****

**Faculty of Informatics**

**Department of Information System**

**Post Graduate Program (Regular)**

**Title: Hybrid approach of Amharic fake news detection on social media using machine learning technique**

**By: Menbere Hailu**

**November 2020**

**Gondar Ethiopia**

**UNIVERSITY OF GONDAR**

**FACULITY OF INFORMATICS**

**DEPARTMENT OF INFORMATION SYSTEMS**

**Hybrid approach of Amharic fake news detection on social media using machine learning technique**

**A Thesis Submitted to**

**The Department of Information systems of University of Gondar in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information systems**

**By MENBERE HAILU**

**Declaration**

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

Name: Menbere Hailu

Signature:

This MSc thesis has been submitted for examination with my approval as a thesis

Advisor Name: Michael Melese (PhD.)

Signature:

Date of Submission: -----------------------

**Approval of Board of Examiners**

We, the undersigned, members of the Board of Examiners of the final open defense by Menbere Hailu have read and evaluated his/her thesis entitled Hybrid approach of Amharic fake news detection on social media using machine learning techniques “and examined the candidate. This is, therefore, to certify that the thesis has been accepted in partial fulfillment of the requirement of the Degree of Masters of Science in Information systems.

------------------------ ----------------------- ------------------------

Supervisor/advisor Signature Date

------------------------ ----------------------- ------------------------

Chairperson Signature Date

------------------------ ----------------------- ------------------------

Internal Examiner Signature Date

------------------------ ----------------------- ------------------------

External Examiner Signature Date

**ACKNOWLEDGMENT**

Nothing is possible in this world without the help of God. I know you are the one who keeps me alive and made me go this far.

Next to God, I would love to give thanks for my mother Amelework Niguse, for the support, encouragement and love that you have given me in my life and to this thesis. Nothing that I have done so far would be nothing if you were not around to see it.

This thesis would not come this far without the encouragement, constructive comments, technical support and guidance that I have got from my advisor Michael Melese (PhD). You were really an amazing mentor and model for me and for any researcher that is working on machine learning technique for the Amharic Language. Finally I would like to thank my collogues Mulualem and G/Michael for supporting with resource and you technical advising to the and all of Gondar Fana news department staffs who participate on the annotation process and for being around and support my work.

Contents

[1. Introduction 1](#_Toc56137919)

[1.1. Background 1](#_Toc56137920)

[1.2. Statement of problem 2](#_Toc56137921)

[1.3. Objectives 4](#_Toc56137922)

[1.3.1. General objectives 4](#_Toc56137923)

[1.3.2. Specific objective 4](#_Toc56137924)

[1.4. Scope of the study 5](#_Toc56137925)

[1.5. Significance of the study 5](#_Toc56137926)

[1.6. Methodology 5](#_Toc56137927)

[1.7. Data collection and analysis 6](#_Toc56137928)

[1.8. Tools and techniques 6](#_Toc56137929)

[1.9. Evaluation 6](#_Toc56137930)

[1.10. Thesis organization 6](#_Toc56137931)

[2. Literature review 8](#_Toc56137932)

[2.1. Definition of fake news 8](#_Toc56137933)

[2.2. Fake news on social media 8](#_Toc56137934)

[2.3. Fake news detection mechanism 8](#_Toc56137935)

[2.3.1. Content based fake news detection approach 9](#_Toc56137936)

[2.3.2. Social-context based fake news detection approach 11](#_Toc56137937)

[2.4. Machine learning for fake news detection 13](#_Toc56137938)

[2.5. Amharic language 15](#_Toc56137939)

[2.5.1. Amharic Writing System 15](#_Toc56137940)

[2.5.2. Amharic Morphology 16](#_Toc56137941)

[2.5.3. Challenges in Amharic Writing System 17](#_Toc56137942)

[2.6. Related wok 17](#_Toc56137943)

[2.6.1. Summary of Related Works 22](#_Toc56137944)

[3. Methodology 25](#_Toc56137945)

[3.1. General approach 25](#_Toc56137946)

[3.2. Data collection 25](#_Toc56137947)

[3.3. Data preparation 26](#_Toc56137948)

[3.4. Data annotation 28](#_Toc56137949)

[3.5. Fake news detection modelling 29](#_Toc56137950)

[3.5.1. Pre-processing 29](#_Toc56137951)

[3.5.1.1. Removing Non-Amharic character, symbols, whitespace, URL and emoji’s, 29](#_Toc56137952)

[3.5.1.2. Normalization 30](#_Toc56137953)

[3.5.1.3. Tokenization 31](#_Toc56137954)

[3.5.1.4. Remove stop words 31](#_Toc56137955)

[3.5.1.5. Stemming/lemmatization 32](#_Toc56137956)

[3.5.2. Feature extraction method 33](#_Toc56137957)

[3.5.3. Machine Learning Classification Algorithm 35](#_Toc56137958)

[3.5.4. Evaluation 36](#_Toc56137959)

[4. Model design and implementation 38](#_Toc56137960)

[4.5. High level architecture 38](#_Toc56137961)

[4.6. Experiment 40](#_Toc56137962)

[4.6.3. Import library 41](#_Toc56137963)

[4.6.4. Read dataset 41](#_Toc56137964)

[4.6.5. Removing irrelevant character, emoji’s, punctuation mark 42](#_Toc56137965)

[4.6.6. Normalization 42](#_Toc56137966)

[4.6.7. Tokenization 43](#_Toc56137967)

[4.6.8. Remove stop word 43](#_Toc56137968)

[4.6.9. Stemming 44](#_Toc56137969)

[4.6.10. Handling categorical data 44](#_Toc56137970)

[4.6.11. Feature extraction 45](#_Toc56137971)

[4.6.11.1. N- Gram weighted by TFI-DF 45](#_Toc56137972)

[4.6.11.2. Word 2vec 46](#_Toc56137973)

[4.6.11.3. Hybrid approach 47](#_Toc56137974)

[4.6.12. Machine learning model 48](#_Toc56137975)

[4.6.13. Model testing and evaluation 49](#_Toc56137976)

[4.7. Experimental result 50](#_Toc56137977)

[4.7.3. Experiment 1: Unigram weighted by TFIDF with three ML algorithms result 50](#_Toc56137978)

[4.7.4. Experiment 2: Bigram weighted by TFIDF with three ML algorithms result 54](#_Toc56137979)

[4.7.5. Experiment 3: Trigram weighted by TFIDF with three ML algorithms result 57](#_Toc56137980)

[4.7.6. Experiment 4: Word2vec With three ML algorithms result 60](#_Toc56137981)

[4.7.7. Experiment 5: Hybrid of unigram and social context feature 63](#_Toc56137982)

[4.7.8. Experiment 6: Hybrid of Bigram and social context feature 65](#_Toc56137983)

[4.7.9. Experiment 7: Hybrid of Trigram and social context feature result 68](#_Toc56137984)

[4.7.10. Experiment 8: Hybrid of Word2vec and social context feature result 71](#_Toc56137985)

[4.7.11. Summary of result 74](#_Toc56137986)

[References 77](#_Toc56137987)

**List of figure**

[Figure 3.1: Labeled data distribution 28](#_Toc56082046)

[Figure 4.1: proposed architecture of fake news detection: 38](#_Toc56082047)

[Figure 4.2: python code for importing library 41](#_Toc56082048)

[Figure 4.3:phyton code for read dataset 42](#_Toc56082049)

[Figure 4.4:phyton code for clean irrelevant character and emoji’s 42](#_Toc56082050)

[Figure 4.5:phyton code for tokenization 43](#_Toc56082051)

[Figure 4.6:phyton code for stop word removal 44](#_Toc56082052)

[Figure 4.7: python code for handling categorical data 45](#_Toc56082053)

[Figure 4.8: phyton code for splitting date to day month and year 45](#_Toc56082054)

[Figure 4.9: Python code for unigram feature extraction 45](#_Toc56082055)

[Figure 4.10: Python code for bigram feature extraction 45](#_Toc56082056)

[Figure 4.11:Python code for trigram feature extraction 46](#_Toc56082057)

[Figure 4.12: Python code for Wrd2vec feature extraction 46](#_Toc56082058)

[Figure 4.13:Python code for averaging word2vec feature extraction 47](#_Toc56082059)

[Figure 4.14:Python code for hybrid feature extractionof unigram and social-context 47](#_Toc56082060)

[Figure 4.15:Python code for split train and test dataset 48](#_Toc56082061)

[Figure 4.16: Python code for logistic regression model 48](#_Toc56082062)

[Figure 4.17: Python code for support vector machine model 49](#_Toc56082063)

[Figure 4.18: Python code for random forest model 49](#_Toc56082064)

[Figure 4.19: Python code for evaluation of confusion matrix for logistic regression model 50](#_Toc56082065)

[Figure 4.20: Python code for classification report 50](#_Toc56082066)

[Figure 4.21: confusion matrix of unigram TFIDF feature with LR 52](file:///C:\Users\MenbiAmel\Music\fake%20news%20detection\Documentation\FAKE%20NEWS%20DETECTION%20(MENBERE%20HAILU%201-3).docx#_Toc56082067)

[Figure 4.22: Confusion matrix of unigram TFIDF feature with SVM 53](#_Toc56082068)

[Figure 4.23: Confusion matrix unigram TFIDF feature with RF 54](file:///C:\Users\MenbiAmel\Music\fake%20news%20detection\Documentation\FAKE%20NEWS%20DETECTION%20(MENBERE%20HAILU%201-3).docx#_Toc56082069)

[Figure 4.24: Confusion matrix of bigram TFIDF feature with LR 56](#_Toc56082070)

[Figure 4.25: Confusion matrix bigram TFIDF feature with SVM 56](#_Toc56082071)

[Figure 4.26: Confusion matrix bigram TFIDF feature with RF 57](#_Toc56082072)

[Figure 4.27: Confusion matrix trigram TFIDF feature with LR 58](file:///C:\Users\MenbiAmel\Music\fake%20news%20detection\Documentation\FAKE%20NEWS%20DETECTION%20(MENBERE%20HAILU%201-3).docx#_Toc56082073)

[Figure 4.28: Confusion matrix trigram TFIDF feature with SVM 59](#_Toc56082074)

[Figure 4.29: Confusion matrix trigram TFIDF feature with RF 59](#_Toc56082075)

[Figure 4.30: Confusion matrix Word2vec feature with LR 61](file:///C:\Users\MenbiAmel\Music\fake%20news%20detection\Documentation\FAKE%20NEWS%20DETECTION%20(MENBERE%20HAILU%201-3).docx#_Toc56082076)

[Figure 4.31: Confusion matrix Word2vec feature with SVM 62](#_Toc56082077)

[Figure 4.32: Confusion matrix Word2vec feature with RF 62](#_Toc56082078)

[Figure 4.33: confusion matrix for Hybrid of unigram and social context feature with LR 64](file:///C:\Users\MenbiAmel\Music\fake%20news%20detection\Documentation\FAKE%20NEWS%20DETECTION%20(MENBERE%20HAILU%201-3).docx#_Toc56082079)

[Figure 4.34: confusion matrix for Hybrid of unigram and social context feature with SVM 64](#_Toc56082080)

[Figure 4.35: confusion matrix for Hybrid of unigram and social context feature with RF 65](#_Toc56082081)

[Figure 4.36 : confusion matrix for Hybrid of bigram and social context feature with LR 67](file:///C:\Users\MenbiAmel\Music\fake%20news%20detection\Documentation\FAKE%20NEWS%20DETECTION%20(MENBERE%20HAILU%201-3).docx#_Toc56082082)

[Figure 4.37: confusion matrix for Hybrid of bigram and social context feature with SVM 67](#_Toc56082083)

[Figure 4.38: confusion matrix for Hybrid of bigram and social context feature with RF 68](#_Toc56082084)

[Figure 4.39: confusion matrix for Hybrid of trigram and social context feature with LR 69](#_Toc56082085)

[Figure 4.40: Confusion matrix for Hybrid of trigram and social context feature with SVM 70](#_Toc56082086)

[Figure 4.41: Confusion matrix for Hybrid of trigram and social context feature with RF 71](#_Toc56082087)

[Figure 4.42: Confusion matrix of hybrid of Word2vec and social context feature with LR 72](#_Toc56082088)

[Figure 4.43: Confusion matrix of hybrid of Word2vec and social context feature with SVM 73](#_Toc56082089)

[Figure 4.44: Confusion matrix of hybrid of Word2vec and social context feature with RF 74](#_Toc56082090)

**List of table**

[Table 1.1:Time table for performing the proposed work 7](#_Toc56081499)

[Table 1.2: Items and cost list needed for preforming the study 8](#_Toc56081500)

[Table 2.1: Summary of related work for fake news detection. 24](#_Toc56081501)

[Table 3.1: Collected data attribute with their description 26](#_Toc56081502)

[Table 3.2: Selected number of tuples from the given pages 28](#_Toc56081503)

[Table 3.3: Confusion matrix of fake news detection 36](#_Toc56081504)

[Table 4.1: Unigram weighted by TFIDF with three ML algorithms result 51](#_Toc56081505)

[Table 4.2: Bigram weighted by TFIDF with three ML algorithms result 54](#_Toc56081506)

[Table 4.3:Trigram weighted by TFIDF with three ML algorithms result 57](#_Toc56081507)

[Table 4.4: Word2vec With three ML algorithms result 60](#_Toc56081508)

[Table 4.5: Hybrid of unigram and social context feature 63](#_Toc56081509)

[Table 4.6: Hybrid of Bigram and social context feature with three ML algorithms result 65](#_Toc56081510)

[Table 4.7: Hybrid of Trigram and social context feature result 68](#_Toc56081511)

[Table 4.8: Hybrid of Word2vec and social context feature result 71](#_Toc56081512)

[Table 4.9: Summary of accuracy result 75](#_Toc56081513)

**List of equation**

[Equation 4.3: Equation of Recall 37](#_Toc57350916)

[Equation 4.4: Equation of Precision 37](#_Toc57350917)

[Equation 4.5: Equation of F-measure 37](#_Toc57350918)

[Equation 4.6: Equation of accuracy 37](#_Toc57350919)

**List of abbreviations**

AI Artificial Intelligence

API Application Programming Interface

BoW Bag-of-words

CV Consonant-Vowel

CBOW Continuous Bag of Word

FP False Positive

FN False Negative

LR Logistic regression

LSTM Long Short Term Memory

NB Naïve Bayes

NLP Natural Language Processing

RF Random Forest

RNNs Recursive Neural

SVM Support Vector Machine

TF Term Frequency

IDF Inverse Document Frequency

TP True Positive

TN True Negative

USML Unsupervised Machine Learning

Word2Vec Word to Vector

# Introduction

## Background

The large increase of social media users in the past few years has led to an overpowering quantity of information available in daily (or even hourly) basis [1]. In addition, the easy accessibility to these platforms whether it’s by a computer, tablet or mobile, allows the consumption of information at a distance of a click.

Despite the advantage provided by social media, the quality of news on social media is lower than the traditional news organization [2]. However, because it is cheap to provide news online and much faster and easier to disseminate through social media, large volume of false information are produced online for the purpose of political and financial gain and those information creates confusion to the users to detect whether the information is real or fake.

Therefore, traditional and independent news media desire to adopt social media to reach a broader audience and gain new clients/consumers [3]. The ease of creating and disseminating content in social networks like Twitter and Facebook has contributed to the emergence of malicious users. In particular, users that infect the network with the propagation of misinformation or rumors.

Fake news is different from news where the source is unsure or has not performed a full search on the subject, which is called misinformation, because it is purposely released to deceive people [4]. As a consequence, these news may be misleading or even harmful, especially when they are disconnected from their origins and original contexts [5].

The sharing of fake news on social media is deliberate, undertaken by an organization or an individual, with the aim of fabricating and disseminating information that is fully or partially false in nature in order to influence opinion or for financial gain. Fake news often includes a grain of truth, but this ‘kernel of true information’ is twisted, taken out of context, surrounded by false details and so on [4].

Therefore, fake news detection on social media has recently become an emerging research that is attracting great attention. Fake news detection on social media presents unique characteristics and challenges that make existing detection algorithms from traditional news media ineffective or not applicable [6]. Fake news is intentionally written to mislead readers to believe false information, which makes it difficult and nontrivial to detect based on news content; therefore, we need to include auxiliary information, such as user social engagements on social media, to help make a determination [7].

Consequently, there is a need for overcoming the problem of detecting fake news from the real one.

## Statement of problem

Now a days in Ethiopia the issue of social media and fake news has been critical and the major issue among concerned stakeholders in society [8]. Social networks and immediate messaging applications allow misleading content to reach to a number of people [9]. Due to the appealing nature, fake news spreads rapidly and influencing people’s perceptions about various subjects, from news stories about unproven scientific studies that confirm half-truths to statements by politicians and celebrities that are distorted and act like a fire in the timelines of social networks.

In this new world of social media, with the increase of network traffic, it still is challenging for fake news detection. Firstly, the content of fake news become more diverse in terms of topics, styles and media platforms [6]. It is difficult to extract obvious features from these rich and colorful contents. Secondly, to invert black and white and mix truth and fiction intentionally, some real events will be incorporated into the fake news. For example, fake news may cite realistic evidence, data to support an inauthentic point or conclusion in the context. Recent surveys have shown that people increasingly get their news from social media than from traditional news sources [10], making it of ultimate importance to limit false information on such platforms. With primary motives of inﬂuencing opinions and earning money [11], the wide impact of false information makes it one of the modern dangers to society, according to the World Economic Forum [12]. Understanding the reasons for why and how false information is created is important to proactively detect it and mitigate its impact.

In this way, fake news have not only influenced political elections around the world, but also caused problems in public healthy (e.g. currently there was a lot of false information spread about corona virus on social media and the people frightened and stressed based on those misinformation) and human [2]. To make things worse, it is important to highlight the human difficulty of detecting not only fake news, but deceptive content in general. Research on this fact has already shown that humans can unsatisfactorily separate true news from fake ones [13].

There are a lot of research done on the other country on different method of fake news detection [7].Traditionally content-based approach has proven effective for the news lacking social information [14]. In [15] authors showed that a simple model based on term frequency-inverse document frequency (TF-IDF) offers a baseline accuracy of 88.5%. In [16] authors have used TF-IDF with six different machine learning classifiers on a 2000 news dataset, obtaining a 92% accuracy. Another study conducted by [17], applying deep learning techniques to discriminate fake news on the Internet using only their text is studied. In order to accomplish that, three different neural network architectures are proposed, one of them based on BERT, a modern language model created by Google which achieves state-of-the-art results.

The main difficulty in applying content-based methods for real-world fake news detection is that these news are “intentionally written to mislead consumers, which makes it nontrivial to detect simply based on news content” [18].Additionally, in [19] showed that they could use the news’ writing style to effectively discriminate hyper partisan news and satire from mainstream news, but they could not claim “to have solved fake news detection via style analysis alone”. These difficulties are probably the reason behind the rather limited use of content-based methods alone for fake news detection on social media

Research has been conducted for developing fake news detector using social features of news. In [20] authors showed that Facebook posts can be classified with high accuracy as hoaxes or non-hoaxes on the basis of the users who “like” or “dislike” them and achieved accuracies exceeding 99% even with very small training dataset, authors have used logistic regression and harmonic Boolean label crowd-sourcing methods. In our country there is also one research done) [21], proposed social based feature for determining the given news is fake or real by using the number of like, share, and comment from the target users post.

