Systems and Soft Computing 7 (2025) 200258

Contents lists available at ScienceDirect

Systems and Soft Computing

journal homepage: www.journals.elsevier.com/soft-computing-letters

Identification of hateful amharic language memes on facebook using deep learning algorithms

Mequanent Degu Belete , Girma Kassa Alitasb *

School of Electrical and Computer Engineering, Debre Markos Institute of Technology, Debre Markos University, Debre Markos, Ethiopia

ARTICLE INFO

Keywords:
Deep learning
BILSTM
BIGRU
Amharic language hate speech

ABSTRACT

Hate speech has been disseminated more frequently on social media sites like Facebook in recent years. On Facebook, hate speech can proliferate through text, image, or video. We suggested a deep learning approach to identify offensive memes posted on Facebook in case of Amharic language'. The research process commenced by manually gathering memes posted by Facebook users. Next came textual data extraction, annotation, preprocessing, splitting, feature extraction, model development and assessment Amharic OCRs were employed to extract textual data. Character normalization, stop word removal, and unnecessary character removal make up the text-preprocessing step. Using Stratified KFold the textual dataset is split into the train set (80 %), the validation set (10 %) and the test set (10 %). Vectors are created from the preprocessed texts using the Bog of words (BOW), TFIDF and word embeddings. Following that, the vectors are fed into Machine learning algorithms: NB, DT, RF, KNN, LSVM and LR, and deep learning models that are based on Dense, BiGRU, and BiLSTM algorithms. The model with the optimal parameters is chosen after numerous experiments. With an accuracy rate of 94 %, the BiLSTM + Dense model, the suggested technique identified nasty meme posts on Facebook written in Amharic.

1. Introduction

The so-called internet has connected people all over the world. As of April 2024, the world had 5.44 billion internet users worldwide, 67.1 % of a total population, with 5.07 billion of those users being social media users, according to DATAREPORTAL [1].

Social media platforms are a great way to keep people in touch. That being said, not all of the information shared on social media is significant. The quantity of hostile content increases along with the user base. Hate speech can spread from secret chat rooms to public posts via text, audio, video, and text picture (memes).

The biggest problem facing Ethiopia right now is hate speech, which spreads via social media, YouTube, and broadcast partnerships and has led to confrontations between nations, nationalities, and ethnic groups [2–4]. Facebook is one of these social networking sites; it has 2.91 billion users worldwide, with 6.8 million of them users residing in Ethiopia [5]. Facebook is being used by extremists to seriously hurt Ethiopian citizens [6] . In order to control such hate speech, the Ethiopian parliament passed a "hate-speech proclamation" on February 13, 2020 [7]. Unfortunately, due to phony identities and the rise in Facebook users, local laws of this kind are unable to control hate speech posted on Facebook walls [6]. However, because of developments in machine learning and computer vision, hate speech on Facebook may now be identified and controlled before it appears. Hence, this paper to develop automated systems that can accurately identify and flag hateful memes on social media platforms, Facebook. This is crucial for mitigating the spread of online hate speech and creating a safer digital environment.

For text submissions and comments in the Amharic language, hateful content detection has been done [8–11] An acoustic hate speech identification model for Amharic movies was created by Debele et al. [12]. Using deep learning techniques, Ayichilie Jigar M. et al. [13]conducted an experiment to identify offensive messages that appear as text-images, or memes. Only 2000 memes have been collected, though. Furthermore, the unimodal displayed subpar accuracy. Because they target specific people directly and receive more views due to their short captions and the fact that they are posted on public pages, text-image postings which for the sake of this study are also referred to as screenshot texts or memes discriminate and abuse more than text posts. As a result, this study used a deep learning methodology to address hostile text-image messages

* Corresponding author.
E-mail addresses: mequanent_degu@dmu.edu.et (M.D. Belete), girma_kassa@dmu.edu.et (G.K. Alitasb).

https://doi.org/10.1016/j.sasc.2025.200258
Received 12 August 2024; Received in revised form 2 November 2024; Accepted 23 April 2025
Available online 24 April 2025

(memes) that became widespread on Facebook.

There are two main responsibilities involved in developing text-image based hate detection systems. Text extraction from text-image posts is the first step, and model construction is the second to determine if the recovered text is free or hateful [14].

2. Related works

Numerous studies are conducted on the identification and classification of hate speech because it has become a significant problem for any online platform that hosts user-generated material. In order to create hate speech recognition models, these studies used machine learning or deep learning techniques, which use deep artificial neural networks to learn abstract feature representations from input data through its various layers [15–17].

Naïve Bayes is one of the machine learning algorithms that is easy to use, quick to train, and works best with tiny amounts of data. perform badly, nevertheless, when dealing with big data sets or requiring a sophisticated machine learning architecture. Deep learning is used to mitigate these drawbacks.

One of the deep learning techniques used in time series prediction and classification, such as sentiment analysis, text classification, and language translation, is the recurrent neural network (RNN), which has memory units to maintain data dependencies [18]. In order to address vanishing gradient issues and deal with both long and short temporal dependencies, the RNN architecture was significantly enhanced. LSTM and GRU are two of these enhanced designs. In order to enhance error flow in the current RNN, the Long Short-Term Memory (LSTM) was created to handle both long and short temporal dependencies [19]. To create a bidirectional-LSTM (BiLSTM), some changes are made to the original LSTM [20]. In contrast, the vanishing gradient problem was intended to be tackled with the Gated Recurrent Unit (GRU) [21]. GRU is expanded to bidirectional-GRU (BiGRU), similar to LSTM.

Because machine or deep learning algorithms require numerical input data, research conducted for hate detection also included text transformers, also known as feature extraction techniques, which convert text into vectors. Translating words into vector space is the process of feature extraction in text. Word embeddings, Term Frequency-Inverse Document Frequency (TF-IDF), and Bag of Words are a few text vectorization approaches used.

A histogram representation of words based on independent attributes is called a Bag of Words (BoW) [22]. Since all words in BoW have the same semantic representation, more significant terms in a document cannot be represented. The drawback of BoW is mitigated by the frequency-based method, TF-IDF [23]. TF-IDF, like BoW, does not retain semantic information, which increases the possibility of overfitting the classification model [24] . Word embedding, a deep learning technique, overcomes the shortcomings of the existing text representation methods. Word embedding is a learnt representation for text in which words with the same meaning have a similar representation. It presents hurdles for natural language processing (NLP) issues. Among the word embedding techniques are Word2vec [25], GloVe [26], Fasttext [27], and BERT [28].

