speech, Hate Speech, and Neutrality: In and Out of the Consulting Room," *Journal of the American Psychoanalytic Association,* 2019.

[4] Naganna Chetty, Sreejith Alathur, "Hate speech review in the context of online social networks," May 2018.

[5] K. Gelber, "Terrorist-Extremist Speech and Hate Speech: Understanding the Similarities and Differences," June 2019.

[6] Christer L. Pettersson, Nigussu Solomon, "Social media and journalism in Ethiopia," 2019.

[7] E. J. Shaw, "The Rwandan Genocide: A Case Study," May 2012.

[8] L. Weldehanna, "Practices and Challenges of Using Social Media as Sources of News in Ethiopia Mainstream Media: Selected Mainstream Media in Focus," 2018.

[9] M. Assefa, "The Role of Social Media in Ethiopia's Recent Political Transition," 2020.

[10] E. Meseret, "Hate speech and disinformation concerns escalate in Ethiopia," 2020.

[11] A. A. Abiodun Salawu, "LANGUAGE POLICY, IDEOLOGIES, POWER AND THE ETHIOPIAN MEDIA," 2015.

[12] G. Dires, "Language Policy of Ethiopia," 2019.

[13] Zewdie Mossie, Jenq, Haur Wang, "Social Network Hate Speech Detection for Amharic Language," 2018.

[14] Y. K. Defar, "Hate Speech Detection for Amharic Language on Social Media Using Machine Learning Techniques," September 2019 .

[15] BBC, "Tech," British Broadcast Corporate, 01 01 2018. [Online]. Available: https://www.bbc.com/news/technology-42510868. [Accessed 02 09 2020].

[16] Simeon O. Edosomwan, et al, "The history of social media and its impact on business," *The Journal of Applied Management & Entrepreneurship ,* 2011.

[17] M. Assefa, "The Role of Social Media in Ethiopia's Recent Political Transition," 2020.

[18] C. S. Agency, "Population Projections for Ethiopia," 2013.

[19] T. W. Workneh, "Social media, protest, & outrage communication in Ethiopia: toward

fractured publics or pluralistic polity?," 2020.

[20] B. Osatuyi, "Information sharing on social media sites," *Computers in Human Behavior,* 2013.

[21] Nicholas J. Belkin, W. Bruce Croft, "Information Filtering and Information Retrieval: Two Sides of the Same Coin?," 1992.

[22] Santosh Kr. Paul, Madhup Agrawal, Shyam Singh Rajput, "An Information Retrieval(IR) Techniques for text Mining on web for Unstructured data," in *Advanced Developments in Engineering and Technology*, 2014.

[23] Ika Alfina, Rio Mulia, Mohamad Ivan Fanany, and Yudo Ekanata, "Hate Speech Detection in the Indonesian Language: A Dataset and Preliminary Study," October 2017.

[24] Samuel C. Silva, Adriane B., Serapião, Ivandré Paraboni, "Hate-speech detection in Portuguese using CNN and psycho-linguistic dictionary," *Journal of Information and Data Management,* p. 1–0??, 2019.

[25] ThomasDavidson, DanaWarmsley, MichaelMacy, IngmarWeber, "Automated Hate Speech Detection and the Problem of Offensive Language," 2017.

[26] Tin Van Huynh, Duc-Vu Nguyen, Kiet Van Nguyen, Ngan Luu-Thuy Nguyen, Anh Gia-Tuan Nguyen , "Hate Speech Detection on Vietnamese Social Media Text using the Bi-GRU-LSTM-CNN Model," 2019.

[27] Shahzad Qaiser, Ramsha Ali, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents," *International Journal of Computer Applications,* 2018.

[28] E. Nguyen, "Text Mining and Network Analysis of Digital Libraries in R," *Data Mining Applications with R,* 2014.

[29] S. Robertson, "Understanding Inverse Document Frequency On theoretical arguments for IDF," *Journal of Documentation,* 2004.

[30] Yin Zhang · Rong Jin · Zhi-Hua Zhou, "Understanding Bag-of-Words Model: A Statistical Framework," *International Journal of Machine Learning and Cybernetics,* 2015.

[31] Dongyang Yan, Keping Li, et al, Network-Based Bag-of-Words Model for Text Classification, 2017.

[32] Prakash M Nadkarni, Lucila Ohno-Machado, Wendy W Chapman, "Natural language processing: an introduction," *Journal of the American Medical Informatics Association ·,* 2016.

