2020-03-16

# SARCASM DETECTION MODEL FOR AMHARIC TEXT

Tsegaw, Miniybel

Wisdom at the source of the Blue Nile

**BAHIR DAR UNIVERSITY**

**BAHIR DAR INSTITUTE OF TECHNOLOGY**

**SCHOOL OF RESEARCH AND POSTGRADUATE STUDIES**

**FACULTY OF COMPUTING**

# SARCASM DETECTION MODEL FOR AMHARIC TEXT

**MSc. Thesis**

**By: - Miniybel Tsegaw Agidaw**

**Program: - Computer Science**

**BAHIR DAR, ETHIOPIA**

**JULY ,2019**

Wisdom at the source of the Blue Nile

BAHIR DAR UNIVERSITY

BAHIR DAR INSTITUTE OF TECHNOLOGY

SCHOOL OF RESEARCH AND POSTGRADUATE STUDIES

FACULTY OF COMPUTING

# SARCASM DETECTION MODEL FOR AMHARIC TEXT

By

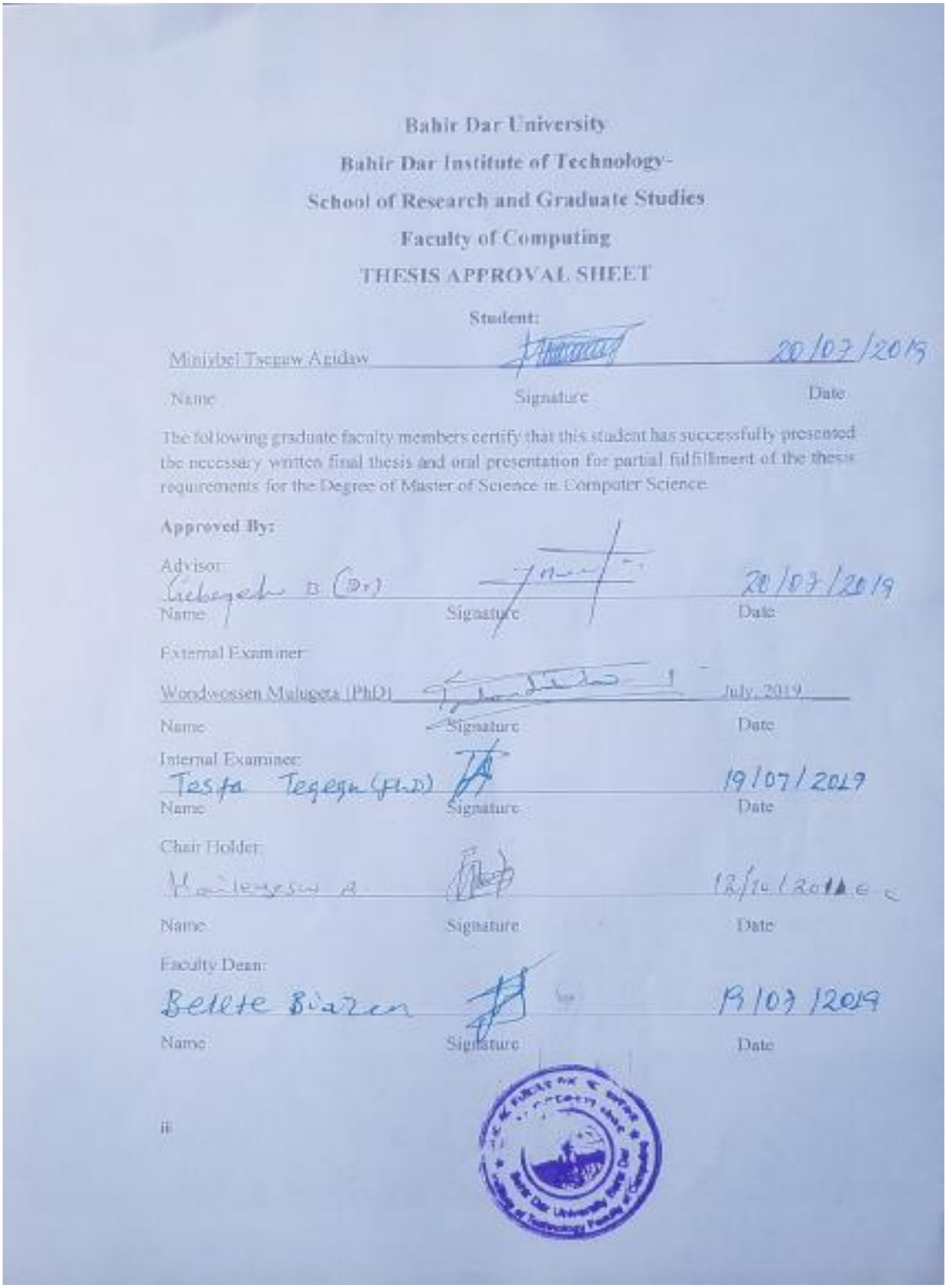
Miniybel Tsegaw Agidaw

A THESIS SUBMITTED TO THE SCHOOL OF RESEARCH AND GRADUATE STUDIES OF BAHIR DAR INSTITUTE OF TECHNOLOGY, BDU IN PARTIAL FULFILLMENT FOR THE DEGREE OF MASTERS OF SCIENCE IN COMPUTER SCIENCE.

Advisor Name: Gebeyehu Belay (Dr.)

Bahir Dar, Ethiopia

July 22, 2019



Bahir Dar University

Bahir Dar Institute of Technology-

School of Research and Graduate Studies

Faculty of Computing

THESIS APPROVAL SHEET

Student:

| | | |
|---|---|---|
| Minixbel Tsegaw Agidaw | | 20/07/2019 |
| Name | Signature | Date |

The following graduate faculty members certify that this student has successfully presented the necessary written final thesis and oral presentation for partial fulfillment of the thesis requirements for the Degree of Master of Science in Computer Science.

Approved By:

Advisor:

| | | |
|---|---|---|
| Gebeyehu B (Dr) | | 20/07/2019 |
| Name | Signature | Date |

External Examiner:

| | | |
|---|---|---|
| Wondwossen Mulugeta (PhD) | | July, 2019 |
| Name | Signature | Date |

Internal Examiner:

| | | |
|---|---|---|
| Tesfa Tegegn (PhD) | | 19/07/2019 |
| Name | Signature | Date |

Chair Holder:

| | | |
|---|---|---|
| Mailegesu A | | 12/11/2011 E.C |
| Name | Signature | Date |

Faculty Dean:

| | | |
|---|---|---|
| Belete Biazen | | 19/07/2019 |
| Name | Signature | Date |

ii

Bahir Dar University Institute of Technology Faculty of Computing

# Declaration

I, the undersigned, declare that the thesis comprises my own work. In compliance with internationally accepted practices, I have acknowledged and refereed all materials used in this work. I understand that non-adherence to the principles of academic honesty and integrity, misrepresentation/ fabrication of any idea/data/fact/source will constitute sufficient ground for disciplinary action by the University and can also evoke penal action from the sources which have not been properly cited or acknowledged.

Name of the student: Miniybel Tsegaw Agidaw Signature _____________

Date of submission: July 22, 2019

Place: Bahir Dar

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

Advisor Name: Gebeyehu Belay (Dr.)

Advisor's Signature: ________________________

Date: 7.22.2019

*To my father and mother*

# Acknowledgements

Firstly, I would like to praise and thank the supernatural power and creature of the entire universe almighty God for helping me to realize this work. Then, I was very lucky enough to have had the support of many people and without them; the completion of this thesis work would have been very difficult.

My special thanks go to my research advisor Dr. Gebeyehu Belay. I thank you my mentor for your encouragement, guidance, understanding and motivation throughout this thesis work. You have showed me how to tackle problems by investing your invaluable time. I will not forget ever, the way you approach to me and warn me to finish the thesis work on time. Really, you are a model advisor in advising the organization of thesis work and the flow of idea in order to better convincing the readers.

Finally, I must express my very profound gratitude to my parents and to my classmates for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing the proposed research. This accomplishment would not have been possible without them. Thank you

# Contents

# LIST OF ACRONYMS

| | |
|---|---|
| SVM | Support Vector Machine |
| NN | Neural Network |
| DT | Decision Tree |
| API | Application Programming Interface |
| NLP | Natural Language Processing |
| ASCII | American Standard Code for Information Interchange |
| RF | Random Forest |
| NLTK | Natural Language Toolkits |
| IR | Information Retrieval |
| TP | True Positive |
| FP | False Positive |
| FN | False Negative |

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ALGORITHMS AND EQUATIONS

# ABSTRACT

Sentiment Analysis is a technique to identify people's opinion, attitude, sentiment, and emotion towards any specific target such as individuals, events, topics, product, organizations; services sarcasm, entailment, etc. Sarcasm is a special kind of sentiment that comprise of words, which mean the opposite of what you really want to say (especially to insult or wit someone, to show irritation, or to be funny). People often express it verbally through the use of heavy tonal stress and certain gestural clues like rolling of the eyes. Which is obviously not available for expressing sarcasm in text? This is a crucial step to sentiment analysis, considering the prevalence and challenges of sarcasm in sentiment-bearing text. Sarcasm detection is the task of predicting sarcasm in text. Therefore, in this thesis we developed a model to detect the presence of sarcasm in Amharic texts. We used primary data's from "Abebe Tolla's" "Mitsetoch", "Silaqoch" and "Shimutoch" essay books and his official facebook blogs. The rest of the data is collected by using FacePager API from other Facebook blogs, and pages, which write about sarcasm elements, and to support the data any related reference such as magazines, newspapers and Amharic literature are used as a dataset. We used lexical (unigram), Semantic and Emoticons (smiley faces etc) features to extract different feature sets as useable inputs for Machine learning. Support Vector Machine (SVM), Neural Network (NN) and Random forest classifiers trained on simple lexical dictionary based approach is used to classify the sarcastic Amharic texts based on the features provided. An accuracy of 80.6%, 80.1 and 79% was obtained on the total collected datasets with the Support Vector Machine, Neural Network, and Random Forest classifier respectively. We found some strong features that characterize sarcastic texts. However, a combination of more subtle dictionary-based features proved more promising in identifying the various facets of sarcasm.

**Key Words and Phrases**: Sarcasm, Sentiment Analysis, Sarcasm detection,

# CHAPTER ONE

## 1.1 INTRODUCTION

The Free Dictionary defines sarcasm as a form of verbal irony that is intended to express contempt or ridicule. The figurative nature of sarcasm makes it an often-quoted challenge for sentiment analysis. (Bing Liu, 2010)

Therefore Sarcasm, in speech is multi-modal, involving tone, body-language, and gestures along with linguistic artifacts used in speech. Sarcasm in the text, on the other hand, is more restrictive when it comes to such non-linguistic modalities. This makes recognizing textual sarcasm more challenging for both humans and machines. Sarcasm detection in the text is a difficult problem and has only recently begun to be successfully examined as an automated natural language processing problem. (Abhijit Mishra, 2017)

Sarcasm detection plays an indispensable role in applications likes online review summarizers, dialog systems, recommendation systems, and sentiment analyzers. Which makes the process is challenging but also interesting to solve such a problem with traditional Natural Language Processing (NLP) tools and techniques (JOSHI, 2017).

As (Ellen Rilo, 2013) Developed a method for detecting sarcasm that examined the contrast between positive sentiments that is paired with a traditionally negative situation the work used tweets tagged with "sarcasm" as a gold standard. The juxtaposition of positive sentiment words with negative situations incorporated the essential idea of world context into detecting sarcasm. The goal is to learn phrases that are implicitly linked with negative sentiment. The algorithm learns by detecting a positive sentiment word (e.g. "love") as a seed word and find a negative situation that follows the word in the sarcastic tweet. Positive sentiment phrases are then learned by looking at adjacent negative situation phrases. These phrases and situations are then used to detect sarcasm in new tweets. Support Vector Machine (SVM) is used unigrams and bigrams of the learned phrases resulting in an F-score of 51%.

Sarcasm has the negative implied sentiment, but may not have a negative surface sentiment. A sarcastic sentence may carry positive surface sentiment (for example, ‘ጓደኞቼ ጋር መዝናናት ያስደስተኛል!’), negative surface sentiment (for example, ‘ያለፈዉ ዉድድር ላይ ጥሩ አልነበረም’ as a response to the criticism of an Olympic medalist) or no surface sentiment (for example, the idiomatic expression ‘የሀገሬ ንጉሥ ነኝ’ is used to express sarcasm). Since sarcasm implies sentiment, detection of sarcasm in a text is crucial to predicting the correct sentiment of the text.

Sarcasm detection in texts can be modeled as a binary document classification task. Two main sources of features have been used. First, most previous work extracts rich discrete features according to the texts content itself (Davidov et al., 2010; Tsur et al., 2010; Gonz´alez-Ib´anez et al., 2011; Reyes et al., 2012; Reyes et al., 2013; Riloff et al., 2013; Pt´aˇcek et al., 2014), including lexical unigrams, bigrams, word sentiment, punctuation marks, emoticons, quotes, character ngrams and pronunciations. Some of these work uses more sophisticated features, including POS tags, dependency-based tree structures, Brown clusters, and sentiment indicators, which depend on external resources. Overall, ngrams have been among the most useful features.

The challenges of sarcasm and the benefit of sarcasm detection to sentiment analysis for Amharic text  have led to an interest in sarcasm detection as a research problem. Sarcasm detection refers to computational approaches that predict, if a given text is sarcastic. Thus, the sentence ‘I love being ignored’ { መረሳት ያስደስተኛል} should be predicted as sarcastic, while the sentence ‘I love it when my son gives me a present’{“ ልጄ ስጦታ ሲሰጠኝ ደስ ይለኛል”} should be predicted as non-sarcastic. This problem is difficult because of nuanced ways in which sarcasm may be expressed. Sarcasm detection from text has now extended to different data forms and techniques. This synergy has resulted in interesting innovations for sarcasm detection in Amharic text.

## 1.2 MOTIVATION

Sarcasm is a sophisticated form of speech act and its recognition is one of the difficult tasks in Natural language processing (NLP). Sarcasm detection can benefit many NLP applications like review summarization, dialogue system, review ranking system. It is obvious that there is no simple rule or algorithm that can capture Sarcasm. This paper investigates the possibility of classifying sarcasm in text reliability and identifies typical textual features from social Media that are important for sarcasm in the process.

## 1.3 STATEMENT OF THE PROBLEM

Sarcasm is a form of speech act in which the speakers convey their message in an implicit way. The inherently ambiguous nature of sarcasm sometimes makes it hard even for humans to decide whether an utterance is sarcastic or not. Unlike a simple negation, a sarcastic sentence conveys a negative opinion using only positive or intensified positive words. The detection of sarcasm is therefore important, for the development and refinement of Sentiment Analysis.

Sarcasm detection in writing is challenging in part due to the lack of intonation and facial expressions. The human comprehension system can often spot a sarcastic sentiment, and reason about what makes it so. Recent advances in natural language sentence generation research have seen increasing interests in measuring negativity and positivity from the sentiment of words or phrases. However, accuracy and robustness of results are often affected by untruthful sentiments that are of sarcasm nature and this is often left untreated. Sarcasm detection is a very important process to filter out noisy data (in these case, sarcastic sentences) from training data inputs, which can be used for natural language sentence generation. [ (Chun-Che Peng, 2015)]

So there is a need for good sampling and classification techniques for these reviews and opinions. For this reason, many researches on sarcasm detection have been done and are being undertaken for English and other languages (Ellen Rilo, 2013).

In addition to that, sarcasm detection for Amharic texts has never been studied even though the amount of sarcasm texts on the web is increasing (Abreham, 2014). Therefore, this study investigates and aims to develop a sarcasm detection model for Amharic texts.

## 1.4 RESEARCH QUESTIONS

This research study tried to answer the following research questions.

1. Which features from the theories contribute most to sarcasm detection for Amharic text?
2. Does our modeling approach work for Amharic texts well?
3. What are the challenges of sarcasm detection for Amharic texts model developments?

## 1.5 OBJECTIVE

The general and specific objectives of this study are given below:

**General Objective:** the general objective of the research is to design and develop a sarcasm detection model for Amharic texts.

**Specific Objectives**: the specific objectives of the proposed research work are:

- To apply appropriate algorithms and classification approach to Amharic sarcasm texts.
- To understand the general structure of Amharic statements related to sarcasm and sentiments in order to identify sarcasm and Nonsarcasm.
- To gather a corpus of reliably labelled texts that can be used to train and accurately evaluate the relative performance of different classifiers.
- To review related works and state of the art in Sarcasm detection to have conceptual understanding and the general principle of how it works.
- To experiment, construct and evaluate the model for Amharic sarcasm texts.

## 1.6 SCOPE AND LIMITATION

Sarcasm detection is a complex and recent research discipline that requires the effective analysis and processing of documents. The system is designed to analyze Amharic sarcasm corpus collected from Abebe Tolla review and identify the text into sarcasm and Nonsarcasm.

The scope of this research is:

- Focused to sarcasm detection (only sarcasm, or not sarcasm) classification.
- Domain-independence is one of the biggest problems in machine learning and classification (Abreham, 2014).The system works for domain specific only for Abebe Tolla books review domain.
- The sentiment analysis holder identification and reasons for positive and negative classifications are not covered in this research work.
- Because of their complicated nature, Amharic proverbs “ቅኔያዊ አነጋገር” are out of the scope of this research work.

## 1.7 METHODOLOGY

### 1.7.1 Data collection and preparation

An important part when working with text classification is getting hold of good quality data, which is difficult in the case of Sarcasm text. It is often necessary to annotate data manually. Most of the datasets used for conducting the experiment are manually collected from ‘Abebe Tolla’s’ “Shimutoch” and “Mitsetoch 1,2” books and from his Facebook blogs. The rest of the dataset is collected from any related reference such as magazines; newspapers and Amharic literature are used.

We have used supervised machine learning approaches, which needs to have primary sentences labeled as sarcastic or as non-sarcastic so that our classifier can build the model.