Although the method proposed by [22] offers very high accuracy, its application is inherently limited to cases in which information about the propagation of news in the network is available. In other words, as the method uses social interactions (i.e. “likes”) as signals to help classifying Facebook posts, it cannot be used when a post has no likes, and it will probably perform worse when a post collects only few social. This condition is also challenged for [23] and the author use news contents only to find clues to diﬀerentiate fake news and true news.

To fill those gaps, this paper will propose uses both content-based and social features of news articles simultaneously for detecting Amharic fake news that is posted on the Facebook.

Another research gaps that are identified from previous work are those researchers are using their own country social media dataset, but according to [24] in building social media the dataset should take language and cultural differences into account. Based on these researchers result is not applicable for Ethiopian Amharic social media dataset. It is also emphasized that a lack of reliable and prepared social media data is a serious problem in our country [23].

Therefore, NLP-based applications try to use linguistic patterns that serve to detect whether information is fake or not. However, much of the difficulty in such NLP-based research lines resides in the fact that it is language dependent and there are very few available corpora to develop and test the systems, mainly if we consider non-English languages [25].

Therefore, the research will intend to get answers for the following research questions.

* How the Amharic news articles can be detected as fake or real?
* Which features are used for detecting Amharic news as real or fake?
* Which feature extraction method should be used to best determine and detect Amharic news post as real and fake?
* Which approach is preferable for Amharic fake news detection?
* Which machine learning algorithm is preferable for detecting Amharic fake news

## Objectives

### General objectives

The general objective of this study is to design a model that can detect Fake News by using both content and context feature of news post on social media specifically on Facebook.

### Specific objective

* To study the current status of fake news in Ethiopia and currently existing application for detecting the fake news in Ethiopia or as a global.
* To collect Amharic posts and related information from Facebook and annotate by expert.
* To identify the features from Facebook data that used for detecting Amharic news.
* To design and develop a fake news detection models using machine learning algorithms.
* To evaluate the research process, findings and the resulting model for the detection of fake news on social media.
* To recommend future research direction for further investigation

## Scope of the study

The aim of this study is detecting the Amharic fake news that are posted on Facebook. By extracting those posts and other related data from the Facebook and labeled by the expert like (journalist) as fake or genuine news that are posted on those Facebook pages. And identify content based (i.e. analyzing the content of the news) of the news and social features of news (i.e. how the news is diffused in the network) from those collected data.

By applying NLP to preprocess and analyzing and incorporating for both content and social context features and classifying the given news as fake or real news by using different machine learning approaches.

## Significance of the study

Because of spreading rate of fake news on social media is high, most of social media user especially Facebook user cannot detect if it is true or false information and they accept as real news. By conveying biased and false information, fake news can destroy society’s faith and beliefs in authorities, experts and the government. This study plays a significant role for detecting fake news for those social media users.

Because of there is no fact-checking website or source in our country most of journalist check if the news [6] is false or true by accessing different websites and communicate with the perspective e domain sectors .and it is very time consuming and fatigue task for them. This study also help journalist easily identify the given news is genuine or fake.

This study also help the researcher to identify the characteristics and nature of fake news and provide the way to detect flow of fake news

## Methodology

This study will follow quantitative research as general approach and experimental analysis as specific method. In the quantitative research it was possible together many posted and other information that were available on Facebook and also, machine learning techniques was used to train and classify the collected dataset. After the classification has been done an evaluation technique was used to check the performance of the proposed experiment

## Data collection and analysis

Because there is no well-known and standard dataset for fake Amharic news and fact- checking websites, the researcher will identify Facebook news pages that contain accepted news and fake news through observation and communicate with the experts. After that collect those news from different Facebook news pages, news agencies website, and sector office official s like health minister websites and the expert like journalist and some exposer will annotate and label the collected data.

## Tools and techniques

Python programming language will use for collect and preprocess the data and also to building the model. Python programming language is one of the most flexible languages and can be used for various purposes. Python does contain special libraries for machine learning name including the latest one Flair and other scipy and numpy which great for linear algebra and getting to know kernel method of machine learning. The language has easy syntax relatively. Because of the above advantage the researcher will uses python for preprocess as well as for model building [29].Natural language processing will use for preprocessing the data like for tokenization, stemming and normalizing the news content and analyzing some other social-context feature of data.

* 1. Evaluation

The aim of evaluation is to evaluate the performance of the proposed experiments after building the prototype. This performance evaluation method can be used regarding the exactness of what we have aimed before (precision), the relevant instances form a given dataset (recall), and the harmonic relationships that the precision and the recall have (F-Measure) within a given dataset.

## Thesis organization

The research paper organized under five chapter in the following ways:

*Chapter one*: It includes the introduction of the study from different perspectives, the statement of problems, the research questions that would be answered in the proposed solutions, the scope that cover this study, the objective of the study, the methodology that this thesis follow, and the application results of the study.

*Chapter two*: presents the literature review and related works on fake news detection, definition of fake news from different perspective, fake news on social media, mechanism or approaches of fake news detection, feature extraction and machine learning classifier, it discussed overview of the Amharic language and finally, it discussed the related work.

*Chapter three:*discusses the research methodology used in this study, methods used to collect, prepare and annotate the data, identify fake news detection modelling technique like preprocessing, feature extraction, machine learning classification and evaluation techniques for this study.

Chapter four: discusses the proposed solution for fake news detection, which is the proposed high level architecture, working environment, implementation and experimentation of preprocessing, feature extraction implementation, models implementations, after conducting and analyzing the experiments, the proper performance measurement has been performed to evaluate the experiments .

Chapter five: Presents conclusion, recommendation and also suggest the insight into what should be done for future relating to this research topic.

# Literature review

## Definition of fake news

Fake news is information that has been deliberately fabricated and disseminated with the intention to deceive and mislead others into believing falsehoods or doubting verifiable facts and it is disinformation that is presented as, or is likely to be perceived as news [30].

## Fake news on social media

Social media has become a primary source of news consumption nowadays. Social media is free of cost, easy to access and helps one to express opinions publicly and hence acts as an excellent way for individuals to consume information [31].Since there is no regulatory authority on social media like Facebook and tweeter, the quality of news pieces spread in social media is often lower than traditional news sources. In other words, social media also enable the wide spread of fake news. [32] [33].

According to [34], fake news have their own characteristics on social media. First, Malicious accounts on social media for propaganda. While many users on social media are legitimate, social media users may also be malicious, and in some cases are not even real humans. The low cost of creating social media accounts also encourages malicious user accounts, such as social bots, cyborg users, and trolls. A social bot refers to a social media account that is controlled by a computer algorithm to automatically produce content and interact with humans (or other bot users) on social media [35]. Second, Echo Chamber Effect. Social media provides a new paradigm of information creation and consumption for users. The information seeking and consumption process are changing from a mediated form (e.g. by journalists) to a more disinter-mediated way [36] . Consumers are selectively exposed to certain kinds of news because of the way news feed appear on their homepage in social media, amplifying the psychological challenges to dispelling fake news.

## Fake news detection mechanism

Fake news detection is defined as the task of categorizing news along a range of veracity with an associated measure of certainty [34]. Detecting fake news in social media has been an extremely important, yet technically very challenging problem. The difficulty comes partly from the fact that even human beings may have difficulty identifying between real news and fake news. In one study, human judges, by a rough measure of comparison, achieved only 50-63 % success rates in identifying fake news. [11]

Fake news detection methods generally focus on using news contents and social contexts [34] [37].

### Content based fake news detection approach

News contents contain the clues to diﬀerentiate fake and real news. For news content based approaches, features are extracted as linguistic features, writing style features, Semantic features. Sentiment features, visual-based features.

**Linguistic-based analysis**: Online fake news is generated intentionally by the fake news creators for financial, political, or other gains [38]. Most of the fake news creators use specific writing strategies to avoid being detected [39]. The primary goal of linguistic analysis is to match the news creator’s language competence by observing the language formats and discovering the writing patterns [40] . “Bag-of-words” and “n-grams” are the most common methods for representing raw news texts [41] [39]. In “bag-of-words”, by regarding each word as a single and equal unit, the raw news text can be represented as the set of its words, disregarding language grammar and the word order. In “n-gram”, the raw news text is represented by a contiguous sequence of n items, the items can be phonemes, syllables, letters, or words. However, the simplicity of these two approaches also leads to some obvious shortcomings for raw text processing, for instance,” n-gram” model is extreme sparsely, and it cannot interpret news samples that contain unknown tokens; “bag-of-words” may lose significant information by ignoring the context and semantic of the words. In recent years, other techniques have been proposed and used for natural language representation and document classification, such as deep syntax analysis [42], word2vec [43] ,long short-term memory (LSTM) neural network [44], sequence-2-sequence based deep neural network [45]and so on.

**Semantic-based analysis**: Semantic-based analysis refers to the process of characterizing the syntactic structures of the news from phrases levels to semantics level. By uniting “n-gram” model with deep syntax model, semantic-based analysis can discover the degree of compatibility and consistency between the news creator’s personal experience and the news content [39]. For example, the fake news creators often use exaggerated title to attract readers’ attention, so the title of the fake news is usually unrelated or in conﬂict with the news content. However, the title of a true news should be consistent with the content of the news body. A fake online review may contain contradictions or mistakes in the comments, since the deceptive reviewers have no experience with the functionalities and services about the products. So semantic-based analysis can provide important clues for assessing the suspicious level of online news. With the combined information from news creator analysis and semantic based analysis, researchers can verify the compatibility between the user’s background and the news content, which has shown good improvement for false information classification and detection [46].

**Knowledge-based analysis:** Knowledge-based analysis refers to the attempts to directly check the truthfulness of the major claims in a news [34]. The aforementioned fact-checking websites like Snopes.com, PolitiFact.com and Fact Check. Org are typical examples of knowledge-based fake news detection websites. In these websites, external and professional resources, like the knowledge from an expert or an organization, are necessary for assigning truthful value for a piece of news.

Knowledge-based analysis is a fundamental component of online fake news detection in terms of the following two perspectives. First, artificial intelligence (AI)-based learning models are feasible solutions for online news evaluation. However, misleading online messages with diﬀerent writing patterns and purposes are emerging every day, which make it difficult for the AI-based models to maintain a high detection performance. Although many novel techniques are proposed recently for automatic fake news detection, nowadays fact-checking tasks are still mainly depend on human’s knowledge. Second, by considering fake news detection as a binary classification task, and build supervised machine learning models for classifying fake news from true news. And the very first task for establishing an automatic machine learning model is to collect a high quality dataset with labels. Online fake news is diverse in terms of topics, purposes, domains, styles, and platforms. So it is difficult to generate a complete fake news dataset for training such models. In addition, the real-world datasets of fake news are always incomplete, unstructured, unlabeled and noisy [34] , which make automatic detection even more problematic. In this case, a large amount of human eﬀorts are essential for collecting and labeling fake news datasets, and knowledge-based analysis is a critical aspect for generating eﬀective machine learning models, and for identifying online fake news.

**Style-based analysis**: Legitimate online users express their opinions, emotions, and feelings toward certain products, events, and services via social media [47]. Whereas, malicious online accounts express deceptive information by intentionally confusing their writing style or attempting to imitate other users [48]. By trying to capture the distinguishing characteristics of writing styles between legitimate users and anomalous accounts, style-based analysis plays an important role in online fake news identification. From the perspective of analytic target, style-based analysis can be divided as physical style analysis and non-physical style analysis. (1) Physical style analysis is the process of extracting inﬂuential physical features to distinguish fake news from honest news. These features can illustrate the writing style, the text syntax and the personal attitude of the news, such as the number of verbs and nouns, the number of emotion words and casual words. The presence of suspicious tokens, such as the number of URLs, hashtags, mentions, and the uppercase words in social communication data are also good features for authorship identification and writing style analysis [31] [49] [50]. (2) Non-physical style analysis analyzes the nonphysical aspects of the news, such as the complexity and readability of the news text [49]. Based on [41] and [51], fake news creators usually take longer time and make more mistakes during their writing. So some specific keystroke patterns can be traced for writing style analysis. For instance, the key “backspace” and “delete” are used more often when a fake news creator want to write some false messages [41] .The authenticity of a news or a document heavily depends on the authenticity of its author [48]. Style-based analysis can provide important information on representing an author’s writing style, therefore, it should obtain more attention in online fake news detection.

### Social-context based fake news detection approach

Beside content-based features of news there are other important features called social-based features which are related to the analysis of news distribution and diffusion patterns through the different social networks, analyze how fast and widely the fake news is spread and how the different users interact with these news and how they share it between them. There are two basic categories of social context-based features: user-based features and network-based features. [52]

.User-based features analysis

The analysis of these features aims to analyze the online user’s features, which they create, share and spread the fake news. By utilizing the unique characteristics of the fake news authors, the fake or bot accounts can be exposed and differentiated from the true accounts [37]. Being exposed to a large amount of unproven messages, online users lack the clues to evaluate the credibility of the social information [53]. Malicious social media accounts committed to manipulate people’s decision and pollute the truth news content by purposely spreading misinformation [54]. So user based feature analysis is a critical aspect for fake news detection. The user-based features analysis approach can be divided into 1) user profiling features analysis, 2) posting behavior and temporal features analysis and 3) credibility features analysis.

**User profiling features analysis**: The basic user profiling information includes the language used by the account, the geographic locations of the account, the account creation time, if the account is verified or not, how many posts/tweets does the account have, and so on [55]. The user profiling analysis describes how active and suspicious a social account is, and has been shown useful for suspicious social account detection [56]. User profiling features include the basic user information such as account name, geo-location information, and the data of registration of the user, verified or not, has description or not, in [33], proposed to understand user profiles from various aspects to diﬀerentiate fake news. In [57] proposed an unsupervised fake news detection algorithm by utilizing users’ opinions on social media and estimating their credibility.

**Posting behavior and temporal features analysis**: Temporal behavior expose the temporal patterns of the online social account, such as the average time between two consecutive posts, the frequency of replying, sharing, mentioning, and so on. Driven by timers or automatic programs, suspicious accounts like social bots and cyborgs are more active in a certain time period [58]. In contrast, legitimate human users have complex timing behaviors [59] .

**Credibility feature analysis**: The numbers of friends and followers are also good features for diﬀerentiating malicious accounts and legitimate users. The number of followers of a legitimate social user is often close to its friends. However, social bots usually have much more friends than followers [60]. [59] Proposes an equation for calculating account reputation with the number of followers. User credibility features record the impact and the credibility of the online account, include the credibility score of the user [59], the number of friends and followers of the user, the ratio between the user’s friends and followers, the total number of tweets/posts of the user [53].

**Network based feature analysis**

Network-based features are extracted via constructing specific networks among the users who published related social media posts. Different types of networks can be constructed. **The stance network** can be built with nodes indicating all the tweets relevant to the news and the edge indicating the weights of similarity of stances [61] . Another type of network is the co-occurrence network, which is built based on the user engagements by counting whether those users write posts relevant to the same news articles [62]. In addition, the **friendship network** indicates the following/followed structure of users who post related tweets [63]. An extension of this friendship network is the diffusion network, which tracks the trajectory of the spread of news [63], where nodes represent the users and edges represent the information diffusion paths among them. That is, a diffusion path between two users’ user1 and user2 exists if and only if (1) user2 follows user1, and (2) user2 posts about a given news only after user1 does so. After these networks are properly built, existing network metrics can be applied as feature representations. For example, degree and clustering coefficient have been used to characterize the diffusion network [63] and friendship network [63].

## Machine learning for fake news detection

Machine learning is an application of Artificial Intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learn for themselves. The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers to learn automatically without human intervention or assistance and adjust actions accordingly [64]. In terms of classifiers, machine learning approaches can be categorized into supervised, unsupervised, and semi-supervised approaches [65].

**Supervised machine learning:** algorithms can apply what has been learned in the past to new data using labeled examples to predict future events. Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system can provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors in order to modify the model accordingly [65].

**Semi-supervised machine learning:** is a combination of supervised and unsupervised machine learning methods. This machine learning used when labeled data is not enough to produce an accurate model and when there is a lack of the ability or resources to get more labeled data, then use semi-supervised techniques to increase the size of training data with unlabeled data. [64]

**Unsupervised machine learning:** algorithms used when the information used to train is neither classified nor labeled. Unsupervised learning studies how systems can infer a function to describe a hidden structure from unlabeled data. The system does not figure out the right output, but it explores the data and can draw inferences from datasets to describe hidden structures from unlabeled data [64].

According to [66] the majority of existing research uses supervised methods while semi-supervised or unsupervised methods are less commonly used. This approach is domain-dependent since it relies on manual labeling of a large volume of text for instant most of the research work reviewed for this research uses a classifier algorithm such as

**Support vector machine (SVM)** is one of the most widely used methods for classification method of supervised learning. It uses the hyper plane to split two classes data point with the maximum margin. Content-based features (e.g. linguistic and visual features) were exploited in most SVM-based approaches to fake news detection. [39] Has trained an SVM for satirical fake news detection with a number of content-based features and obtaining an F1 of 0.87.

**Naive-Bayes** model is that all features are independent of each other. This is a particularly strong hypothesis in the case of text classification because it supposes that words are not related to each other. But it knows to work well given this hypothesis. [67] Use this model used for analyzing the number of words associated with the wording used in the titles of the sites for purposes of determining which of them contains false information.

**Random forests** is an ensemble of several decision trees. The mode of all the predictions from each tree is the final output. Comparative studies on various machine learning algorithms for rumors and fake news tasks have outlined random forests as a strong performer. [68] Has implemented a random forest with a set of temporal, structural and linguistic features for rumor classification in a tweet graph, obtaining an accuracy of 0.90. Random forest is also exploited for stance detection. In [69] author has used only content-based features of tweets and has obtained an average accuracy between 0.83 and 0.88 on a manually collected dataset.

**Logistic regression** models have been also employed in a number of studies, in particular for rumor stance classification tasks. As for decision trees and random forests, studies comparing different approaches in the context of rumors and fake news have reported competitive performances for logistic regression. It was used for stance classification of news articles based on headlines and claims in [70] .The proposed method has obtained an accuracy of 0.73 on the Emergent dataset. [71] Has performed a study on linguistic predictors of rumor veracity by exploiting a logistic regression to identify the most significant ones with respect to true and false rumors. [72] Has exploited logistic regression in analyzing the credibility of Bulgarian true and false news, achieving an accuracy of 0.75 on the hardest presented dataset.

**BayesNet (Bayesian Network)** is a classifier that uses different search methods and various quality criteria. Bayes theorem based BayesNet is a probabilistic graphical model used to represent conditional dependencies between multiple sets of variables. The generated Bayesian network consists of a directed graph and probability tables. [67] Use this model used for analyzing the number of words associated with the wording used in the titles of the sites for purposes of determining which of them contains false information.