[33] Tutorialspoint, Natural Language Processing, 2019.

[34] G. A. Fink, Markov Models for Pattern Recognition, London, 2014.

[35] D. Hiemstra, N-Gram Models, Netherlands: University of Twente, Enschede, 2016.

[36] S. Pyysalo, Part-of-Speech Tagging, Tokyo, Japan: University of Tokyo, 2013.

[37] B. Yimam, Yeamaregna sewasew, addis abeba: culture and society of ethiopia, 2000 E.C.

[38] L. P. H. R. LLUIS MARQUEZ, "A Machine Learning Approach to POS Tagging," 2000.

[39] Sanjiv K. Bhatia · Krishn K. et al, Advances in Intelligent Systems and Computing, Allahabad, Uttar Pradesh India: Springer, 2016.

[40] Chang-shuai Xing, Gang Zhou, Ji-Cang Lu, and Feng-juan Zhang, "A Word Embeddings Training Method Based," in *Cloud Computing and Security*, Springer, 2018.

[41] Jeffrey Pennington, Richard Socher, Christopher D. Manning, "GloVe: Global Vectors for Word Representation," 2014.

[42] Long Ma, Yanqing Zhang, "Using Word2Vec to Process Big Text Data," 2016.

[43] Chris Aldrich, Lidia Auret, Unsupervised Process Monitoring and Fault Diagnosis with Machine Learning Methods, London: Springer, 2013.

[44] Shai Shalev-Shwartz, Shai Ben-David, "1 Introduction," in *Understanding Machine Learning*, New York, Cambridge University Press, 2014.

[45] Rodrigo Fernandes de Mello, Moacir Antonelli Ponti, Machine Learning A Practical Approach on the Statistical Learning Theory, Switzerland: Springer International Publishing, 2018.

[46] Qiong Liu, Ying Wu, "SUPERVISED LEARNING," 2015.

[47] P. Cunningham, "Unsupervised Learning and Clustering," 2017.

[48] Alejandro Cholaquidis, Ricardo Fraimana, Mariela Sued , "Semi-supervised learning: When and why it works," 2018.

[49] P. V. B. E. R. Y C A Padmanabha Reddy, "Semi-supervised learning: a brief review," *International Journal of Engineering & Technology,* 2018.

[50] V. Heidrich-Meisner, M. Lauer, C. Igel, M. Riedmiller, "Reinforcement Learning in a Nutshell," 2014.

[51] S. Abe, Support Vector Machines for Pattern Classification, Verlag London: Springer Limited, 2005.

[52] Ingo Steinwart, Andreas Christmann, Support Vector Machines, Springer Science+Business Media, LLC, 2008.

[53] J. Vikramaditya, "Tutorial on Support Vector Machine," 2003.

[54] R. Russell, Step-by-Step Guide To Implement, Rudolph russell, 2018.

[55] Shai Shalev-Shwartz, Shai Ben-David, Understanding Machine Learning:, Cambridge

University Press., 2014.

[56] E. Alpaydın, Introduction to machine learning, London, England: The MIT Press, Massachusetts Institute of Technology, 2010.

[57] D. Berrar, "Bayes' Theorem and Naive Bayes Classifier," *Machine Learning for Soccer,* 2019.

[58] Jiangtao Ren, Sau Dan Lee, et al, "Naive Bayes Classification of Uncertain Data," 2014.

[59] Adele Cutler, D. Richard Cutler and John R. Stevens, "Random Forests," 2014.

[60] Yitna Firdyiwek, Daniel Yaqob, "The System for Ethiopic Representation in ASCII," 2016.

[61] Junanda Patihullah, Edi Winarko, "Hate speech detection for Indonesia tweets using word embedding and Gated recurrent unit," 2019.

[62] ShervinMalmasi, MarcosZampieri, "DetectingHateSpeechinSocialMedia," *Proceedings of Recent Advances in Natural Language Processing,* p. 467–472, 2017.

[63] Hossin, M., Sulaiman, M.N, "A REVIEW ON EVALUATION METRICS FOR DATA CLASSIFICATION EVALUATIONS," *International Journal of Data Mining & Knowledge Management Process (IJDKP) ,* 2016.

[64] K. J. Danjuma, "Performance Evaluation of Machine Learning Algorithms in Post-operative Life Expectancy in the Lung Cancer Patients," 2018.