### 1.7.2 Model development technique and tools

In this research, we used a supervised approach, which is one of the well-known techniques for sentiment classification (Belete, 2013).Five main processes (phases) are employed in applying sarcasm detection for Amharic texts.

1. Preprocessing: - the text of a document has to be converted into data that a ML algorithm can analyze. Since Amharic writing system has homophone characters which mean characters with the same sound have different symbols, such type of inconsistency in writing words were handled by replacing characters of the same sound by a common symbol using normalization. The text is broken down into discrete units using tokenization and then several operations are applied for removal of stop words, removal of punctuation, and stemming for effectiveness and efficiency improvements. This process has been done using python 3.6. We initiated to use Python because; Python's syntax is clear and readable. The way Python's syntax is organized imposes some order to programmers. Experts and beginners can easily understand the code. Because the block structures in Python are defined by indentations, it has much less likely to have bugs in

codes caused by incorrect indentation. It is simple to get support and fast to code**.** Python provides fast feedback in several ways. Python also encourages program reusability by implementing modules and packages. A large set of modules has already been developed and is provided as the standard python library, which is part of the Python distribution. One can easily share functionality between your programs by breaking the programs into modules, and reusing the modules as components of other programs. Python is also portable programming language. (K. D.Lee, 2011)

2. The second step creating lexical words by using unigram words. This lexicon construction is concerned with building a dictionary of sarcasm sentence and assigning prior polarity value. Since we applied supervised techniques, the preparation of training data set annotating by manually.

3. Weight assignment and propagation in this step, every polarity word and modifier get the initial weight defined in the sentiment lexicon. If the word is linked to a modifier, the polarity value is multiplied by a coefficient (Xiaoying, 2009.) Or some value is added to the initial value (Inkpen, 2006.)

4. Feature selection (FS) is the important step in text classification; feature selection is to construct vector space, which improves the scalability, efficiency and accuracy of a text classifier. The main idea of feature selection is to select subset of features from the original data. Feature selection is performed by keeping the words with highest score according to predetermined measure of the importance of the word. In next step, before using ML methods in text categorization, it is essential to choose which features are the most suited for this task. In text categorization, FS algorithms are a strategy to make text classifiers more efficient and accurate. It will attempt to reduce the dimension considered in a task so as to improve performance on some dependent measures. If the dimensionality of a domain expands, the number of features will increase (Varela, 2012).

   The features considered for detecting sarcasm comes from the Amharic text that was conducted in the beginning of this study and are based on different aspects that we believe may display sarcasm. All the different features are grouped together and always tested together since we believe that they represent similar aspects. In total we have three groups of features:

3 Lexical (vocabulary)

4 Semantic and

5 Punctuation (emoji)

In total we have 8 features and an additional number of unigrams depending of the feature selection conducted after extracting all the features.

5. The final step is applying machine learning algorithms to classify Amharic sarcasm texts into pre specified categories (sarcasm and Nonsarcasm) by using Natural language processing toolkit (NLTK).

Three machine learning algorithms were employed for classify the collected data as sarcasm and Nonsarcasm such as Support Vector Machine, Neural Network and Random Forest. Testing with more than one classification algorithms were provided comparison clues for determining best performed algorithms for Amharic sarcasm text in the domain because of many research works in sentiment analysis achieved high performance using them.

**Support Vector Machine (SVM):-** SVM are a set of supervised learning methods for machine learning that can be used for classification, anomaly detection and regression. One of SVMs strong suits is that they are very efficient when your data is high dimensional and also effective in cases where number of dimensions are greater than the number of samples. SVMs are especially good to solve text and hypertext categorization since the application decreases the use of label training occasions (Witten, 2005). Disadvantages with SVM are that they do not provide an estimate of the probability. To get these a calculation of an expensive cross-validation may be performed. In cases when the number of samples is significantly less than the number of features is likely that the method gives a poor result.

**Artificial Neural Network (ANN)** The recent vast research activities in classification have established that neural networks are a promising alternative to various conventional classification methods. The advantage of neural networks lies in the following theoretical aspects. First, neural networks are data driven self-adaptive methods in that they can adjust themselves to the data without any explicit specification of functional or distributional form for the underlying model. Second, they are universal functional approximates in that neural networks can approximate any function with arbitrary accuracy

**Random Forest (RF)** A random forest is a Meta estimator that fits a number of decision tree classifier on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

This research needs a programming language that is convenient to use for particular machine learning methods. Python is a versatile programming language that supports both object oriented programming and functional programming. It provides wide spectra of libraries that supports a lot of different programming tasks and is also open source. The libraries are easy to install and well documented. One more advantage with Python is that it is easy to read because the use of indents and new lines instead of more brackets. The libraries described below are some of the most common that have been used in the research. More information about Python and the libraries can be found at Pythons own webpage.

- **Numpy** gives support for high-level mathematical functions, large arrays and matrices. When using linear algebra and random number capabilities are Numpy very useful. Numpy also has utilities for integrating FORTRAN and C/C++ code.
- **Scikit-learn** are a Machine Learning library that Google started working on in 2007. The first public release was in 2010 and has been updated continuously. It is built on the math packages numpy and scipy. Among many other features it includes a variety of regression, classification and clustering algorithms such as naive Bayes, random forests, support vector machines, boosting algorithms etc. (Pedregosa, 2011)

- **NLTK** is an abbreviation for Natural Language Toolkits; it is a Python library that is useful when you work with human language data. The library consists of more than 50 different lexical assets and corpora including such as WordNet, parsing and tagging. NLTK supports assignments of varying difficulty and scope. NLTK also provides students with a flexible framework for advanced projects, such as developing a multi-component system, by integrating and extending NLTK components, and adding on entirely new components. Here NLTK helps by providing standard implementations of all the basic data structures and algorithms, interfaces to standard corpora, substantial corpus samples, and a flexible and extensible architecture. Thus, as we have seen, NLTK offers a fresh approach to NLP pedagogy, in which theoretical content is tightly integrated with application. (Steven Bird, 2008)

### 1.7.3 Evaluation

To assess the effectiveness of the proposed model testing corpus of sarcasms has been prepared and classified in correct classes specifically sarcasm and Nonsarcasm. Evaluation has two primary functions. Primarily, it helps to predict how well the final model will work. Secondly, evaluation is an integral part of many learning methods and helps to explore the model that best represents the training data.

The different classification models developed in this research were evaluated using classifier accuracy in test dataset. Similarly for the evaluation of opinion classification using regular expression is usually measured by precision, recall and F-measure. Precision is the fraction of relevant retrieved instances, while recall is the fraction of retrieved relevant instances. F-measure also used to fairly treat recall and precision which is brings recall and precision into a single measure. Therefore precision, recall and F-measure had been used based on an understanding and measure of relevance.

**Error Estimation**

Some interesting estimations in classification problems are accuracy, precision and recall. In statistical terms recall the ratio between the number of true positives and the sum of true positives and negative positives. Precision is the ratio between the true positive and the sum of true positives and false positives. Finally the F1-score is an accuracy measure that can be interpreted as weighted mean of the precision and recall:

## 1.8 SIGNIFICANCE OF THE STUDY

Detecting sarcasm is a very important field in Natural Language Processing. Sentiment analysis and text summarization taking sarcastic statements literally might result in a completely incorrect analysis of the data. Sarcasm detection can therefore, help to improve NLP systems like product review summarization (text summarization), brand monitoring, dialogue systems, and sentiment analysis. Sarcasm detection can also help in conflict resolution - Many conflicts arise between writers and politicians/celebrities due to a misunderstanding of sarcasm in written text, which can be avoided through effective Sarcasm Detection.

In the current business and political situations, knowing what other people think is a determinant factor in decision making. The results of this research can be used as an input to the development of full-fledged sentiment analysis system for Amharic language or any other Ethiopian languages that make use of the same Ethiopic alphabets such as Tigrinya, Guragignya, and others. Hence, the Amharic sarcasm detection model can be used for different purposes. Some of them are:

- Business and organizations (product review mining and service analysis, market intelligence) can use the system to reduce the money spent to find consumer’s sentiment and opinions.
- Individuals (who are interested in other’s opinion, can use it when purchasing a product, using a service or finding opinions on political topics).
- Government intelligence can use the system for sarcasm detections (analysis sentiments) of people on a particular issue.

## 1.9 THESIS ORGANIZATION

This thesis is organized into five chapters consisting of Introduction, Literature review, Sarcasm detection techniques and algorithms, Experimentation and Evaluation metrics and Conclusion and Recommendations.

The first chapter gives the general introduction of the thesis that contains an overview of the study, statement of the problem, objectives, methodology, Scope, and limitation of the study and Significance of the study. The second chapter presents reviews made on different literatures regarding sarcasm detection model together with its approaches and different machine learning techniques as well as previous related works for both English and Non-English documents in varies domains. Chapter Three illustrates sarcasm detection techniques and algorithms which contains corpus preparation and preprocessing, system architecture, feature selection methods, classification techniques, and performance measurement. The fourth chapter discusses the experimentation and discussion of the findings of how these experiments and methodologies were implemented. Finally, chapter five deals with the conclusion and the recommendation drawn from the findings of the study.

# CHAPTER TWO

# LITERATURE REVIEW

## 2.1. INTRODUCTION

Sarcasm is well studied by psychologists, behavioral scientists, and linguists (Rajadesingan, 2015). However for text mining automatic detection of sarcastic sentences is a difficult task (González-Ibánez, 2011) and it has been addressed by few researchers. (B. Pang, 2008) Had provided a comprehensive study in the field of opinion mining, Tepperman et al. (2006) had identified the sarcasm based on 'yeah-right' pronunciation in spoken form. Kreuz and Caucci (2007) studied the influence of different lexical factors like interjections and punctuation symbols in recognizing sarcasm in written form. (Filatova., 2012) Presented a detailed description of sarcasm corpus creation with sarcasm annotations of Amazon product reviews

## 2.2 BASIC COMPONETNTS OF SARCASM

Sarcasm as a 6-tuple/components consisting of <S, H, C, u, p, p'>

Where: S = Speaker, H = Hearer/Listener, C = Context, u = Utterance, p = Literal Proposition, and p' = Intended Proposition. (Pexman, 2003)

The components can be read as 'Speaker S generates an utterance u in Context C meaning proposition p but intending that hearer H understands p'.

For example, if a teacher says to a student, "That's how assignments should be done!" and if the student knows that they have barely completed the assignment, they would understand the sarcasm. In context of the 6-tuple above, the properties of this sarcasm would be:

S: Teacher

H: Student

C: The student has not completed his/her assignment.

u: "That's how assignments should be done!"

p: The student has done a good job at the assignment.

p': The student has done a bad job at the assignment.

## 2.3 TYPES OF SARCASM

According to (Camp., 2012) showed that there are four types of sarcasm:

1. **Propositional**: In such situations, the statement appears to be a proposition but has an implicit sentiment involved. For example ‘Your plan sounds fantastic!’ is sentence may be interpreted as non-sarcastic, if the context is not understood.
2. **Embedded**: this type of sarcasm has an embedded incongruity in the form of words and phrases themselves. For example ‘John has turned out to be such a diplomat that no one takes him seriously’. The incongruity is embedded in the meaning of the word ‘diplomat’ and rest of the sentence.
3. **Like-prefixed**: A like-phrase provides an implied denial of the argument being made. For example, ‘Like you care!’ is a common sarcastic retort.
4. **Illocutionary**: this kind of sarcasm involves non-textual clues that indicate an attitude opposite to a sincere utterance. For example, rolling one’s eyes when saying ‘Yeah right’. In such cases, prosodic variations play a role.

## 2.4 SARCASM DETECTION TECHNIQUES

There are different approaches to the problem of sarcasm detection. The most commonly applied techniques for sarcasm detection are described as follows.

### 2.4.1 MACHINE LEARNING TECHNIQUES

Machine learning treats sentiment classification simply as a special case of topic based categorization (with the two topics or classes being positive and negative sentiment). The traditional topic based categorization attempts to sort documents according to their subject matter (e.g. sports vs. politics).A system capable of acquiring and integrating the knowledge automatically is referred to as machine learning. The systems that learn from analytical observation, training, experience, and other means, results in a system that can exhibit self-improvement, effectiveness and efficiency.

#### 2.4.1.1 Supervised learning

Supervised learning, the classes are predetermined. The classes are previously known. In a supervised learning, let the domain of instances be X and the domain of labels be Y. Let P(x, y) be an unknown joint probability distribution on instances and labels X × Y. Given a training sample $\{(x_i, y_i)\}$, supervised learning trains a function f: X → Y in some function family F, with the goal that f (x) predicts the true label y on future data x (Goldberg, 2009)

According to (O. Chapelle, 2006.) The goal of supervised learning is to learn a mapping from x to y, given a training set made of pairs $(x_i, y_i)$. Here, the $y_i \in Y$ are called the labels or targets of the examples $x_i$. The process in supervised learning is that a certain part of data will be labeled with known classifications. The machine learner's task is to search for patterns and construct mathematical models. These models then are evaluated on the basis of their predictive capacity in relation to measures of variance in the data itself.

Supervised learning is the current dominant technique for addressing sentiment analysis in text classification techniques includes Hidden Markov Models (HMM), Decision Trees, Maximum Entropy Models (ME), Support Vector Machines (SVM) and Random Forest (RF). These are all variants of the supervised learning approach, which typically feature a system that reads a large annotated corpus, memorizes lists of entities, and creates disambiguation rules based on discriminative features (D. Nadeau, 2007)

#### 2.4.1.2 Unsupervised learning

Unsupervised learning algorithms work on a training sample with *n* instances $\{\mathbf{x}_i\}_{i=1}^{n}$. There is no teacher providing supervision as to how individual instances should be handled, Common unsupervised learning tasks include, clustering, where the goal is to separate the *n* instances into groups, which identifies the few instances that are very different from the majority and dimensionality reduction, which aims to represent each instance with a lower dimensional feature vector while maintaining key characteristics of the training sample (Goldberg, 2009)

In unsupervised learning, there is no such supervisor and we only have input data. The aim is to find the regularities in the input. There is a structure to the input space such that certain patterns occur more often than others, and we want to see what generally happens and what does not (Alpaydin, 2004)

According to (Alpaydin, 2004) the goal of unsupervised learning is to group data into clusters. In fact, the basic task of unsupervised learning is to develop classification labels automatically. Unsupervised algorithms seek out similarity between pieces of data in order to determine whether they can be characterized as forming a group. These groups are termed clusters, and there is a whole family of clustering machine learning techniques.

### 2.4.2 LINGUISTIC TECHNIQUES

Sarcasm is a form of figurative language where the literal meaning of words does not hold, and instead the opposite interpretation is intended  (H Paul Grice, 1975). Sarcasm is closely related to irony - in fact, it is a form of irony. (Gibbs., 1994) State that 'verbal irony is recognized by literary scholars as a technique of using incongruity to suggest a distinction between reality and expectation'. They define two types of irony: verbal and situational. Verbal irony is irony that is expressed in words. For example, the sentence 'Your paper on grammar correction contains several grammatical errors.' is ironic. On the other hand, situational irony is irony that arises out of a situation. For example, a situation where a scientist discovers the cure for a disease but herself succumbs to the disease before being able to apply the cure is a situational irony.

(Raymond W Gibbs., 1994)  Refereed to sarcastic language as 'irony that is especially bitter and caustic' there are two components of this definition: (a) presence of irony, (b) being bitter. Both together are identifying features of sarcasm. For example, 'I could not make it big in Hollywood because my writing was not bad enough'. This example from (Aditya Joshi V. T., 2016) is sarcastic, because: (a) it contains an ironic statement that implies a writer in Hollywood would need to be bad at writing, (b) the appraisal in the statement is in fact bitter/contemptuous towards the entity 'Hollywood'.

(Chernet, 2014) Stated sarcasm as image building expressed on posting photo based on Abebe Tolla reviews. See the extract below which takes from his one of the essays

እንደ ወዳጄ ገለፃ ከሆነ አምባገነን መሪዎች በየአደባባዩ ፎቶግራፋቸውን ፤ በየጓዳ ጎድጓዳው ደግሞ ህዝባቸውን ይሰቅላሉ፡፡ (ምፀቶች ገፅ 63)

“*As my dear explained that dictators known by posting their photo in every square and hung their people in every hidden place.* (MITSETOCHI p.63)”

As it seen on the above extract Abebe told to his reader about the dictators who had striven to build their image in front of the public and also how they killed many people in the hidden place. On the other hand Abe expressed his advices to the government by saying this

“ህዝቡ ይወደኛል” ብሎ መዘናጋት ለጋዳፊም አልበጀም፡፡ ለማንኛውም ተዘጋጅቶ መጠበቅ ነው፡፡ (ምፀቶች ገፅ 110)

*“Believing the people by saying they love me even not used for Gaddfi. Being readiness is advisable. (MITSETOCH p. 110)”*

As it knows the Arab revolution took away Gaddfi from his power. But he was preaching the passion of the people towards him: they removed and killed him. The sarcastic aware the government not cheated by the people love.

In general, sarcasm is a verbal irony that has an intention to be mocking / ridiculing towards an entity. However, what context is required for the sarcasm to be understood forms a crucial component. Compare the sarcastic example ‘I love being ignored’ with another ‘I love solving math problems all day’. The former is likely to be sarcastic for all speakers. The latter is likely to be sarcastic for most speakers. However, for authors who do really enjoy math, the statement is not sarcastic. The sarcasm in the latter may be conveyed through an author’s context or paralinguistic cues (as in the case of illocutionary sarcasm). Thus, sarcasm understanding and automatic sarcasm detection are contingent on what information (or context) is known.

#### 2.4.2.1 Relationship with irony, deception, metaphor and humor

Sarcasm is related to other forms of incongruity or figurative language. Sarcasm has an element of ridicule that irony does not (Katz., 1998).