## Amharic language

### Amharic Writing System

Amharic is one of the Semitic language spoken in many parts of Ethiopia and it is an official working language for Federal Democratic Republic of Ethiopia [73] . It is also a working language of several regional states like Amhara, South Nation and Nationalities and etc. thus it have nationwide coverage. Amharic is written using a writing system called Fidel **(ፊደል),** adapted from the one used by Ge'ez language. This unsystematic borrowing from Ge’ez has resulted in redundant characters in the Amharic fidel. These create different symbol which have the same pronunciation. Although these different fidels give each word different meaning in Ge’ez, while in Amharic language they have been used interchangeably. These fidels ሀ, ሐ, ኀ and ኸ all pronounced as (hä), ሰ and ሠ refers (sä), አ and ዐ refers (a) and, ጸ and ፀ refers (tsä). Amharic took the whole Ge’ez alphabet (fidel) and uses it in the Amharic writing system [74]. Amharic language writing system contains 34 base characters each of which occurs in a basic form and six other forms known as **orders** [73]. The seven orders represent syllable combinations consisting of a consonant following vowel. The vowels are fused to the consonant form in the form of diacritic markings. The diacritic markings are strokes attached to the base characters to change their order. In Amharic writing system there is no capital-lower case distinction. The 34 basic characters and their orders give 238 distinct symbols. In addition to the 238 symbols, there are other symbols with a special feature usually representing labialization, for example ሏ፣ሟ፣ሷ፣ሯ etc. According to [75] Amharic alphabet has around 290 letters and the alphabet/character has no any capital letter and small letter distinction. Amharic has its own way of writing having numerals and punctuations [74]. In Amharic language, different punctuation marks are used for different purposes. Most text processing works with sentence and word based unit therefore larger blocks of text such as paragraphs or whole article is split into single sentences and sentences. It starts with a sequence of characters to identify the elementary parts of natural language such as words, punctuation marks and separators. In the old writing system, a colon (፡) has been used to separate two words. Nowadays the two dots are replaced with whitespace. An end of a statement is marked with four dots (፡፡) while netela serez (፣) is used to separate lists or ideas just like the comma in English Single phrases and dirib serez **(ድርብ-ሰረዝ)** used to separate more than single phrases like semicolon in English.

Numbers in Amharic consist of single characters for one to ten, for multiples of ten (twenty to ninety), hundred, and thousand. According to [76], these characters are derived from Greek letters, and some were modified to look like Amharic ‘Fidel’. Each of the symbols has a horizontal stroke above and below. There is no symbol for zero in the Amharic script. Thus, arithmetical computations using the symbols are very difficult, if ever done. As a result, people generally use the Hindu-Arabic numerals. Ethiopic numbers are used mostly in writing dates and page numbers in text [76].

### Amharic Morphology

Amharic is one of the most morphologically complex language [73]. It shows a root-pattern morphological phenomenon [77] . Root is a set of consonants which has a basic lexical meaning. According to [73], Amharic words are categorized under five categories based on the use of morphology and position of the word in sentence. Amharic word categories are noun, verb, adjective, adverb and preposition. Nouns are words used to name or identify any class of things, people, places or ideas or a particular of these. Verbs are the most important part of speech because it shows the action or state, word that tells the listener or reader what is happening in the sentence and have more to do with mental processes and perceptions. Adjective is a word that comes before a noun and add some kind of qualification to the noun. Adverb is a word that qualifies the verb by adding extra idea from time, place and situations point of view. Preposition is a word which can be placed before a noun and perform adverbial operations related to place, time, cause and etc.

### Challenges in Amharic Writing System

There are numerous challenges observed in the writing system of the Amharic language. Most researches show that these challenges are attributed to the very nature of the language [75] [78] [79] [80]. Some of these are:

* Amharic is morphologically rich language
* Amharic borrowed most of its scripts from Ge’ez without selecting those symbols that are only necessary for its consonants. As a result there are phonemes with different symbols, where they have meaning in Ge’ez, but their meaning is not known in Amharic [75]
* The proper use of symbols in Amharic is not studied exhaustively and there is no standard dictionary to refer to. These and other reasons cause the following problems to appear in Amharic information retrieval tasks in general and classification in particular.

## Related wok

Similar research has been carried out by authors in [81], they have built an automated system called “FakeNewsTracker” for understanding and detection of fake news. This system collects news contents and social context automatically and they propose Social Article Fusion (SAF) model that uses the linguistic features of news content and features of social context to classify fake news. The major functionalities of FakeNewsTracker are first, collecting news contents and social context automatically, which provides valuable datasets for the study of fake news. Second extracting useful features and build various machine learning models to detect fake news. Third, presenting the characteristics of fake news dissemination through effective visualization techniques. For data collection first, they collect the verified fake news and true news from fact-checking websites like PolitiFact on daily basis. Then, using the Twitter’s advanced search API, they gathered the tweets related to the fake/real news that spread them in Twitter. And they incorporate social engagements of users such as replies of tweet, retweet, and favorites through Twitter APIs. Social Article Fusion (SAF) model that uses the linguistic features of news content and features of social context to classify fake news. They use Auto-encoders features and the doc2vec for content of the article and they used softmax model and RNN for social context finally they achieved the accuracy of 71.7 %.

To overcome the problem that is stated in [22], the authors in [82] have used content-based methods only when the social based methods perform poorly. They have built the model based entirely on only one type of feature at a time and tested this model on real-world data and have obtained an accuracy of 81.7%.

On the other study [62] , detecting fake news based on the response of the users and text present in the piece of news; it uses a recurrent neural network (RNN) to capture the temporal pattern of user activity on a given article. The second module learns the characteristics of the source based on the behavior of users, and in the third module, the previous two modules are integrated to classify an article as fake or not. This work combines text, response and source user information. This model detects fake news from the Twitter and from Weibo .The results of the two networks are then concatenated and use for ﬁnal classiﬁcation. As textual features they used doc2vec. They did test their model on two datasets, one from Twitter dataset with an accuracy of 89.2% and the other one from Weibo, which a Chinese equivalent of Twitter. Compared to simpler models with an accuracy of 95.3%, CSI performs better, with 6% improvement over simple GRU networks.

Fake news detection model based on content of news, in [47] they collected 12,600 truthful news articles from Reuters.com (News website) for real news articles. As for the fake news, they were collected 12,600 fake news articles from .those articles focus only on political news articles and the news articles from both fake and truthful categories happened in the same timeline, specifically in 2016. Each of the articles length is bigger than 200 characters. The author use n-gram analysis and machine learning techniques. They investigate and compare two different features extraction techniques (TF-IDF and N-gram) and six different machine classification techniques (Stochastic Gradient Descent (SGD), Support Vector Machines (SVM), Linear Support Vector Machines (LSVM), K-Nearest Neighbor (KNN) and Decision Trees (DT)). Experimental evaluation yields the best performance using Term Frequency-Inverted Document Frequency (TF-IDF) and unigram as feature extraction technique, and Linear Support Vector Machine (LSVM) as a classifier, with an accuracy of 92%.

Deep learning approaches, which have gained ground in the past few years, have also been used in fake news detection. Yang et al [83], have used both text and image information to train a model named as the text and image information based convolutional neural network (TI-CNN). They have used sentiment and lexical diversity for text. For images, they observed that real news had more images of faces whereas fake news had more irrelevant images. In their model, they used two parallel CNNs to extract latent features from both textual and visual in-formation and achieved the highest precision of 0.92 and recall of 0.9227. CNNs however, require a large dataset and using them to analyze both text and images tends to be computationally expensive.

In [84] the authors propose a deep convolutional neural network (FND-Net) for fake news detection. Instead of relying on hand-crafted features, their model (FND-Net) was designed to automatically learn the discriminatory features for fake news classification through multiple hidden layers built in the deep neural network. They create a deep convolutional neural network (CNN) to extract several features at each layer and they compare the performance of the pro-posed approach with several baseline models. Benchmarked datasets were used to train and test the model, and the proposed model achieved state-of-the-art results with an accuracy of 98.36% on the test data. Various performance evaluation parameters such as Wilcoxon, false positive, true negative, precision, recall, F1, and accuracy, etc. were used to validate the results. These results demonstrate significant improvements in the area of fake news detection as compared to existing state-of-the-art results and affirm the potential of their approach for classifying fake news on social media.

On the other hand ,in [85] the author provide a means to separate related from unrelated headlines and further classifying the related headlines. They use Fake News Challenge (FNC1) dataset on stance detection and the dataset consists of a set of headlines and articles that are combined with each other (multiple times, in different combinations) and annotated for one of four classes: “unrelated”, “agree”, “disagree”, “discuss”, indicating the stance of the headline towards the content of the article .then they apply a procedure to decide whether a particular headline/article combination is related or unrelated. This is done based on n-gram matching of the lemmatized input (headline or article), using the Core NLP Lemmatizer .The number of matching n-grams (where n=1...6) in the headline and article is multiplied by length and IDF value of the matching n-gram (n-grams containing only stop words or punctuation are not considered), then divided by the total number of n-grams. If the resulting score is above some threshold (we established 0.0096 as the optimal value), the pair is taken to be related. This procedure is rule-based and only relies on finding an optimal value for the threshold, based on the data. If the data arrive at an optimal value, they used all data and did not separate it into training and test sets. And the classification task based on the machine learning techniques and achieved accuracy score of 89.59%.

In [86] the author used two methods for identifying fake news on social media has been proposed, focusing on fake news .First step of the method, a number of pre-processing is applied to the data set to convert unstructured data sets into the structured data set. The texts in the data set containing the news are represented by vectors using the obtained TF weighting method and Document-Term Matrix. In the second step, twenty-three supervised artificial intelligence algorithms have been implemented in the data set transformed into the structured format with the text mining methods. In this work, an experimental evaluation of the twenty-three intelligent classification methods has been performed within existing three public data sets such as, first dataset called BuzzFeed dataset. The data set contains 1627 news articles related to the 2016 U.S. election gathered from Facebook and these classification models have been compared depending on four evaluation metrics and achieved accuracy score of 65% by zeroR algorithm. Second dataset called Random Political News Data, this political news data set dataset created by Horne and Adali [49]. They have collected fake news from Zimdars’ list of fake and misleading news websites and real news from Business Insider’s “Most Trusted” list .This data set consists of 75 news articles collected from different sources and these classification models have been compared depending on four evaluation metrics and achieved accuracy score of 69.3% by Sequential Minimal Optimization algorithm. Third dataset called ISOT Fake News Data set. This data set contains two types of news, fake and real news. Fake and real news are obtained from real-world sources. The real news articles were collected from Reuters.com, while the fake news articles were collected from unreliable various websites such as Politifact and Wikipedia. The data set consists of a total of 44898 data, 21417 real-labeled data and 23481 false-labeled data and these classification models have been compared depending on four evaluation metrics and achieved accuracy score of 96.8 % by decision tree algorithm.

In [87] the author aims to build a novel machine learning model based on NLP technique for detection of fake news by using both content-based features and social features of news. Content-based features of fake news consist of a headline and body. After preprocessing these features are modeled using the Bag of Words Model. These features mostly emphasize on the content portion and don’t incorporate any contextual meaning. Data Preprocessing (Normalization, Stop Word Removal, and Lemmatization) is applied only on the content-based part of the news (heading, body, etc.). Social features of news don’t require any preprocessing. Each of these content-based parts of news present in the dataset is modeled using the Bag of Words Model which is a simplified representation used in Natural Language Processing (NLP). Social features play a significant role in proving the authenticity of the article. Because the internet acts as a medium for the diffusion of the news. Thus by examining the social features of the news articles, its authenticity can be established by using (Facebook Page ID, Source, and Facebook App ID).

The proposed model has been tested over the publically available dataset FakeNewsNet that consist both the PolitiFact and BuzzFeed datasets. The BuzzFeed dataset consists of 182 news articles (half of them are labeled as fake) labeled on the basis of the expert opinion of the journalists from BuzzFeed. The PolitiFact dataset consists of 240 news articles (half of them are labeled as fake) labeled by the well-recognized fact-checking website PolitiFact. Both the datasets provide, the content-based features (headline, body) along with social media based features (how this news article was shared/posted on social media websites like Facebook, Twitter).the classification task performed by using probalistic classifier and achieved average accuracy of 90.62% with F1 Score of 90.33% the above dataset.

In [67] authors Detect Fake News in Social Media Networks by using simple and carefully selected features of the title and post to accurately identify fake posts. They also develop a tool that can identify and remove fake sites from the results provided to a user by a search engine or social media news feed. The tool will come into operation and run through the sites that have been retrieved by the search engine before they are delivered to the user. In doing so, the extension will identify sites whose links contain words that may have a misleading effect on the reader, including those that are characterized by a lot of hyperbole and slang phrases. Additionally, the tool will also use the number of words associated with the wording used in the titles of the sites for purposes of determining which of them contains false information. The first step was to locate a credible click baits database, then compute the attributes and produce the data files for WEKA. In the second step, after gathering URLs in a file, a python script computed the attributes from the title and the content of the web pages. Finally, they extracted the features from the web pages. They use WEKA Ten-fold Cross-validation was used in all experiments. In order to validate the solution. They use the BayesNet, Logistic, Random Tree, naive Bayes algorithm. The experimental results show a 99.4% accuracy using the logistic classifier.

In [22] focus on using social network features in order to improve the reliability of fake news detector. This paper showed that Facebook posts can be classified with high accuracy as hoaxes or non-hoaxes on the basis of the users who “like” them. They applied two algorithms (logistic regression and harmonic Boolean label crowdsourcing) using the IDs of users as features to classify posts. For the training they used cross-validation, dividing the dataset into 80% for training and 20% for testing and performing 5-fold cross-validation, reaching 99% of accuracy in both cases. To validate their approach they using three different datasets. This allows to easily compare the accuracy of our method with the accuracy of a purely social-based method. The dataset consists of the public posts and posts’ likes of a list of Facebook pages (selection based on [1]) belonging in two categories: scientific news sources vs. Conspiracy news sources. The resulting dataset is composed of 15,500 posts, coming from 32 pages (14 conspiracy pages, 18scientific pages), with more than 230,000 likes by 900,000+users. 8,923 (57.6%) posts are hoaxes and 6,577(42.4%) are non-hoaxes. The second and third datasets come from the FakeNewsNet dataset that consist both PolitiFact and BuzzFeed.

### Summary of Related Works

The next Table presents a summary of related work in fake news detection research.

|  |  |  |  |
| --- | --- | --- | --- |
| **Authors & year** | **Title** | **Feature extraction** | **ML Method & accuracy** |
| Shu et al… (2018) [81] | FakeNewsTracker: a tool for fake news, collection, detection and visualization | **Content feature**: Auto-encoders and doc2vec.  **Social context**: SAF Model | KNN with accuracy of 71.7 % |
| Ruchansky et al (2017) [62] | CSI: A Hybrid Deep Model for Fake News Detection | **Content feature**: doc2vec.  **Social context** by using user responses on the articles | RNN with accuracy of 95.3% |
| Ahmed et al… (2017) [47] | Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques | TF-IDF and N-gram | Stochastic Gradient Descent (SGD), Support Vector Machines (SVM), Linear Support Vector Machines (LSVM), K-Nearest Neighbors (KNN) and Decision Trees (DT). Finally achieved 92% with unigram feature and SVM |
| Yang et al… (2018) [83] | CNN: Convolutional Neural Networks for Fake News Detection. | Text and image based information CNN (TI-CNN) and lexical and semantic features. | CNN with accuracy of 92 % |
| Bourgonje et al (2017) [85] | From Click bait to Fake News Detection: An Approach based on Detecting the Stance of Headlines to Articles | Content based feature  N- gram | Logistic regression with accuracy of 89.59% |
| Altunbey et al (2019) [86] | Fake News Detection within Online Social Media Using Supervised Artificial Intelligence Algorithms | TF weighting method and Document-Term Matrix. | achieved accuracy score of 65% by zeroR , 69.3% by Sequential Minimal Optimization and of 96.8 % by decision tree tested on  BuzzFeed, Random Political News and ISOT Fake News dataset respectively. |
| Shu et al  (2019) [87] | Natural Language Processing based Hybrid Model for Detecting Fake News Using Content-Based Features and Social Features | For content based bag of words (BOW) are used.  For Social context, relating attributes with class label | Probabilistic classifier with accuracy of 90.62% with F1 Score of 90.33% |
| Aldwairi et al...  (2018) [67] | Detecting fake news in social media networks | Features of the title and post to accurately identify fake posts. Then develop tool that use the number of words associated with the wording used in the titles of the sites for purposes of determining which of them contains false information. | Bayes Net, Logistic, Random Tree, naive Bayes algorithm.  Finally they achieved 99.4% accuracy by using logistic regression. |
| Tacchini et al.  (2017) [22] | Some Like it Hoax: Automated Fake News Detection in Social Networks | IDs of users as features to classify posts. For the training they used cross-validation | Performing 5-fold cross-validation and achieved 99% accuracy in both logistic regression and harmonic Boolean label crowdsourcing. |
| Abinet and Tibebe et al … (2020) [21] | Fake news detection model using machine learning approach: the case of Amharic news on social media | Use number of like ,share and comment as a feature and partial style based feature | CART, Random forest and ensemble method (bagging) and achieved scores an accuracy, precision and recall 94.4%, 94.4% and 94.7% respectively. |

Table 2.1: summary of related work for fake news detection.

# Methodology

## General approach

In this thesis, a quantitative research as general approach and perform experimental analysis as specific method has been used. In the quantitative research it was possible together many posted and other information that were available on Facebook and also, machine learning techniques was used to train and classify the collected dataset. After the classification has been done an evaluation technique was used to check the performance of the proposed experiment.

## Data collection

Data collection is the first stage in machine learning technique. Fake news detection can also be one of the area that machine learning can be used. But it requires certain kind of data should be collected. Online news can be collected from different World Wide Web sources, such as news agency website, television and radio station websites and social media websites like Facebook, Tweeter...etc. there are many publicly available dataset for fake news detection that contain content news and other social context information with their perspective labels that performed by expert journalists, Fact-checking websites, industry detector and crowd sourced workers. Some of those datasets are PolitiFact, BuzzFeedNews, LIAR, and Fake News Challenge (FNC1). However those public dataset didn’t consist Amharic content news and as a researchers knowledge there is no publicly available datasets which contain local language especially Amharic content news, so we should collect data from the Facebook platform and make ready to expert (journalist) in the area for labeling (annotation).

The study gathers Amharic posts and other social-context information from different categories of a public page because Facebook privacy policy does not allow to the public content of private page. Those categories are News media and Broadcasting pages, Bloggers and Journalists pages, Political Party, Politician and Government Office pages in general.

For collecting data from different website, any researcher can use different methods based on the platform that she/he has chosen. The first one is scraping data by using API. If someone collect data from twitter can use twitter API and the same thing for other social media platform like Facebook, however specifically Facebook denied the privilege of accessing the collection of data through this application program interface due to security reasons. The second one is that can be proposed is a python program like beautiful soup and scrappy can be used by writing the code on different python platforms. Nowadays based on Facebook’s privacy issue it is not easy to collect data as much as needed from Facebook. The third one is a websites that give permission to scrape the different news content and different social context that are available in this specific social media sites. For this thesis “www.data-miner.io” scrapping website has been used for collect for post contents and other social-context information. It is Google chrome and Microsoft edge browser extension that help to scrap data from web pages and save into a CSV file Excel spreadsheet.