[65] N. Lavesson, Evaluation and Analysis of Supervised Learning Algorithms and Classifiers, SWEDEN: Blekinge Institute of Technology, 2006.

[66] Pushkar Mishra, Marco Del Tredici, Helen Yannakoudakis, Ekaterina Shutova, "Author Profiling for Hate Speech Detection," Feb 2019.

[67] Muhammad Sajjad, Fatima Zulifqar, Muhammad Usman Ghani Khan, Muhammad Azeem, "Hate Speech Detection using Fusion Approach," 2019.

[68] Fabio Del Vigna, Andrea Cimino, Felice Dell'Orletta, "Hate me, hate me not: Hate speech detection on Facebook," 2017.

[69] MicheleCorazza,StefanoMenini, PinarArslan, RacheleSprugnoli, ElenaCabrio, SaraTonelli, SerenaVillata, "Comparing Different Supervised Approaches to Hate Speech Detection," 2018.

[70] P. B. S. K. Sreelakshmi ka, "Detection of Hate Speech Text in Hindi-English Code-mixed Data," *Third International Conference on Computing and Network Communications,* 2020.

[71] Xhemal Zenuni, Jaumin Ajdari, Florije Ismaili, Bujar Raufi, "AUTOMATIC HATE SPEECH DETECTION IN ONLINE CONTENTS USING LATENT SEMANTIC ANALYSIS," June 2017.

[72] DIGITAL2020, "reports," 17 02 2020. [Online]. Available: https://datareportal.com/reports/digital-2020-ethiopia. [Accessed 06 12 2020].

[73] Brittan Heller, and Larry Magid, "COMBATING HATE SPEECH".

[74] YouTube, "Transparency Report," Google, 01 09 2020. [Online]. Available: https://support.google.com/youtube/answer/2801939. [Accessed 09 09 2020].

[75] C. Dictionary, "Dictionary," Cambridge University, 02 09 2020. [Online]. Available: https://dictionary.cambridge.org/dictionary/english/hate-speech. [Accessed 02 09 2020].

[76] U. N. Secretary-General, "UNITED NATIONS STRATEGY AND PLAN OF ACTION ON HATE SPEECH," 2019.

[77] C. o. european, "media," 2020. [Online]. Available: https://www.coe.int/en/web/freedom-expression/hate-speech#:~:text=Over%20the%20years%2C%20the%20Council,manners%20to%20counter%20hate%20speech.&text=According%20to%20the%20Committee%20of,of%20hatred%20based%20on%20intolerance.. [Accessed 6 12 2020].

[78] "company," 2 12 2020. [Online]. Available: https://blog.twitter.com/en_us/topics/company/2019/hatefulconductupdate.html. [Accessed 6 12 2020].

[79] worldometer, "population," worldometer, 08 02 2021. [Online]. Available: https://www.worldometers.info/world-population/ethiopia-population/. [Accessed 08 02 2021].

[80] L. O.-M. W. W. C. Prakash M Nadkarni, "Natural language processing: an introduction," 2016.

Appendixes

Appendix 1: Hate speech law

Definition

- "**Hate speech**" means speech that deliberately promotes hatred, discrimination or attack against a person or a discernable group of identity, based on ethnicity, religion, race, gender or disability;
- "**Discrimination**" means any act of denying including legally accepted and attack against a person or a group based on ethnicity, religion, gender or disability;
- "**Violence**" means any injury of property, body or life against on an individual or a group of people;

Prohibition of Hate Speech

Any person disseminating hate speech by means of broadcasting, print or social media using text, image, audio or video is prohibited

Criminal Liability

1. Any person who commits acts proscribed under Article 4 shall be punished with simple imprisonment not exceeding two years or a fine not exceeding 100,000 birr.
2. If an attack against a person or a group has been committed as a result of a hate speech, the punishment shall be simple imprisonment not exceeding from one year up to five years

Appendix 2: sample labeled dataset

	label	message
364	Ethnic-hate	ቆሻሻ ትግሬ
530	Non-hate	እንኳን ለብርሀነ መስቀሉ በሰላም አደረሰህ አደረሰን
979	Religion-hate	ሻፊዒ ደንጋይ ሙስሊም እኮ መጤ ነዉ የኦርቶዶክስ እምነት ተከታዮች ባይቀበ...
517	Non-hate	እንኳንም ኢትዮጵያዊይ ሆንሽ እህቴ እንደ አንች አይነት ጀግናና ሳበዝ እም...
775	Religion-hate	የማብፃውያንን ፎቶ ተቃጠለ ወይስ አምላኬ ወይም ታቦት የምትሉትማራር እንጨ...