Deception also appears to be closely related to sarcasm. If a person says ‘I love this soup’, they could be speaking the truth (literal proposition), they could be lying (deception) or they could be sarcastic (sarcasm). The difference between a literal proposition and deception lays in the intention of the speaker while the difference between sarcasm and deception lies in shared knowledge between speaker and listener. If the speaker saw a fly floating on the soup, the statement above is likely to have a sarcastic intention. Whether or not the listener understands the sarcasm depends on whether the listener saw the fly in the soup and whether the listener believes that the presence of a fly in a soup makes it bad (Gibbs., 1994)

Sarcasm as a form of aggressive humor Thus, the peculiarity that distinguishes sarcasm from another form of incongruent expression, humor, is the element of mockery or ridicule. (Stefan Stieger, 2011)

(Gibbs., 1994) Distinguished between metaphor and sarcasm in terms of the plausibility of the statement they state that a metaphor is never literally plausible. For example, A says to B, ‘You are an elephant’ to imply that B has a good memory is metaphorical because a human being cannot literally be an elephant. However, sarcasm, as in the case of ‘You have a very good memory’ may be plausible for people with a good memory, but sarcastic if said to a forgetful person. These characteristics of sarcasm relate it to these linguistic expressions like humor or metaphor. It is also these characteristics such as incongruity, shared knowledge, plausibility, and ridicule that form the basis of my work in sarcasm detection for Amharic texts.

### 2.4.3 LEXICON BASED TECHNIQUES

In the lexical based technique, the definition of sentiment is based on the analysis of individual words and/or phrases; emotional dictionaries are often used: emotional lexical items from the dictionary are searched in the text, their sentiment weights are calculated, and some aggregated weight function is applied (Etstratios K., 2013)**.**When using the lexical approach there is no need for labeled data and the procedure of learning, and the decisions taken by the classifier can be easily explained. However, this usually requires powerful linguistic resources (e.g., emotional dictionary), which is not always available, in addition, it is difficult to take the context into account. Dictionaries for lexicon-based approaches can be created manually or automatically, using seed words to expand the list of words. Much of the lexicon-based research has focused on using adjectives as indicators of the semantic orientation of text (Etstratios K., 2013)**.**

## 2.5 SARCASM DETECTION APPROACHES

In this section, we described the approaches used for sarcasm detection. In general, approaches to sarcasm detection can be classified into rule-based, statistical and deep learning-based approaches

### 2.5.1 RULE-BASED APPROACH

Rule-based approaches attempt to identify sarcasm through specific evidence. This evidence is captured in the form of rules that rely on indicators of sarcasm. (Hao, 2010) Identified sarcasm in similes using Google searches in order to determine how likely a simile is. They present a 9-step approach where at each step/ rule, a simile is validated as non-sarcastic using the number of search results. To demonstrate the strength of their rules, they present an error analysis corresponding to each rule.

1. A simile is classified as non-ironic if there is lexical/ morphological similarity between: i) the vehicle and the ground (e.g., as manly as a man); ii) between the

vehicle and a synonym of the ground (e.g., as masculine/manly as a man); or iii) between the vehicle and an adjective that is a frequently conjoined with the ground as a co-descriptor (e.g., as cold [and snowy] as snow).

2. If the web frequency of "about as GROUND as a VEHICLE" is more than half that of "as GROUND as a VEHICLE" (i.e., the "about" form is predominant), then the simile is classified as ironic and noted as an ironic precedent.
3. If this simile is recognizable as a direct variation of an ironic precedent (see 2 above), then this simile is also classified as ironic.
4. If this simile is recognizable as an inverse variation of an ironic precedent (see 2 above), then this simile is inversely classified as non-ironic.
5. If the ad-hoc category pattern "GROUND * such as VEHICLE" is found on the web, then the simile is considered non-ironic and is noted as a non-ironic precedent.
6. If the simile is a direct variation of a non-ironic precedent, it is deemed non-ironic.
7. If the simile is an inverse variation of a non-ironic precedent, it is deemed ironic.
8. If the simile has a web-frequency of 10 or more, it is classified as non-ironic and is also noted as a non-ironic precedent.
9. If the simile has a web-frequency less than 10, it is classified as ironic.

The research work (Greenwood., 2014.) Proposed that hashtag sentiment is a key indicator of sarcasm Hash tags are often used by tweet authors to highlight sarcasm, and hence, if the sentiment expressed by a hashtag does not agree with rest of the tweet, the tweet is predicted as sarcastic. They use a hashtag tokenizer to split hash tags made of concatenated words.

As (Santosh Kumar Bharti, 2015) Presented two rule-based classifiers the first uses a parse–based lexicon generation algorithm that creates parse trees of sentences and identifies situation phrases that bear sentiment. If a negative phrase occurs in a positive sentence, then the sentence is predicted as sarcastic. The second algorithm aims to capture hyperbolic sarcasm (i.e., by using interjections (such as '(wow)'and intensifiers (such as 'absolutely') that occur together.

According to (Ellen Rilo, 2013) Presented rule-based classifiers that look for a positive verb and a negative situation phrase in a sentence the set of negative situation phrases are extracted using a well-structured, iterative algorithm that begins with a bootstrapped set of positive verbs and iteratively expands both the sets (namely, positive verbs and negative situation phrases). They experiment with different configurations of rules such as restricting the order of the verb and situation phrase.

### 2.5.2 FEATURE SET APPROACH

In this section, we review the set of features that have been reported for statistical sarcasm detection. Most approaches use bag-of-words as features. However, in addition to these, several other sets of features have been reported. Oren Tsur, 2010 designed pattern-based features that indicate the presence of discriminative patterns (such as ‘as fast as a snail’) as extracted from a large sarcasm-labeled corpus. To prevent overftting of patterns, these pattern-based features take real values based on three situations: exact match, partial overlap, and no match.

As (González-Ibánez, 2011) Used sentiment lexicon-based features and pragmatic features like emoticons and user mentions similarly, (Delia Iraz´u Herna´ndez Far´ıas, 2016) Used features derived from multiple affective lexicons such as AFINN, SentiWordNet, General Inquirer, etc. In addition, they also use features based on semantic similarity, emoticons, counter factuality, etc.

According to (Reyes, 2013) Introduced features related to ambiguity, unexpectedness, emotional scenario, etc. Ambiguity features cover structural, morphosyntactic, semantic ambiguity, while unexpectedness features measure semantic relatedness.

### 2.5.3 DEEP LEARNING-BASED APPROACHES

As architectures based on deep learning techniques gain popularity in Natural Language Programming (NLP) applications, a few such approaches have been reported for automatic sarcasm detection as well. (Aditya Joshi V. T., 2016) Used similarity between word embedding’s as features for sarcasm detection. They augment these word embedding-based features with features from four prior works. The inclusions of past features are Key because they observe that using the new features alone does not suffice for good performance.

# CHAPTER THREE
# SARCASM DETECTION DESIGN AND IMPLEMENTATION

## 3.1 INTRODUCTION

In this chapter, the design and implementation of the proposed sarcasm detection model for opinionated Amharic texts are described in detail. The proposed model has the following components: pre-processing, sentiment word detection for Amharic sarcasm texts, weight manipulation, polarity classification and polarity strength (post-polarity classification analysis). Each component is composed of sub components which are the building blocks of the system. Pre-processing is responsible for normalization of texts and words segmentation. In the sentiment words detection for Amharic sarcasm texts component, all possible sentiment words and contextual valence shifter terms are checked for existence in the Amharic word lexicon. The weight manipulation component contains sub systems: weight assignment and polarity propagation. After the weight manipulation is completed, the next step is the polarity classification of the texts.

The strength of the polarity (whether how much it is positive or negative) is rated in the post-classification analysis step. The sentiment word detection for Amharic sarcasm texts and weight manipulation activities are fully dependent on the lexicon of Amharic lexicon terms that contains opinion terms, punctuation marks and emojis tagged with readily interpretable values. The procedures of building the sentiment lexica, the types of lexicon, the guidelines and principles followed during the sentiment lexicon building process are also described in this chapter. In addition, tools used for implementing the system and the proposed algorithms are also presented.

## 3.2 ARCHITECTURE OF SARCASM DETECTION FROM AMHARIC TEXTS

The general architecture of sarcasm detection model development for Amharic texts is given in Figure 3-1. The architecture has six major components; these are collected document, Document Preprocessing, Amharic word lexicon corpus preparing, sentiment word detection with weight assignment and Feature selection and Classification task.

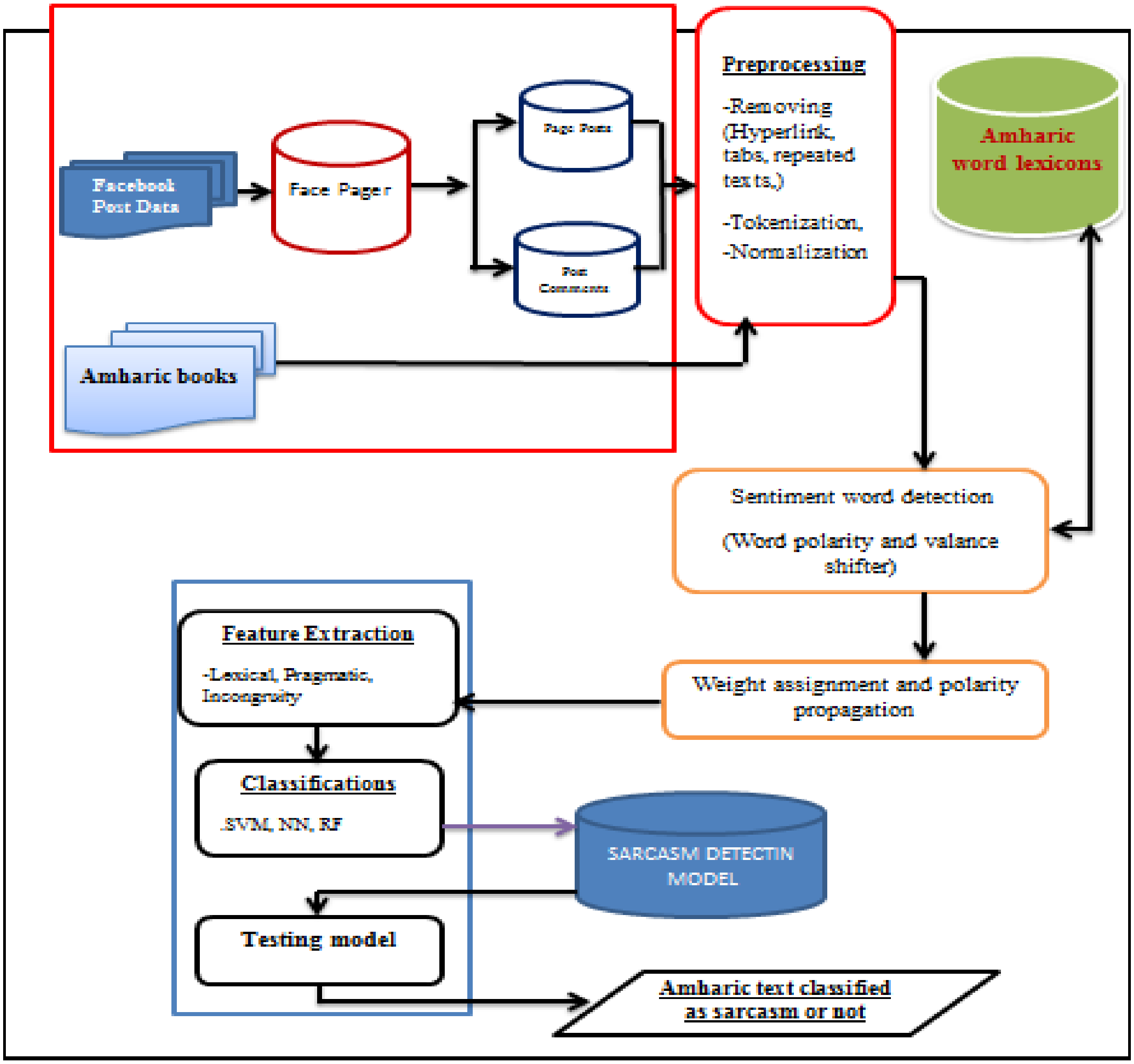


Figure 3-1 Architecture of Sarcasm detection for Amharic texts

The model takes manually annotated Amharic sarcasm text as an input. The texts are preprocessed to make the system efficient and effective in its performance. Then the supervised learning algorithms (SVM, NN and RF) are applied to build the model from the labeled training set. The next task is the evaluation of the model and performance status of the model.

### 3.2.1 Data Collections

Supervised machine learning for sarcasm classification tasks requires an annotated corpus to train and test a classifier. For the Amharic language there is no standardized corpus for sentiment analysis. So the construction of labeled corpus is a very important step because it would allow for more experiments, especially with supervised classification.

Totally, we gathered 800 comments and book reviews on facebook pages and Abebe Tolla's books. The data set consists of 800 Amharic sarcasm sentences and facebook comment text reviews on Sarcasm, from which 400 are labeled Sarcasm, 400 are labeled Non-Sarcasm Amharic texts so that there will be balanced class distribution. For the purpose of the experiment from the total Sarcasm, texts review 20% (20/80) is randomly selected for testing. Most of the data is collected from Abebe Tolla's books and the rest of the dataset is collected from Facebook pages, magazines, newspapers and Amharic literatures'.

### 3.2.2 DATA PREPROCESSING FOR SARCASM DETECTION

The second phase of the sarcasm detection model for Amharic text is the preprocessing component. The pre-processing activity is important to improve the accuracy, efficiency, and scalability of the classification process. Preprocessing activity involves normalization and tokenization. The input for this process is a text data, but not every word in the text is meaningful for categorization. For this reason, the data must be processed and represented to a concise and an identifiable format or structure. Non-standard words such as numbers, abbreviations and dates are removed from the dataset. In this research, the different characteristics or features of the

Amharic language were considered in the development of the algorithm. The preprocessing task was implemented using the python programming language (Python 3.6).

#### 3.2.2.1 Tokenization

Tokenization refers to the process of splitting the text into a set of tokens (usually words). This process detects the boundaries of a written text. The Amharic language uses a number of punctuation marks which demarcate words in a stream of characters which include *‘huletneTb’* (:), ‘*aratneTb’* (::), ‘*deribsereze’* (፤), ‘netelaserez’ (፥), exclamation mark ‘!’ and question mark ’?’  Punctuation marks one of the most relevance in sarcasm detection task and has to be used mostly in Amharic texts. Amharic text tokenization process can be done by using the algorithm shown in algorithm 3-1.

```
1. While the Amharic texts input exists
        Read Character
        If the character is white space or Amharic punctuation marks
            Append token
        Else
            Assign token with token + Character
2. While end
3. Pass token to next processing
4. Close files
```

***Algorithm 3-1 Algorithm of Amharic Text Tokenizer***

#### 3.2.2.2 Normalization

Amharic writing system has homophone characters which mean characters with the same sound have different symbols for example; it is common that the character ሰ and ሠ are used interchangeably as ስራ and ሥራ to mean “work”. These different symbols must be considered as similar because they do not have an effect on meaning .Such type of inconsistency in writing words will be handled by replacing characters of the same sound by a common symbol. Thus, for example, if the character was one of ሐ፣ሓ፣ሃ፣ኀ፣ኃ or ኸ (all of them with a similar sound h) then it was converted to ሀ. By the same symbol, all orders of ሠ (with the sound s) were changed to their equivalent respective orders of ሰ, all orders of ፀ (with the sound tse) were changed to their equivalent respective orders of ጸ. A sample of such normalized Amharic words is shown in Algorithm3-2. Such inconsistency of characters may cause unnecessary increase in the number of document representative words that causes large data size processing.

```
1.  Open the Amharic texts
2.  While not end of file is reached do
3.  Read Character
    For each character in the Amharic text
            If the character is one of ሀ, ሐ, ኀ
                Replace them with ሀ
          Else if character is one of ሰ, ሠ
                Replace them with ሰ
         Else if character is one of ፀ, ጸ
               Replace them with ፀ
       End if
    End for
4.  End while
5.  Close files
```

***Algorithm 3-2 Algorithm of Amharic text Normalization***

### 3.2.2.3 DETECTION OF SARCASM BY USING OPINION WORDS IN AMHARIC TEXTS

This activity is responsible for detecting sarcasm sentence by used polarity terms and contextual valence shifter terms. After the data is preprocessed, every valid term in the text is checked whether it is sentiment word or not. This is done by a simple detection mechanism where the whole lexicon is scanned for every term. If the term exists in the dictionary, then the term is a polarity word (positive or negative) or a contextual valence shifter (negation or intensifier). Polarity words are terms that can express opinions towards an object such as 'ጥሩ' (good) that expresses a positive opinion and 'መጥፎ' (bad) that expresses a negative opinion towards an object. These terms are properly tagged in the lexicon with computer interpretable values as '2' for positive opinion terms and '-2' for negative opinion terms. Then, if a term is found in the lexicon and if it is corresponding value is '+', then this opinion term is positive. Similarly, if a term is found in the lexicon and if it is corresponding value is '-', then this opinion term is negative.

The general purpose of the Amharic words lexicon is used for opinion mining system in any domain. This is because the opinion terms in this lexicon are not restricted to the specific domain rather it contains any opinion terms in the Amharic language. As a result, the valid terms in the collected data are first preprocessed and saved into the dictionary. Then if at least a single term is found in the Amharic words lexicon, the process continues to the next step (weight assignment and polarity propagation) otherwise the general lexicon is scanned for further search. If the term taken from the collected data is not found in Amharic words lexicon, this term is considered as non-sentiment word and it is discarded as such terms are not important in the detection of sarcasm word.