The following is a summary of some of the most recent research on the identification of hate speech on social media using the previously described methods.

Schmidt and Wiegand [29] conducted a survey on the use of natural language processing (NLP) for the detection of hate speech. They therefore proved the direct connection between sentiment analysis and hate speech. A lethal natural language processing optimization ensemble deep learning strategy is used to automatically identify hate speech from Twitter utilizing the sentiment-based feature of Al-Makhadmeh and Tolba's [15] work. In addition to sentiment-based characteristics, [15] also used three other features: semantic, unigram, and pattern. However, Z. Zhang et al. [16] suggested using "skipped" GRU structures to find implicit properties that might be helpful in recognizing hateful tweets.

The Voting Based Ensemble Classifier, which was developed by S. Madisetty et al. [30] and composed of three deep learning techniques: CNN, LSTM, and Bi-LSTM showed that the performance of the ensemble approach for social media aggression detection outperformed that of the individual techniques. Nevertheless, in their work, the test size is too small and the classes in the test set are not balanced to allow for reliance on the suggested model's performance.

Singh et al. [31] and Abhishek et al. [32] employed BERT and multimodal models to categorize nasty memes. On the other hand, Konstantinos and Goutsos [33] used residual neural networks and RoBERT in conjunction with text and image modalities to identify hate speech in Greek social media.

To detect hateful postings and comments in the Amharic language on the Facebook network, machine-learning algorithms were utilized by Mossie and Wang [8] and Kenenisa [9]. For both TF-IDF and word2vec feature extractions, NB fared better in [8] than RF did. However, in [9], RF performed better than NB. This suggested that there has never been a machine learning algorithm that is superior. However, Kuluo H. [34] suggested using the SVM model in conjunction with word2vec rather than the LR, DT, and NB models based on TF-IDF and word2vec to filter text content in the Amharic language into non-offensive, Sol-offensive, Pol-offensive, and Rel-offensive categories. When it came to classifying comments and postings on Facebook as hateful or free, Tesfaye S [10]. offered the LSM model, which performed with an accuracy of 97.1 % instead of GRU, despite the use of established machine learning algorithms for hate classification on social media for the Amharic language, such as [8,9],34],. Nevertheless, duplicate samples were found in the testing and training datasets. As a result, the accuracy might not be as claimed when the duplicate samples are eliminated.

Author Hailemichalel E [35]. had applied LSTM, BiGRU, CNN, BiLSTM and BiGRU to develop fake news detection models for Amharic language. As a result, they recommend that, BiGRU, achieved 94 % followed by BILSTM, 93 % accuracy. On the other hand, Bewuketu Molla [11]Performed Amharic language stance detection using CNN, LSTM, CNN+LSTM and BiLSTM algorithm and it proposed BiLSTM algorithm achieved better performance, which is 0.93 accuracy. Despite hateful contented identification for textual posts and comments on social media, Ayichlie Jigar [13] applied multimodal analysis in detecting Amharic hate speech. It paired CNN and BiLSTM algorithms to achieved 0.75 accuracy score in case of multimodal (picture and text) and 0.65 accuracy score in case of unimodal (texts alone).

In addition to systematic review of Demilie et al. [36] who recommended deep learning approaches to challenge hate speech detection for Ethiopian languages, from the presented related works, it is observed that BiGRU and BiLSTM neural networks out performed for text information filtering, hate speech detection and fake text classification for different languages.

Debele et al. [12].utilized BILSTM to automate multimodal Amharic language hate speech. In this work, BILSTM performed accuracy of 88.15 %, however, the datasets are too small to generalize.

3. Methodology

3.1. Introduction

In order to create the suggested model, we first gathered text-image postings from Facebook, then we annotated the data, preprocessed the text, extracted features, constructed models, and finally evaluated the finished models.

3.2. Data gathering and annotation

Following the collection of 5000 memes postings from different sources across Facebook platform in the image format seen in Fig. 2, messages are extracted.