The data for this study have been collected from the Facebook public pages that uses Amharic language most frequently for posts contents and starting from April 10/2020 up to August 25/2020 which covers four months of data collection. During data collection from 30 pages **9828** data with six attributes are collected.

|  |  |
| --- | --- |
| **Collected data Attribute name** | **Description** |
| Post\_content | Posted news content on the selected web pages |
| Posted\_date | The date of news posted |
| No\_of\_like | News posted pages number of like |
| No\_of\_follower | News posted pages number of follower |
| Page\_created\_on | News posted page created date |
| Is verified | If the news posted page is verified or not |
| Label | Annotation result of posted news as Real/Fake |

Table 3.1: Collected data attribute with their description

## Data preparation

After the data collected, a preparation process has been performed, which are cleaning, filtering, and merging data into one file or data table. This study used MS Excel as preparation tools. Filtering and cleaning the data primarily use for the next stage, which is the annotation of the news content in the dataset and then after used for designing a training model for Amharic fake news detection. The following tasks performed to prepare the dataset for annotation:

* Removing all non-Amharic post contents and all non-textual posts which are video link
* Removing all Null, blank value and whitespace
* Removing rows that give non sense or meaning
* Joining data of each page in to one dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Catagories | Page name | No of tuples before removed | No of tuples after removed |
| 1 | News media and broadcasting page | Amharic News | 608 | 393 |
| 2 | Ethiopia HOT News | 106 | 56 |
| 3 | Ethiopian News Agency | 720 | 500 |
| 4 | Ethiopia News Today | 370 | 300 |
| 5 | ECADF Ethiopian News | 448 | 330 |
| 6 | BBC News Amharic | 746 | 400 |
| 7 | waltainfo.com | 220 | 200 |
| 8 | FBC | 551 | 469 |
| 9 | ሰበር ዜና Ethiopian Breaking News | 671 | 500 |
| 10 | Bloggers page | EthiopianDJ የኢትዮጵያ ሙዚቃ | 594 | 300 |
| 11 | Ethio lsan | 481 | 300 |
| 12 | Eyoha Media | 204 | 136 |
| 13 | Yegna Tube - የኛ ቲዩብ | 483 | 320 |
| 14 | General interest and activist page | Finfinne Power | 441 | 390 |
| 15 | Kr Qeerroo | 214 | 100 |
| 16 | Muna Oromiyaa Page | 163 | 150 |
| 17 | Tekvah Eth | 380 | 290 |
| 18 | Tigray News Channel | 180 | 150 |
| 19 | Yehulachin የሁላችን | 280 | 190 |
| 20 | Yenepost | 235 | 185 |
| 21 | ልሣነ ዐማራ- Amhara Press | 359 | 100 |
| 22 | ቤተ አማራ | 47 | 20 |
| 23 | ፌደራሊስት ሀይሎች | 200 | 180 |
| 24 | Getachew reda (ጌታቸው-ረዳ) | 141 | 70 |
| 25 | Tigray 24 News | 200 | 100 |
| 26 | Sutafe media | 100 | 61 |
| 27 | Tigray independent | 187 | 100 |
| 28 |  | Qerro baha Africa | 250 | 186 |
| 29 |  | Oromo activist በአማርኛ | 300 | 192 |
| 30 |  | Tigray daily | 290 | 145 |
| Total | | | **9828** | **7253** |

Table 3.2: selected number of tuples from the given pages

## Data annotation

The purpose of the data annotation is to label the collected posts with meaningful classes manually before giving to the machine which is going to be used by the supervised machine learning and modeling technique. Large dataset in any domain which require extensive data labeling by human factors professionals can be time-consuming. As indicated earlier the collected data doesn’t have labels, however, manually determining the veracity of news usually requiring annotators with domain expertise who performs a careful analysis of claims and additional evidence, context, and reports from authoritative sources. Therefore, the collected data were given to professionals or experts (journalists) in the area in order to make labels or to make an annotation for the data as real and fake. In such case, six journalist who are worked as news editor in Gondar Fana FM 98.1 and in FM web new editor was performs careful analysis of claims and additional evidence and labeling collected data as real and fake. During annotation process some post that didn’t give any meaning and contain Tigrigna language been removed. Because of Tigrigna and Amharic language use same alphabets or letters it’s difficult to cleaned by preprocessing step. Generally 4590 posts has been annotated by journalists. From those posts 2211 posts have been labeled as Fake and 2379 posts have been labeled as Real.

Figure 3.1: Labeled data distribution

## Fake news detection modelling

### Pre-processing

For content based, Amharic text processing method used to clean up the news posts based on Amharic language and basic text mining preprocess technique. This methods used for making the dataset ready for feature extraction method. In addition to content based, other attributes which contain social context information and labeled attribute in the form of categorical data must be transformed to numerical data. The reason why this transformation process which makes the machine easily understand the data’s. In this study removing irrelevant and Non-Amharic character, symbols, whitespace, URL and emoji’s, text normalization, tokenization, remove stop words and stemming has been performed as text processing and data transformation has been performed for categorical data .

#### Removing Non-Amharic character, symbols, whitespace, URL and emoji’s,

The dataset is created from social media news posts, which contains a special character, punctuation, symbol, and emoji’s to express different outlooks. This cleaning task is important for removes all irrelevant special characters, symbols, and emoji’s. Also, remove all non-Amharic characters.

**Algorithm: 4.1 Removing Non-Amharic character, symbols, whitespace, URL and emoji’s**

**Input: Actual\_Post column in a dataset**

**Output: clean text**

**Begin:** 1. Read the text in the Actual\_Post column in a dataset

2. While (! end of the text in a dataset):

If the text contains special character [,’! @#$%^&\*] then

Remove special character

If the text contains symbol [,’! @#$%^&\*] then

Remove symbol and add space

If text contain English word = [a-z A-Z] then

Remove English word

If text contain number= [0-9] then

Remove number

If a text contains emoji= [...👉 🙏 😂....] then

Remove emoji

If a text contains geez number [፩, ፪, ፫, ፬, ፭, ፮, ፯, ፰, ፱, ፲] then

Remove geez number

If a text contains Amharic Punctuation='[፤።፡፣, • ፨] then

Remove Amharic Punctuation

3. Return clean text;

**End**:

#### Normalization

Normalization used to solve the redundancy problem of Amharic language, the same sound character with different character form. It is a process of changing words into a single form by performing character replacement with a similar sound to one common form of character. The change of the character into one common representation does not cause a meaning difference, but it decreases the chance of getting the same feature with different characters, which leads to duplication of a feature

**Algorithm: 4.2 Normalization**

**Input: Actua\_Post column in a dataset**

**Output: Normalized Actua\_Post**

**Begin:** 1. Read the text in the Actua\_Post column in a dataset

2. While (! end of the text in a dataset):

If text contain characters [ሐ] [ሑ] [ሒ] [ሓ] [ሔ] [ሕ] [ሖ]; then

Replace characters with [ሀ] [ሁ] [ሂ] [ሀ] [ሄ] [ህ] [ሆ] [ሀ] # respectively

If text contain characters [ኀ] [ኁ] [ኂ] [ኃ] [ኄ] [ኅ] [ኆ], then

Replace the characters with [ሀ] [ሁ] [ሂ] [ሀ] [ሄ] [ህ] [ሆ]

If text contain characters [ሠ] [ሡ] [ሢ] [ሣ] [ሤ] [ሥ] [ሦ], then

Replace the characters with [ሰ] [ሱ] [ሲ] [ሳ] [ሴ] [ስ] [ሶ]

If text contain characters [ዐ] [ዑ] [ዒ] [ዓ] [ዔ] [ዕ] [ዖ] [ኣ], then

Replace the characters with [አ] [ኡ] [ኢ] [አ] [ኤ] [እ] [ኦ] [አ]

If text contain character [ጸ] [ጹ] [ጺ] [ጻ] [ጼ] [ጽ] [ጾ], then

Replace the character with [ፀ] [ፁ] [ፂ] [ፃ] [ፄ] [ፅ] [ፆ]

3. Return replaced text;

**End**

#### Tokenization

After the cleaning and normalizing tasks, the Tokenization method follows, which splits the actual post text into individual words or tokens by using spaces between words or punctuation marks this is important because the meaning of text generally depends on the relations of words in that text and help the feature extraction methods to get appropriate feature form the dataset.

**Algorithm: 4.3 Tokenization**

**Input: clean and normalized text**

**Output: token words**

**Begin:** 1. Read clean and normalized text in a dataset

2. Split the text using nltk library

3. Remove words less than two character

4. Return tokens

**End**:

#### Remove stop words

A stop word is a commonly used word that is mainly used as a connector word in order to aid the flow of the given sentence, stop words do not add any value to the given news article in order to help the given analysis techniques in order to identify whether a given article is reliable or not (Usman Malik, 2020). The main reason as for removing stop words as-well as the previously stated reasons is because various vectorization techniques weight each given word based on their occurrence within the given corpus, due to stop words being extremely common within a given corpus these such words would hold a greater weight against an actual meaningful word; even though they are considered as connector words and don’t carry any meaning.

**Algorithm: 4.4 Remove stop-word**

**Input: Tokenized words in the dataset**

**Output: Token words not contain stop words**

**Begin:** 1**.** Take tokenized words and read stop words

2. While (! end of the tokens in a dataset):

If words not in stop words then

Append words

3. Return words not in stop words

**End**

#### Stemming/lemmatization

Lemmatization essentially analyses the given vocabulary of each word within the given corpus. In order to find the given words morphology linguistic makeup. The objective of lemmatization is to remove the form off the endings of the given words in order to return the base or root origin of the words. This aspect of finding the root of the word is known as lemma. Due to lemmatization being a more dictionary orientated, it requires a greater corpus size and greater computational linguistic power to achieve better resolutions. Stemming is also an approach attempts to achieve the same outcome as lemmatization as it reduces the inflectional form of each word into a common base route/stem. Stemming however goes about it a different way as to lemmatization. In this approach the stemming algorithm works by cutting off the end or the beginning of the word, taking only into account a list of common prefixes and suffixes which all words can share. This study uses Hornmorpho for this task. HORNMORPHO is a Python program that analyzes Amharic, Oromo, and Tigrinya words into their constituent morphemes (meaningful parts) and generates words, given a root or stem and a representation of the word’s grammatical structure [85]. It is part of the L3 project at Indiana University. Install the software and import to python as a library.

### 3.5.2. Feature extraction method

In this study feature extraction methods are basically performed for content-based analysis. Feature extraction methods based on state-of-the-art text mining; techniques applied for reducing redundant features and dimensionality. The methods involved in selecting a subset of relevant features that would help in identifying fake and real news from the dataset and can be used in the modeling of the detection problems. The N-gram weighted by TF-IDF and word2vec feature extraction used in this study because they are better and popular feature extraction methods used fake news detection studies and text classification problems.

**N-Gram**

N-grams are one of the most used techniques in fake news detection. N-gram is a word prediction model using probabilistic methods to predict the next word after observing N-1 words. The most common N-grams approach consists in combining sequential words into lists with size N, where N is the number of words used in the probability sequences. E.g. if N=1, N=2, and N=3 referred them as unigram, bigram, and trigram, respectively. This study uses a word N-gram method to create N-gram of post content features.

In this thesis we apply word level N-gram analysis of Words feature extraction in the approach to detect fake news. The reason for this is because at different N-gram levels words take up a new meaning within a more or less restricted context. This could potentially have an impact on the how well the given models are able to detect fake news. As using n-grams analysis allows us to see the actual relationship various words have with one-another whether it’s examining words that occur after one-another, or words that tend to co-occur within the same document. N-gram analysis adds an extra level of textual understanding during the feature extraction technique.

**TF\_IDF**

This method tells us the relative importance of the word in a corpus of documents, based on its frequency of occurrence in the document. It has two parts: the first one is Term frequency (TF) the second one is Inverse Document frequency (IDF). The term frequency indicates the frequency of each of the words present in the document or dataset. So, its equation is given as follows:

Equation 4.1: Equation of term frequency

TF (t) = (Number of times term t appears in a document) / (Total number of terms in the document).

The second part is — inverse document frequency. IDF actually tells us how important the word is to the document. This is because when we calculate TF, we give equal importance to every single word. If the word appears in the dataset more frequently, then its term frequency (TF) value is high while not being that important to the document. Thus, we need to define some weighing down of the frequent terms while scaling up the rare ones, which decides the importance of each word. It’s achieved by the given equation.

Equation 4.2: Equation of inverse document frequency

Finally, TF\_IDF is a product of TF and IDF. Words that occur across all documents equally indicate a higher importance and thus it will get a high TF-IDF value.in this study using N-gram weighted by TF\_IDF which means simply put the TF-IDF values instead of just pure counts of how many times certain words (or n-grams) appear in a sentence. Then normalize these values using the Euclidean norm:

**Word Embedding**:-**Word2vec**

Word Embedding are vector representations of a text in n-dimensional space. Each word is encoded as a vector. The main objective behind word embedding is to cluster words having similar meaning together that is, if we wish to visualize the words in a feature vector space, words having similar meaning will have close spatial positions. The angle between these similar word vectors will be close to 0. Word2vec converts the input text and produces a corresponding vector space for each word, each consisting of n-dimensions. These word vectors will be positioned in the vector space in such a way that words that have similar contexts/meaning in the text, to be closely located in the vector space. Word2vec learns semantic similarity between words and the actual meaning of a word and it have basically two architecture .The first one is continuous bag of word (CBOW) which uses a set of continuous words in a sentence and the next possible word could be predicted. The second one is Skip-Gram which is the reverse of CBOW and it take one word and the consecutive set of words in a sentence can also be predict. In Word embedding, the position of a word with in the vector space is learned from text and is based on the words that surround the word when it is used. For fake-news detection, using word embedding makes more sense, as understanding the contextual meaning of a word becomes highly important. A word separately could be of a different meaning and a word surrounded by a group of words in a paragraph, could have slightly different meaning.

### Machine Learning Classification Algorithm

Machine Learning Classification Algorithm The objective of this study is to detect Amharic fake new using a machine learning algorithm on labeled datasets. We use a multiple supervised machine learning algorithm for comparison and accuracy of the detection. The algorithms are selected because of their good classification performance, and those are frequently used for solving fake news detection problem on previous work on other foreign language [47] [85] [69]. The following three machine learning algorithms used to build detection models.

**Support Vector Machine (SVM):** is a supervised machine learning algorithm that can be used in classification problems .SVM algorithm performs classification by finding hyper plane in n-dimensional space that separates the classes or data points in feature space. Hyper planes are decision boundaries that help classify the data points. A hyper plane in n-dimensions is an affine subspace of dimension n-1 [88]. In general, the main goal of SVM is to divide the datasets into classes to find maximum marginal hype plane and it can be done in the following two steps. First, SVM will generate hyper planes iteratively that segregates the classes in the best way. Then, it will choose the hyper plane that separates the classes correctly.

**Random Forest:** is supervised learning algorithm which used for both classification and regression. However it is mainly used for classification problem. Random forest algorithm creates decision tree on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than single decision tree because it reduces the over fitting by averaging the result. Random forest can be done in the following step. First, start with the selection of random sample from a given dataset, then construct a decision tree for every sample and get the prediction result from every decision tree. Then voting will be performed for every predicted result. Finally select the most voted prediction result as the final prediction result.

**Logistic regression: -**is supervised learningclassification algorithm used to predict the probability of a target variable. The nature of dependent variable would be only two possible classes.it means the dependent variable is binary in nature having data coded as either 1/0. It is widely used for various classification problems such as spam detection, cancer detection.

### Evaluation

For model performance evaluation we use confusion matrix, precision, recall and f1-score. The confusion matrix is a method for summarizing the quality of a classification algorithm. The number of correct and incorrect predictions are summarized with count values and broken down by each class. It gives insight not only into the errors being made by the classifier but, more importantly, the types of errors that made. A confusion matrix table consists of four outcomes:

**True Positive**- it is the news content and social-context information which are classified as Real and been predicted by the machine learning technique as Real

**False Positive-**while the machine learning algorithms predicts them as false, it is the news content and social context data that are labeled as true.

**True Negative**- it is the news content and social-context information which are classified as Fake and predicted by the machine learning technique as Fake.

**False Negative**- it is the news content and social-context information which are classified as fake but they are predicted by the machine learning technique as real.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Predicted value | |
| Real | Fake |
| Actual Label | Real | True Real | False Real |
| Fake | False Fake | False Fake |

Table 3.3: Confusion matrix of fake news detection

**Recall/Sensitivity** Recall is also known as the True Positive rate and is the measure of all the positive classes that were predicted correctly by the given classifier.

Equation 4.3: Equation of Recall

**Precision** measures the predicted value true, and it shows how many times the model predicts true. Precision answer what proportion of identifications was correct [88]. To Precision calculated using the equation:

Equation 4.4: Equation of Precision

**F-measure (F1-score or F-score)** is a measure of a test's accuracy and defined as the weighted harmonic mean of the precision and recall of the test.F1 is more useful than accuracy when the dataset contains uneven class distribution [88].This score is calculated using this equation:

Equation 4.5: Equation of F-measure

**Accuracy** shows the classification problem correct prediction value and calculates as the total number of the model correct prediction divide by all number of data instances used for the model [88] . Accuracy calculated using the equation

Equation 4.6: Equation of accuracy

# Model design and implementation

## High level architecture

The proposed architecture used to detect Amharic news content and social-context information as real or fake. This architecture shows everything from dataset collection up to model evaluation.