Figure 0.1 sample labeled dataset

Appendix 3: sample dictionary words

```
properties={'religions':['ሙስሊሞች','ፕሮቴስታንት','ፖስተር','አምላኪዎቹ','መሲህዎች','ወሀቢይ','ኡስታዝ','በኦርቶዶክስ','በቤተክርስቲያን','ክርስቶስ','ሰባ
            'ethnicities':['ሐበሻ','አማራ','ኢትዮጵያ','አባገዳ','ኦሮሞ','ትግራይ','ህዝብ','ኦሮሞ','አፋር','አገው','አኝዋክ','አርጎባ','አዊ','ባና','ዳውሮ
            'races':['ነጮች', 'ጥቁሮች', 'ጥቁር አሜሪካዉያን'],
            'pronouns':['እንተ', 'እንቺ', 'እናንተ', 'እሱ', 'እሷ', 'እነሱ','እርሶ'],
            'hatred':['አፈናቃይ','ገዳይ','ቆራጭ','ክፋት','ዘረኞች','ቅሪቅምቦ','ዲያቢሎስ','ኤጭ','ጅብ','ቀዳዳ','ደንቆሮ','ባንዳ','የጠነባ','እርኩስ','ነፍ
            'discrimination':['አይገባዉም','አይችልም','አይጠቅምም','አይገባቹሁም','አይረግጠዉም'],
            'attack':['አፈናቅል','ግደል','ደምስስ','ማፍረስ','ይጥፋ','አዋርደሽ','ይውደም','ያጥፋችሁ','መግደል','ማሸበር','ሲጨፈጭፉ','አቃጥሎ','ተቃጠ
            'positive_state':['ይሻላል','ያፀድቃል','ነው', 'ነች', 'ናቸው', 'ነኝ', 'ይሆናል', 'ትሆናለች', 'አለበት',],
            'negative_state':['አያፀድቅም','አያድንም','አይደለም','የለባቸውም', 'አትሆንም']
           }
```

Figure 0.2 Sample dictionary words

Appendix 4: sample codes

Appendix 4.1: sample codes for organizing dataset

```
for i in files:
    name_only=os.path.splitext(i)[0]  #removing the extension
    #creating folder and naming the category
    directory=os.path.join("C:/Users/yes/Desktop/implementation/excel/",name_only)
    if not os.path.exists(directory):
        os.mkdir(directory)
    PATH = ("C:/Users/yes/Desktop/implementation/excel/"+i)
    dataset = pandas.read_excel(PATH, sheet_name='Sheet1') #reading the dataset
    for r in criteria:
        ee=dataset['message'].str.contains(r,na=False) #selecting datasets that contains the
criteria
        filtered=dataset.loc[ee]
filtered.to_excel("C:/Users/yes/Desktop/implementation/excel/"+name_only+"/"+r+'.xlsx')
#writing the selecting dataset to excel and saving it to it's category
```

Appendix 4.1: sample codes for organizing dataset

Appendix 4.2: sample codes for counting dataset by its category

```
In [12]: filess='C:/Users/yes/Desktop/implementation/excel/'
for x in os.listdir(filess):
    if os.path.isdir(os.path.join(filess, x)):
        folder = "C:/Users/yes/Desktop/implementation/excel/"+x+"/"
        files_in_folder = os.listdir(folder)
        print("================================================================================")
        print("Page Name:----------",x,"-----------------------")
        print("================================================================================")
        count=0
        for y in files_in_folder:
            dataset = pandas.read_excel(folder+y, sheet_name='Sheet1')
            count=count+len(dataset)
            print("File name_____",y,"contains__________",len(dataset),"_____________datasets")
        print("--------------------------------------------------------------------------------")
        print("Total count_____________________________________",count)
        print("--------------------------------------------------------------------------------\n\n\n")
```

```
================================================================================
Page Name:---------- abiy ahmed -----------------------
================================================================================
File name_____ UላՈ.xlsx contains__________ 2 _____________datasets
File name_____ Uማር.xlsx contains__________ 0 _____________datasets
File name_____ Uረረ.xlsx contains__________ 1 _____________datasets
File name_____ UՈሾች.xlsx contains__________ 0 _____________datasets
File name_____ Uዲያ.xlsx contains__________ 7 _____________datasets
File name_____ ሃርላ.xlsx contains__________ 0 _____________datasets
File name_____ ሙሃይሙ.xlsx contains__________ 1 _____________datasets
File name_____ ሙንታ.xlsx contains__________ 37 _____________datasets
File name_____ ሙንታዉ.xlsx contains__________ 0 _____________datasets
File name_____ ሙንታዎቹ.xlsx contains__________ 0 _____________datasets
File name_____ ሙንታዎች.xlsx contains__________ 3 _____________datasets
File name_____ ሙካን.xlsx contains__________ 1 _____________datasets
File name_____ ሙዉደም.xlsx contains__________ 0 _____________datasets
File name_____ ሙደብደብ.xlsx contains__________ 12 _____________datasets
File name_____ ሙግደል.xlsx contains__________ 177 _____________datasets
File name_____ ሙፈናቀል.xlsx contains__________ 24 _____________datasets
File name_____ ሙርሌ.xlsx contains__________ 0 _____________datasets
```