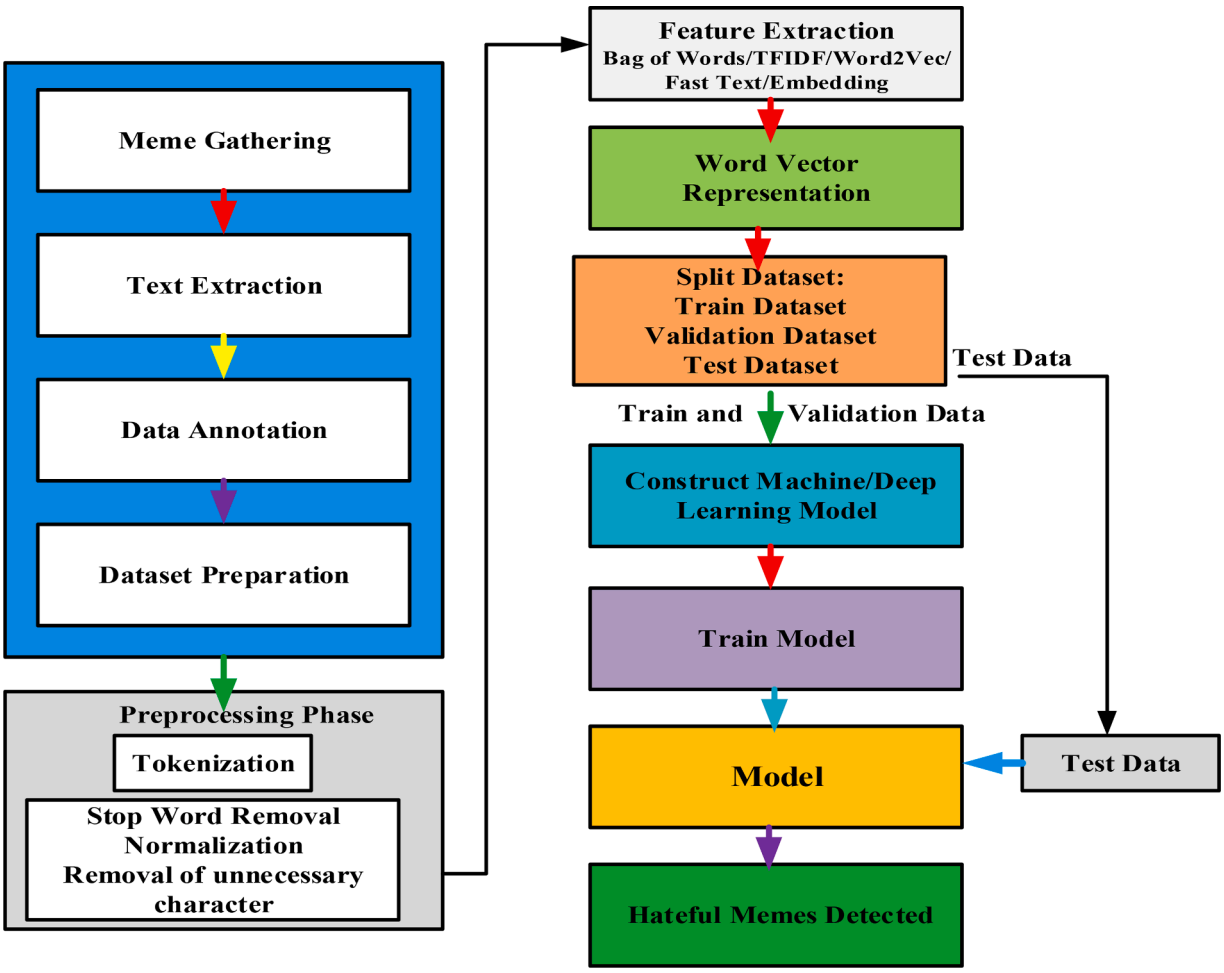

Fig. 1. System Architecture.

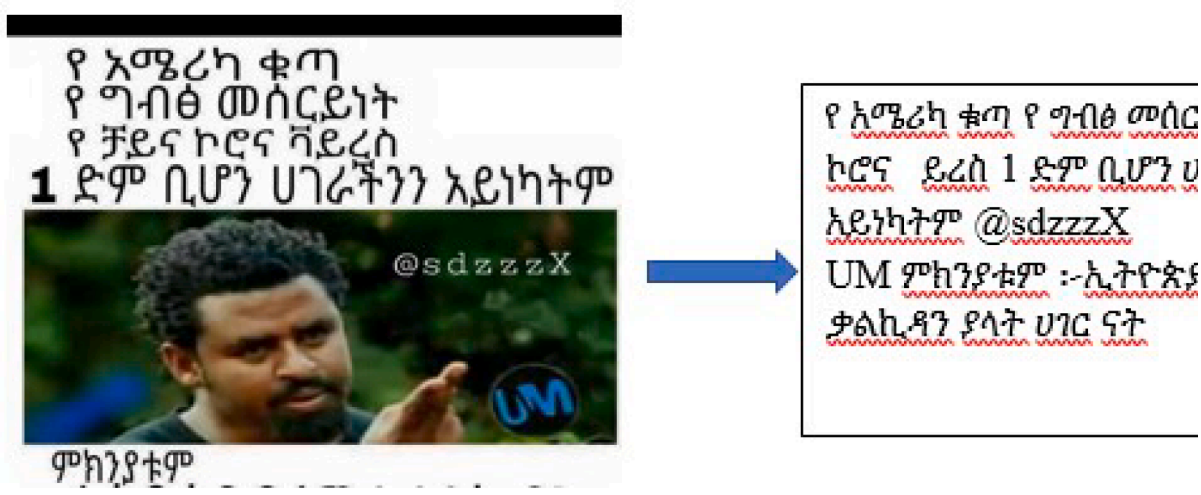

Fig. 2. Text-image post (left side) and extracted text (right side).

Fig 1

In this research, the method of identifying a given text as hateful or free based on predefined guidelines is called data annotation [8]. With a Kappa value of 0.61, the degree of agreement between the two designated annotators is good [37] Thus, of 5000 samples, 45 % were marked as free and 55 % as hateful. Mendeley Data has the textual dataset that was used in the work that is being presented [38] .

- The guidelines for the annotation task are as follows:

I. Determining a text or sentence's discourse is the first stage. This indicates that text's substance is classified as personal, political, religious, or ethnic.

Table 1
Guideline for data annotation.

Is insult	Is offensive		Label /Class
	Act of terrorism	Act of attack	
Yes	Yes or No	Yes or No	hate
Yes or No	**Yes**	Yes or No	hate
Yes or No	Yes or No	**Yes**	hate
No	No	No	free

II. Content identification comes next after discourse analysis. Is offensive? Is it objectionable? or neither. Following labeling, the following Table 1 is used.

3.3. Text preprocessing

The process of eliminating stop words, pronouns, conjunctions, and unnecessary characters, as well as normalizing and tokenizing text is known as text preprocessing [8,9,39,40] .

Eliminating unnecessary characters: The document is edited to eliminate non-Amharic characters, Emojis, URLs, and punctuation [41].

Tokenization: is the process of dividing a document into distinct tokens.

Normalization: Because Amharic is a language, rich in morphologies, distinct character morphs but same sounds are standardized. For instance, ሀ, ሐ and ኀ are all normalized to ሀ and have the same sound (ha); similarly, ዐ (ae) becomes አ; ሠ (se) becomes ሰ and ፀ (tse) (tse) becomes ጸ [9,34].

Removing of irrelevant characters: Punctuation marks, Emoji's, URL's and @'s, non-Amharic characters are removed from the document [41].

Tokenization: it is break down a document into meaning full tokens.

Normalization: Amharic language is rich of morphs so different morphs of characters but same sounds are standardized. For example, ሀ, ሐ and ኀ have same pronunciation (ha) and they are normalized to ሀ; similarly, ዐ (ae) into አ; ሠ (se) into ሰ; into ፀ (tse) into ጸ [9,34].

Stop word removal: Prefixes including "ስለ" (sile: about), "የ'' (ye: the), and "በ'' (be: by) are examples of stop words that need to be removed. Suffixes like "ዎች'' (woch: plural form); verbs like "ነው" (new: is) and "ነበር" (neber: was); pronouns like "እኔ" (enie: my), "እነሱ" (esu: he), and conjunctions like "ስለዚህ" (slezih: so) and "ነገርግን" (negergn: but). The Amharic word's left pronunciation and its English meaning are shown by the associated term inside brackets (right side). Null values are eliminated from the document following preprocessing.