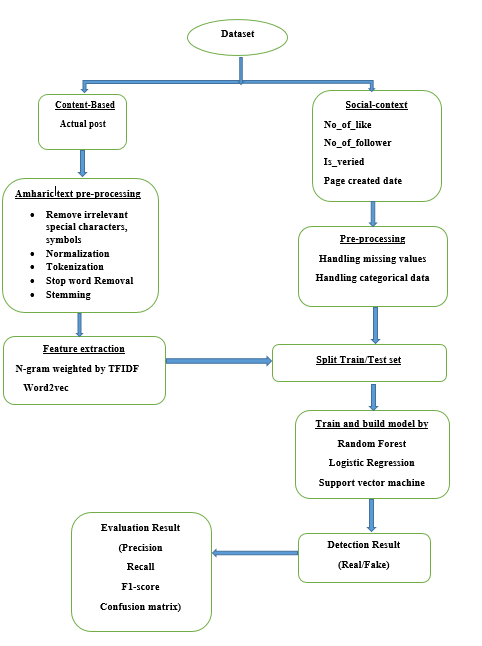


Figure 4.1: proposed architecture of fake news detection:

The above architecture shows, it takes the dataset as input .The dataset contain content-based data like posted news content and social-context information like posted page information. Amharic pre-processing technique like remove irrelevant special characters, symbols, normalization, tokenization, stop word removal and stemminghave been done for content-based of actual post. As well as other preprocessing technique like handling missing value and categorical data have been done for social-context information. After all the preprocessing, then feature extraction takes place to extract features using N-gram weighted by TFIDF and word2vec for content-based(actual post). The output of feature extraction of content based and preprocessed of social context information were combined together and it makes an important feature vector of the dataset for training the model. After feature extraction have been done, splitting the dataset for training and testing set takes place. After split the dataset, models developed and trained using LR, SVM, and RF machine learning algorithms. The classification was done by combining the feature extraction and the machine learning technique together, the next step of the architecture shows the performance evaluation techniques was performed by accuracy, precision, recall and F-measure. The architecture showed all the necessary steps need to follow to build a fake news detection model. After the models have been built a prototype have been prepared to take new dataset and classify a given news content and other information as fake or not. This prototype followed the same kinds of preprocessing, feature extraction and machine learning method. The proposed design works on any kinds of computer that have Jupyter Notebook with a predefined built in functions. But for this thesis a personal computer with a specification listed below was used. The next table show the specification of the computer that is used to conduct the experiment.

|  |  |
| --- | --- |
| Computer name | DELL |
| Operating system | Windows 10 pro |
| Process | Intel® Core™ i5-7200c CPU @2.50.GHz-2.70GHz |
| Installed memory(RAM) | 12.00GB |
| Storage | 1TB |
| System type | 64-bit operating system,x64-base processor |

Table 4.1-1: Computer specification

## Experiment

This study used several development tools and packages to implement the proposed solution, Amharic fake news detection. This study uses python programing language for implementing and experimenting with each proposed solution from the data preprocessing to the model building phase. Also, to evaluate the implemented proposed classifiers model. Python used because of its programming language of choice for developers, researchers and data scientists who need to work in machine learning models. The next table shows the list of tools and python packages with their version and description which is used in this study.

|  |  |  |
| --- | --- | --- |
| Tool | | |
| Tools | Version | Description |
| Anaconda Navigator | 1.9.6 | Allow us to launch development applications and easily manage conda packages, environment and channels without the need to use command line codes. |
| Jupiter notebooks | 5.7.8 | An open-source web application that allows us to create and share documents that contain live code, equations, visualizations, and narrative text. Uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, and machine learning. |
| Python | 3.7 | Easy to learn, powerful programming language to develop a machine learning application. |
| Microsoft Excel | 2013 | Used data preparation tasks in cleaning filtering and sorting the gather data. Also, used to manage the annotation task. |
| Python Packages | | |
| Scikit-learn | 0.20.1 | A set of python modules for machine learning and data mining. This study uses it for feature extraction and training and testing model. The name of the package is called sklearn. |
| Pandas | 0.24.2 | High-performance, easy-to-use data structures, and data analysis tools. This study uses it for data reading, manipulation, writing and handling the data frame. |
| NumPy | 1.15.4 | Array processing for number, strings, and objects. This study uses it for handling converting the text to numeric data for features and training and testing the model. |
| RegEx (Re) |  | A regular expression (or RE) specifies a set of strings that matches it; the functions in this module let you perform string matching, removal, replace, etc. package used in this study to perform preprocessing of Amharic text. |
| Genism | 3.4.0 | Python library for topic modeling document indexing and similarity retrieval with large corpora. This study uses it for the constricting word2vec model. |
| Nltk | 3.4 | Build python programs to work with human language data. This study uses it for tokenization. |
| Matplotlib | 3.0.3 | Publication quality figures in python. This study uses it for data and results visualization. |
| Flask | 1.0.2 | Micro framework for making web services in python. This study uses it for implementing the prototype for selected models. |

Table 4.2: Tools and library packages used for fake news detection model

### Import library

Import basic library that used for preprocessing, feature extraction, model building and evaluation tasks most of libraries are stated in table 4.2-1

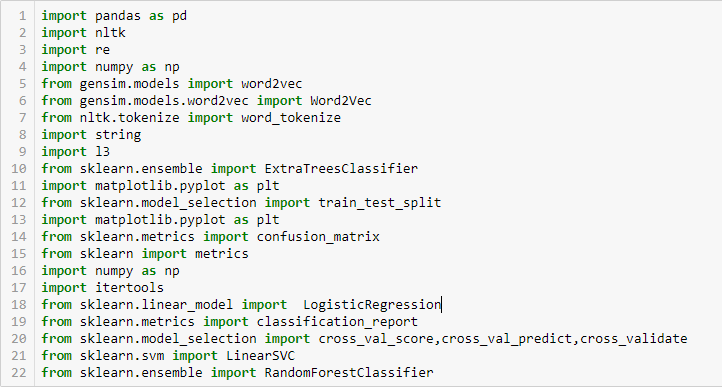


Figure 4.2: python code for importing library

### Read dataset

The data reading part was the second step after importing the libraries. This process has been used to read the file that is given weather its .csv file or .xls file. In this case since the collected data’s was being represented in .xls format we gave the file location and the file format to decrease ambiguity to what kind of data is being read. The loaded dataset using panda is used throughout the whole implementation of this study.



Figure 4.3:phyton code for read dataset

### Removing irrelevant character, emoji’s, punctuation mark

In order to clean the Actual\_post on the dataset, we implement a function clean \_irrelevant that performs removal and replace the punctuation mark, symbols ,emoji’s and special characters ,etc. the function is written using python re module. The function accept Actual \_Post in the dataset as a parameter and return the cleaned text and append to corpus array.



Figure 4.4:phyton code for clean irrelevant character and emoji’s

### Normalization

To normalize the Amharic characters with the same sound into one selected common character form, we implement this method using the Python with the re module. The function normalization () accepts the Actual\_post text in the corpus array, replace the characters into one common character form. The method returns the normalizing Actual\_post. The code for normalizing the Actual\_post in the dataset shown in Appendix

### Tokenization

To get token or word of the text in the dataset, we used a python module nltk. We used python splits function and nltk method for word tokenize. The output of this function is tokens or words, which is input for feature extraction methods.

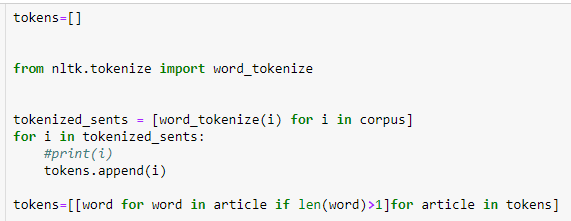


Figure 4.5:phyton code for tokenization

### Remove stop word

Words such as articles and some verbs are usually considered stop words because they don’t help us to find the context or the true meaning of a sentence. These are words that can be removed without any negative consequences to the final model that you are training.

Reducing the data set size is without any doubt a way of increasing performance. Training models takes time and if you have less tokens to be trained, the training time should decrease.

In addition, techniques such as TF-IDF give more value to rare words than to very repetitive tokens. So, let’s say that you are working on a problem for a bank about one of their products. The name of this product could be a word with high frequency on your corpus, therefore, you could consider it a stop word.

Common noisy words called ‘stop words’ are less important words when it comes to text feature extraction, they don’t contribute towards the actual meaning of a sentence and they only contribute towards feature dimensionality and may be discarded for better performance.

The researcher prepare the stop words from the given corpus that are mostly frequent words and that doesn’t give any meaning words are selected from the corpus .

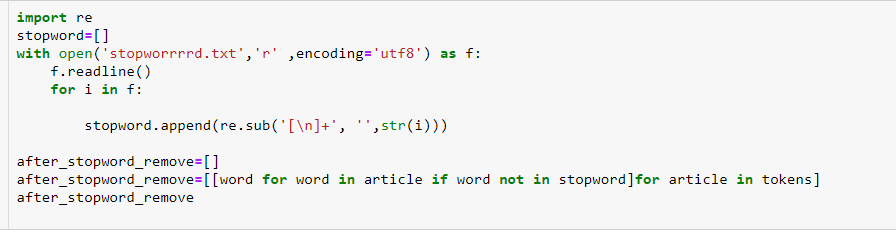


Figure 4.6:phyton code for stop word removal

### Stemming

After removing stop words, the next step is to transform the tokens into a standard form. Stemming simply is changing the words into their original form, and decreasing the number of word types or classes in the data. We use stemming to make classification faster and efficient. Furthermore, we use Hornmorpho that is a python library to analyze Amharic, oromifa and Tigrigna language. One part of analysis is changing the word to their root, which is the most commonly used stemming algorithms due to its accuracy. The code for stemming is shown in Appendix.

The above Amharic text pre-processing are used for content-based news article that are under Actua\_post column in the dataset. For social-context information handling missing value and categorical data has been done. According to this dataset there is no missing value .and directly goes to handling categorical data.

### Handling categorical data

Handling the categorical data is very important thing in the machine learning because categorical data’s are not a numerical data and the machine cannot understand easily so it must be converted to numerical value. In this paper we use one of handling categorical data method called one hot encoding .one hot encoding is One Hot Encoding is a process in the data processing that is applied to categorical data, to convert it into a binary vector representation for use in machine learning algorithms. This study uses pandas get.\_dummies () function to convert categorical to binary vector



Figure 4.7: python code for handling categorical data

Another variable of social context information is date. Date and time are rich source of information that can be used with machine learning model however these Date time variable do require some feature engineering to turn them in to numerical data .so in this paper use pandas DataFrame to create date time feature by splitting the date time into multiple column like day, month and year.



Figure 4.8: phyton code for splitting date to day month and year

### Feature extraction

#### N- Gram weighted by TFI-DF

In this study we apply varying word level N-gram analysis weighted by TF-IDF of Words feature extraction in the approach to detect fake news. The reason for this is because at different N-gram levels words take up a new meaning within a more or less restricted context. In line with that we use unigram, bigram and trigram word level feature extraction and use TfidfVectorizer () class of sci-kit learns package. This class converts a news articles of Actual post of the dataset to a matrix of TF-IDF features vectors. Which contain the word frequency and their importance in the dataset. For feature modeling with TF-IDF, we instantiate the *Tfidf Vectorizer* class and pass the stemmed news article in the dataset to fit\_transform() method of the class and use n-gram range to vectorized in the form of unigram, bigram and trigram word level .

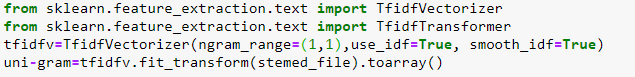


Figure 4.9: Python code for unigram feature extraction

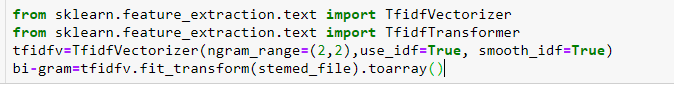


Figure 4.10: Python code for bigram feature extraction

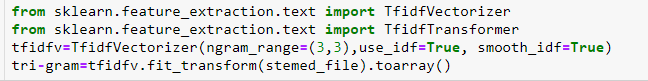


Figure 4.11:Python code for trigram feature extraction

#### Word 2vec

This study uses a python Genism module which is used to implement different embedding methods. Feature modeling with genism word2Vec is straightforward. First import and instantiate word2Vec class with necessary parameter and builds the vocabulary, and train Word2Vec model using the Actual\_post that are tokens in the dataset with four parameter such as feature size, window context, minimum word count and the downsample setting for frequent words.

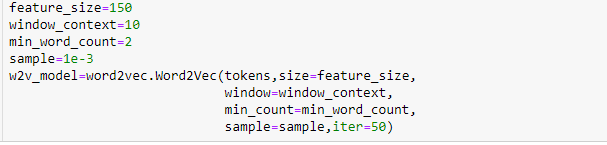


Figure 4.12: Python code for Wrd2vec feature extraction

We would need to get the document level embedding from each of the words present in each document. One strategy would be to average out the word embedding for each word in a document. This is an extremely useful strategy to get features for each document.

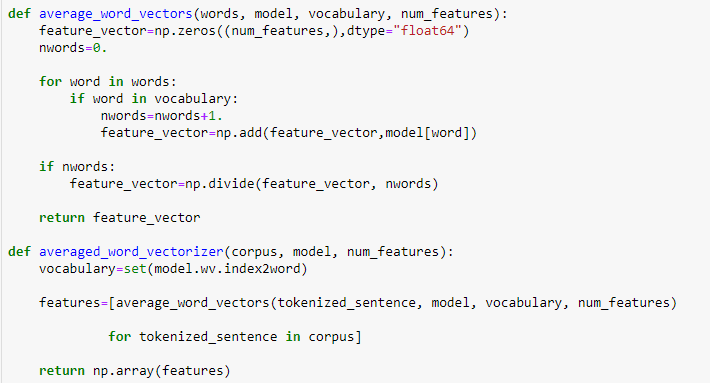


Figure 4.13:Python code for averaging word2vec feature extraction

#### Hybrid approach

The basic aim of this study is hybrid the content based approach and social context approach for detecting fake news .to make hybrid use both social context feature and content based feature for training.

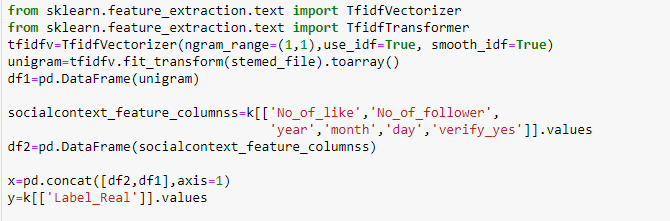


Figure 4.14:Python code for hybrid feature extractionof unigram and social-context

This hybrid approaches are worked for all content based features and social context features like bigram, trigram, and word 2vec to social context features.

After feature extraction the data splitting takes place by using sklean model selection package of train\_test split class “X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(x, y, test\_size=0.2)” code had been written. In this line of code it is possible to generalize that we are going to have two kinds of data. The first one is the one that we can use it to train which is X-train used in the machine learning and the second one is the data that is used to represent the test which is X-test by comparing the similarity that the train data and test data have we can conclude their performance. In the data splitting part we have used 0.2 to show the test size is taking 20% of the total dataset. ”x” means the extracted features, while “y” means the class label.



Figure 4.15:Python code for split train and test dataset

### Machine learning model

After the splitting the dataset as train and test ,build and train the model using three classification algorithm logistic regression , random forest and support vector machine. Those three algorithms build and trained by each features. To do that we use library package for modeling and metrics for model evaluation and train using the fit () method with the appropriate set parameter for each classifier instantiate the class.

**Logistic regression**

When the experiment is done using logistic regression the first thing is calling the library from sklearn. To do that we are going to use “from sklearn.linear\_model import LogisticRegression” this line of code import the model logistic regression from sklearn linear model. Then by specific what kind of classifier are we using ,in this case using (classifier=Logistic Regression()) with some parameters and give the X\_train (which is the features ) and the y\_train (which is a class label) like (classifier.fit(X\_train, y\_train)) then start the machine learning training.



Figure 4.16: Python code for logistic regression model

**Support vector machine**

When the experiment is done using support vector machine the first thing is calling the library from sklearn. To do that we are going to use “from sklearn.svm import LinearSVC” this line of code import the model LinearSVC from sklearn SVM model. Then by specific what kind of classifier are we using ,in this case using (classifier= Linear SVC ()) with some parameters and give the X\_train (which is the features ) and the y\_train (which is a class label) like (classifier.fit(X\_train, y\_train)) then start the machine learning training.

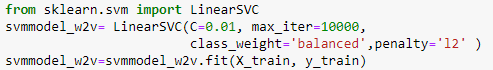


Figure 4.17: Python code for support vector machine model

**Random forest**

When the experiment is done using random forest the first thing is calling the library from sklearn. To do that we are going to use “from sklearn.ensemble import RandomForestClassifier” this line of code import the model RandomForestClassifier from sklearn ensemble model. Then by specific what kind of classifier are we using ,in this case using (classifier= RandomForestClassifier ()) with some parameters and give the X\_train (which is the features ) and the y\_train (which is a class label) like (classifier.fit(X\_train, y\_train)) then start the machine learning training.



Figure 4.18: Python code for random forest model

### Model testing and evaluation

After training the model, this study use confusion matrix, precision, recall and f1-score for evaluate and test the model. To do that we are using the sklearn classification\_report() and confusion\_matrix () method. To test the model and use confusion matrix first predict the test data based on the model then based on that calculate the score by using the predicted value and y\_test (which contain test class label) and plot the confusion matrix to evaluate the model.

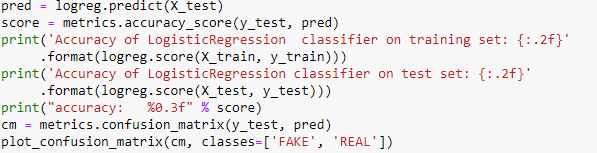


Figure 4.19: Python code for evaluation of confusion matrix for logistic regression model

For other evaluation methods like precision, recall, f1-score we use “from sklearn.metrics import classification\_report” this line of code import the model classification\_report from sklearn metrics and test based on predict and y\_test values and display precision, recall, and f1-score values for each classification model.



Figure 4.20: Python code for classification report

Finally, a prototype for detecting fake news is implemented using the selected best model then deployed using flask web services for it to be able to accept a new input actual post and other social context information and then return classification result.

## Experimental result

In this study total 8 experiments have been conducted with three machine learning algorithms. Experiment1, will focus on unigram weighted by TF-IDF feature extraction, Experiment2, will focus on bigram weighted by TF-IDF feature extraction, Experment3, will focus on trigram by TF-IDF feature extraction and Experiment 4, will focus on Word2Vec feature extraction. Experiment 5, will focus on hybrid of unigram weighted by TF-IDF and social context features. Experiment 6, will focus on hybrid of bigram weighted by TF-IDF and social context features. Experiment 7, will focus on hybrid of trigram weighted by TF-IDF and social context features. Experiment 8, will focus on hybrid of Word2Vec feature and social context features. All those 8 experiments are modeled with three LR, RF, and SVM consecutive algorithms.

### Experiment 1: Unigram weighted by TFIDF with three ML algorithms result

The first experiment is conducted content based unigram feature weighed by term frequency- inverse document frequency and modeled with three algorithm logistic regression, support vector machine, and random forest and each model evaluated based on confusion matrix, recall, precision, and f1-score. The accuracy and evaluation result of unigram TFIDF feature with three algorithms are shown in table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Unigram with TFIDF | LR | 93.4 | 93 | 94 | 94 | 93 | 93 | 93 |
| SVM | 89.8 | 86 | 94 | 94 | 85 | 90 | 90 |
| RF | 91.2 | 89 | 93 | 93 | 89 | 91 | 91 |

Table 4.1: Unigram weighted by TFIDF with three ML algorithms result

**Precision of logistic regression**

The precision is calculated individually for both labels which means for fake and real. The one which is labeled as fake we have got a precision of 0.93 which means to calculate we are going to divide the result that we got from true negatives (Actual\_post which are classified as fake and predicted as fake) with the summation of true positive (Actual\_post which are classified as real and predicted as real) and true negative (Actual\_post which are classified as fake and predicted as fake). For the one which is classified as real we have got a precision of 0.94 which means to calculate we are going to divide the result that we get from true positive (Actual\_post which are classified as real and predicted as real) with the summation of true positive (Actual\_post which are classified as real and predicted as real) and true negative (Actual\_post which are classified as fake and predicted as fake).

**Recall for logistic regression**

Like the precision the recall is calculated individually for both fake and real labels. In order to do that for the actual post which are labeled as fake the researcher have got a recall of 0.94 to do that, we are going to divide the result that we got from true positive (Actual post which are classified as real and predicted as real) with the summation of true positive (Actual post which are classified as real and predicted as real ) and false negative (Actual post which are classified as real while they are predicted as fake). For the one which are classified as real the researcher have got a recall of 0.93 which means to calculate we are going to divide the result we get from true negative (Actual post which are classified as fake and predicted as fake) with the summation of true positive (Actual post which are classified as real and predicted as real) and false negative (Actual post or posts which are classified as real while they are predicted by as fake).