## 3.3 WEIGHT ASSIGNMENT AND POLARITY PROPAGATION

In this phase, the main activities are weight assignment and polarity propagation. By used the Senti-strength tool, the polarity of each word is generated. The value generated lies in the range [-5, 5]. If the value is positive, it is taken as a word with positive polarity. Similarly, if it is negative it is taken to be a word with negative polarity. Using these values two features are generated namely the total count of the words with positive and negative polarity. All possible positive sentiment terms are tagged in the Amharic word lexical by giving '+' and given a default value of +2 at run time. All the negative sentiment terms are tagged by '-' and given a default value of -2. Before the final average polarity weight is calculated, the polarity propagation is done which is used to modify the initial value of the sentiment terms and followed the Amharic grammatical rules like: all of these rules are finding from (Gebremeskel, 2010) and we used as for this thesis to complete the overall polarity feature set.

**Rule 1:** if any polarity term is followed by a negation term, the initial polarity value or weight of the term will be changed.

For example in the sentence ‘ጥሩ አይደለም’ the sentiment term ‘ጥሩ’ (good) is given an initial value of **+2**. But due to the negation term ‘አይደለም’ (not), the polarity value of the term is changed into **-2**. Similarly, in the sentence ‘‘መጥፎ አይደለም’ the sentiment term ‘መጥፎ’ (bad) is given an initial value of **-2**. But due to the negation terms ‘አይደለም’ (not), the polarity value of that sentiment is changed into **+2**.

**Rule 2:** if a positive sentiment term is preceded by an overstatement term, then the initial value of that term is propagated from +2 to +3. For example in the sentence ‘በጣም ጥሩ ነዉ’ due to the overstatement term ‘በጣም’ (very), the initial polarity value of the sentiment term ‘ጥሩ’(good) is increased by **+1** from +2 to +3.

**Rule 3:** if a positive sentiment term is followed by an understatement term, the initial value of that term is decreased from +2 to +1. For example in the sentence ‘ጥሩ ቢሆንም’ the polarity weight of the sentiment term ‘ጥሩ’ (good) is decreased from the initial value +2 to +1 due to the understatement term ቢሆንም’.

**Rule 4:** if a negative sentiment term is preceded by an overstatement term, then the initial value of the term is decreased by -1 from -2 to -3. For example in the sentence ‘በጣም መጥፎ ነዉ’ due to the overstatement ‘በጣም’ (very), the initial weight of the sentiment word ‘መጥፎ’( bad) is decreased from -2 to -3.

**Rule 5:** if a negative sentiment term is followed by an understatement term, the initial weight of that term is increased by +1. For example, in the sentence ‘መጥፎ ቢሆንም the initial weight of the sentiment term is increased from -2 to -1 due to the understatement term.

**Rule 6:** if a sentiment term is not linked to any contextual valence shifting term, the initially assigned weight is considered for further process.

## 3.4 CLASSIFICATION TECHNIQUES

In any Machine Learning task features are of central importance. The quality of the classification depends on the features selected. Carefully designed and chosen features play a big role in improving the results both qualitatively and quantitatively.

Sarcasm detection is a non-trivial task. Usually, sarcasm is cleverly embedded in a sentence which has a positive sentiment. The context also plays a role in determining whether sarcasm is present as a hidden sentiment or not. Hence, it is a linguistically complex task in the domain of Natural Language Processing.

### 3.4.2 SUPERVISED LEARNING

Supervised machine learning techniques involve the use of a labeled training corpus to learn a certain classification function and involve learning a function from examples of its inputs and outputs (schrauwen, 2010.). The output of this function is either a continuous value (‘regression’) or can predict a category or label of the input object (‘classification’). A classifier is called supervised if it is built based on training corpora containing the correct label for each input.

The process is based on learning a model given a set of correctly classified data. The aim of supervised learning is to train a model to recognize discriminant attributes (in statistical literature supervised learning is sometimes known as discriminant learning) in the data (Michie, 1994). Let us say that we want to build a model that can separate news articles about soccer from religion. With a given set of labeled data, the model can be trained to e.g. learn that some words (attributes) are used solely, or more frequently, in one of the classes. The model might have learned that articles containing words such as “referee”, “player” or “goaltender” is more likely to be an article about the sport. Conversely, words such as “god”, “church” and “Islam” are more likely to occur in an article about religion. New unseen news articles can then be predicted to be an article about soccer or religion depending on the frequency of its words.

### 3.4.3 UNSUPERVISED LEARNING

Supervised methods cannot always be used, because labeled corpora are not always available. Unsupervised and weakly-supervised methods are another option for machine learning that does not require pre-tagged data. Unsupervised involve learning patterns in the input when no specific output values are supplied (Norvig, 2003.). This means that the learner only receives an unlabelled set of examples. Unsupervised methods can also be used to label a corpus that can later be used for supervised learning. Examples of unsupervised learning methods are (k-means) clustering or cluster analysis and the expectation-maximization algorithm, an algorithm for finding the maximum likelihood.

### 3.4.4 SEMI-SUPERVISED LEARNING

Semi-supervised learning is based on the fact that labelled data can be hard to obtain and unlabelled data is cheap. The idea is to combine a small set of labelled data and expand it using unlabelled data with the help of unsupervised learning. The result is then a big set of labelled data, perhaps containing some noise that can be used for supervised classification. (Lin C. et. al., 2011)

In the remainder of this section, we present three classifiers that have been used during this research work, every single one is used for supervised learning.

## 3.5 IMPLEMENTATION

In this sub section, the sarcasm detection for Amharic texts lexicon building issues, the tools used for implementing the system, the procedures to integrate the different components, the proposed algorithm, the input review, output result and other related issues are described.

### 3.5.2 BUILDING SARCASM LEXICON

Several approaches use lexical sentiment as a feature to the sarcasm classifier. It must, however, be noted that these approaches require ‘surface polarity’: the apparent polarity of a sentence.

(Santosh Kumar Bharti K. S., 2015) Described a rule-based approach that predicts a sentence as sarcastic if a negative phrase occurs in a positive sentence

Sarcasm consists of: (i) the use of irony, and (ii) the presence of ridicule. Based on the theories described here, understanding sarcasm (by humans or through programs) can be divided into the following components:

A. Identification of shared knowledge: the sentence ‘I love being ignored’ cannot be understood as sarcastic without the shared knowledge that people do not like being ignored. This specifically holds true in case of specific context. For example, the sentence ‘I love solving math problems all weekend’ may be perceived as non-sarcastic by a listener who loves math or by a listener who knows that the speaker loves math. A listener, in these situations, would either look for a dropped negation or an echoic reminder, as given by theories above,
B. Identification of what constitutes ridicule: the ridicule may be conveyed through different reactions such as laughter, change of topic, etc. (JOSHI, 2017)

### 3.5.3 TOOLS

In order to achieve our objective, we used different environments and tools. Python programming language is used to develop the model. It is an interpreter, object oriented, high level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding; make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together (python).

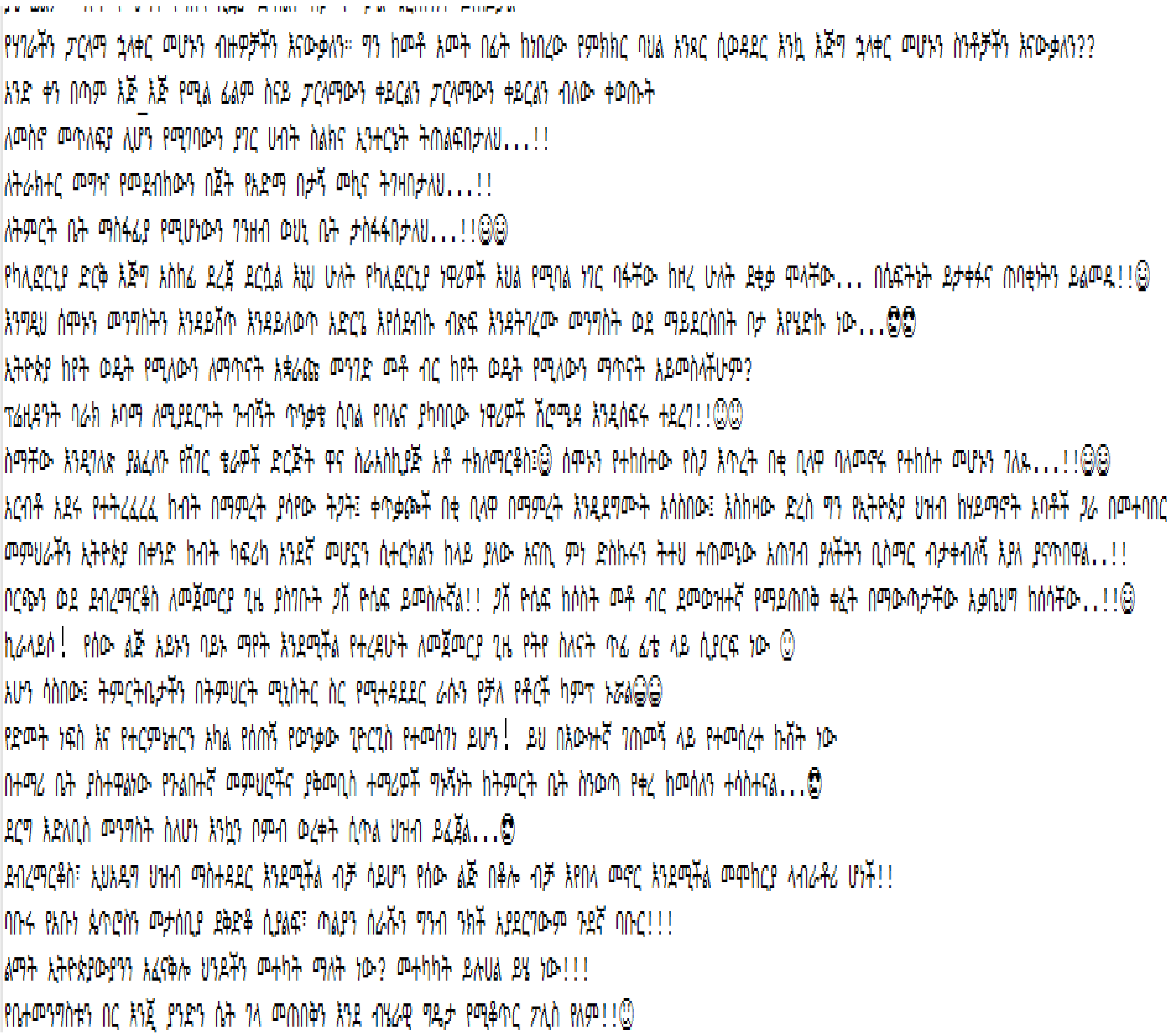
The python programming language is a dynamically typed, object oriented, interpreted language and it is great for natural language processing (NLP) because it is simple, easy to debug (exceptions and interpreted language), easy to structure (modules and object oriented) and powerful for string manipulation.

We used python 3.6.6 version because it is possible to use encodings different than ASCII in python source files. As a result, Amharic language characters are directly interpreted by python3.6.6 and above versions without the need to go for transliteration or feeding the Unicode representation of the characters. All the source codes and rules of the model are written in python3.6.6 compatible format because this version doesn’t support backward compatibility.

**Dictionary representation**

Dictionary is a useful built in data type into python. Regular python dictionaries iterate over a key: value pairs in an arbitrary order. Dictionaries are sometimes found in other languages as “associative memories” or “associative arrays”. Unlike sequences, which are indexed by a range of numbers, dictionaries are indexed by keys, which can be any immutable type; strings and numbers can always be keys. Tuples can be used as keys if they contain only strings, numbers, or tuples. If tuple contains any mutable object directly or indirectly, it cannot be used as keys. Dictionaries in python are an unordered set of keys: values pairs, with the requirement that the keys are unique (with one dictionary). A pair of braces creates an empty dictionary: {}. Placing a comma separated list of key: value pairs within the braces adds initial key: value pairs to the dictionary. The main operations on a dictionary are storing a value with some key and extracting the value given by the keys (Fred L. Drake, 2003).

The lexicon of Amharic word lexicon (dictionary) can be put within the source code or can be imported as a text file at run time. Therefore for each key, its corresponding value is returned for further process. As a result for each Amharic sentiment terms in the input review (key), the whole dictionary is scanned for its corresponding value. A sample of the input text data is given in figure 3-2.



የሃገራችን ፓርላማ ኋላቀር መሆኑን ብዙዎቻችን እናውቃለን። ግን ከመቶ አመት በፊት ከነበረው የምክክር ባህል አንጻር ሲወዳደር እንኳ እጅግ ኋላቀር መሆኑን ስንቶቻችን እናውቃለን??
አንድ ቀን በጣም እጅ_እጅ የሚል ፊልም ስናይ ፓርላማውን ቀይርልን ፓርላማውን ቀይርልን ብለው ቀወጡት
ለመስኖ መጥለፍያ ለሆን የሚገባውን ያገር ሀብት ስልክና ኢንተርኔት ትጠልፍበታለህ...!!
ለትራክተር መግዣ የመደብከውን በጀት የአድማ በታኝ መኪና ትገዛበታለህ...!!
ለትምርት ቤት ማስፋፊያ የሚሆነውን ገንዘብ ወህኒ ቤት ታስፋፋበታለህ...!!😃😃
የካሊፎርኒያ ድርቅ እጅግ አስከፊ ደረጃ ደርሷል እነሀ ሁለት የካሊፎርኒያ ነዋሪዎች እህል የሚባል ነገር በፋቸው ከዞረ ሁለት ደቂቃ ሞላቸው... በሰፍትነት ይታቀፉና ጠበቂነትን ይልመዱ!!😃
እንግዲህ ሰሞኑን መንግስትን እንዳይሸጥ እንዳይለውጥ አድርጌ እየሰደብኩ ብዕፍ እንዳትገረሙ መንግስት ወደ ማይደርስበት ቦታ እየሄድኩ ነው...😃😃
ኢትዮጵያ ከየት ወዴት የሚለውን ለማጥናት አቋራጩ መንገድ መቶ ብር ከየት ወዴት የሚለውን ማጥናት አይመስላችሁም?
ፕሬዚደንት ባራክ ኦባማ ለሚያደርጉት ጉብኝት ጥንቃቄ ሲባል የቦሌና ያካባቢው ነዋሪዎች ሽሮሜዳ እንዲሰፍሩ ተደረገ!!🙂🙂
ስማቸው እንዳገለጽ ያልፈለጉ የሸገር ቄራዎች ድርጅት ዋና ስራአስኪያጅ አቶ ተክለማርቆስ፤😃 ሰሞኑን የተከሰተው የስጋ እጥረት በቂ ቢላዋ ባለመኖሩ የተከሰተ መሆኑን ገለጹ...!!😃😃
አርብቶ አደሩ የተንረፈረፈ ከብት በማምረት ያሳየው ትጋት፤ ቀጣፊዎች በቂ ቢላዋ በማምረት እንዳደግሙት አሳስበው፤ እስከዛው ድረስ ግን የኢትዮጵያ ህዝብ ከሃይማኖት አባቶች ጋራ በመተባበር
መምህራችን ኢትዮጵያ በቀንድ ከብት ካፍሪካ አንደኛ መሆኗን ሲተርክልን ከላይ ያለው አናጢ ምን ድስኩሩን ትተህ ተጠመኔው አበገብ ያለችትን ቢስማር ብታቀብለኝ እያለ ያናጥበዋል..!!
ቦርጭን ወደ ደብረማርቆስ ለመጀመርያ ጊዜ ያስገቡት ጋሽ ዮሴፍ ይመስሉኛል!! ጋሽ ዮሴፍ ከሰስት መቶ ብር ደመወዝተኛ የማይጠበቅ ቁፈት በማውጣታቸው አቃቤህግ ከሰሳቸው..!!😃
ኪራላይሶ! የሰው ልጅ አይኑን ባይኑ ማየት እንደሚችል የተረዳሁት ለመጀመርያ ጊዜ የትየ ስለናት ጥፊ ፊቴ ላይ ሲያርፍ ነው 🙂
አሁን ሳስበው፤ ትምርትቤታችን በትምህርት ሚኒስትር ስር የሚተዳደር ራሱን የቻለ የቶርቸ ካምፕ ኦሯል😃😃
የድመት ነፍስ እና የተርምኔተርን አካል የሰጠኝ የወንቃው ጊዮርጊስ የተመሰገነ ይሁን! ይህ በእውነተኛ ገጠመኝ ላይ የተመሰረተ ኩሸት ነው
በተማሪ ቤት ያስተዋልነው የንልበተኛ መምህሮችና ያቅመቢስ ተማሪዎች ግኑኝነት ከትምርት ቤት ስንወጣ የቁረ ከመሰለን ተሳስተናል...😃
ደርግ እደለቢስ መንግስት ስለሆነ እንኳን በምብ ወረቀት ሲጥል ህዝብ ይፈጀል...😃
ደብረማርቆስ፣ ኢህአዴግ ህዝብ ማስተዳደር እንደሚችል ብቻ ሳይሆን የሰው ልጅ በቆሎ ብቻ እየበላ መኖር እንደሚችል መሞከርያ ላብራቶሪ ሆነች!!
ባቡሩ የአቡነ ጴጥሮስን መታሰቢያ ደቅድቆ ሲያልፍ፣ ጣልያን ስራሹን ግንብ ንክች አያደርገውም ጉደኛ ባቡር!!!
ልማት ኢትዮጵያውያን አፈናቅሎ ህንዶችን መተካት ማለት ነው? መተካካት ይሉሀል ይሄ ነው!!!
የቢተመንግስቱን በር እንጂ ያንድን ቤት ገላ መጠበቅን እንደ ብሄራዊ ግዴታ የሚቆጥር ፖሊስ የለም!!🙂

Figure 3-2 Sample inputs for sarcasm detection

### 3.5.4 THE PROPOSED ALGORITHMS

In this paper, we presented a novel supervised learning algorithm for sarcasm identification. The algorithm has two modules: (i) Feature-extraction, and (ii) classification.

#### *3.5.4.3 Feature -extraction*

**Lexical Features**

N-grams are a commonly used feature set for NLP related tasks in Machine Learning. Certain words or phrases like "Yeah right!" may be prevalent in sarcastic tweets. Presence of such words can be a strong indicator for sarcasm. We use unigrams in order to extract the lexical information contained in the text.