Ultimately, the dataset was divided into train and test groups, with 90 % of the dataset designated as train data and 10 % as test data. Subsequently, the train group was further divided into train and validation groups, with 80 % of the dataset designated as train data and 10 % as validation data, utilizing Stratified KFold [42].

3.4. Feature extraction build and evaluate models

Bag of Words (BoW), TFIDF, word2vec, Fasttext and BERT are some of feature extraction techniques. BoW represents text as a bag of words without considering word order or semantic relationships. TFIDF improves upon BoW by weighting words based on their importance within a document and across the entire corpus. Word Embeddings such as Word2Vec and FastText, handles semantic and syntactic relationships between words by learning dense vector representations. The advanced technique, BERT (Bidirectional Encoder Representations from Transformers) is a powerful language model that handles contextual information of words within a sentence and the entire document; however, it requires larger datasets. Because of factor such as data size and quality, we chose Bag of Words (BoW), TFIDF, word2vec and Fasttext feature extraction techniques.

Text (words) must be converted into vectors in order to be fed into deep learning algorithms; an embedding layer with an embedding value of 100 is used to do this. In addition to embedding layer fasttext is utilized with window size=10 and epoch 200. When creating a deep learning model, factors such as selecting the optimal activation functions and dropout values, calculating the number of neurons in each hidden layer, and determining the number of hidden layers overall are taken into consideration. After the model is generated, hyperparameters including optimizers, learning rates, loss functions, and accuracy measures are passed through to build the model. The assembled model is then fitted. Training and validation datasets, epochs, batch size, and callbacks for early stop are supplied as arguments when the model is being fitted.

Three layers are possible for the models. The input, hidden, and output layers are these.

Input layer: the input layer is the embedding layer that accepts integers, ids of tokens and outputs vector representation of tokens. The input layer forwards its output to the first hidden layer.

Output layer: this layer sums the outputs of the last hidden layer and reads the sum. The output layer for this paper is dense layer with units=1 and we chose activation='sigmoid' because the actual output of the model is either one or zero

Hidden layer: the main task in building model with deep neural network is estimating the constraints of hidden layer

We used the following procedures to determine the hidden layer(s).

I. **Choosing Algorithm:** We chose Bi-LSTM, Bi-GRU and Dense

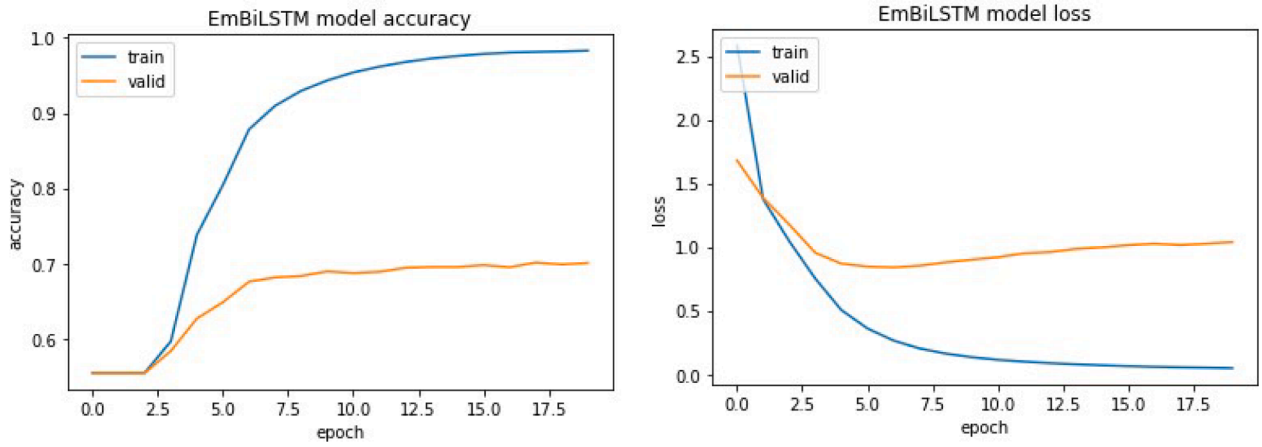

Fig. 3. BiLSTM model without the addition of dropout and regularizer.

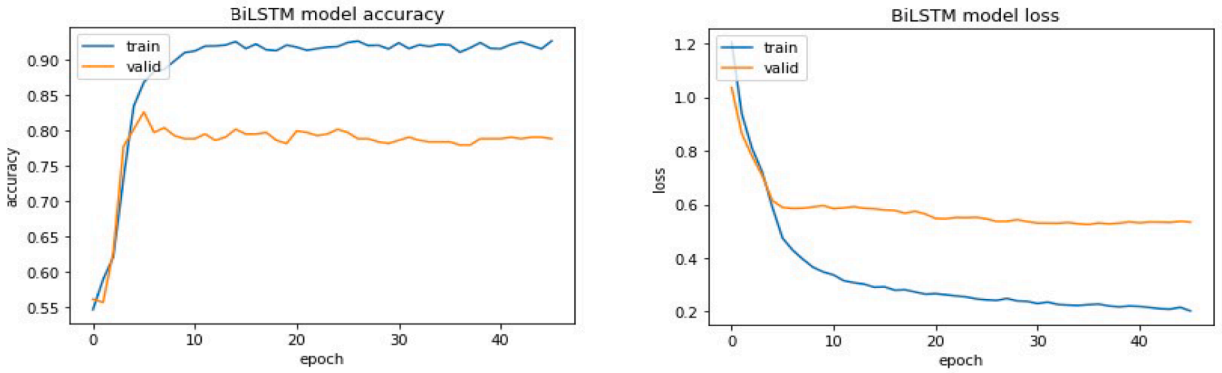

Fig. 4. BiLSTM with dropout the addition of and regularizer.