**F1-score for logistic regression**

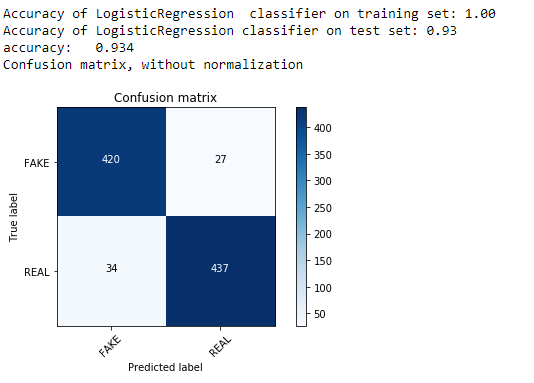
The F-Measure was also calculated for both fake and real labels. In order to calculate for actual posts which are fake it is going to multiply the recall and the precision and divided it by their summation finally multiplying it with 2 by doing this calculation for the one which are labeled as fake the F-Measure was 0.93. The one which is labeled as real the result for the F-measure was 0.93.

Figure 4.21: confusion matrix of unigram TFIDF feature with LR

**Confusion matrix for logistic regression**

After splitting the dataset out of the whole database size 4590, the shape of the testing set was 918 and the training set is 3672. So the confusion matrix shows that 420 of the testing set was predicted as fake when its actual value is labeled as fake. 34 of the testing set was classified as real but predicted as fake. 437 of the testing set has been classified as real and predicted as real. 27 of the testing set was classified as fake but predicted as real.

Finally the accuracy of unigram content based feature with logistic regression achieved 93.4%

**Precision for support vector machine**

Support vector machine for unigram have got a precision of 0.86 for the one which are classified as fake. Which means out of the testing dataset 86% of them have been positively classified as fake. A 0.94 of the total test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum that have been classified as truly positive and falsely positive.

**Recall for support vector machine**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that 0.94 have been correctly identified as fake and 0.85 as real.

**F-measure support vector machine**

For the F-Measure by taking the result of recall and precision. It was able to get 0.90 for the one which were classified as fake and 0.90 for the one that has been classified as real. This result have been affected with the one that we have got as a precision and recall.

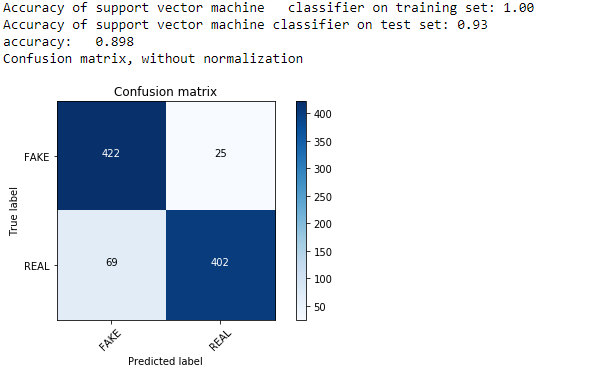


Figure 4.22: Confusion matrix of unigram TFIDF feature with SVM

**Confusion matrix for support vector machine**

After splitting the dataset out of the whole dataset size 4590, the shape of the testing set was 918 and the training set is 3672. So the confusion matrix shows that 422 of the testing set was predicted as fake when its actual value is labeled as fake. 69 of the testing set was classified as real but predicted as fake. 402 of the testing set has been classified as real and predicted as real.25 of the testing set was classified as fake but predicted as real.

Finally the accuracy of unigram content based feature with support vector machine achieved 89.9%

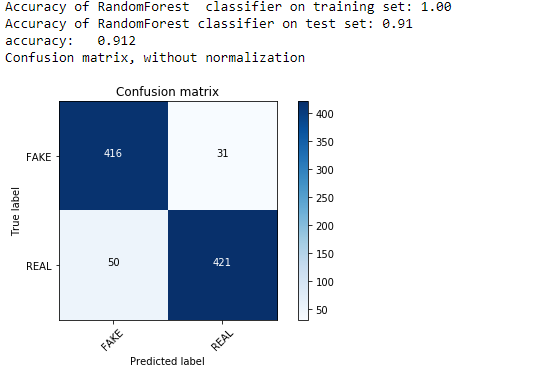


Figure 4.23: Confusion matrix unigram TFIDF feature with RF

After splitting the dataset out of the whole dataset size of 4590, the shape of the testing set was 918 and the training set is 3672. So the confusion matrix shows that 416 of the testing set was predicted as fake when its actual value is labeled as fake. 50 of the testing set was classified as real but predicted as fake. 421 of the testing set has been classified as real and predicted as real. 31 of the testing set was classified as fake but predicted as real.

### Experiment 2: Bigram weighted by TFIDF with three ML algorithms result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Bigram with TFIDF | LR | 89.3 | 84 | 96 | 97 | 82 | 90 | 89 |
| SVM | 81.7 | 74 | 95 | 96 | 68 | 84 | 79 |
| RF | 86.7 | 80 | 97 | 97 | 77 | 88 | 86 |

Table 4.2: Bigram weighted by TFIDF with three ML algorithms result

**Precision value using Bigram with TFIDF for LR, SVM and RF**

The logistic regression using Bigram with TFIDF have got a precision of 84% for the one which are classified as fake. Which means out of the testing set 84% of them have been positively classified as fake. A 96% of the test data have been classified as positively real. For support vector machine using bigram with TFIDF have got a precision of 74% for the one which are classified as fake. Which means out of the testing set 74% of them have been positively classified as fake. A 95% of the test data have been classified as positively real. For random forest using bigram with TFIDF have got a precision of 80% for the one which are classified as fake. Which means out of the testing set 80% of them have been positively classified as fake. A 97% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using Bigram with TFIDF for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 97% have been correctly identified as fake and 82% as real. For support vector machine 95% have been correctly identified as fake and 96% as real. For random forest 97% have been correctly identified as fake and 77% as real.

**F-measure value using Bigram with TFIDF for LR, SVM and RF**

For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 80% for the one which were classified as fake and 97% for the one that has been classified as real. For support vector machine was able to get 97% for the one which were classified as fake and 77% for the one that has been classified as real. For Random forest was able to get 88% for the one which were classified as fake and 86% for the one that has been classified as real.

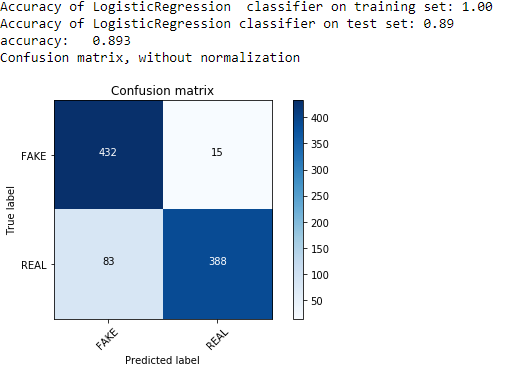


Figure 4.24: Confusion matrix of bigram TFIDF feature with LR

The confusion matrix shows that out of 918 testing set 432 of the testing set was predicted as fake when its actual value is labeled as fake. 15 of the testing set was classified as real but predicted as fake. 388 of the testing set has been classified as real and predicted as real. 83 of the testing set was classified as fake but predicted as real.

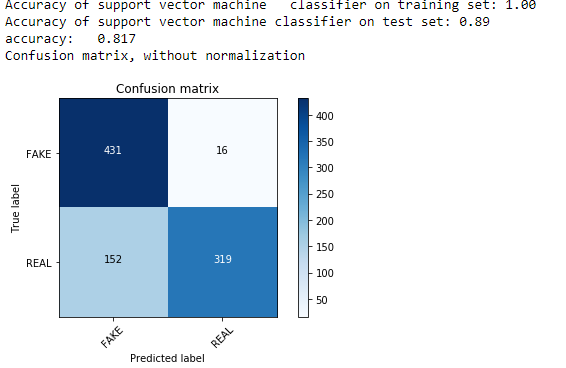


Figure 4.25: Confusion matrix bigram TFIDF feature with SVM

The confusion matrix shows that out of 918 testing set 431 of the testing set was predicted as fake when its actual value is labeled as fake. 16 of the testing set was classified as real but predicted as fake. 319 of the testing set has been classified as real and predicted as real. 152 of the testing set was classified as fake but predicted as real

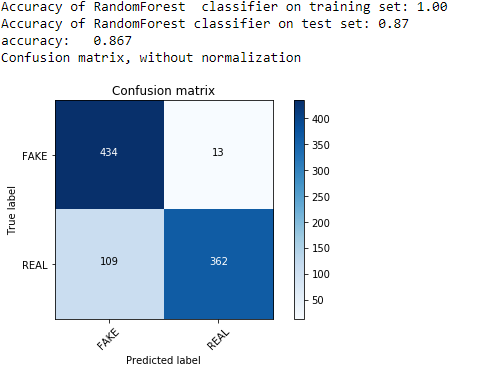


Figure 4.26: Confusion matrix bigram TFIDF feature with RF

The confusion matrix shows that out of 918 testing set 434 of the testing set was predicted as fake when its actual value is labeled as fake. 13 of the testing set was classified as real but predicted as fake. 362 of the testing set has been classified as real and predicted as real. 109 of the testing set was classified as fake but predicted as real.

### Experiment 3: Trigram weighted by TFIDF with three ML algorithms result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Trigram with TFIDF | LR | 76.6 | 68 | 99 | 99 | 55 | 81 | 71 |
| SVM | 74.9 | 66 | 98 | 99 | 52 | 79 | 62 |
| RF | 79.5 | 71 | 99 | 99 | 61 | 82 | 75 |

Table 4.3: Trigram weighted by TFIDF with three ML algorithms result

**Precision value using trigram with TFIDF for LR, SVM and RF**

The logistic regression using trigram with TFIDF have got a precision of 68% for the one which are classified as fake. Which means out of the testing set 68% of them have been positively classified as fake. A 99% of the test data have been classified as positively real. For support vector machine using trigram with TFIDF have got a precision of 66% for the one which are classified as fake. Which means out of the testing set 66% of them have been positively classified as fake. A 98% of the test data have been classified as positively real. For random forest using trigram with TFIDF have got a precision of 71% for the one which are classified as fake. Which means out of the testing set 71% of them have been positively classified as fake. A 99% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using trigram with TFIDF for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 97% have been correctly identified as fake and 82% as real. For support vector machine 95% have been correctly identified as fake and 96% as real. For random forest 97% have been correctly identified as fake and 61% as real.

**F-measure value using trigram with TFIDF for LR, SVM and RF**

For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 81% for the one which were classified as fake and 71% for the one that has been classified as real. For support vector machine was able to get 79% for the one which were classified as fake and 62% for the one that has been classified as real. For Random forest was able to get 82% for the one which were classified as fake and 65% for the one that has been classified as real.

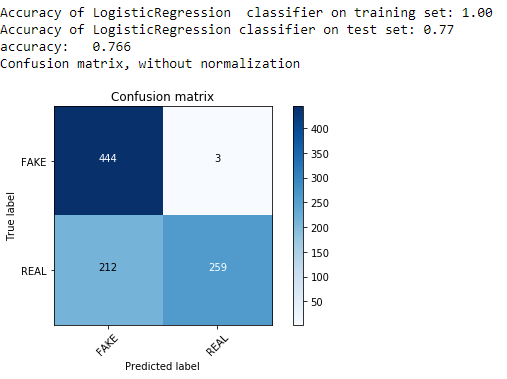


Figure 4.27: Confusion matrix trigram TFIDF feature with LR

The confusion matrix shows that out of 918 testing set 444 of the testing set was predicted as fake when its actual value is labeled as fake. 3 of the testing set was classified as real but predicted as fake. 259 of the testing set has been classified as real and predicted as real. 212 of the testing set was classified as fake but predicted as real.

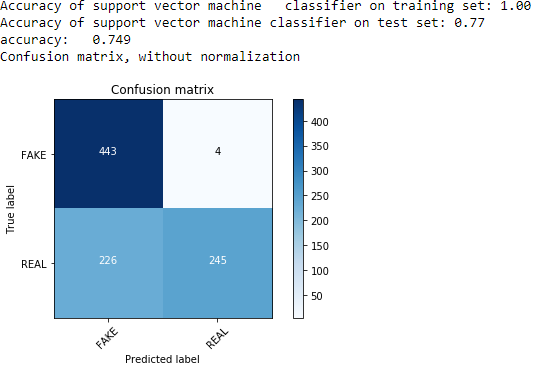


Figure 4.28: Confusion matrix trigram TFIDF feature with SVM

The confusion matrix shows that out of 918 testing set 443 of the testing set was predicted as fake when its actual value is labeled as fake. 4 of the testing set was classified as real but predicted as fake. 245 of the testing set has been classified as real and predicted as real. 226 of the testing set was classified as fake but predicted as real.

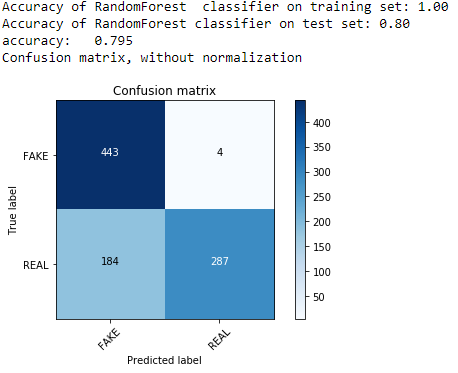


Figure 4.29: Confusion matrix trigram TFIDF feature with RF

The confusion matrix shows that out of 918 testing set 443 of the testing set was predicted as fake when its actual value is labeled as fake. 4 of the testing set was classified as real but predicted as fake. 287 of the testing set has been classified as real and predicted as real. 184 of the testing set was classified as fake but predicted as real.

### Experiment 4: Word2vec With three ML algorithms result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Word2vec | LR | 92.5 | 93 | 92 | 91 | 93 | 92 | 93 |
| SVM | 94.2 | 95 | 93 | 93 | 96 | 94 | 94 |
| RF | 94 | 95 | 93 | 93 | 95 | 94 | 94 |

Table 4.4:Word2vec With three ML algorithms result

**Precision value using Word2vec for LR, SVM and RF**

The logistic regression using Word2vechave got a precision of 93% for the one which are classified as fake. Which means out of the testing set 93% of them have been positively classified as fake. A 92% of the test data have been classified as positively real. For support vector machine using Word2vechave got a precision of 95% for the one which are classified as fake. Which means out of the testing set 95% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. For random forest using Word2vec have got a precision of 95% for the one which are classified as fake. Which means out of the testing set 95% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using trigram with TFIDF for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 91% have been correctly identified as fake and 93% as real. For support vector machine 93% have been correctly identified as fake and 96% as real. For random forest 93% have been correctly identified as fake and 95% as real.

**F-measure value using trigram with TFIDF for LR, SVM and RF**

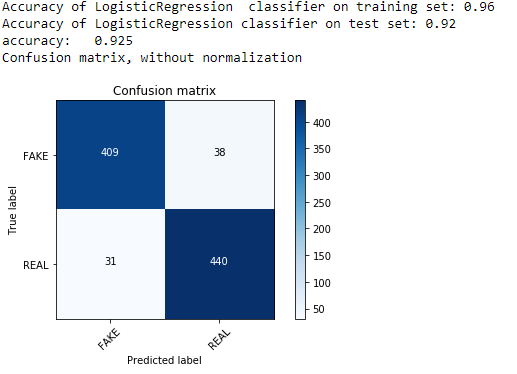
For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 92% for the one which were classified as fake and 93% for the one that has been classified as real. For support vector machine was able to get 94% for the one which were classified as fake and 94% for the one that has been classified as real. For Random forest was able to get 94% for the one which were classified as fake and 94% for the one that has been classified as real.

Figure 4.30: Confusion matrix Word2vec feature with LR

The confusion matrix shows that out of 918 testing set 409 of the testing set was predicted as fake when its actual value is labeled as fake. 38 of the testing set was classified as real but predicted as fake. 440 of the testing set has been classified as real and predicted as real. 31 of the testing set was classified as fake but predicted as real.

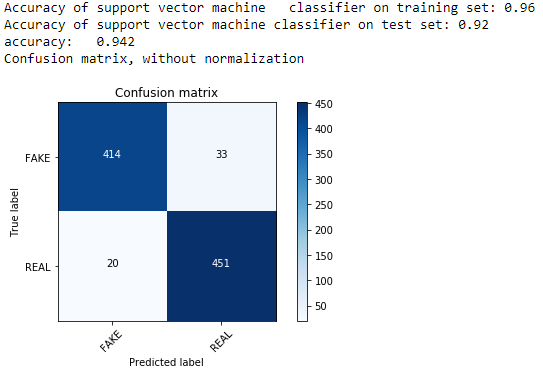


Figure 4.31: Confusion matrix Word2vec feature with SVM

The confusion matrix shows that out of 918 testing set 414 of the testing set was predicted as fake when its actual value is labeled as fake. 33 of the testing set was classified as real but predicted as fake. 451 of the testing set has been classified as real and predicted as real. 20 of the testing set was classified as fake but predicted as real.

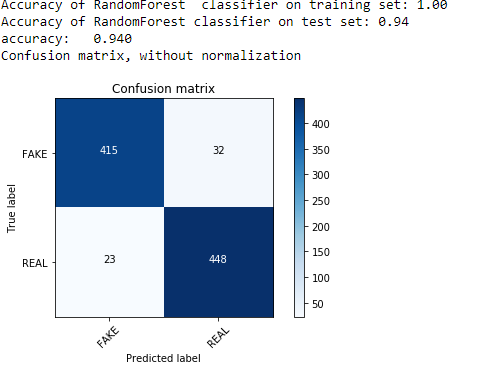


Figure 4.32: Confusion matrix Word2vec feature with RF

The confusion matrix shows that out of 918 testing set 415 of the testing set was predicted as fake when its actual value is labeled as fake. 32 of the testing set was classified as real but predicted as fake. 448 of the testing set has been classified as real and predicted as real. 23 of the testing set was classified as fake but predicted as real.

### Experiment 5: Hybrid of unigram and social context feature

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Unigram with TFIDF  +  Social context features | LR | 96.8 | 100 | 94 | 94 | 100 | 97 | 97 |
| SVM | 56.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.6 | 100 | 99 | 99 | 100 | 100 | 100 |

Table 4.5: Hybrid of unigram and social context feature

**Precision value using** **Hybrid of unigram and social context feature for LR, SVM and RF**

The logistic regression using Hybrid of unigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 94% of the test data have been classified as positively real. For support vector machine using Hybrid of unigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. For random forest using Hybrid of unigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 99% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using Hybrid of unigram and social context feature for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 94% have been correctly identified as fake and 100% as real. For support vector machine 93% have been correctly identified as fake and 100% as real. For random forest 99% have been correctly identified as fake and 100% as real.

**F-measure value using Hybrid of unigram and social context feature for LR, SVM and RF**

For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 97% for the one which were classified as fake and 97% for the one that has been classified as real. For support vector machine was able to get 96% for the one which were classified as fake and 97% for the one that has been classified as real. For Random forest was able to get 100% for the one which were classified as fake and 100% for the one that has been classified as real.

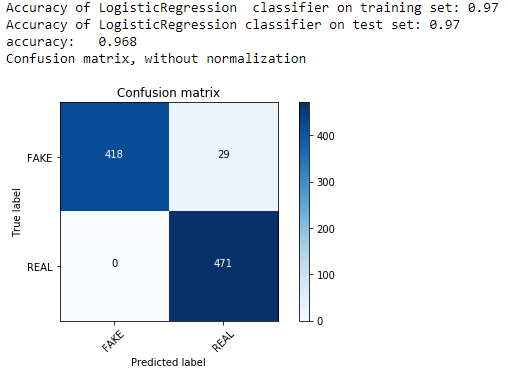


Figure 4.33: confusion matrix for Hybrid of unigram and social context feature with LR

The confusion matrix shows that out of 918 testing set 415 of the testing set was predicted as fake when its actual value is labeled as fake. 32 of the testing set was classified as real but predicted as fake. 448 of the testing set has been classified as real and predicted as real. 23 of the testing set was classified as fake but predicted as real.