Using the training corpus a dictionary is created. Each unique word is mapped onto a particular ID. These ID numbers are used as the feature numbers. The value corresponding to each such feature number is the frequency of occurrence of that particular word in the text for which we are generating the feature values. The dictionary would be large owing to the vocabulary available in the corpus. The text would contain only a few words from this large vocabulary. Hence, the feature vector would naturally contain a lot of 0's corresponding to the words that are not appearing in the sentence but are present in the dictionary. The ID's with value (frequency) 0 can be discarded since we are looking for the presence of words prevalent in sarcastic texts which can be a potentially important indicator while the absence of words conveys no information here.

**Punctuation Features**

These features determine when the sentiment of the sentence differs from the emoticons or similes. For example: - “እድሜ ጠገቡ ትምህርት ቤት ‘ለልማት’ ፈረሰ! ☹” In the example, the sentiment of the sentence is supposed to be positive but the use of the frowning face simile makes it sarcastic.

These features are associated with the grammatical hints. Grammar plays an important role in any kind of language analysis. These include:-

### *Number of emoticons*

Emoticons are commonly used across social media platforms to express sentiments. As a feature, they can be captured using UTF-8 encoding. Using the 'codecs' module in Python, files containing emoticons in UTF-8 format can be opened and read. The emoticons present in them can be captured using regular expressions.

### *Number of slang laughter expressions*

In the Amharic language there is popular slang expressions such as 'ኪኪኪኪ', 'ሃሃሃ' and 'ቂቂቂ' are fairly well used. Various variants of these have also been accounted for. The frequency of occurrence of these expressions is a feature. Since sarcasm is intended to have an element of humor, higher occurrence of these is potentially indicative of sarcasm.

### *Number of punctuation marks*

The Exclamation mark is often used to lay extra emphasis on the underlying emotion like a surprise, shock or dismay. Even in sarcastic texts, such use of exclamation marks is prevalent especially that of “!”, “?” and “…”. The count of such exclamation marks is hence used as a feature.

## Semantic feature

These features determine whether there exist words which contradict or are nearly opposite of each other. The linguistic theory of Context Incongruity suggests that the common form of sarcasm expression consists of a positive sentiment which is contrasted with a negative situation. For example: - “ድንቁርና በረከት ነው…!” In example two nearly opposite words are used which makes the utterance sarcastic. In these features there are also some particular features are included

### *Number of words with positive and negative polarity*

First, it takes tokenized and normalized Amharic texts terms and checks them if they bear Amharic sentiment words in the Amharic word lexical corpus. This is done by checking the existence of the terms in the dictionary of Amharic sentiment terms and by checking the existence of graphical expressions in emoticon lookup tables. Next, the sentiment terms are assigned initial polarity weight and polarity propagation is done if the sentiment terms are linked to contextual valence shifter terms. Using the Senti-strength tool, the polarity of each word is generated. The value generated lies in the range [-5, 5]. If the value is positive, it is taken as a word with positive polarity. Similarly, if it's negative it is taken to be a word with negative polarity. Using these values two features are generated namely the total count of the words with positive and negative polarity.

### *Lexical Polarity*

This is the overall polarity of the entire sentence. Owing to the theory of lexical incongruity, it can be observed that a text which has an overall strong positive polarity is more likely to be sarcastic compared to a text with overall negative polarity. This is because in general sarcasm tends to be caustic.

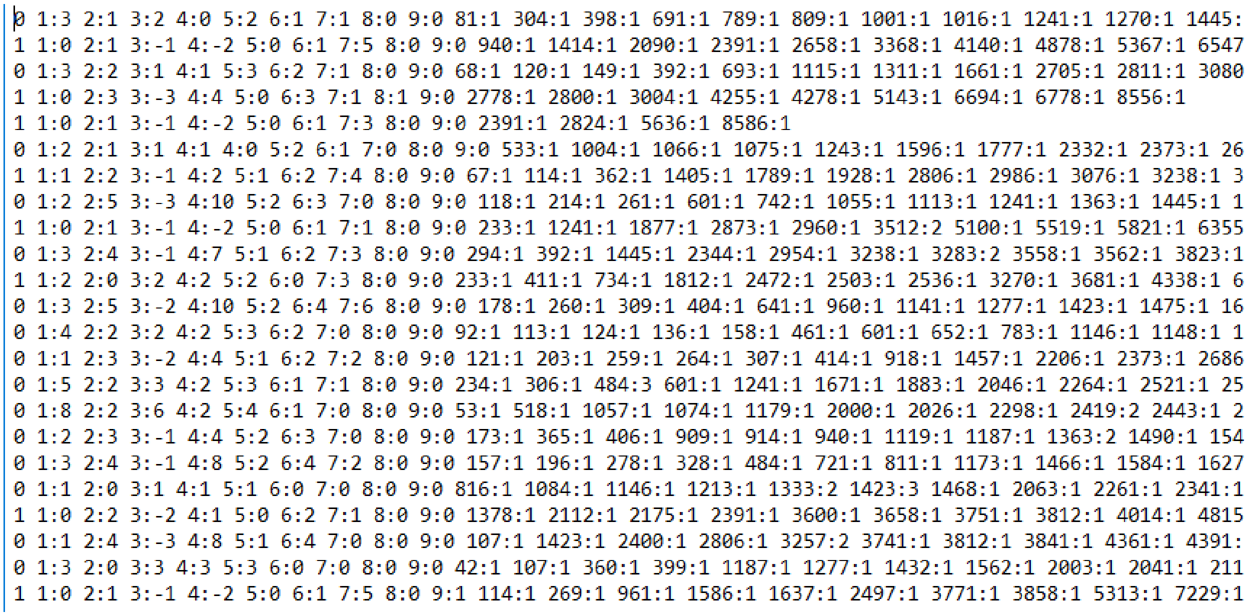
#### 3.5.4.4 Classification

In order to perform the classification based on the features mentioned above, we explore a set of standard classifiers typically used in text classification research. We used a support vector machine SVM with a linear kernel in the implementation provided by LIBSVM. We also used a Neural Network (NN) classifier and Random Forest classifier (RFC) as comparisons to each other. For error estimating it is common to divide the entire data set in to train set and test set. The model that obtains the best result on the test data is then used as how it performs.

Finally, the text assigned into predefined categories Sarcasm and Non-Sarcasm based on the total weight obtained from the previous step. The high level view of the proposed algorithms that show how the sentiment terms, punctuation marks and laugher expression (emojis) are detected and how the sentiment polarity value is propagated as algorithm 3-3

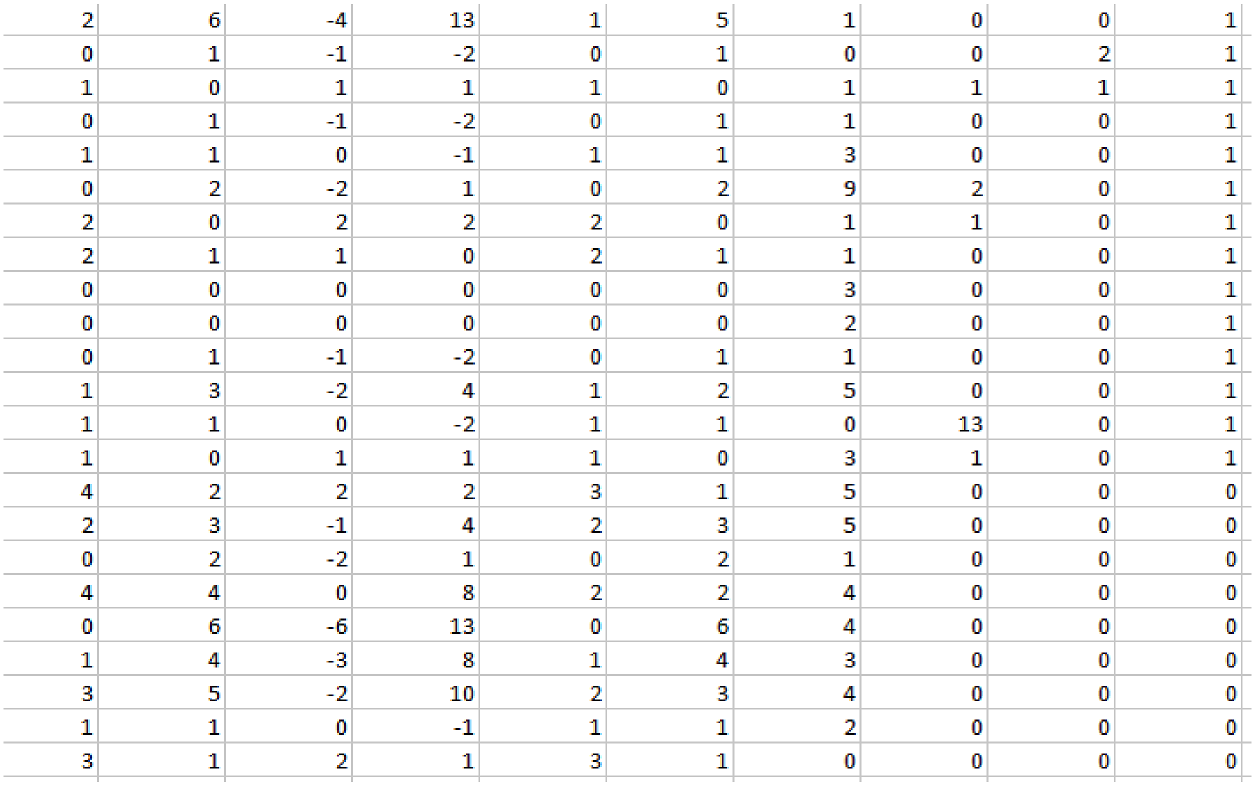
```
1. For every pre-processed Amharic sarcasm texts S
2. For every term T in the texts S, checks its existence in the Amharic word lexicon D
3. If a term T exists in the dictionary D
   3.1. Its corresponding initial polarity weight Tpi is given
   3.2. If it is linked to a contextual shifter term C
        3.2.1.The polarity value Tpi of the term is propagated
              3.2.1.1. If a sentiment term T is linked to a negative contextual valence term C, then the
                       prior polarity value of the term T is reversed from Tpi to –Tpi.
              3.2.1.2. If the sentiment term T is linked to overstatement contextual valence shifter term
                       C, then the prior polarity value of the term T is modified from Tpi to Tpi +1 (for
                       positive sentiment terms) and from Tpi to Tpi-1(for negative sentiment terms)
              3.2.1.3. If the sentiment term T is linked to understatement contextual valence shifter C,
                       then the prior polarity value of the term is modified from Tpi to Tpi-1( for positive
                       sentiment terms) and from Tpi to Tpi+1 (for negative sentiment terms)
   3.3. Add all the polarity weights of the individual terms to get text polarity value Rp
   3.4. If the total polarity weight Rp is greater than 0, then the text is categorized into
        predefined category positive (+)
   3.5. If the total polarity weight Rp is less than zero, then the text is assigned into a predefined
        category negative (-)
   3.6. Else the text is assigned into a predefined category of neutral.

4. Else the text is assigned into a unclassified class because there are no sentiment terms Ts in
   the given text
```

Algorithm 3-3 Algorithm for sarcasm detection by used text polarity propagation



```
0 1:3 2:1 3:2 4:0 5:2 6:1 7:1 8:0 9:0 81:1 304:1 398:1 691:1 789:1 809:1 1001:1 1016:1 1241:1 1270:1 1445:
1 1:0 2:1 3:-1 4:-2 5:0 6:1 7:5 8:0 9:0 940:1 1414:1 2090:1 2391:1 2658:1 3368:1 4140:1 4878:1 5367:1 6547
0 1:3 2:2 3:1 4:1 5:3 6:2 7:1 8:0 9:0 68:1 120:1 149:1 392:1 693:1 1115:1 1311:1 1661:1 2705:1 2811:1 3080
1 1:0 2:3 3:-3 4:4 5:0 6:3 7:1 8:1 9:0 2778:1 2800:1 3004:1 4255:1 4278:1 5143:1 6694:1 6778:1 8556:1
1 1:0 2:1 3:-1 4:-2 5:0 6:1 7:3 8:0 9:0 2391:1 2824:1 5636:1 8586:1
0 1:2 2:1 3:1 4:1 4:0 5:2 6:1 7:0 8:0 9:0 533:1 1004:1 1066:1 1075:1 1243:1 1596:1 1777:1 2332:1 2373:1 26
1 1:1 2:2 3:-1 4:2 5:1 6:2 7:4 8:0 9:0 67:1 114:1 362:1 1405:1 1789:1 1928:1 2806:1 2986:1 3076:1 3238:1 3
0 1:2 2:5 3:-3 4:10 5:2 6:3 7:0 8:0 9:0 118:1 214:1 261:1 601:1 742:1 1055:1 1113:1 1241:1 1363:1 1445:1 1
1 1:0 2:1 3:-1 4:-2 5:0 6:1 7:1 8:0 9:0 233:1 1241:1 1877:1 2873:1 2960:1 3512:2 5100:1 5519:1 5821:1 6355
0 1:3 2:4 3:-1 4:7 5:1 6:2 7:3 8:0 9:0 294:1 392:1 1445:1 2344:1 2954:1 3238:1 3283:2 3558:1 3562:1 3823:1
1 1:2 2:0 3:2 4:2 5:2 6:0 7:3 8:0 9:0 233:1 411:1 734:1 1812:1 2472:1 2503:1 2536:1 3270:1 3681:1 4338:1 6
0 1:3 2:5 3:-2 4:10 5:2 6:4 7:6 8:0 9:0 178:1 260:1 309:1 404:1 641:1 960:1 1141:1 1277:1 1423:1 1475:1 16
0 1:4 2:2 3:2 4:2 5:3 6:2 7:0 8:0 9:0 92:1 113:1 124:1 136:1 158:1 461:1 601:1 652:1 783:1 1146:1 1148:1 1
0 1:1 2:3 3:-2 4:4 5:1 6:2 7:2 8:0 9:0 121:1 203:1 259:1 264:1 307:1 414:1 918:1 1457:1 2206:1 2373:1 2686
0 1:5 2:2 3:3 4:2 5:3 6:1 7:1 8:0 9:0 234:1 306:1 484:3 601:1 1241:1 1671:1 1883:1 2046:1 2264:1 2521:1 25
0 1:8 2:2 3:6 4:2 5:4 6:1 7:0 8:0 9:0 53:1 518:1 1057:1 1074:1 1179:1 2000:1 2026:1 2298:1 2419:2 2443:1 2
0 1:2 2:3 3:-1 4:4 5:2 6:3 7:0 8:0 9:0 173:1 365:1 406:1 909:1 914:1 940:1 1119:1 1187:1 1363:2 1490:1 154
0 1:3 2:4 3:-1 4:8 5:2 6:4 7:2 8:0 9:0 157:1 196:1 278:1 328:1 484:1 721:1 811:1 1173:1 1466:1 1584:1 1627
0 1:1 2:0 3:1 4:1 5:1 6:0 7:0 8:0 9:0 816:1 1084:1 1146:1 1213:1 1333:2 1423:3 1468:1 2063:1 2261:1 2341:1
1 1:0 2:2 3:-2 4:1 5:0 6:2 7:1 8:0 9:0 1378:1 2112:1 2175:1 2391:1 3600:1 3658:1 3751:1 3812:1 4014:1 4815
0 1:1 2:4 3:-3 4:8 5:1 6:4 7:0 8:0 9:0 107:1 1423:1 2400:1 2806:1 3257:2 3741:1 3812:1 3841:1 4361:1 4391:
0 1:3 2:0 3:3 4:3 5:3 6:0 7:0 8:0 9:0 42:1 107:1 360:1 399:1 1187:1 1277:1 1432:1 1562:1 2003:1 2041:1 211
1 1:0 2:1 3:-1 4:-2 5:0 6:1 7:5 8:0 9:1 114:1 269:1 961:1 1586:1 1637:1 2497:1 3771:1 3858:1 5313:1 7229:1
```

Figure 3-3 Combine data from Sarcasm and Nonsarcasm files to give it as input to the SVM classifier



| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 2 | 6 | -4 | 13 | 1 | 5 | 1 | 0 | 0 | 1 |
| 0 | 1 | -1 | -2 | 0 | 1 | 0 | 0 | 2 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | -1 | -2 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | -1 | 1 | 1 | 3 | 0 | 0 | 1 |
| 0 | 2 | -2 | 1 | 0 | 2 | 9 | 2 | 0 | 1 |
| 2 | 0 | 2 | 2 | 2 | 0 | 1 | 1 | 0 | 1 |
| 2 | 1 | 1 | 0 | 2 | 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 1 |
| 0 | 1 | -1 | -2 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 3 | -2 | 4 | 1 | 2 | 5 | 0 | 0 | 1 |
| 1 | 1 | 0 | -2 | 1 | 1 | 0 | 13 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 3 | 1 | 0 | 1 |
| 4 | 2 | 2 | 2 | 3 | 1 | 5 | 0 | 0 | 0 |
| 2 | 3 | -1 | 4 | 2 | 3 | 5 | 0 | 0 | 0 |
| 0 | 2 | -2 | 1 | 0 | 2 | 1 | 0 | 0 | 0 |
| 4 | 4 | 0 | 8 | 2 | 2 | 4 | 0 | 0 | 0 |
| 0 | 6 | -6 | 13 | 0 | 6 | 4 | 0 | 0 | 0 |
| 1 | 4 | -3 | 8 | 1 | 4 | 3 | 0 | 0 | 0 |
| 3 | 5 | -2 | 10 | 2 | 3 | 4 | 0 | 0 | 0 |
| 1 | 1 | 0 | -1 | 1 | 1 | 2 | 0 | 0 | 0 |
| 3 | 1 | 2 | 1 | 3 | 1 | 0 | 0 | 0 | 0 |

Figure 3-4 Combine data from Sarcasm and Nonsarcasm files to give it as input to the NN and RF classifiers

# CHAPTER FOUR
# EXPERIMENTAL RESULTS

## 4.1 INTRODUCTION

Three supervised techniques are used for conducting the experiments: The Support Vector Machine (SVM), Random Forest (RF) and Neural Network (NN) classifiers. We tested each technique individually and evaluate its performance. The procedure is, as is standard in supervised machine learning tasks, first training a classifier on pre-classified training data and then evaluating the performance of the classifier on un labeled set of test data. We selected to work with the Natural Language Toolkit (NLTK). This package is equipped with several classifiers (i.e. SVM, RF, and NN). All programming has been done in the Python programming language and executed in the programming environment Window 10 python interactive shell.