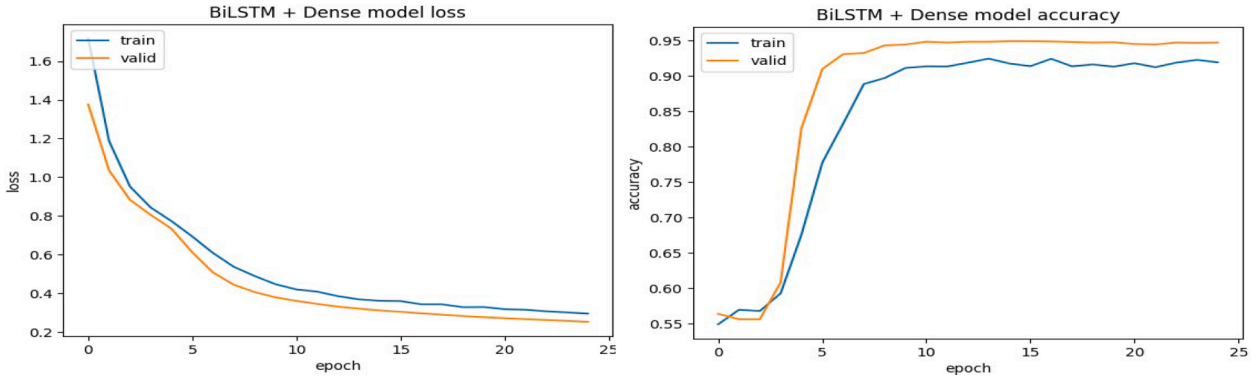

Fig. 5. BiLSTM +Dense model with the addition of dropout and regularizer (proposed).

II. **Estimating network complexity:** guess the network size of hidden layers, number of neurons per each hidden layer, and then adjust.
III. **Tuning hyperparameters:** tune activation functions, optimizers, batch sizes, learning rates and epochs.
IV. **Regularization:** adding dropouts and regularizers to fit the network.

The models that are constructed and compared in this study are BiLSTM, BiGRU, BiLSTM + BiGRU, BiLSTM + Dense, and BiGRU + Dense. The first and second hidden levels are distinguished by the + symbol. Lastly, because we have binary classes hate and free models are assessed using binary cross entropy and binary accuracy to assess loss and accuracy of models in each training session.

4. Experimental results

Much effort has gone into estimating the network complexity of models in order to produce accurate and well-fitting models. The creation of a fitted model was impossible without dense layers, regularizers, and dropouts. To address this issue: L2-regularizer, dropout, early drop and introduction of dense layer are applied. The L2-regularizer technique adds the sum of the squares of all the weights in the model, a penalty term to the loss function that helps to prevent overfitting. On the other hand, dropout is used to randomly drop out neurons in the network, forcing the model to learn more robust features. The model is trained on a validation set, and training is stopped when the validation accuracy starts to decrease. Since dropout, regularizer and early drop were not enough to address overfitting, a dense layer is introduced to mitigate overfitting. In the absence of regularizers and dropouts, the BiLSTM model started to overfit after the third epoch, as shown in Fig 3. The inclusion of regularizers and dropouts, as shown in Fig. 4, did, however, improve validation accuracy from 0.75 to 0.81 and reduce the difference between validation and train accuracies but the overfitting problem remains unresolved. Thus, the used dropout and regularizers solved overfitting problem. The validation accuracy of this model, BiLSTM + Dense, was 0.94 when the Dense layer was included as a second hidden layer in addition to dropouts and regularizers resolved during fitting difficulty, as shown in Fig 5. This also held true for the

Table 2
Accuracy result of models.

Model name	Word embedding	Hidden layer	Output layer	Accuracy
BiLSTM	Embedding layer	Bidirectional LSTM	Dense	0.75
BiGRU	Embedding layer	Bidirectional GRU	Dense	0.92
BiLSTM + Bi-GRU	Embedding layer	Bidirectional LSTM	Bidirectional GRU	0.90
BiLSTM + Dense	Embedding layer	Bidirectional LSTM and Dense	Dense	**0.94**
FsBiGRU + Dense	Embedding layer	Bidirectional GRU and Dense	Dense	0.80
FsBiLSTM	Fasttext	Bidirectional LSTM	Dense	0.75
FsBiGRU	Fasttext	Bidirectional GRU	Dense	0.74
FsBiLSTM + Bi-GRU	Fasttext	Bidirectional LSTM	Bidirectional GRU	0.74
FsBiLSTM + Dense	Fasttext	Bidirectional LSTM and Dense	Dense	0.73
Bidirectional GRU and Sense	Fasttext		Dense	0.73

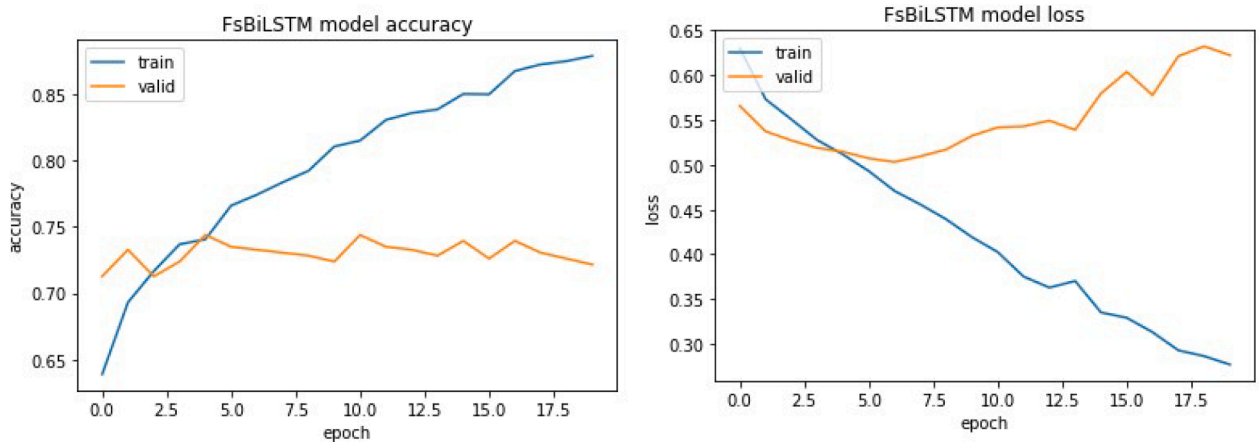

Fig. 6. FsBiLSTM without the addition of regularizers and dropouts.