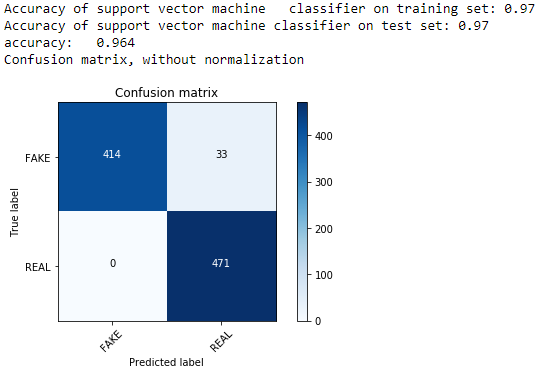


Figure 4.34: confusion matrix for Hybrid of unigram and social context feature with SVM

The confusion matrix shows that out of 918 testing set 415 of the testing set was predicted as fake when its actual value is labeled as fake. 32 of the testing set was classified as real but predicted as fake. 448 of the testing set has been classified as real and predicted as real. 23 of the testing set was classified as fake but predicted as real.

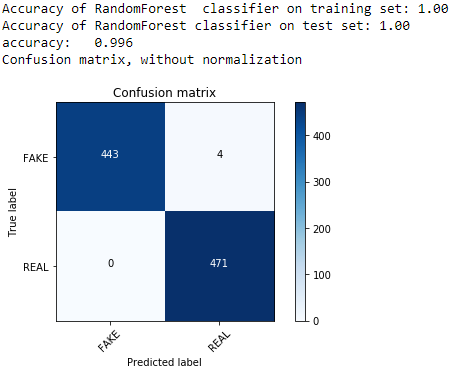


Figure 4.35: confusion matrix for Hybrid of unigram and social context feature with RF

The confusion matrix shows that out of 918 testing set 415 of the testing set was predicted as fake when its actual value is labeled as fake. 32 of the testing set was classified as real but predicted as fake. 448 of the testing set has been classified as real and predicted as real. 23 of the testing set was classified as fake but predicted as real.

### Experiment 6: Hybrid of Bigram and social context feature

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Bigram with TFIDF  +  Social context feature | LR | 96.8 | 100 | 94 | 94 | 100 | 97 | 97 |
| SVM | 96.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.3 | 100 | 95 | 95 | 100 | 99 | 99 |

Table 4.6: Hybrid of Bigram and social context feature with three ML algorithms result

**Precision value using** **Hybrid of bigram and social context feature for LR, SVM and RF**

The logistic regression using Hybrid of bigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 94% of the test data have been classified as positively real. For support vector machine using Hybrid of bigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. For random forest using Hybrid of bigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 95% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using Hybrid of bigram and social context feature for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 94% have been correctly identified as fake and 100% as real. For support vector machine 93% have been correctly identified as fake and 100% as real. For random forest 95% have been correctly identified as fake and 100% as real.

**F-measure value using Hybrid of bigram and social context feature for LR, SVM and RF**

For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 97% for the one which were classified as fake and 97% for the one that has been classified as real. For support vector machine was able to get 96% for the one which were classified as fake and 97% for the one that has been classified as real. For Random forest was able to get 99% for the one which were classified as fake and 99% for the one that has been classified as real.

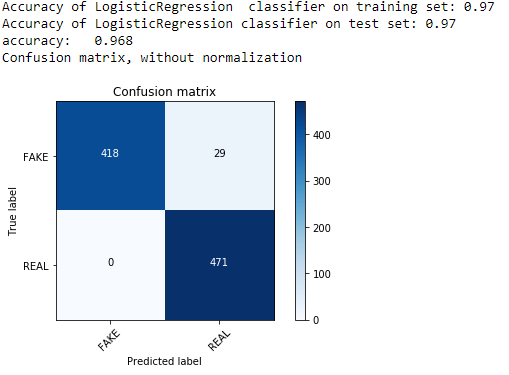


Figure 4.36 : confusion matrix for Hybrid of bigram and social context feature with LR

The confusion matrix shows that out of 918 testing set 418 of the testing set was predicted as fake when its actual value is labeled as fake. 29 of the testing set was classified as real but predicted as fake. 471 of the testing set has been classified as real and predicted as real. 0 of the testing set was classified as fake but predicted as real.

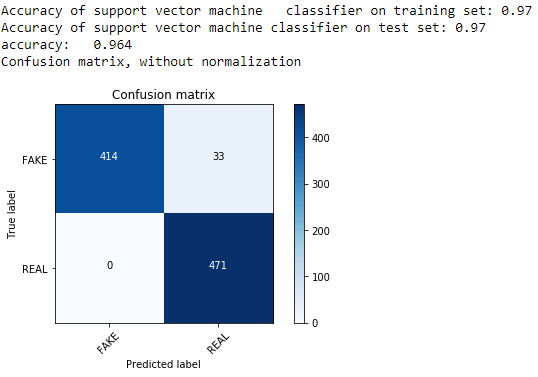


Figure 4.37: confusion matrix for Hybrid of bigram and social context feature with SVM

The confusion matrix shows that out of 918 testing set 414 of the testing set was predicted as fake when its actual value is labeled as fake. 33 of the testing set was classified as real but predicted as fake. 471 of the testing set has been classified as real and predicted as real. 0 of the testing set was classified as fake but predicted as real.

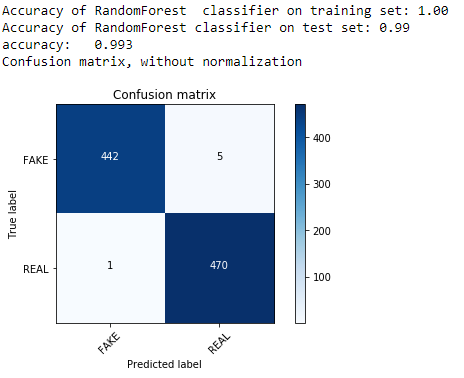


Figure 4.38: confusion matrix for Hybrid of bigram and social context feature with RF

The confusion matrix shows that out of 918 testing set 442 of the testing set was predicted as fake when its actual value is labeled as fake. 5 of the testing set was classified as real but predicted as fake. 470 of the testing set has been classified as real and predicted as real. 1 of the testing set was classified as fake but predicted as real.

### Experiment 7: Hybrid of Trigram and social context feature result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Trigram with TFIDF  +  Social context | LR | 96.8 | 100 | 94 | 94 | 100 | 97 | 97 |
| SVM | 96.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.3 | 100 | 99 | 99 | 100 | 99 | 99 |

Table 4.7: Hybrid of Trigram and social context feature result

**Precision value using** **Hybrid of trigram and social context feature for LR, SVM and RF**

The logistic regression using Hybrid of trigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 94% of the test data have been classified as positively real. For support vector machine using Hybrid of trigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. For random forest using Hybrid of trigram and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 99% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using Hybrid of bigram and social context feature for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 94% have been correctly identified as fake and 100% as real. For support vector machine 93% have been correctly identified as fake and 100% as real. For random forest 99% have been correctly identified as fake and 100% as real.

**F-measure value using Hybrid of bigram and social context feature for LR, SVM and RF**

For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 97% for the one which were classified as fake and 97% for the one that has been classified as real. For support vector machine was able to get 96% for the one which were classified as fake and 97% for the one that has been classified as real. For Random forest was able to get 99% for the one which were classified as fake and 99% for the one that has been classified as real.

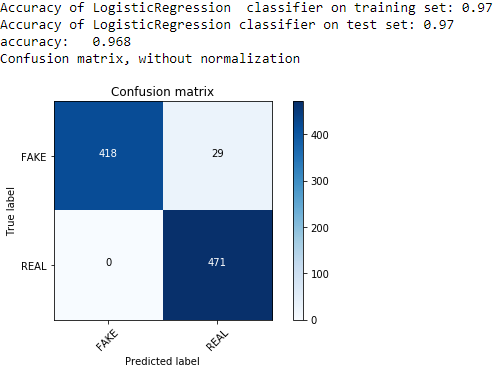


Figure 4.39: confusion matrix for Hybrid of trigram and social context feature with LR

The confusion matrix shows that out of 918 testing set 418 of the testing set was predicted as fake when its actual value is labeled as fake. 29 of the testing set was classified as real but predicted as fake. 471 of the testing set has been classified as real and predicted as real. 0 of the testing set was classified as fake but predicted as real.

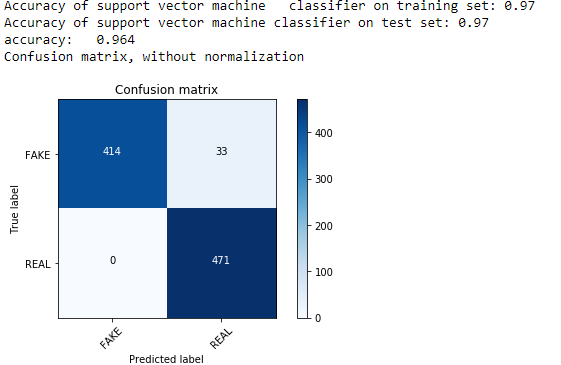


Figure 4.40: Confusion matrix for Hybrid of trigram and social context feature with SVM

The confusion matrix shows that out of 918 testing set 414 of the testing set was predicted as fake when its actual value is labeled as fake. 33 of the testing set was classified as real but predicted as fake. 471 of the testing set has been classified as real and predicted as real. 0 of the testing set was classified as fake but predicted as real.

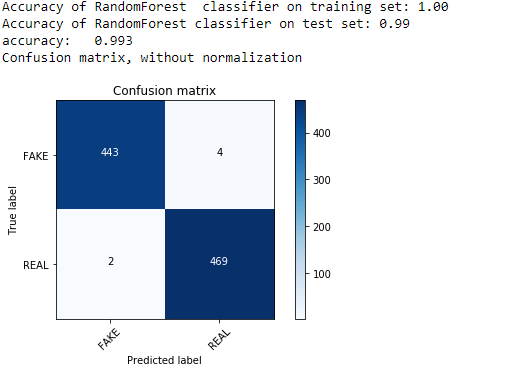


Figure 4.41: Confusion matrix for Hybrid of trigram and social context feature with RF

The confusion matrix shows that out of 918 testing set 443 of the testing set was predicted as fake when its actual value is labeled as fake. 4 of the testing set was classified as real but predicted as fake. 469 of the testing set has been classified as real and predicted as real. 2 of the testing set was classified as fake but predicted as real.

### Experiment 8: Hybrid of Word2vec and social context feature result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Word2vec  +  social context feature | LR | 93.8 | 94 | 93 | 93 | 94 | 94 | 94 |
| SVM | 96.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.7 | 100 | 99 | 99 | 100 | 100 | 100 |

Table 4.8: Hybrid of Word2vec and social context feature result

**Precision value using** **Hybrid of Word2vec** **and social context feature for LR, SVM and RF**

The logistic regression using Hybrid of Word2vec and social context feature have got a precision of 94% for the one which are classified as fake. Which means out of the testing set 94% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. For support vector machine using Hybrid of Word2vec and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 93% of the test data have been classified as positively real. For random forest using Hybrid of Word2vec and social context feature have got a precision of 100% for the one which are classified as fake. Which means out of the testing set 100% of them have been positively classified as fake. A 99% of the test data have been classified as positively real. In order to get this result we have taken the one that have been classified truly positive and divide it with the sum of that have been classified as truly positive and falsely positive.

**Recall** **value using Hybrid of Word2vec** **and social context feature for LR, SVM and RF**

The recall is calculated by dividing the one which are classified as truly positive by the summation of the one that has been classified as truly positive and falsely negative. While doing that for logistic regression 93% have been correctly identified as fake and 94% as real. For support vector machine 93% have been correctly identified as fake and 100% as real. For random forest 99% have been correctly identified as fake and 100% as real.

**F-measure value using Hybrid of Word2vec** **and social context feature for LR, SVM and RF**

For the F-Measure by taking the result of recall and precision. For logistic regression was able to get 94% for the one which were classified as fake and 94% for the one that has been classified as real. For support vector machine was able to get 96% for the one which were classified as fake and 97% for the one that has been classified as real. For Random forest was able to get 100% for the one which were classified as fake and 100% for the one that has been classified as real.

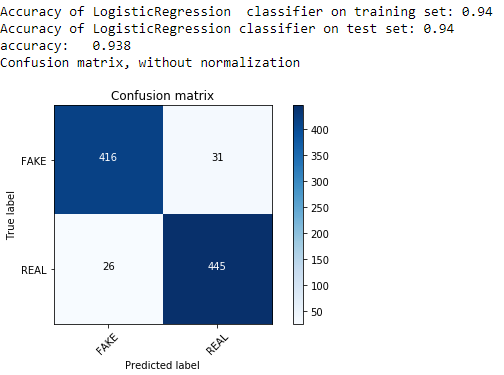


Figure 4.42: Confusion matrix of hybrid of Word2vec and social context feature with LR

The confusion matrix shows that out of 918 testing set 416 of the testing set was predicted as fake when its actual value is labeled as fake. 31 of the testing set was classified as real but predicted as fake. 445 of the testing set has been classified as real and predicted as real. 26 of the testing set was classified as fake but predicted as real.

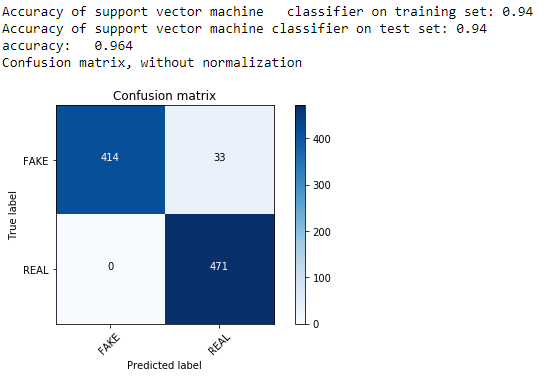


Figure 4.43: Confusion matrix of hybrid of Word2vec and social context feature with SVM

The confusion matrix shows that out of 918 testing set 414 of the testing set was predicted as fake when its actual value is labeled as fake. 33 of the testing set was classified as real but predicted as fake. 471 of the testing set has been classified as real and predicted as real. 0 of the testing set was classified as fake but predicted as real.

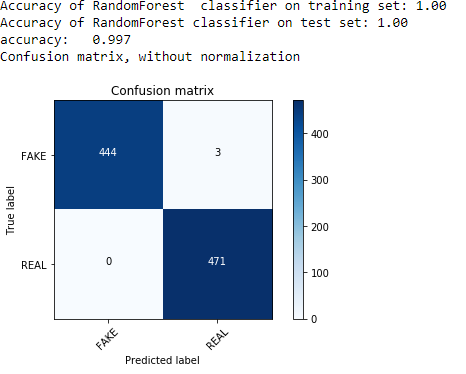


Figure 4.44: Confusion matrix of hybrid of Word2vec and social context feature with RF

The confusion matrix shows that out of 918 testing set 444 of the testing set was predicted as fake when its actual value is labeled as fake. 3 of the testing set was classified as real but predicted as fake. 471 of the testing set has been classified as real and predicted as real. 0 of the testing set was classified as fake but predicted as real.

### Summary of result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Features | Algorithm | Accuracy  (100%) | Precision (100% ) | | Recall (100%) | | F1-score (100) | |
| Fake | Real | Fake | Real | Fake | Real |
| Unigram with TFIDF | LR | 93.4 | 93 | 94 | 94 | 93 | 93 | 93 |
| SVM | 89.8 | 86 | 94 | 94 | 85 | 90 | 90 |
| RF | 91.2 | 89 | 93 | 93 | 89 | 91 | 91 |
| Bigram with TFIDF | LR | 89.3 | 84 | 96 | 97 | 82 | 90 | 89 |
| SVM | 81.7 | 74 | 95 | 96 | 68 | 84 | 79 |
| RF | 86.7 | 80 | 97 | 97 | 77 | 88 | 86 |
| Trigram with TFIDF | LR | 76.6 | 68 | 99 | 99 | 55 | 81 | 71 |
| SVM | 74.9 | 66 | 98 | 99 | 52 | 79 | 62 |
| RF | 79.5 | 71 | 99 | 99 | 61 | 82 | 75 |
| Word2vec | LR | 92.5 | 93 | 92 | 91 | 93 | 92 | 93 |
| SVM | 94.2 | 95 | 93 | 93 | 96 | 94 | 94 |
| RF | 94 | 95 | 93 | 93 | 95 | 94 | 94 |
| Unigram with TFIDF  +  Social context features | LR | 96.8 | 100 | 94 | 94 | 100 | 97 | 97 |
| SVM | 56.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.6 | 100 | 99 | 99 | 100 | 100 | 100 |
| Bigram with TFIDF  +  Social context feature | LR | 96.8 | 100 | 94 | 94 | 100 | 97 | 97 |
| SVM | 96.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.3 | 100 | 95 | 95 | 100 | 99 | 99 |
| Trigram with TFIDF  +  Social context feature | LR | 96.8 | 100 | 94 | 94 | 100 | 97 | 97 |
| SVM | 96.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| RF | 99.3 | 100 | 99 | 99 | 100 | 99 | 99 |
| **Word2vec**  **+**  **social context feature** | LR | 93.8 | 94 | 93 | 93 | 94 | 94 | 94 |
| SVM | 96.4 | 100 | 93 | 93 | 100 | 96 | 97 |
| **RF** | **99.7** | **100** | **99** | **99** | **100** | **100** | **100** |

Table 4.9: Summary of accuracy result

The accuracy result have shown us which feature extraction technique, approach and model perform well than other .The result that has been observed while doing the experiment shows which technique should be used to build the prototype which will receive a new data contain content news and social context. From the above 8 experiments, content based approach of word2vec feature extraction technique with support vector machine achieves the highest accuracy 94.2% among other context based feature extraction technique like N-gram levels with TFIDF. Because Word2Vec retains the semantic meaning of different words in a document. The context information is not lost. Another great benefit of Word2Vec approach is that the size of the embedding vector is very small. Each dimension in the embedding vector contains information about one aspect of the word. We do not need huge sparse vectors, unlike the bag of words and TF-IDF approaches. Whereas when we use hybrid approach which means integrate content based feature with social context feature the accuracy is higher than only content based features. When we incorporate social context features to the content based feature it performs higher that only content based feature. The proposed model achieves its highest accuracy when using hybrid of word2vec with social context and Random Forest classifier. The highest accuracy score is 99.7%

1. Conclusion and recommendation

## 5.1. Conclusion

This study proposed a solution for detecting fake news detection on social media especially on Facebook using hybrid approach of fake news detection mechanism and machine learning classification techniques for Amharic online news.

To effectively execute the study, it was essential to understand what fake news mean, fake news detection mechanism and approach and what the characteristics of Amharic language is and related work that are performed by different scholars to understand workflow and so on that are discussed on chapter two. This study proposed hybrid approach of fake news detection that integrate the content /linguistic feature of news content and social context that represent the news content posted page information like no of follower that posted page have, number of like that the posted page have, verification of that page, if the page is verified means it get the verification symbol or blue thick by the Facebook company, page created data .totally 4590 data’s are collected form 30 Facebook pages by using data\_miner tool that are posted within three months. Manually annotation of dataset as real or fake class performed by professional journalist. And different preprocessing, feature extraction (n -gram weighted by tfidf and word2vec), model training with machine learning algorithms (svm, logistic regression and random forest) and model testing has been performed. Finally the models are evaluated and compared by evaluation metric using confusion matrix, f1-score, precision and recall.