Figure 0.3 sample codes for counting dataset by its category

Appendix 4.3: sample codes for counting dataset by its type

```
religions  =properties['religions']
ethnicities=properties['ethnicities']
races= properties['races']
hatred     =properties['hatred']
attack     =properties['attack']
count_religions  =0
count_ethnicities=0
count_races      =0
count_hatred     =0
count_attack     =0
files="C:/Users/yes/Desktop/implementation/excel/"
for i in os.listdir(files):
   if os.path.isdir(os.path.join(files,i)):
      folder = "C:/Users/yes/Desktop/implementation/excel/"+i+"/"
      files_in_folder = os.listdir(folder)

print("============================================================")
      print("Page name::::::: ",i)

print("============================================================")
      count_r=0
      count_e=0
```

```
        count_rc=0
        count_h=0
        count_a=0
        for y in files_in_folder:
            name=os.path.splitext(y)[0]
            if name in religions:
                dataset = pandas.read_excel(folder+y, sheet_name='Sheet1')
                print(name,"---is religious Word---number of data contains this word
is===",len(dataset))
                count_r=count_r+len(dataset)
            elif name in ethnicities:
                dataset = pandas.read_excel(folder+y, sheet_name='Sheet1')
                print(name,"---is ethnic name---number of data contains this word is
====",len(dataset))
                count_e=count_e+len(dataset)
            elif name in races:
                dataset = pandas.read_excel(folder+y, sheet_name='Sheet1')
                print(name,"---is race name---number of data contains this word is
======",len(dataset))
                count_rc=count_rc+len(dataset)
            elif name in hatred:
                dataset = pandas.read_excel(folder+y, sheet_name='Sheet1')
                print(name,"---is a hatered word---number of data contains this word is
==",len(dataset))
                count_h=count_h+len(dataset)
            elif name in attack:
                dataset = pandas.read_excel(folder+y, sheet_name='Sheet1')
                print(name,"---is an attack word---number of data contains this word is
==",len(dataset))
                count_a=count_a+len(dataset)
        count_religions+=count_r
        count_ethnicities+=count_e
        count_races+=count_rc
        count_hatred+=count_h
        count_attack+=count_a

        print("\nTotal number of data contains religious word in this
page============",count_r)
        print("Total number of data contains ethnicity words in this
page=============",count_e)
        print("Total number of data contains race types in this
page================",count_rc)
        print("Total number of data contains hatered words in this
page==============",count_h)
        print("Total number of data contains attack words in this
page===============",count_a)
        print("Total number of datasets filtered in this
page=====",count_r+count_e+count_rc+count_h+count_a)
```

```
        print("--------------------------------------------------------------------------------------------------
-----")

print("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%")
print("%%%%%%%%%%%%%%%%%%%%%%%%%%%%          SUMMERY
%%%%%%%%%%%%%%%%%%%%%%%%%%")
print("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%")
print("Total number of data contains religious
words================",count_religions)
print("Total number of data contains ethnic
words==================",count_ethnicities)
print("Total number of data contains racia words===================",count_races)
print("Total number of data contains hate words===================",count_hatred)
print("Total number of data contains attack words==================",count_attack)
print("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%")
print("Total number of data contains religious
words=========",count_religions+count_ethnicities+count_races+count_hatred+count
_attack)
print("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%")
```

```
===========================================================================================
Page name:::::::  BBC news
===========================================================================================
ሀላባ ---is ethnic name---number of data contains this word is ========= 0
ሀበሾች ---is ethnic name---number of data contains this word is ========= 1
ሀዲያ ---is ethnic name---number of data contains this word is ========= 2
ሃርላ ---is ethnic name---number of data contains this word is ========= 0
መሃይም ---is a hatered word---number of data contains this word is ========= 0
መንጋ ---is a hatered word---number of data contains this word is ========= 10
መንጋው ---is a hatered word---number of data contains this word is ========= 0
መንጋዎቹ ---is a hatered word---number of data contains this word is ========= 0
መንጋዎች ---is a hatered word---number of data contains this word is ========= 0
መጨደም ---is an attack word---number of data contains this word is ========= 0
መደብደብ ---is an attack word---number of data contains this word is ========= 4
መግደል ---is an attack word---number of data contains this word is ========= 48
መፈናቀል ---is an attack word---number of data contains this word is ========= 7
ሙርሌ ---is ethnic name---number of data contains this word is ========= 0
ሙርሲ ---is ethnic name---number of data contains this word is ========= 0
ሙስሊም ---is religious Word---number of data contains this word is ========= 79
⋮
    Total number of data contains religious word in this page=============== 164
    Total number of data contains ethnicity words in this page================ 649
    Total number of data contains race types in this page===================== 66
    Total number of data contains hatered words in this page================== 566
    Total number of data contains attack words in this page=================== 69
    Total number of datasets filtered in this page============================ 1514
    ---------------------------------------------------------------------------------
    ---------------------------------------------------------------------------------
⋮
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%                SUMMERY          %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Total number of data contains religious words=================== 14748
Total number of data contains ethnic words=================== 92209
Total number of data contains racia words=================== 3004
Total number of data contains hate words=================== 29629
Total number of data contains attack words=================== 5007
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Total number of data contains religious words=================== 144597
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 0.4 Sample result of counting dataset by its type

Appendix 4.4: Sample codes for cleaning the dataset

Removing latin alphabets