Table 3
Machine learning algorithms evaluation.

Text vectorization		NB	DT	RF	KNN	LSVM	LR
BoW(bag of words)	Accuracy	0.68	0.68,	0.70	0.62	0.70,	0.73
TFIDF	Accuracy	0.79	0.62	0.73	0.62	0.72	74
Word2Vec	Accuracy	0.55	0.62	0.69	0.61	0.63	0.64
FastText	Accuracy	0.55	0.59	0.65	0.58	0.63	0.62

GRU + Dense model that Table 2 displays. The dropout values for the kernel and recurrent layers were 0.5 and 0.2, respectively, and for the dropout layers that were inserted after the first and second hidden layers, they were 0.5. The type of regularizers that were passed internally for all hidden layers in all models were the L2- regularizers with lambda values of 0.05.

Fig 6

BiLSTM performed better than FsBiLSTM(fasttext + BiLSTM) when we compared the BiLSTM and BiGRU models with FsBiLSTM and FsBiGRU, as shown in Table 2. BiGRU fared better than FsBiGRU(fasttext + BiGRU) in a similar way. So, the embedding layer performed better than the extraction of fasttext features.

We assessed machine learning algorithms in addition to deep learning models since it has been suggested that for small datasets, machine learning algorithms are more effective than deep learning algorithms. For the purpose of converting text into numbers, we used bag of words (BoW), TFID, word embedding (word2vec and fasttext), Naïve Bayes (NB), Decision Tree (DT), Random Forest (RF), Linear Support Vector Machine (LSVM), and Logistic Regression (LG).

TFIDF and the NB algorithm together produced an accuracy of 0.79, as shown in Table 3 and Fig 7. Word2Vec (0.69) and BoW (0.74) and FastText (0.65) had the best results when combined with LR, RF, and RF, in that order. However, when word embeddings (both word2vec and fasttext) did poorly with the NB approach, scored 0.55 accuracy. When combined with TFIDF, all machine-learning algorithms functioned optimally, whereas Word2Vec produced the lowest accuracy. When trained using the presented machine learning, bag of words and TFIDF represented texts more accurately than word embeddings as of [40] . TFIDF and Bag of words outperformed word embedding such as Word2Vec because high dimensionality of word representation, which increased the number of features.

The NB method combined with TFIDF performed better than the published machine learning algorithms, despite [40] having multiclass (60 classes) whereas the presented work had binary classes. Still, NB did not perform well when combined with word embeddings.

When trained with the described machine learning, utilizing bag of words and TFIDF performed more accurately than word embeddings (in this case, 5000 sample) as of [40].

4.3. Comparison with related works

Despite the fact that various writers utilized different authors have evaluated the various system performances in Table 5.

As shown in Table 5, this study achieved the highest accuracy score for text only 94 %) in the case of Ethiopian languages. Based on the confusion matrix in Table 4, the proposed model achieved precision and recall scores of 0.95 and 0.94, respectively.

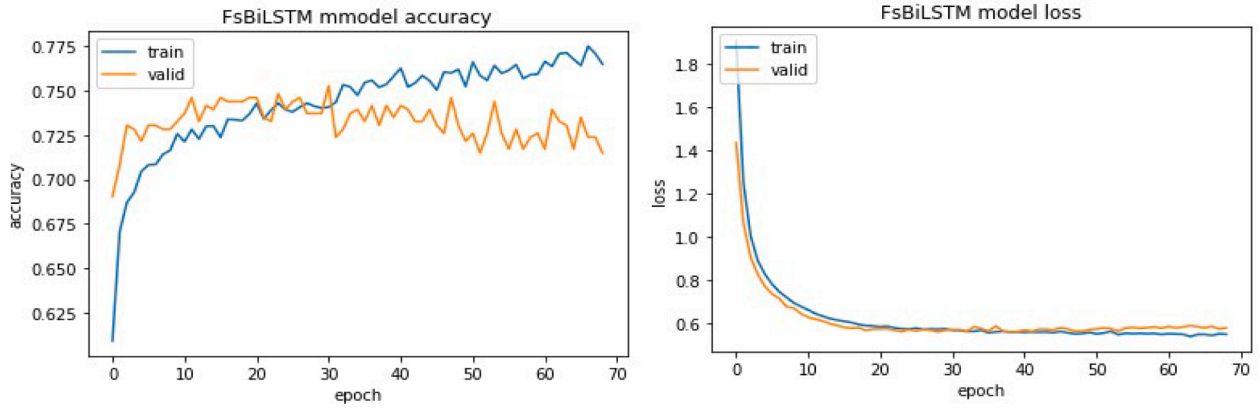

Fig. 7. FsBiLSTM with the addition of regularizers and dropouts.

Table 4
Confusion matrix of the proposed model.

predicted			
actual		hate	free
	hate	517	33
	free	27	433

Table 5
Comparison of the proposed approach with related works.

Author	Method	Algorithm	Accuracy
Ayichlie Jigar [13]	Unimodal (text alone)	BiLSTM	0.63
Ayichlie Jigar [13]	Unimodal (image alone)	CNN	0.69
Ayichlie Jigar [13]	Multimodal(picture and text)	BiLSTM	0.75
Debele et al. [12].	Multimodal(audio and text)	BILSTM	0.88
Proposed	Unimodal(text a lone)	BiLSTM + Dense + Dense	0.94

5. Conclusion and recommendation

5.1. Conclusion

This research is done to determine whether a certain text-image post is free or hate speech, since hate speech is a major problem on social media platforms like Facebook. In order to accomplish this, we first gather memes from Facebook. Following the analysis of the gathered memes, we used the bag of words, TFIDF, embedding layer and fasttext to extract the features and the machine learning algorithms NB, DT, RF, KNN, LSVM and LR, and BiLSTM, BiGRU, and Dense algorithms to create deep learning models.

Traditional text to number transformers (TFIDF and BoW) achieved more than word embedding. The TFIDF achieved better than BoW. As a result, TFIDF + NB other the presented machine learnings whereas word embedding + NB performed poorly.

The embedding layer represented texts better than and fasttext approaches. In addition, the fasttext representations trained slowly than the Embedding layer as depicted in the Figs. 5 and 8.