In this study total 8 experiments have been conducted with three machine learning algorithms. Experiment1, will focus on unigram weighted by TF-IDF feature extraction, Experiment2, will focus on bigram weighted by TF-IDF feature extraction, Experment3, will focus on trigram by TF-IDF feature extraction and Experiment 4, will focus on Word2Vec feature extraction. Experiment 5, will focus on hybrid of unigram weighted by TF-IDF and social context features. Experiment 6, will focus on hybrid of bigram weighted by TF-IDF and social context features. Experiment 7, will focus on hybrid of trigram weighted by TF-IDF and social context features. Experiment 8, will focus on hybrid of Word2Vec feature and social context features. All those 8 experiments are modeled with three LR, RF, and SVM consecutive algorithms.

The proposed model achieves its highest accuracy when using hybrid of word2vec with social context and Random Forest classifier. The highest accuracy score is 99.7%

## 5.2. Recommendation

In this thesis, one of the main and time-consuming problems was not getting a well-organized dataset that we could use to do the experiments. If this thing is available in some kind of library, it is possible to reduce and transfer the time that any researcher can waste when scrapping to the time it will be needed to train and change the technique over a given time.

There are several steps in the preprocessing steps, but the investigator did not get a well-organized stop word from the steps. So it is in the best interest of making all the stop words available for the future to improve some kind of machine learning.

There have been several forms of preprocessing that we need to pursue while training a given dataset. One of the main preprocessing results from a given dataset to a certain form of norm. As one of the preprocessing processes, changing the words to their root word will be better for machine learning techniques. There is no algorithm or built-in feature to stem the words for the language in the machine learning process for the Amharic language. Since the machine will think of words that do not stem and stem to have different meanings. So it would make a difference in the outcome if we had a built-in feature for this preprocessing.

A supervised learning algorithm with a text mining feature extraction approach was used for the proposed solution for content base news and other social context attributes used for model building. Using unsupervised or deep learning algorithm models, it is better to see the difference in output outcomes.

The last is that Ethiopia has many kinds of languages that are used to connect with these languages in all kinds of social media, given that false news can be disseminated with these languages in social media. Since only the Amharic language is considered by the researchers as one of the languages chosen for this thesis. In the future, it would be good to consider other languages like Tigregna, Oromifa, or Sidamegn to create a variety of detection methods.

# References

|  |  |
| --- | --- |
| [1] | M. Tsagkias, M. d. Rijke and W. Weerkamp, "Linking Online News and Social Media," *ACM,* 2011. |
| [2] | S. Tasim, M. M. Hossain and H. Mazumder, "Impact of rumors or misinformation on coronavirus disease (COVID-19) in social media," *TX77843,USA,* 2020. |
| [3] | B. Gottfried and E. Shearer, "News Use Across Social Media Platforms," *Pew Research Center, Sept 2017(News Use Across Social Media Platforms 2017,* 2017. |
| [4] | D. Lazer, M. Baum, Y. Benkler, A. Berinsky, K. Greenhill, F. Menczer, M. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, M. Schudson, S. Sloman, C. R. Sunstein, E. A. Thorson, D. J. Watts, M. Schudson, S. Sloman and Suns, "The science of fake news," *Science,* no. 359, p. 1094–1096, 2018. |
| [5] | Rubin, "Pragmatic and cultural considerations for deception detection in asian languages,," *Talip perspectives, guest editorial commentary:,* no. 13, p. 10:1–10:8., 2014. |
| [6] | K. Shu, A. Sliva, S. Wang, J. Tang and H. Liu, "Fake News Detection on Social Media: A Data Mining Perspective.," *KDD exploration newsletter,* 2017.. |
| [7] | X. Zhang and A. A. G. , "An overview of online fake news: Characterization, detection, and discussion," *Elsevier Ltd,* 2019. |
| [8] | D. K. Abeselom, "THE IMPACTS OF FAKE NEWS ON PEACE AND DEVELOPMENT IN THE WORLD:IN THE CASE OF ETHIOPIA," *Internal journal of current research ,* 2018. |
| [9] | S. Vosoughi, D. Roy and S. Aral, " The spread of true and false news online. ,," *Science,* no. 359, p. 2018, , 1146–1151. |
| [10] | A. Perrin., " Social media usage. .," *Pew Research Center,* 2015. |
| [11] | M. Aldwairi and A. Alwahedi, "Detecting Fake News in Social Media Networks," *Elsevier Ltd,* © 2018 . |
| [12] | L. Howell, " Digital wildfres in a hyperconnected world," *WEF Report,* p. 3:15–94, 2013. |
| [13] | C. F, J. Bond and B. M. & DePaulo, "Accuracy of deception judgments.," *Personality and Social Psychology Review, ,* no. 10, p. 214–234, 2006. |
| [14] | M. L. D. Vedova, E. Tacchini, S. Moret, G. Ballarin and M. a. L. d. Alfaro, "Automatic Online Fake News Detection Combining Content and Social Signals," *2018 22nd Conference of Open Innovations Association (FRUCT),* pp. pp. 272-279, 2018. |
| [15] | B.Riedel, I.Augenstein and S.Riedel, "A simple but tough-to-beat baseline for the Fake News Challenge stance detection task," *arXiv preprint arXiv:1707.03264,* 2017. |
| [16] | H. Ahmed, I. Traore and a. S. Saad, "Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques,," *“” in Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments, ser. Lecture Notes in Computer Science. Springer, Cham,,* pp. , pp. 127–138., Oct. 2017. |
| [17] | Á. I. Rodríguez and L. L. Iglesias, "FAKE NEWS DETECTION USING DEEP LEARNING," *CS.CL,* 2019. |
| [18] | K. Shu, S. Wang and a. H. Liu, "Exploiting Tri-Relationship for Fake News Detection," *arXiv:1712.07709 [cs],,* Dec. 2017. |
| [19] | M. Potthast, J. Kiese, K. Reinart, J. Bevendorff and a. B. Stein, "A Stylometric Inquiry into Hyper partisan and Fake News," *arXiv preprint arXiv:1702.05638,,* 2017. |
| [20] | G. B. M. L. D. V. S. M. a. L. d. A. E. Tacchini, "Some Like it Hoax: Automated Fake News Detection in Social Networks," *arXiv preprintarXiv:1704.07506,* 2017. |
| [21] | A. Tlahun and D. Beshah, "Fake news detection model using machine learning approach: the caseof Amharic news on social media," *unpublished masters thesis ,University of Gondar, Department of Information systems,* February 2020. |
| [22] | E. Tacchini, G. Ballarin, M. Vedova, Della, S. More and L. d. Alfato, "Some Like it Hoax: Automated Fake News Detection in Social Networks," *arXiv preprintarXiv:1704.07506,* 2017. |
| [23] | A. Tilahun, "Fake news detection model using machine learning approach:the case of Amharic news on social media," 2020. |
| [24] | S. Ahmed, K. H. and F. C. , "Combining Machine Learning with Knowledge Engineering to detect Fake News in Social Networks-a survey," *Proceedings of the AAAI,* 2019. |
| [25] | R. M. Silvaa, Roney L. S. Santosb, T. A. Almeida and T. A. S. , "Towards Automatically Filtering Fake News in Portuguese," *Expert Systems with Applications,* 2020. |
| [26] | K. Lotze, "Python programming language: Best resources for developers and managers," *Tech republic,* February 28, 2020. |
| [27] | T. McGonagle, "Fake news: False fears or real concerns?," *Netherlands Quarterly of Human Rights,* vol. 35(4), p. 203–209, 2017. |
| [28] | C. Castillo, M. Mendoza and a. B. Poblete., " Information credibility on twitter.," *In Proceedings of the 20th international conference on World wide web. ACM, ,* p. 675–684, 2011. |
| [29] | S. Kwon, M. Cha, K. Jung, W. Chen, Wang and Yajun, "Prominent features of rumor propagation in online social media. In Data Mining (ICDM),," *IEEE 13th International Conference on. IEEE,.,* p. 1103–1108, 2013. |
| [30] | S. W. a. H. L. Kai Shu, " Understanding User Profles on Social Media for Fake News Detection.," *IEEE Conference on Multimedia Information Processing and Retrieval (MIPR).,* In 2018. |
| [31] | K. Shu, A. Sliva, S. Wang, J. Tang and a. H. Liu., " Fake News Detection on Social Media: A Data Mining Perspective.," *KDD exploration newsletter ,* 2017.. |
| [32] | O. V. C. D. Emilio Ferrara, F. Menczer and a. A. Flammini., "The rise of social bots.," *Communications of the ACM, ,* 2016. |
| [33] | M. D. Vicario, G. Vivaldo, A. Bessi, F. Zollo, A. Scala, G. Caldarelli and a. W. Quattrociocchi., "Echo chambers: Emotional contagion and group polarization on facebook.," *Scientific Reports,* 2016.. |
| [34] | X. Zhou, R. Zafarani, K. Shu and a. H. Liu., "Fake News: Fundamental Theories, Detection Strategies and Challenges.," *In WSDM’,* 2019. |
| [35] | Z. Zhao, R. P and .. M. Q, "Enquiring minds: Early detection of rumors in social media from enquiry posts," *Proceedings of the 24th international conference on advanced visual interfaces,* 2015. |
| [36] | Conroy, Rubin and Chen, "Automatic deception detection: Methods for fnding fake news.," *Proceedings of the Association for Information Science and Technology,* 2015. |
| [37] | V. Raskin, " Linguistics and natural language processing. Machine translation: Theoretical and methodological issues," *Cambridge University Press,* no. Cambridge42–58, 1987. |
| [38] | H. Ahmed, "Detecting opinion spam and fake news using n-gram analysis and semantic similarity Ph.D. thesis," 2017. |
| [39] | P. M and Z. Z., " Tectomt: Modular NLP framework," *International conference on natural language processing.,* no. pringer293–304, 2010. |
| [40] | G. Y and L. O., "word2vec explained: Deriving Mikolov et al.’s negative-sampling word-embedding method," *ArXiv e-prints,,* 2014. |
| [41] | S. M, S. R and N. H., "Recurrent neural network based language model," vol. 10, p. 1045–1048, 2010. |
| [42] | S. I and V. O., " Sequence to sequence learning with neural networks," *Advances in neural information processing systems,* p. 3104–3112, 2014. |
| [43] | Feng and G. Hirst, "Detecting deceptive opinions with profle compatibility.," *Proceedings of the sixth international joint conference on natural language processing,* p. 338–346, 2013. |
| [44] | H. Ahmed, I. Traore and Saad, "Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques," *in Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments, ser. Lecture Notes in Computer Science. Springer, Cham,,* p. 127–138., October 2017. |
| [45] | S. Afroz, M. Brennan and R. Greenstadt, "Detecting hoaxes, frauds, and deception in writing style online.," *Security and privacy (sp),* no. IEEE461–475, 2012. |
| [46] | Horne and S. Adali, "This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news," *ArXiv eprints,* 2017. |
| [47] | Z. Jin, J. Cao, Y. Zhang and J. Luo, "News verifcation by exploiting conﬂicting social viewpoints in microblogs," *AAAI2972–2978,* 2016. |
| [48] | Banerjee, S. Feng, S. Kang and Y. Choi, "Key stroke patterns as prosody in digital writings: A case study with deceptive reviews and essays," *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP),* 2014. |
| [49] | A. Hussein, F. K. Ahmad and S. S. Kamaruddin, "Content-Social Based Features for Fake News Detection Model from Twitter," *International Journal of Advanced Trends in Computer Science and Engineering,* vol. 8, December 2019. |
| [50] | C. Castillo, B. Mendoza and Poblete, "Information credibility on twitter," *Proceedings of the 20th international conference on world wide we ACM,* p. 675–684, 2011. |
| [51] | D. Varol, E. Ferrara, A. Flammini and F. Menczer, "A system to evaluate social bots.," *Proceedings of the 25th international conference,* 2016. |
| [52] | E. Ferrara, O. Varol, C. Davis, F. Menczer and A. & Flammini, "The rise of social bots," *Communications of the ACM,* vol. 59(7), p. 96–104., 2016. |
| [53] | J. Zhao, N. Cao, Z. Wen, Y. Song, Y.-R. Lin and Collins, "Visual analysis of anomalous information spreading on social media," *IEEE Transactions on Visualization and Computer Graphics, ,* p. 1773–1782., 2014. |
| [54] | S. Yang, K. Shu, S. Wang, R. Gu, F. W and H. Liu., "Unsupervised Fake News Detection on Social Media: A Generative Approach.," *In AAAI’,* 2019. |
| [55] | S. Gianvecchio, M. Xie, Z. Wu and H. Wang, "Measurement and classifcation of humans and bots in internet chat.," *Usenix security symposium,* p. 155–170., 2008. |
| [56] | Z. Chu, S. Gianvecchio, H. Wang and S. Jajodia, "Detecting automation of twitter accounts: Are you a human, bot, or cyborg?," *IEEE Transactions on Dependable Iand Secure Computing,,* p. 811–824, 2012. |
| [57] | A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel and B. & Bhattacharjee, " Measurement and analysis of online social networks," *Proceedings of the 7th acm sigcommconference on internet measurement,* 2007. |
| [58] | E. Tacchini, G. Ballarin and M. L, "Some like it hoax: Automated fake news detection in social networks," *. arXiv preprint arXiv:1704.07506,,* 2017. |
| [59] | N. Ruchansky, S. Seo and Y. Liu, "CSI: A Hybrid Deep Model for Fake News Detection," *CIKM’17,* November 6-10, 2017. |
| [60] | Sejeong Kwon, M. Cha, Kyomin Jung and a. Y. W. WeiChen, " Prominent features of rumor propagation in online social media. 13, pages," *In ICDM. IEEE,* pp. 1103-1108, 2013. |
| [61] | A. Smola and S. Vishwanathan, Introduction to Machine Learning, Cambridge: Cambridge University Press, 2008. |
| [62] | T. O. Ayodele, "Types of Machine Learning Algorithms," in *Machine Learning Algorithms*. |
| [63] | R. Oshikawa, J. Qian and W. Y. Wang, "A Survey on Natural Language Processing for Fake News Detection," *arXiv,* 5 Mar 2020. |
| [64] | M. Aldwairi and A. Alwahedi, "Detecting fake news in social media networks," *Procedia Comput. Sci,* vol. 141, p. 215–222, 2018. |
| [65] | S. Kwon, M. Ch, K. Jung, W. Chen and Y. Wang, "Prominent features of rumor propagation in online social media," *Proceedings of the IEEE Thirteenth international conference on data mining ,* pp. 1003-1008, 2013. |
| [66] | L. Zeng, K. Starbird and E. Spiro, "classifying rumor stance in crisis-related social media messages,," *in: Proceedings of the Tenth International AAAI Conference on Web and Social Media,,* p. 747–750, 2016. |
| [67] | W. Ferreira and A. Vlachos, ", Emergent: a novel data-set for stance classification,," *in: Proceedings of the Conference of the North American Chapter of the association of linguestic .Human language technology ,* pp. 1163-1168, 2016. |
| [68] | A. Chua and S. Banerjee, "Linguistic predictors of rumor veracity on the internet," *in: Proceedings of the International Multi Conference of Engineers and computer science,* 2016. |
| [69] | M. Hardalov, I. Koychev and P. Nakov, "In search of credible news," *,in: Proceedings of the International Conference on Artificial Intelligence: Methodology,Systems and Application ,* 2016. |
| [70] | B. A. a. F. Getahun, "Event Modeling from Amharic news article," *Unpublished Master’s Thesis, Department of Computer Science, Addis Ababa University,,* February 2018. |
| [71] | M. A. a. S. H/Mariam, "Named Entity Recognition for Amharic Language," *Unpublished Master’s Thesis, Department of Computer Science, Addis Ababa University,* Nov, 2010.. |
| [72] | Z. Sintayehu, "Automatic Classification of Amharic News Items: The Case of Ethiopian News Agency," *School of Information Studies for Africa, Addis Ababa University, Addis Ababa,,* 2001. |
| [73] | M. Bender, J. Bowen and R. a. F. C. Cooper, " the Ethiopian writing system.," *Oxford University Press: London. ,* 1976. |
| [74] | M. A. a. Y. Assabie, "“Development of Amharic Morphological Analyzer Using Memory-Based Learning”,," *In Proceedings of the 9th International Conference on Natural Language Processing (PolTAL2014), Springer Lecture Notes in Artificial Intelligence (LNAI),,* Vols. 8686,, pp. 1-13,, Warsaw, Poland, 2014.. |
| [75] | A. Atelach, " Automatic Sentence Parsing for Amharic Text an Experiment Using Probabilistic Context Free Grammars.," *Addis Ababa University, Masters Thesis,,* 2002. |
| [76] | S. Teklu., " Automatic categorization of Amharic news text: A machine learning approach. Department of Information Science, Addis Ababa University, Masters Thesis,," *Department of Information Science, Addis Ababa University, Masters Thesis,,* 2003. |
| [77] | Y. A., " Automatic Amharic news text classification using Support Vector Machine approach," *Department of Information Science, Addis Ababa University, Master’s Thesis,,* 2007. |
| [78] | K. Shu, D. Mahudeswaran and H. Liu, "FakeNewsTracker: a tool for fake news collection, detection,and visualization," *Springer,* 13 october 2018. |
| [79] | D. Vedova, E. Tacchini, S. Moret, G. Ballarin and M. D. a. L. d. Alfaro, "Automatic Online Fake News Detection Combining Content and Social Signals," *22nd Conference of Open Innovations Association (FRUCT),* pp. 272-279, 2018. |
| [80] | Y.Yang, L.Zheng and J.W.Zhang, "CNN:Convolutional Neural Networks for Fake News Detection,," *[Online],Available:https://arxiv.org/abs/180 00749,,* August1–20,2018. |
| [81] | R. K. Kaliyar, A. Goswami, P. Narang and S. Sinha, "FNDNet- A Deep Convolutional Neural Network for Fake News Detection," *Cognitive Systems Research,* 2020. |
| [82] | P. Bourgonje, J. M. Schneider and G. Rehm, "From Clickbait to Fake News Detection:An Approach based on Detecting the Stance of Headlines to Articles," *EMNLP Workshop on Natural Language Processing meets Journalism,,* p. 84–89, September 7, 2017.. |
| [83] | O. F. Altunbey and B. Alatas, "Fake News Detection within Online Social Media Using Supervised Artificial Intelligence Algorithms," *Physica A,* 2019. |
| [84] | B. Shubham, B. Vijay, P. Jain and M. Chawla, "Natural Language Processing based Hybrid Model forDetecting Fake News UsingContent-Based Features and Social Features," *Information Engineering and Electronic Business,* vol. 4, pp. 1-10, 2019. |
| [85] | P. Harrington, "Machine Learning in Action," *IEEE,* vol. 37, no. 3, 2012. |
| [86] | D. Novakovi, "Evaluation of Classification Models in Machine Learning," *Theory Appl. Math. Comput. Sci.,* vol. 7, p. 39–46, 2017. |