In this research, three experiments had done by using unigram words and most informative words with the three learning algorithms SVM, Decision tree and Neural Network classifiers. All the results are presented in the subsequent section. All works are done using NLTK classification packages and python programming.

## 4.2 PROCEDUER AND EXPERIMENTAL RESULT

To evaluate the Sarcasm detection model for Amharic texts, we used procedures and setups that include data collection, methods and manual classifications. These are described in the subsequent sections.

### 4.2.1 SARCASM DATA PRESENTATION

As presented in the previous chapters, for conducting all experiments, we have considered the Abebe Tolla’s books and some sarcasm related facebook pages as reviews domain. The main reason why we used those as a domain, due to the lack of readily available reviews written in the Amharic language electronically such as in web, blogs and online forums in others domain. As a result, it is relatively more easy and manageable to collect Amharic sarcasm texts manually than any other domains.

In addition, Amharic entertainment viewers can write comments freely as compared to other domains such as politics, products, etc. Hence, most of the sarcasm detection for Amharic texts we used for conducting the experiments is collected manually. After collecting the data from those domains source the sarcasm texts are coding into a computer and categorize in to two labeled classes. These are Sarcasm and Non-Sarcasm. As a result, a total of 800 data are collected from all the sources described above. After the data is collected, preprocessing tasks are applied to construct the final data set (data that are used as input for the modeling tool) from the initial raw data. Data preparation tasks are usually performed multiple times depending on the quality and size of the initial data set. A task includes cleaning; normalization and tokenization of the data were performed to come up with the final suitable dataset for the selected algorithms.

### 4.2.2 MANUAL DATA CLASSIFICATION

This activity is concerned with labeling the data for the experimental purpose. All the 800 data are manually categorized by an independent individual from the data source. If the given data is not related with the topic in the target, it is assigned into the Non-Sarcasm category. As a result, 400 of the total data are labeled as Sarcasm and the rest 400 of them are labeled as Non-Sarcasm. The manually classified reviews helped us in crosschecking with the results obtained from the proposed Sarcasm detection model for Amharic texts.

Table 4-1 Number of Data classification Manually from Different data sources

| DOMAIN | TOTAL NUMBER OF SARCASM TEXTS | TOTAL NUMBER OF NON-SARCASM TEXTS |
|---|---|---|
| books | 100 | 100 |
| facebook | 100 | 100 |
| magazines | 50 | 50 |
| Social medias | 50 | 50 |

## 4.3 EXPERIMENTAL ANALYSIS

This activity is responsible for describing the evaluation parameters of the designed model and its results. Evaluation of the system is made with the evaluation parameter that compares the number of the data which, are categorized correctly and incorrectly. The comparison is done between the data categorized by the proposed model system and that of the manually labeled (categorized) data. Then after the precision, recall and F-measure compute as follows.

**Precision (P)** is the number of true positives divided by the total number of elements labeled as belonging to that class. A high precision means that the majority of items labeled as for instance 'positive' indeed belong to the class 'positive' and it defined as.

$$P = \frac{TP}{TP + FP}$$ where: TP = True Positive, and FP= False Positive **(4-1)**

- **True positives** are positive items that we correctly identified as positive for positive class and Negative items that we correctly identified as Negative for negative class.
- **False positives** (or **Type I errors**) are negative comments that we incorrectly identified as positive for positive class and positive comments that we incorrectly classified as negatives for negative class

.

**Recall (R)** is the number of true positives divided by the total number of items that actually belong to that class. A high recall means that the majority of the 'positive' items were labeled as belonging to the class 'positive'.

$$R = \frac{TP}{TP + FN}$$ Where: FN= False Negative **(4-2)**

- **True negatives** are irrelevant items that we correctly identified as irrelevant. (negative comments not classified under positive for positive class and vice versa )

- **False negatives** (or **Type II errors**) are relevant items that we incorrectly identified as irrelevant.(positive comments that incorrectly not classified under positive for positive class and negative comments that incorrectly not classified under negative for negative class.)

**F-measure** is a measure that combines Recall and Precision into a single measure of performance, this is just the product of Precision and Recall divided by their average.

$$F = \frac{2PR}{P + R} \qquad \textbf{(4-3)}$$

## 4.4 FEATURE VECTORS

In order to perform machine learning, it is necessary to extract clues from the text that may lead to correct classification. So in this research, we are very interested to use simple, superficial unigram feature vectors: all unigram words and most informative words in the corpus used as a feature set. In the first stage, all bag-of-words of the corpus are used to perform experiments. The same experiment is performed on different feature subset in the second stage, which comprises the most informative features of the corpus: the distribution of each word over the different output classes is calculated, and the words with the lowest entropy (or highest information gain) are considered the most relevant features for the classifiers. To measure the frequency distribution of a feature (in this case a word, punctuation and emojis figures are covered) over the output classes is computed.

## 4.5 RESULTS

Classification works by learning from labeled feature sets, or training data. Text feature extraction is the process of transforming what is essentially a list of words into a feature set that is used by a classifier. The NLTK classifiers expect *dict* style feature sets, so we must therefore; transform our text into a *dict*. The ***bag-of-words*** model is the simplest method and that constructs a *dict* from the given words, where every word gets the value true where each word becomes a key with the value True. In our study all unigrams in the file are transferred to ***dict***, and consider as a feature. In order to examine the applicability of machine learning algorithm to classify the sarcasm detection model for Amharic text classification SVM, Random Forest and Neural Network algorithms are compared with the same dataset and feature categories. Most of the research in the area of sentiment analysis and sarcasm detection for text classification used 30% of the data for testing and the remaining data are used for training. Based on the other researcher methodologies we also apply 30% of the total data used for testing and the others for training for our research works.

### 4.5.1 Experimental result using Support Vector Machine (SVM)

The support vector machine (SVM) is a supervised learning method that generates input-output mapping functions from a set of labeled training data. The mapping function can be either a classification function, i.e., the category of the input data, or a regression function. For classification, nonlinear kernel functions are often used to transform input data into a high-dimensional feature space in which the input data become more separable compared to the original input space. Maximum-margin hyperplanes are then created. The model thus produced depends on only a subset of the training data near the class boundaries. Similarly, the model produced by Support Vector Regression ignores any training data that is sufficiently close to the model prediction. SVMs are also said to belong to “kernel methods”. (Lipo Wang ed., 2005)

We conducted the first experiment by using the SVM algorithm by used a simple bag-of-words approach with all unigram words in the corpus. In each stage, the results will be presented. We discuss the accuracy results and performance analysis by computing recall, precision and F-measure.

Table 4-2 Accuracy Results for SVM

| Class | Precision | Recall | F-measure | Accuracy |
|---|---|---|---|---|
| Sarcasm | 0.77 | 0.93 | 0.84 | 80.69 |
| Non-Sarcasm | 0.88 | 0.67 | 0.76 | |

As it is shown in Table 4-2 above, we achieved an accuracy of 80.7% by using a Supportive Vector Machine. In a file given, Sarcasm classified class is 77% likely to be correct. High precision causes only 23% false positive for the Sarcasm class relatively. The f-measure of the RF classifier with all unigram features for the class Sarcasm is 84%.

In the case of a Non-Sarcasm class, the text given a Non-Sarcasm classification is 88% likely to be correct. This is a good precision leads to 12% of a false positive for the negative categories. The f-measure of the Random Forest classifier with all unigram features for the class Non-Sarcasm *is* 76%.

### 4.5.2 Experimental result using Random Forest Classifier

A random forest is a Meta estimator that fits a number of decision tree classifier on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

During training, the Random forest Classifier creates a tree where the child nodes are also instances of Random forest Classifier. The *train ()* class method builds this tree from the ground up, starting with the leaf nodes. It then refines itself to minimize the number of decisions needed to get with a label by putting the most informative features at the top.

In the table 4-3 Decision Tree experimentation accuracy, precision and recall are presented. The feature set used in this experiment is all unigram words (all bag-of-words.)

Table 4-3 Accuracy Result for RF

| Class | Precision | Recall | F-meas | Acc. |
|---|---|---|---|---|
| Sarcasm | 0.85 | 0.77 | 0.81 | 80.0 |
| Non-Sarcasm | 0.81 | 0.80 | 0.80 | |

As it is shown in Table 4-3 above, we achieved an accuracy of 80.0% by using a decision tree classifier. In a file given, Sarcasm classified class is 85% likely to be correct. High precision causes only 15% false positive for the Sarcasm class relatively. The f-measure of the RF classifier with all unigram features for the class Sarcasm is 81%.

In the case of a Non-Sarcasm class, the text given a Non-Sarcasm classification is 81% likely to be correct. This is a good precision leads to 19% of a false positive for the negative categories. The f-measure of the Random Forest classifier with all unigram features for the class Non-Sarcasm *is* 80%.

### 4.5.3 Experimental result using Neural Network Classifier

Classification is one of the most active research and application area of neural networks. Neural Networks are considered a robust classifier. The field of Neural Networks has arisen from diverse sources, ranging from the fascination of mankind with understanding and emulating the human brain, to broader issues of copying human abilities such as Classification, it is one of the most frequently encountered decision making tasks of human activity. Classification is an essential feature to separate large datasets into classes for the purpose of Rule generation, Decision Making, Pattern recognition, Dimensionality Reduction, Data Mining etc. The Neural networks have emerged as an important tool for classification.

The neural network is given the target outputs on to which it should map its inputs, i.e. it is given in paired data of input and output. The error arising from the discrepancy between the network output and the target is used to optimize the network parameters. Once the network has been trained, it is used to produce an output for unseen data. In this research, we used multilayer feed forward neural network (FFNN).

FFNNs are a kind of multilayer neural network which allows signals to travel one way only, from input to output. First, the network is trained on a set of paired data to determine input-output mapping. The weights of the connections between neurons are then fixed and the network is used to determine the classifications of a new set of data. During classification, the signal at the input units propagates all the way through the net to determine the activation values at all the output units. Each input unit has an activation value that represents some feature external to the net. Then, every input unit sends its activation value to each of the hidden units to which it is connected. Each of these hidden units calculates its own activation value and this signal are then passed on to output units. The activation value for each receiving unit is calculated according to a simple activation function. The function sums together the contributions of all sending units, where the contribution of a unit is defined as the weight of the connection between the sending and receiving units multiplied by the sending unit's activation value. This sum is usually then further modified, for example, by adjusting the activation sum to a value between 0 and 1 and/or by setting the activation value to zero unless a threshold level for that sum is reached.

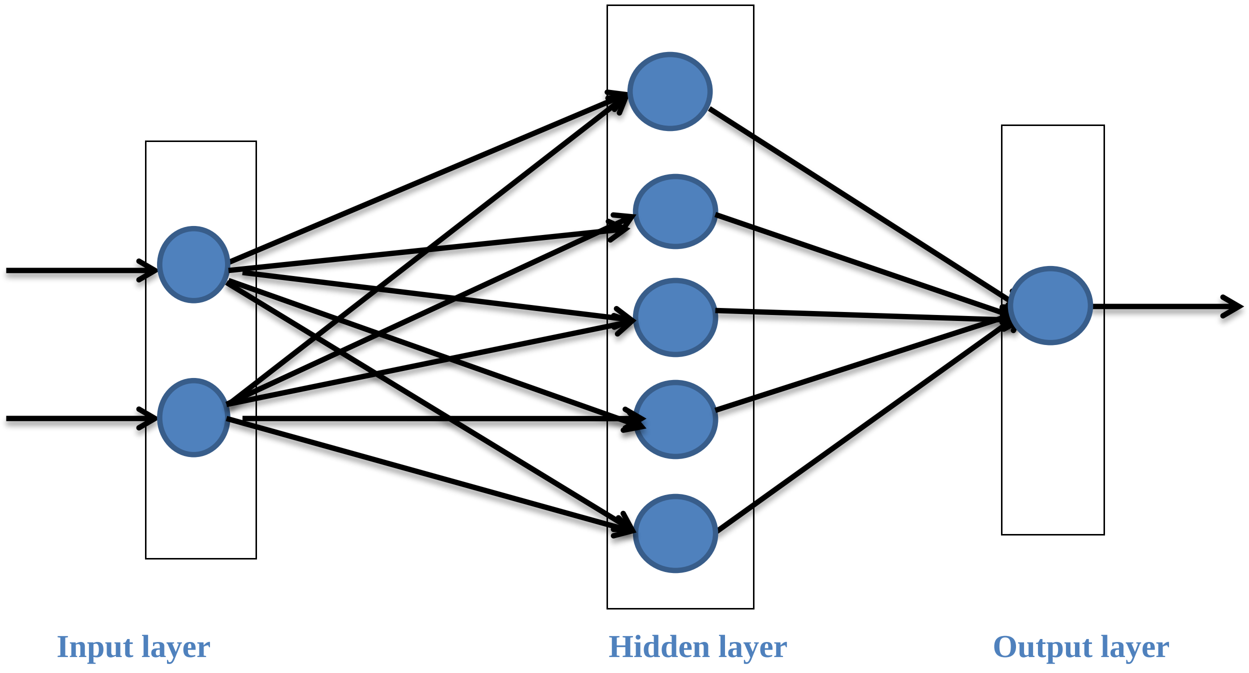


Figure 4-1 Multilayer Perceptron (Feed-forward Artificial Neural Net)

The third classification algorithm, we experimented with is the *Feed-forward Artificial network Classifier*. We used the same train set and test set from the corpus that we constructed before, including unigram features to a Neural Network by using Keras Deep Learning library for Theano and TensorFlow that was not practical due to the high dimensionality of feature vectors involved. We used a single hidden layer with 5 numbers of neurons. Because of the nature used data we use this NN than the others like CNN and RNN

Table 4-4 Accuracy Results for ANN

| Class | Precision | Recall | F-measure | Accuracy |
|---|---|---|---|---|
| Sarcasm | 0.83 | 0.79 | 0.81 | 80.15 |
| Non-Sarcasm | 0.77 | 0.81 | 0.79 | |

As it is shown in Table 4-4 above, we achieved an accuracy of 80.2% by using a decision tree classifier. In a file given, Sarcasm classified class is 83% likely to be correct. High precision causes only 17% false positive for the Sarcasm class relatively. The f-measure of the RF classifier with all unigram features for the class Sarcasm is 81%.

In the case of a Non-Sarcasm class, the text given a Non-Sarcasm classification is 77% likely to be correct. This is a good precision leads to 23% of a false positive for the negative categories. The f-measure of the Random Forest classifier with all unigram features for the class Non-Sarcasm *is* 79%.

### *Recurrent Neural Networks (RNN)*

The RNN uses an architecture that is not dissimilar to the feed forward NN. The difference is that the RNN introduce the concept of memory, and it exists in the form of a different type of link. Unlike a feed forward NN, the outputs of some layers are fed back into the input of the previous layer. This addition allows for the analysis of sequential data, which is something that the Feed Forward NN is incapable of. Also, Feed Forward NN is limited to a fixed length input, whereas the RNN has such restrictions. The inclusion of links between layers in the reverse direction allows for feedback loops, which are used to help learn concept based on context. RNN is applied successfully in many types of tasks. Some of this are image classification, Automatic language translation, Natural Language processing such as sentiment analysis, text classification etc. (EXEXXACT, 2019)

### *Convolutional Neural Networks (CNN)*

The defining features of the CNN are that it performs the convolution operation in certain layer-Hence, the name Convolutional Neural Network. The architecture varies slightly from the feed forward NN. In CNN, the first layer is always a convolutional layer. These are defined using the three spatial dimensions: length, width, and depth. These layers are not fully connected meaning that the neurons from one layer do not connect to each and every neuron in the following layer. The output of the final convolution layer is the input to the first fully connected layer. The most common application for CNN is in the general field of computer vision. Examples of this are medical image analysis, image recognition, face detection and recognition system and full motion video analysis.