```
In [85]: posts_and_comments = [re.sub(r"[A-Za-z]", "", str(post)) for post in posts_and_comments]
```

Removing Numbers

```
In [86]: posts_and_comments = [re.sub("[0-9]", "", str(post)) for post in posts_and_comments]
```

Removing emojis

```
In [87]: posts_and_comments = [re.sub("🦴|😨|👅|🤬|😈|🦄|😬|⏰|🐂|🍾|👆|👆|♥|□□□□□□□□|🌼|🌻|👍|🌺|😉|😛|🙂|👇|😤|😤|❗|😕|…|□□□□□□
```

Remove Punctuations

```
In [88]: punctuations = "፡!\"#$%&'()*+,-./:;<=>?@[\]^_`{|}~።፣፤¡"
         posts_and_comments = [str(post).translate(str.maketrans('', '', punctuations)) for post in posts_and_comments]
```

Remove extra white space

```
In [89]: posts_and_comments = [(" ".join([ single for single in str(post).split()])) for post in posts_and_comments]
```

Figure 0.5 Sample codes for cleaning the dataset

Appendix 4.5: Sample codes for normalization

```
posts_and_comments = [re.sub("ሐ|ሃ|ኀ|ኃ", "ሀ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሑ|ኁ",     "ሁ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሒ|ኂ",     "ሂ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሔ|ኄ",     "ሄ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሕ|ኅ",     "ህ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ኆ|ሖ",     "ሆ", str(post))for post in posts_and_comments]

posts_and_comments = [re.sub("ሠ","ሰ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሡ","ሱ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሢ","ሲ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ሣ","ሳ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub( "ሤ","ሴ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub( "ሥ","ስ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub( "ሦ","ሶ" , str(post))for post in posts_and_comments]


posts_and_comments = [re.sub("ዐ|ዓ|ኣ", "አ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ዑ", "ኡ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ዒ", "ኢ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub( "ዔ", "ኤ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub( "ዕ", "እ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub( "ዖ", "ኦ", str(post))for post in posts_and_comments]


posts_and_comments = [re.sub("ጸ" , "ፀ",  str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ጹ" , "ፁ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ጺ" , "ፂ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ጻ" , "ፃ",  str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ጼ" , "ፄ", str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ጽ" , "ፅ",  str(post))for post in posts_and_comments]
posts_and_comments = [re.sub("ጾ" , "ፆ" , str(post))for post in posts_and_comments]
```

Figure 0.6 Sample codes for normalization

Appendix 4.6: Sample codes for feature generation (created feature extraction method)