Fig 9

Overfitting was a major problem while training models for the BiLSTM and BiGRU algorithms; these models did not fit even after regularizers and dropouts were added. On the other hand, these models fit when a dense layer is introduced as a second hidden layer. The introduction of Dense improves not only overfitting but also training time. To sum up, a two-layered deep learning model called BiLSTM + Dense is suggested, combining the best features of both models to identify text-image postings on Facebook are hate or free.

5.2. Recommendation

The performance of the suggested model can be enhanced by a large dataset. Therefore, by expanding the dataset and using multimodal analysis, this work can be improved even more. Additionally, the effort addresses Facebook's hate speech identification feature. Hate speech is, nevertheless, also spreading through other social media sites and Telegram groups. Thus, future studies might concentrate on identifying hate speech in Telegram chats.

Funding

Authors declare no funding for this research.

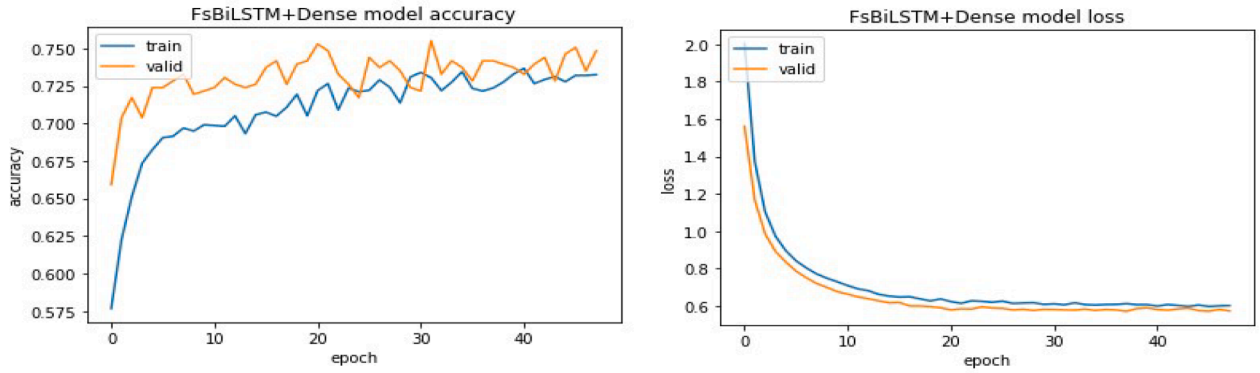

Fig. 8. FsBiLSTM +Dense with the addition of regularizers and dropouts.

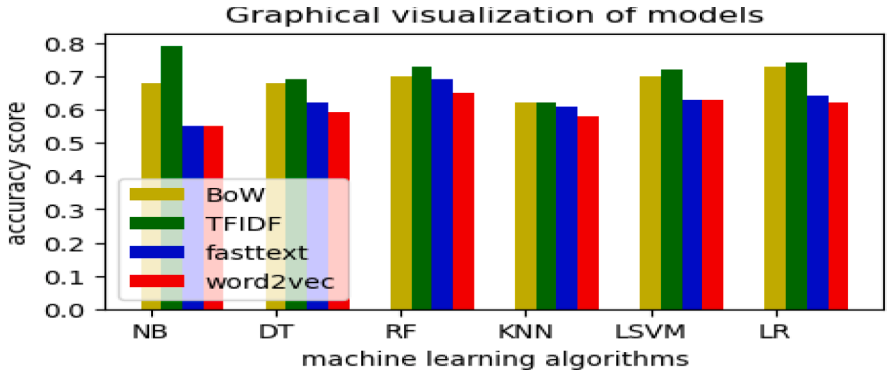

Fig. 9. Machine learning algorithms evaluation.

CRediT authorship contribution statement

Mequanent Degu Belete: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Girma Kassa Alitasb:** Writing – review & editing, Writing – original draft, Visualization, Supervision, Resources, Project administration, Methodology, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data sharing is applicable to this article as datasets were used during the current study [37]. 10.17632/gw3fdtw5v7.2

References

[1] "Digital Around The World Advertisement." [Online]. Available: https://datareportal.com/global-digital-overview.

[2] "Uprooting Hate Speech: The Challenging Task Of Content Moderation In Ethiopia." [Online]. Available: https://openinternet.global/news/uprooting-hate-speech-challenging-task-content-moderation-ethiopia.

[3] "Addis Ababa University College Of Law And Governance Studies School Of Law Ll, M program Master of laws (LL, M) in Human right law adequacy of the prevailing regulatory framework relating to hate speech on social Media in Ethiopia RahwaWeldeghebriel Supervisor:. MesenbetAssefa (PhD) a thesis submitted in partial fulfillment of the requirements for the degree of master of law (llm) in human rights," 2020.

[4] "SOCIAL MEDIA, Mass atrocities, and atrocity prevention 2023 sudikoff interdisciplinary seminar on genocide prevention".

[5] "DIGITAL 2024: Ethiopia 23 Fe Bruary 2024 · Simon Ke Mp." [Online]. Available: https://datareportal.com/reports/digital-2024-ethiopia?rq=facebook.

[6] S. Benesch, "But Facebook's not a country: how to interpret Human rights law for social Media companies." [Online]. Available: https://perma.cc/YW6C-DKJ2].

[7] "ሥጋት ያጫረው የጥላቻ ንግግርና ሐሰተኛ መረጃ ሕግ ጸደቀ የኢትዮጵያ የሕዝብ ተወካዮች ምክር ቤት የጥላቻ ንግግርን እና ሐሰተኛ መረጃን ለመቆጣጠር የወጣውንና የኤክሳይስ ታክስ ረቂቅ አዋጆችን ዛሬ ባደረገው ልዩ ስብሰባው አጽድቋል። የጥላቻ ንግግርን እና ሐሰተኛ መረጃን ለመቆጣጠር የረቀቀው ሕግ ከጠቅላይ ሚኒስትር ዐብይ አህመድ ወደስልጣን ከመጡ በኋላ ኢትዮጵያ ውስጥ ከወትሮው የተሻለ ይዞታ ላይ ይገኛል የሚባልለትን ሐሳብን የመግለጽ ነፃነት ላይ አደጋን ይጋርጣል በሚል ስጋታቸውን የሚሰነዝሩ የመኖራቸውን ያህል፤ ሕጉ ምን ያህል ሊተገበር ይችላል በሚለው ጉዳይ ላይ ጥርጣሬ እንዳላቸው የሚገልፁም አልጠፉም። house of peoples representative." [Online]. Available: https://www.bbc.com/amharic/51485031.