Recurrent and Convolutional Neural Networks are common place in the field of deep learning. RNN and CNN architecture has advantage and disadvantage that are dependent upon the type of data that is being modeled. When choosing one framework over the other, or alternatively creating a hybrid approach, the type of data and the job at hand are the most important points to consider. (EXEXXACT, 2019)

## 4.6 DISCUSSION OF THE RESULT

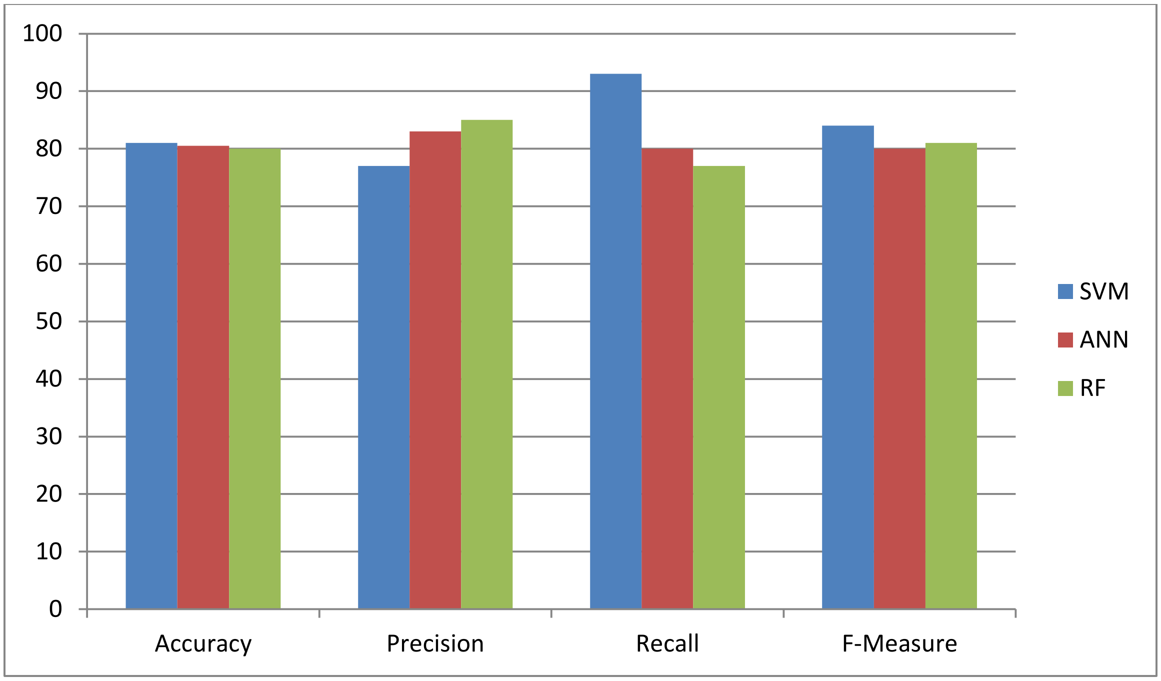


Figure 4-2 Model comparison for the three classifiers

Deciding on the Accuracy, Precision, Recall and F-values, we observed that the SVM classifier (with a bag of words) performs better than the other classifiers. Including these unigram features to a Neural Network by used Keras: Deep Learning library for Theano and TensorFlow was not practical due to the high dimensionality of feature vectors involved. This was however practical with the LibSVM library as in LibSVM: A Library for Support Vector Machines. The Random Forest is trained using these unigram features by used sklearn library from python. Further insights from the Feature Importance index (based on a decrease in Gini Impurity) reveal that features pertaining to Punctuation (e.g. no of '!') , laughter expressions (e.g. 'ኪኪኪኪ', 'ሃሃሃ') and emoticons are important in judging the sarcastic content from Amharic sentiment texts.

# CHAPTER FIVE
# CONCLUTION AND RECOMMENDATION

## 5.1 CONCLUTION

In this thesis, we have built a model for detecting sarcasm form Amharic sentiment text. The model is able to process raw text as input and outputs whether the text is sarcasm or not. The final model accuracy of 80.6%, 80.1 and 79% was obtained on the total collected datasets with the Support Vector Machine, Neural Network and Random Forest classifier respectively. We found some strong features that characterize sarcastic texts. From this, we can draw the conclusion that the model worked for these specific data sets and could identify the majority of the sarcasm texts.

In this research we conducted three different experiments to come up with a better performance. The result of the three learning algorithms was presented with whole unigram words as a feature and with informative features. Information gain was used as a feature selection technique to choose the most informative words. As we can observe from the experiments made using unigram words have a great impact in the classifier accuracy. The SVM algorithm with unigram words outperforms all algorithms with 80.6% accuracy.

This research work has tried to go through the techniques of sentiment mining for sarcasm detection in Amharic texts. To classify a given sarcasm text into predefined classes, the sarcasm text passes through pre-processing, detection of sentiment words, weight assignment and polarity classification processes. Pre-processing involves normalization and tokenization. The detection of sentiment words is a process of detecting polarity words and contextual valence shifters based on the sentiment lexicon. Weight assignment and polarity propagation is responsible for assigning an initial weight for detected sentiment terms and propagating polarity value of sentiment terms that are linked to contextual valence shifters. Polarity classification is concerned with categorizing a given sarcasm texts into predefined categories based on the weights obtained from the weight assign and polarity propagation process.

In order to detect the sentiment terms from a given sarcasm texts, assign initial value to the sentiment terms and propagate the initial polarity values, lexica of properly tagged Amharic sentiment terms are used.

The results of the lexicon-based sentiment mining model for Amharic sarcasm detection model using the processes explained above are encouraging. However, further work can be done to improve the proposed model's results.

## 5.2 RECOMMENDATION

When looking at the experimental results, it is clear that the bag of the word is the most discriminant feature for finding irony in the data sets that have been used. So this is important for future work to test other features that we did together with the bag of word and vocabulary to see if it can give the model a higher accuracy.

Researches in sentiment analysis in other language use linguistic resources like Thesaurus, Lexicon Word Net, spelling checking, speech of tagger machine readable dictionaries and machine translation software. This is a good idea to include it in the future work to facilitate Sentiment analysis related research for Amharic text.

The performance of the classifier needs to be improved for designing more efficient applicable system. We considered only unigram words as a feature. It would be interesting to investigate the feature type like bigram words as a feature.

For Amharic language, there is no standardized corpus for sarcasm detection. So the construction of a labeled corpus is very important because it would allow for more experiments, especially with supervised classifications. This is a good idea to include it in the future work.

In all supervised approaches, reasonably high accuracy can be obtained subject only to the source of the data that test data be similar to training data This dependency on annotated training data is one major shortcoming of all supervised methods. Unsupervised approaches are recommended as future work.

Sarcasm detection model by using another machine learning techniques can also be another research work direction which is concerned with identification and extraction of sarcasm comments and determining the sarcasm towards the given data sets.

Then after, we recommend that more research in the area should be done in this area. There is still a long way until a proper Sarcasm detector could be used in general situations.

## 5.3 FUTURE WORKS

Future research should focus on the development of approaches analyzing the vocabulary used in the Amharic sarcasm text in a deeper fashion. Our impression is that many sarcastic and ironic Amharic texts use words and phrases which are non-typical for the specific domain or product class. We propose that future research should focus on analyzing the specific vocabulary and develop semantic similarity measures which we assume to be more promising than approaches taking into account lexical approaches only.

Most work has been performed on text sets from one source like facebook, books, reviews, etc. Some of the proposed features mentioned in this paper or previous publications are probably transferable between text sources. However, this still need to be proven and further development might be necessary to actually provide automated domain adaption for the area of irony and sarcasm detection.

We approached the problem mainly from the data-driven perspective (annotation, feature engineering, error analysis). There are also possible extensions to the lexical/morphological features – either in the direction of semi-supervised learning and adding for example features based on latent semantics, topic models, or graphical models popular in the sentiment analysis field or the direction of deeper linguistic processing in terms of, e.g., syntax/dependency parsing These deserve further investigation and are planned in future work.

Hence, it should be noted that the corpus is actually a mixture of ironic and sarcastic Amharic texts. Irony and sarcasm are not fully exchangeable and can be assumed to have different properties. Further investigations and analyses regarding the characteristics that can be transferred are necessary

# REFERENCE

1. *EXEXXACT*. (2019, April 17). Retrieved April 27, 2019, from exxactcorp.com: https://blog.exxactcorp.com

2. Abhijit Mishra, D. K. (2017). *Harnessing Cognitive Features for Sarcasm Detection.* Bombay , India: Indian Institute of Technology.

3. Abreham, G. (2014). *OPINION MINING FROM AMHARIC ENTERTAINMENT TEXTS.* Addis Ababa: Addis Ababa University.

4. Aditya Joshi, V. T. (2016.). Are Word Embeddingbased Features for Sarcasm Detection? *EMNLP 2016* .

5. Afework, Y. (2007). *Automatic Amharic text categorization.* Addis Ababa: Addis Ababa university,computer science.

6. Alexander O'Neill. (2009). *Sentiment Mining for Natural Language Documents.* Canberra: Australian National University.

7. Alpaydin, E. (2004). *Introduction to machine learning.* The MIT Press.

8. Amir Silvio, B. C. (2016.). Modelling Context with User Embeddings for Sarcasm Detection in Social Media. *CoNLL 2016*, 167.

9. Arti B., e. a. (2017). Opinion mining and Analysis: A survey. *International journal on Natural language computing*.

10. B. Pang. (2008). Sentiment classification using machine learning techniques. *in Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

11. BayeYemam. (1987 ዓ.ም::). *የአማርኛ ሰዋሰው*፡ Addis Ababa: ት.መ.ማ.ማ.ድ.

12. Belete, M. (2013). *Sentiment Analysis for Amharic opinionated text.* Addis Ababa, Ethiopia: Ababa university.

13. Bender, L. a. (1976.). *The Ethiopian Writing System.* London: Oxford University Press.

14. Bing Liu. (2010). Sentiment analysis and subjectivity. In Bing Liu, *Handbook of natural language processing 2* (pp. 627–666). Chicago: Morgan & Claypool.

15. Camp., E. (2012). Sarcasm, Pretense, and the Semantics/Pragmatics Distinction. 587–634.

16. Chernet, Y. A. (2014). Political Satire in Abebe Tola's "Yabe Tokichaw Shimutochi" and "Yabe Tokichaw Mitsetochi" Essays. *International Journal of Literature and Arts* , 240-251.

17. Chun-Che Peng, M. L. (2015). Detecting Sarcasm in Text: An Obvious Solution to a Trivial Problem. *Writeup for Stanford CS 229 Machine Learning Final Project.*, 1.

18. D. Nadeau. (2007). *"Semi-Supervised Named Entity Recognition: Learning to Recognize 100 Entity Types with Little Supervision,"100 Entity Types with Little Supervision,".* Owtawa,: University of Owtawa.

19. Dawkins, C. (1969.). *The Fundamentals of Amharic, Sudan interior mission.*

20. Delia Iraz´u Herna´ndez Far´ıas, V. P. (2016). Irony Detection in Twitter: The Role of Affective Content. *ACM Trans. Internet Technol*, 24 pages.

21. Ellen Rilo, A. Q. (2013). Sarcasm as Contrast between a Positive Sentiment and Negative Situation. *In EMNLP*, 704–714.

22. Etstratios K., C. B. (2013). Ontology-based sentiment analysis of Twitter posts. *Expert system with applications*, 4065-4074.

23. Filatova., E. (2012). Irony and Sarcasm: Corpus Generation and Analysis Using Crowdsourcing. *In LREC*, 392-398.

24. Fred L. Drake, J. (2003). *Python Tutorial Release 2.3.3.* Python Software Foundation.

25. Gebremeskel, S. (2010). *SENTIMENT MINING MODEL FOR OPINIONATED AMHARIC TEXTS.* Addis Ababa: MSC Thesis.

26. Gibbs., R. W. (1994). *The poetics of mind: Figurative thought, language, and understanding.* London: Cambridge University press.

27. Goldberg, X. Z. (2009). "Introduction to semi-supervised learning," Synthesis. *Synthesis lectures on artificial intelligence and machine learning,*, 1-130.

28. González-Ibánez, R. S. (2011). Identifying sarcasm in Twitter: a closer look. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics* (pp. short papers-Volume 2). Association for Computational Linguistics.

29. Greenwood., D. M. (2014.). Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. *In Proceedings of LREC.*

30. H Paul Grice, P. C. (1975). .Syntax and semantics. *Logic and conversation 3(1975),*, 41–58.

31. Hao, T. V. (2010). Detecting Ironic Intent in Creative Comparisons. *In European Conference on*, 765–770.

32. Inkpen, A. K. ( 2006.). Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters. *Computational Intelligence,,*, Volume 22.

33. JOSHI, A. B. (2017). Automatic Sarcasm Detection: A Survey. *ACM Comput. Surv. 0, 0, Article 1000 ( 2017)*, 22 pages.

34. K. D.Lee, ,. (2011). *Python Programming Fundamentals.* 1st Edition, Springer-verlog press.

35. Katz., C. J. (1998). The differential role of ridicule in sarcasm and irony. In *Metaphor and symbol* (pp. 1–15).

36. Konstantin Buschmeier, P. C. (2014). An impact analysis of features in a classification approach to irony detection in product reviews.

37. Leslau, W. (2000). *Introductory Grammar of Amharic.*

38. Liebrecht, C. K. (2013). The perfect solution for detecting sarcasm in tweets #not. *In Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media*, (pp. pp. 29–37,).

39. Lin C. et. al. (2011). Sentence subjectivity detection with weakly-supervised learning. *proceeding of the 5th international joint conference on Natural language processing* (pp. pages 1153-1161). UK, london: University of Exter.

40. Lipo Wang ed. (2005). *Support Vector Machines: Theory and Applications.* Berlin: Springer.

41. Lunando, E. a. (2013). Indonesian social media sentiment analysis with sarcasm detection. *In 2013 International Conference on Advanced Computer Scienceand Information Systems (ICACSIS),.*

42. Michie, D. S. (1994). *Machine Learning, Neural and Statistical Classification.* Broke Books.

43. Norvig, R. S. (2003.). *Artificial Intelligence: A Modern Approach",.* Prentice hall,.

44. O. Chapelle. ( 2006.). *Semi-supervised learning.* MIT press Cambridge.

45. Oren Tsur, D. D. (2010). Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews. *ICWSM-A Great Catchy Name*.

46. Pedregosa, F. V. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2825–2830.

47. Peng Liu, W. C. (2014). Sarcasm Detection in Social Media Based on Imbalanced Classification. *In Web-Age Information Management*, 459-471.

48. Pexman, S. L. (2003). Context incongruity and irony processing. *Discourse Processes 35*, 241–279.

49. Pt´aˇcek, T. H. (2014). Sarcasm detection on czech and english twitter. *In The 25th International ConConference on Computational Linguistics*.

50. python. (n.d.). Retrieved 2018, from Python Programming Language: "http://www.python.org/doc/essays/blu, last

51. Rajadesingan, A. R. (2015). Sarcasm detection on Twitter: A behavioral modeling approach. *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining.*

52. Raymond W Gibbs., J. G. (1994). *poetics of mind: Figurative thought, language, and understanding.* London, Cambridge : Cambridge University Press.

53. Reyes, A. R. (2013). A multidimensional approach for detecting irony in twitter. *Language Resources and Evaluation*, 239–268.

54. Rosso, A. R. (2014). On the difficulty of automatically detecting irony: beyond a simple case of negation. *Knowledge and Information Systems*, 595-614.

55. Santosh Kumar Bharti, K. S. (2015). Parsing-based Sarcasm Sentiment Recognition in Twitter Data. *In Proceedings of the 2015 IEEE*, 1373–1380.

56. Santosh Kumar Bharti, K. S. (2015). Parsing-based Sarcasm Sentiment Recognition in Twitter Data. . *In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015* (pp. 1373–1380). ACM.

57. schrauwen, S. (2010.). *Machine learning approach to sentiment analysis using the Dutch Netlog corpus.* Computational linguistics and psycholinguistic technical report series(CLIPS).

58. Sebsibeet. (2004). Unit selection for Amharic using FESTVOX,5th ISCA speech synthesis workshop. *Language technology research center*.

59. Solomon Teferra Abate, W. M. (2005). An Amharic speech corpus for large vocabulary continuous speech recognition. *Ninth European onference on speech communication and technology.* France: isca-speech.org.

60. Soujanya Poria, E. C. (2016). A deeper look into sarcastic Tweets using deep convolutional neural networks. *arXiv preprint arXiv:1610.08815*.

61. Stefan Stieger, A. K. (2011). Humor styles and their relationship to explicit and implicit self-esteem. In *Personality and Individual Differences 50* (pp. 747-750).

62. Steven Bird, E. K. (2008). *Natural Language Processing in Python.*

63. Tadesse, B. (1994). The Ethiopian Writing System. *Paper presented at the 12th International Conference of Ethiopian Studies.* Michigan : Michigan State University.

64. Varela, P. d. (2012). *Sentiment Analysis.* Indian journal of computer science and Engineering.

65. Veale., A. G. (2016). Fracking Sarcasm using Neural Network. *WASSA NAACL (2016)*.

66. Walk, S. L. (2013). Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. *In Proceedings of the Workshop on Language Analysis in Social*, (pp. 30–40).

67. Witten, I. H. (2005). *Practical machine learning tools and techniques.*

68. woldekirkos, M. (1934.). *“አማርኛ ሰዋሰው*: Addis Ababa: Berhanenaselam printing press.