```
m=x1.active
m.cell(row=1, column=1).value="label"
m.cell(row=1, column=2).value="message"
```

```
m.cell(row=1, column=3).value="entity"
m.cell(row=1, column=4).value="entity_type"
m.cell(row=1, column=5).value="hate_word"
m.cell(row=1, column=6).value="hate_adj"
m.cell(row=1, column=7).value="hate_vb"
m.cell(row=1, column=8).value="hate_adv"
m.cell(row=1, column=9).value="pos_state"
m.cell(row=1, column=10).value="neg_state"
for b in range(len(posts_and_comments)):
   label_2=labels[b]
   words=nltk.word_tokenize(posts_and_comments[b])
   q=bigram_tagger.tag(words)
   #print(q)
   leng=len(q)
   m.cell(row=b+2, column=1).value=label_2
   m.cell(row=b+2, column=2).value=posts_and_comments[b]
   m.cell(row=b+2, column=3).value=0
                        ….
   m.cell(row=b+2, column=10).value=0
   for j in range(leng):
      if((q[j][0] in list_of_ethnicities) or (q[j][0] in list_of_races) or (q[j][0] in
list_of_religions) or (q[j][0] in list_of_informal_names)):
         m.cell(row=b+2, column=3).value=1
         if((q[j][0] in list_of_ethnicities) or (q[j][0] in list_of_races) or (q[j][0] in
list_of_informal_names)):
            m.cell(row=b+2, column=4).value=2
         elif((q[j][0] in list_of_religions)):
            m.cell(row=b+2, column=4).value=1
         elif(q[j][0] in list_of_genders):
            m.cell(row=b+2, column=4).value=3
      elif(((q[j][0] in list_of_hatred) or (q[j][0] in list_of_discrimination) or (q[j][0] in
list_of_attack))):
         m.cell(row=b+2, column=5).value=1
      if((q[j][1]=='ADJ') and (q[j][0] in list_of_hatred )):
         for x in range(j,leng):
            if((q[x][1]=='ADJ') or (q[x][1]=='CONJ')):
               continue
            if(((q[x][1]=='NN') or (q[x][1]=='NN_OBJ') or (q[x][1]=='NN_SUB') or
(q[x][1]=='NN_CONJ')) and ((q[x][0] in list_of_ethnicities) or (q[x][0] in list_of_races) or
(q[x][0] in list_of_religions) or (q[x][0] in list_of_genders) or (q[x][0] in
list_of_informal_names))):
               m.cell(row=b+2, column=6).value=1
            else:
               break
      if(((q[j][1]=='VB') or (q[j][1]=='VB_CONJ')) and (q[j][0] in list_of_attack)):
         for y in range(j,1,-1):
            if(((q[y-1][1]=='NN') or (q[y-1][1]=='NN_OBJ') or (q[y-1][1]=='NN_SUB') or
(q[y-1][1]=='NN_CONJ')) and ((q[y-1][0] in list_of_ethnicities) or (q[y-1][0] in
```

```
list_of_races) or (q[y-1][0] in list_of_religions) or (q[y-1][0] in list_of_genders) or (q[y-1][0] in list_of_informal_names))):
                m.cell(row=b+2, column=7).value=1
            elif(((q[y-1][1]=='NN') or (q[y-1][1]=='NN_OBJ') or (q[y-1][1]=='NN_SUB') or (q[y-1][1]=='NN_CONJ')) and ((q[y-1][0] not in list_of_ethnicities) or (q[y-1][0] not in list_of_races) or (q[y-1][0] not in list_of_religions) or (q[y-1][0] not in list_of_genders) or (q[y-1][0] not in list_of_informal_names))):
                break
            else:
                continue
        if(((q[j][1]=='ADV') or (q[j][1]=='ADV_CONJ')) and (q[j][0] in list_of_hatred)):
            for z in range(j,1,-1):
                if(((q[z-1][1]=='NN') or (q[z-1][1]=='NN_OBJ') or (q[z-1][1]=='NN_SUB') or (q[z-1][1]=='NN_CONJ')) and ((q[z-1][0] in list_of_ethnicities) or (q[z-1][0] in list_of_races) or (q[z-1][0] in list_of_religions) or (q[z-1][0] in list_of_genders) or (q[z-1][0] in list_of_informal_names))):
                    m.cell(row=b+2, column=8).value=1
                elif(((q[z-1][1]=='NN') or (q[z-1][1]=='NN_OBJ') or (q[z-1][1]=='NN_SUB') or (q[z-1][1]=='NN_CONJ')) and ((q[z-1][0] not in list_of_ethnicities) or (q[z-1][0] not in list_of_races) or (q[z-1][0] not in list_of_religions) or (q[z-1][0] not in list_of_genders) or (q[z-1][0] not in list_of_informal_names))):
                    break
                else:
                    continue
            for k in range(j,leng):
                if((q[k][1]=='VB') and (q[k][0] in list_of_positive_state)):
                    m.cell(row=b+2, column=9).value=1
                    break
                elif((q[k][1]=='VB') and (q[k][0] in list_of_negative_state)):
                    m.cell(row=b+2, column=10).value=1
x1.save("generated_features.xlsx")
```