69. Xiaoying. (2009.). Categorizing Terms’ Subjectivity and Polarity Manually for Opinion Mining in Chinese. *IEEE*.

# ANNEXES

## 3.6 APPENDEX A: Sample representation of sentiment terms in our dictionary

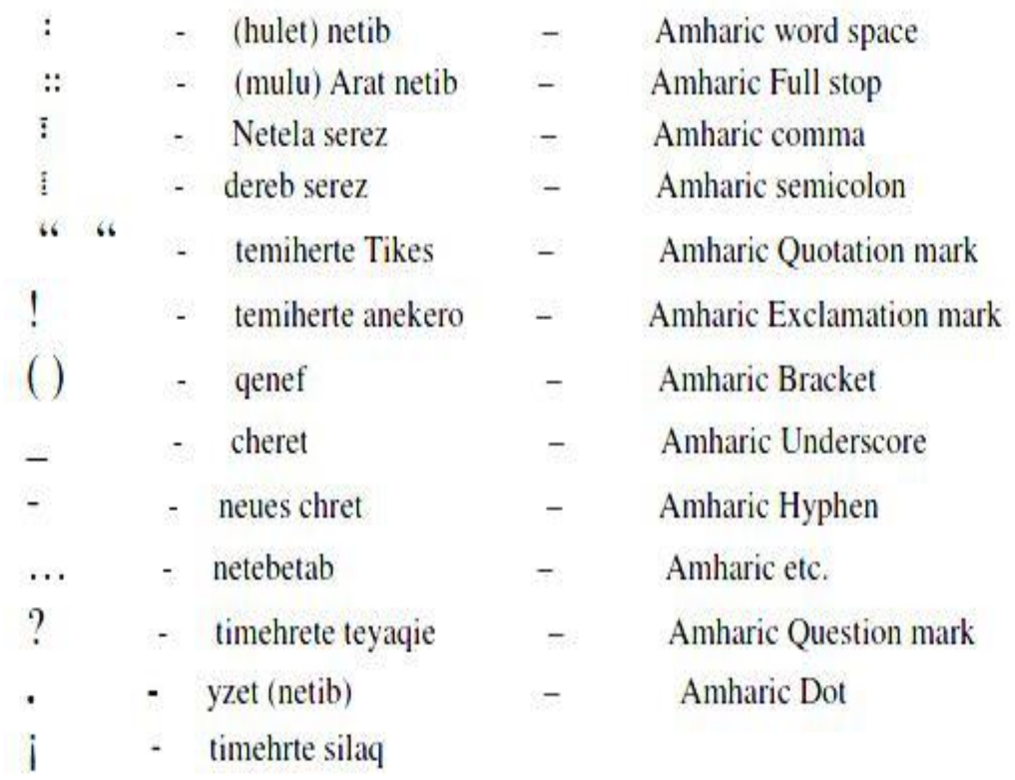
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ሀላፊነት* | 2 | ለልማት* | 2 | ልእልና* | 2 | ርካሽ* | -2 |
| ሀመልማል* | 2 | ለመተዛዘብ* | -2 | ልክ* | 2 | ርካሽ* | -2 |
| ሀሜት* | -2 | ለመከልከል* | -2 | ልዩ* | 2 | ሰለባ* | -2 |
| ሀሞቱ_ፈሰሰ | -2 | ለመፍቀድ* | 2 | ልዩነታችን* | -2 | ሰላማዊ* | 2 |
| ሀረግ_መዘዘ | -2 | ለመፍቀድ* | 2 | ልግመኛ* | -2 | ሰላም* | 2 |
| ሀራም* | -2 | ለማስገደድ* | -2 | ልፈፋ* | -2 | ሰላም* | 2 |
| ሀርነት* | 2 | ለማኝ* | -2 | ልፍያ* | -2 | ሰላምተኛ* | 2 |
| ሀሰተኛ* | -2 | ለማዋረድ* | -2 | ሎሌ* | -2 | ሰበነክ | -2 |
| ሀሰት* | -2 | ለማዋረድ* | -2 | ሎጋ* | 2 | ሰብአዊ* | 2 |
| ሀሴት* | 2 | ለማይል* | -2 | ሐራጅ* | -2 | ሰናይ* | 2 |
| ሀቀኛ* | 2 | ለምፍ* | -2 | መሀይምነት* | -2 | ሰውረን* | -2 |
| ሀቅ* | 2 | ለቅሶ* | -2 | መላህክ* | 2 | ሰውረኝ* | -2 |
| ሀብታም* | 2 | ለችግር* | -2 | መላምታዊ* | -2 | ሰይጣን* | -2 |
| ሀብት* | 2 | ለእውነት* | 2 | መልእክት* | 2 | ሰይጣን* | -2 |
| ሀብት* | 2 | ለዘብተኛ* | -2 | መልእክት* | 2 | ሲራብ* | -2 |
| ሀካይ* | -2 | ለዛ* | 2 | መልከመልካም* | 2 | ሲሰረቅ* | -2 |
| ሀዋሪያ* | 2 | ለዛ_ሙጥጤ | -2 | መልካም* | 2 | ሲሰርቅ* | -2 |
| ሀዘን* | -2 | ለጋስ* | 2 | መልካም* | 2 | ሲሳይ* | 2 |
| ሀያል* | 2 | ለጋሽ* | 2 | መማቀቅ* | -2 | ሲቀር* | -2 |
| ሀያል* | 2 | ለጠላትህ* | -2 | መሞቱን* | -2 | ሲቃጠል* | -2 |
| ሀይለኛ* | 2 | ለጠላትህ* | -2 | መሞታቸውን* | -2 | ሲተረማመሱ* | -2 |
| ሀይማኖተኛ* | 2 | ለጨለማዋ* | -2 | መሞት* | -2 | ሲኦል* | -2 |
| ሁነኛ | 2 | ለፈፈ* | -2 | መረዳዳት* | 2 | ሲዘበዝብ* | -2 |
| ሁከት* | -2 | ለፍላፊ* | -2 | መረጋጋት* | 2 | ሲያለቅስ* | -2 |
| ሃዘን* | -2 | ላባ_ቀረሸ | -2 | መራራ* | -2 | ሲያለቅስ* | -2 |
| ሃጥያቱ* | -2 | ሌሊት* | -2 | መራር* | -2 | ሲያረጅ | -2 |
| ህሊና* | 2 | ሌባ* | -2 | መርዘኛ* | -2 | ሲያረጅ* | -2 |
| ህመሙ* | -2 | ሌቦች* | -2 | መርዝ* | -2 | ሲገደል* | -2 |
| ህመም* | -2 | ልመና* | -2 | መሰረት* | 2 | ሲጮኹ* | -2 |
| ህቅታ* | -2 | ልመና* | -2 | መሳካት* | 2 | ሳያመነታ* | 2 |
| ህብረት* | 2 | ልሙጥ* | 2 | መሳጭ* | 2 | ሳይሆን* | -2 |
| ህክምና* | 2 | ልማታዊ* | 2 | መስተፋቅር* | 2 | ሳይሆን* | -2 |
| ህዝባዊ* | 2 | ልማት* | 2 | መሸወድ* | -2 | ሳይሰበሰብ* | -2 |
| ህያው* | 2 | ልምላሜ* | 2 | መሻሻል* | 2 | ሳይሳካ* | -2 |
| ህገወጥ* | -2 | ልምድ* | 2 | መቀለጃ* | -2 | ሳፋሪ* | -2 |
| ህገወጥነት* | -2 | ልቅ* | -2 | መቃምያ* | -2 | ሴረኛ* | -2 |
| ህጋዊ* | 2 | ልበ_ቀና | 2 | መቃምያ* | -2 | ሴራ* | -2 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ህግ* | 2 | ልበ_ቢስ | -2 | መቃብር* | -2 | ሴት_አውል | -2 |
| ህግ_ገባ | -2 | ልቡ_ሰባ | -2 | መቃወም* | -2 | ስህተት* | -2 |
| ህፀፅ* | -2 | ልቡ_ቆመ | -2 | መቃጠል* | -2 | ስህተት* | -2 |
| ሆደ_ሰፊ | 2 | ልቡ_ተነሳ | -2 | መቅሰፍት* | -2 | ስህተቶች* | -2 |
| ሆዱን_ሰጠ | -2 | ልቡ_አበጠ | -2 | መቅጫ* | -2 | ስለእውነት* | 2 |
| ለሀሰት* | -2 | ልቡ_ወለቀ | -2 | ሙብት* | 2 | ስላልሆነ | -2 |
| ለለቅሶ* | -2 | ልቡ_ጠፋ | -2 | መተዛዘኛ* | 2 | ስላልሆነ* | -2 |
| ለለቅሶ* | -2 | ልብ* | 2 | መተጋገዝ* | 2 | ስላልቻሉ* | -2 |
| ለለቅሶ* | -2 | ልብ_ገዛ | 2 | መታመምን* | -2 | ስላጣ* | -2 |
| ለልመና* | -2 | ልእልና* | 2 | መታወኩ* | -2 | ስልት* | 2 |
| ማስፈራሪያ* | -2 | ምላሽ* | -2 | ሞልቃቃ* | -2 | ስልጡን* | 2 |
| ማሸንክ* | -2 | ምሩቅ* | 2 | ሞልጣፋ* | -2 | ስልጣኔ* | 2 |
| ማተቡን_በጠሰ | -2 | ምሩቅ* | 2 | ሞቅ* | 2 | ስመ_ጥር* | 2 |
| ማታለል* | -2 | ምራቁን_የዋጠ | 2 | ሞት* | -2 | ስመጥር* | 2 |
| ማነስ* | -2 | ምርቃት* | 2 | ሞትን* | -2 | ስመጥር* | 2 |
| ማአረግ* | 2 | ምርጥ* | 2 | ሞኝነት* | -2 | ስሜታዊ* | -2 |
| ማአረግ* | 2 | ምርጥ* | 2 | ሞያ* | 2 | ስምምነት* | 2 |
| ማዋረድ* | -2 | ምሳሌ* | 2 | ሞገስ* | 2 | ስር_በጣሽ | -2 |
| ማይገኝ* | -2 | ምስኪን* | -2 | ሟሽሟጠጥ* | -2 | ስርቅታ* | -2 |
| ማግለል* | -2 | ምስጋና* | 2 | ረብሻ* | -2 | ስርዝ* | -2 |
| ማጣት* | -2 | ምቀኛ* | -2 | ረዣዥም* | 2 | ስስታም* | -2 |
| ማጭበርበር* | -2 | ምቅኝነት* | -2 | ረጋ* | 2 | ስቃይ* | -2 |
| ምሁራንና* | 2 | ምቹ* | 2 | ረጋ_ሰራሽ | 2 | ስብእና* | 2 |
| ምሁራዊ* | 2 | ምቾት* | 2 | ሩሁሩ* | 2 | ስትቃጠል* | -2 |
| ምሁር* | 2 | ምክር* | 2 | ሩህሩህ* | 2 | ስነ_ስርአት* | 2 |
| ምህረት* | 2 | ምዝበራ* | -2 | ራእይ* | 2 | ስንክሳሩ_በዛ | -2 |
| ምህዳር* | 2 | ምጣኔ* | 2 | ሬሳ* | -2 | ስንዱ* | 2 |
| ምላስ_አወጣ | -2 | ምፀት* | -2 | ርህራሄ* | 2 | ስካር* | -2 |
| ስደተኛ* | -2 | ቀላል* | 2 | ቃጠሎ* | -2 | በሀሰት* | -2 |
| ስደተኞች* | -2 | ቀልቃላ* | -2 | ቅልጣን* | -2 | በሃዘን* | -2 |
| ስድ* | -2 | ቀልጣፋ* | 2 | ቅልጥ* | 2 | በህገወጥ* | -2 |
| ስጋት* | -2 | ቀማኛ* | -2 | ቅልጥፍና* | 2 | በሊታ* | -2 |
| ስጦታ* | 2 | ቀርፋፋ* | -2 | ቅማላም* | -2 | በሙስና* | -2 |
| ሽሌ* | -2 | ቀርፋፋ* | -2 | ቅሬታ* | -2 | በሚገባ* | 2 |
| ሽረኛ* | -2 | ቀሳፊ* | -2 | ቅር* | -2 | በማጥፋት* | -2 |
| ሽረከተ_ | -2 | ቀሽም* | -2 | ቅናታም* | -2 | በምቀኛ* | -2 |
| ሽርሙጣ* | -2 | ቀሽት* | 2 | ቅናት* | -2 | በረከቱ* | 2 |
| ሽባ* | -2 | ቀናተኛ* | -2 | ቅን* | 2 | በረከት* | 2 |
| ሽባ* | -2 | ቀንበር_ሰበረ | -2 | ቅንጦት* | 2 | በሬ_ወለደ | -2 |
| ሽክም* | -2 | ቀኝ* | 2 | ቅዱስ* | 2 | በርቱ* | 2 |

## 3.7 APPENDEX B: Sample of Amharic sarcasm texts

1. ኢትዮጵያውያን እና ማንዴላን የሚያመሳስላቸው ሁለቱም ከ27 የጨለማ እስር በኋላ መፈታታቸው ነው!!
2. አዲስአበባ ውስጥ ባለቤት አልባ ህንጻዎች እየበዙ ስለሆነ መንግስት ይስጠኝ እና ላሳድጋቸው!!
3. የአንድ ጥሩ ፕሬዚዳንቱ ምልክት መጀመሪያ ቀድመው ተኩሰው ሰው ያስገድሉ እና ይቅርታ ይጠይቃል !
4. ኮሌጅ ማለት እናትህን ለ4 አመት ከመጠጥ መሸጫ ቤት ማጣት ማለት ነው!
5. ምንም አይነት አስተያየት ለማየት ፈቃደኛ ያልሆነ ሰው ጋር አብሬ ስራ በመስራቴ ደስ ይለኛል!... 😟😟
6. እንደ መጥፎ ሰው ሁኘ ሁልጊዜ መታወሴን እወደዋለሁ። አዎን ሙሉ በሙሉ ! እኔ በስጋ ሰይጣን ነኝ!! የእኔን መንገድ ተከተሉ! ደስ ይለኛል።
7. ላንች ብየ የምሞት ከሆነ በጣም እወድሻለው ማለት ነው... ቂቂቂቂቂቂቂ!!!!
8. ስማቸው እንዲገለጽ ያልፈለጉ የሸገር ቄራዎች ድርጅት ዋና ስራአስኪያጅ አቶ ተክለማርቆስ፤😄 ሰሞኑን የተከሰተው የስጋ እጥረት በቂ ቢላዋ ባለመኖሩ የተከሰተ መሆኑን ገለጹ....!!😄😄
9. አርብቶ አደሩ የተትረፈረፈ ከብት በማምረት ያሳየው ትጋት፤ ቀጥቃጮች በቂ ቢላዋ በማምረት እንዲደግሙት አሳስበው፤ እስከዛው ድረስ ግን የኢትዮጵያ ህዝብ ከሃይማኖት አባቶች ጋራ በመተባበር ተጨማሪ ጦም እንዲያውጅ አሳስበዋል...!!😄😄
10. መምህራችን ኢትዮጵያ በቀንድ ከብት ካፍሪካ አንደኛ መሆኗን ሲተርክልን ከላይ ያለው አናጢ ምነ ድስኩሩን ትተህ ተጠመኔው አጠገብ ያለችትን ቢስማር ብታቀብለኝ እያለ ያናጥበዋል..!!
11. ቦርጭን ወደ ደብረማርቆስ ለመጀመርያ ጊዜ ያስገቡት ጋሸ ዮሴፍ ይመስሉኛል!! ጋሸ ዮሴፍ ከሶስት መቶ ብር ደመወዝተኛ የማይጠበቅ ቀፈት በማውጣታቸው አቃቤህግ ከሰሳቸው..!!😄
12. ኪራላይሶ ! የሰው ልጅ አይኑን ባይኑ ማየት እንደሚችል የተረዳሁት ለመጀመርያ ጊዜ የትየ ስለናት ጥፊ ፊቴ ላይ ሲያርፍ ነው 🙂
13. አሁን ሳስበው፤ ትምርትቤታችን በትምህርት ሚኒስትር ስር የሚተዳደር ራሱን የቻለ የቶርች ካምፕ ኑሯል😁😁
14. የድመት ነፍስ እና የተርምኔተርን አካል የሰጠኝ የወንቃው ጊዮርጊስ የተመሰገነ ይሁን ! ይህ በእውነተኛ ገጠመኝ ላይ የተመሰረተ ኩሸት ነው
15. በተማሪ ቤት ያስተዋልነው የጉልበተኛ መምህሮችና ያቅመቢስ ተማሪዎች ግኑኝነት ከትምርት ቤት ስንወጣ የቀረ ከመሰለን ተሳስተናል...😎
16. ደርግ እድለቢስ መንግስት ስለሆነ እንኳን ቦምብ ወረቀት ሲጥል ህዝብ ይፈጃል...😎
17. ባገራችን በዋናነት የሃብት ምንጭ ጉልበትና ህገወጥነት ነው!😟
18. መኖር ማለት የማይበርድ ናፍቆት ማለት ይሆን? መልመድ ይቅደም! ናፍቆት ይውደም!! ያለምንም ደም!!!
19. ህገሬ ገብቼ ከልማቱም ከግማቱም ብሳተፍ አይሻልም? አልኩት በቁርጠኝነት!
20. አደራ ገዥውን ፓርቲም ሆነ ጉምዥውን ፓርቲ እንዳትመርጥ!!! ዝምብለህ በኤፍ ኤም ደውለህ ለፍቅረኛህ ዘፈን ምረጥ..!!😄
21. አትልፉ መሪዎች የሚወገዱት በምርጫ ወይም በትጥቅ ትግል ሳይሆን በኮሌስትሮል ነው..!!😄
22. የጊዜ ጉዳይ ነው! ጌቶች በህዝብ ግፊት ባይወገዱ በደም ግፊት ይወገዳሉ!!!
23. ጅግናው የውሃ ልማት ሰፈር ህዝብ ሆይ! ተው ምረጠኝ ብትመርጥኝ የኩራዝ መለዋዋጫ እቃዎች አቀርባለሁ ተች ስክሪን ፋኖስ አስገባለሁ
24. የታደለ አገር ሁለመናው ጋዝ ያልታደለው አገር ሁለመናው መጋዝ!! እኛ አገር ቢቆፈር የሚወጣው የሰማእታት አጽም ብቻ ነው!😟
25. ወደብ ባይኖራትም መርከብ አላት ኤሌክትሪክ ባይኖራትም የኤሌክትሪክ ባቡር አላት..!! አይ ኢትዮጲያ..!!
26. ሀገር ማለት ሰው ነው፤ እስኪ ሙት በለኝ!!😄 ባክህ ይሄ እንኳን የጋማ ከብቶች ወሬ ነው..! ሎሎሎሎልልልልል..!

## 3.8 APPENDEX C: Amharic Punctuation Marks, Source: Bender et al. 1976



| | | | | |
|---|---|---|---|---|
| ፡ | - | (hulet) netib | – | Amharic word space |
| ። | - | (mulu) Arat netib | – | Amharic Full stop |
| ፣ | - | Netela serez | – | Amharic comma |
| ፤ | - | dereb serez | – | Amharic semicolon |
| “ “ | - | temiherte Tikes | – | Amharic Quotation mark |
| ! | - | temiherte anekero | – | Amharic Exclamation mark |
| ( ) | - | qenef | – | Amharic Bracket |
| _ | - | cheret | – | Amharic Underscore |
| - | - | neues chret | – | Amharic Hyphen |
| … | - | netebetab | – | Amharic etc. |
| ? | - | timehrete teyaqie | – | Amharic Question mark |
| . | - | yzet (netib) | – | Amharic Dot |
| ¡ | - | timehrte silaq | | |