```
                continue
        if(((q[j][1]=='ADV') or (q[j][1]=='ADV_CONJ')) and (q[j][0] in list_of_hatred)):
            for z in range(j,1,-1):
                if(((q[z-1][1]=='NN') or (q[z-1][1]=='NN_OBJ') or (q[z-1][1]=='NN_SUB') or (q[z-1][1]=='NN_CONJ')) and ((q[z-1][
                    m.cell(row=b+2, column=8).value=1
                elif(((q[z-1][1]=='NN') or (q[z-1][1]=='NN_OBJ') or (q[z-1][1]=='NN_SUB') or (q[z-1][1]=='NN_CONJ')) and ((q[z-1
                    break
                else:
                    continue
            for k in range(j,leng):
                if((q[k][1]=='VB') and (q[k][0] in list_of_positive_state)):
                    m.cell(row=b+2, column=9).value=1
                    break
                elif((q[k][1]=='VB') and (q[k][0] in list_of_negative_state)):
                    m.cell(row=b+2, column=10).value=1

#x1.save("test all/generated features.xlsx")
```

```
--------------------------------------------------------------------------------------------------------------
Hate in this text_____ ጥንብ  ትግሬ
--------------------------------------------------------------------------------------------------------------
Given text:
--------------------------------------------------------------------------------------------------------------
[('ጥንብ', 'ADJ'), ('ትግሬ', 'NN'), ('የቆሸሸ', None), ('ልጅ', 'VB'), ('ሙሳ', None), ('ከዚህ', None)]
==============================================================================================================

--------------------------------------------------------------------------------------------------------------
Hate in this text_____ እበት  ሀዛብ
--------------------------------------------------------------------------------------------------------------
Given text:
--------------------------------------------------------------------------------------------------------------
[('ጥሩነሙ', None), ('ያ', None), ('እበት', 'ADJ'), ('ሀዛብ', 'NN'), ('በሰፈሙ', None), ('ሰሙ', None), ('ለመሪድ', None), ('እና', 'CONJ'), ('ለማግኑ', None), ('ይመቹዋል', None)]
==============================================================================================================

--------------------------------------------------------------------------------------------------------------
Hate in this text_____ ጥምብ  ጋለ
--------------------------------------------------------------------------------------------------------------
```

Appendix 4.7: classification result and demo

```
In [18]: print(classification_report(y_test, y_predict))
                precision    recall  f1-score   support

   Ethnic-hate       0.95      0.99      0.97       173
      Non-hate       0.96      1.00      0.98       172
Religion-hate       0.99      0.90      0.95       173

      accuracy                           0.97       518
     macro avg       0.97      0.97      0.96       518
  weighted avg       0.97      0.97      0.96       518
```

```
In [19]: print(accuracy_score(y_test, y_predict))
0.9652509652509652
```

```
In [20]: from sklearn.model_selection import cross_val_score
cv=cross_val_score(classifier,X_train, y_train,cv=5)
avg=0
for i in range (0,len(cv)):
    print(i+1," fold cv == ",cv[i])
    avg+=cv[i]
print("Average == ",(avg/5))
1  fold cv ==  0.9590361445783132
2  fold cv ==  0.9662650602409638
3  fold cv ==  0.9710144927536232
4  fold cv ==  0.9565217391304348
5  fold cv ==  0.9589371980676329
Average ==  0.9623549269541936
```

Figure 0.7 Classification performance of final model

Figure 0.8 sample social media application