[8] Z. Mossie, J.H. Wang, Social Network Hate Speech Detection for Amharic Language, Academy and Industry Research Collaboration Center (AIRCC), 2018, pp. 41–55, https://doi.org/10.5121/csit.2018.80604. Apr.

[9] Y.Kenenisa Defar, "Hate speech detection for Amharic language on social Media using machine learning techniques," 2019.

[10] S.G. Tesfaye and K.Kekeba Tune, "Tesfaye and tune automated Amharic hate speech posts and comments detection model using recurrent neural network".

[11] "Amharic stance classification using deep learning Girma, Bewketu Molla," 2021. [Online]. Available: http://dspace.orghttp://ir.bdu.edu.et/handle/123456789/13206.

[12] A.G. Debele, M.M. Woldeyohannis, Multimodal amharic hate speech detection using Deep learning, in: 2022 International Conference on Information and Communication Technology for Development for Africa (ICT4DA), IEEE, 2022, pp. 102–107, https://doi.org/10.1109/ICT4DA56482.2022.9971436. Nov.

[13] M. Ayichlie Jigar, A.A. Ayele, S.M. Yimam, and C. Biemann, "Detecting hate speech in Amharic using multimodal analysis of social Media memes." [Online]. Available: https://t.me/hateSpeech_image_data_c.

[14] M. Degu, A. Tesfahun, H. Takele, Amharic language hate speech detection system from Facebook memes using deep learning system, SSRN Electr. J. (2023), https://doi.org/10.2139/ssrn.4389914.

[15] Z. Al-Makhadmeh, A. Tolba, Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach, Comput. 102 (2) (Feb. 2020) 501–522, https://doi.org/10.1007/s00607-019-00745-0.

[16] Z. Zhang, L. Luo, Hate speech detection: a solved problem? Challeng. Case Long Tail Twitter (2018). Feb. [Online]. Available, http://arxiv.org/abs/1803.03662.

[17] M.D. Belete, L.G. Shiferaw, G.K. Alitasb, T.S. Tamir, Enhancing word sense disambiguation for Amharic homophone words using bidirectional long short-Term Memory network, Intellig. Syst. Applic. 23 (2024), https://doi.org/10.1016/j.iswa.2024.200417. Sep.

[18] J. Wang, C. Zhang, Software reliability prediction using a deep learning model based on the RNN encoder–decoder, Reliab. Eng. Syst. Saf. 170 (2018) 73–82, https://doi.org/10.1016/j.ress.2017.10.019. Feb.

[19] Gradient flow in recurrent nets: the difficulty of learning LongTerm dependencies, A Field Guide to Dynamical Recurrent Networks, IEEE, 2009, https://doi.org/10.1109/9780470544037.ch14.

[20] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional LSTM networks for improved phoneme classification and recognition," 2005, pp. 799–804. doi: 10.1007/11550907_126.

[21] Z. Qin, S. Yang, Y. Zhong, Hierarchically gated recurrent neural network for sequence modeling, in: A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), Advances in Neural Information Processing Systems, Curran Associates, Inc., 2023, pp. 33202–33221 [Online]. Available, https://proceedings.neurips.cc/paper_files/paper/2023/file/694be3548697e9cc8999d45e8d16fe1e-Paper-Conference.pdf.

[22] W.A. Qader, M.M. Ameen, B.I. Ahmed, An overview of Bag of words;importance, implementation, applications, and challenges, in: Proceedings of the 5th International Engineering Conference, IEC 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 200–204, https://doi.org/10.1109/IEC47844.2019.8950616. Jun.

[23] H. Liang, X. Sun, Y. Sun, Y. Gao, Text feature extraction based on deep learning: a review, EURASIP. J. Wirel. Commun. Netw. 2017 (1) (2017) 211, https://doi.org/10.1186/s13638-017-0993-1. Dec.

[24] R. Dzisevic, D. Sesok, Text classification using different feature extraction approaches, in: 2019 Open Conference of Electrical, Electronic and Information Sciences (eStream), IEEE, Apr. 2019, pp. 1–4, https://doi.org/10.1109/eStream.2019.8732167.

[25] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," Jan. 2013, [Online]. Available: http://arxiv.org/abs/1301.3781.

[26] J. Pennington, R. Socher, and C.D. Manning, "GloVe: global vectors for word representation." [Online]. Available: http://nlp.

[27] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information." [Online]. Available: http://www.isthe.com/chongo/tech/comp/fnv.

[28] J. Devlin, M.W. Chang, K. Lee, K.T. Google, and A.I. Language, "BERT: pre-training of deep bidirectional transformers for language understanding." [Online]. Available: https://github.com/tensorflow/tensor2tensor.

[29] A. Schmidt and M. Wiegand, "A Survey on Hate Speech Detection using Natural Language Processing," Association for Computational Linguistics. [Online]. Available: https://en.wikipedia.org/wiki/List_.

[30] S. Madisetty and M.S. Desarkar, "Aggression detection in social media using deep neural networks," 2018. [Online]. Available: https://github.com/JohnLangford/vowpal.

[31] B. Singh, N. Upadhyay, S. Verma, and S. Bhandari, "Classification of hateful memes using multimodal models," 2022, pp. 181–192. doi: 10.1007/978-981-16-6460-1_13.

[32] A. Das, J.S. Wahi, and S. Li, "Detecting hate speech in multi-modal memes," Dec. 2020, [Online]. Available: http://arxiv.org/abs/2012.14891.

[33] K. Perifanos, D. Goutsos, Multimodal hate speech detection in Greek social Media, Multimodal. Technol. Interact. 5 (7) (2021) 34, https://doi.org/10.3390/mti5070034. Jun.

[34] H.W. Kululo, "Information filtering of social media Amharic texts based on sentiment analysis.".

[35] E.N. Hailemichael, "Fake news detection for amharic language using deep Learning Adama, Ethiopia," 2021.

[36] W.B. Demilie, A.O. Salau, Detection of fake news and hate speech for ethiopian languages: a systematic review of the approaches, J. Big. Data 9 (1) (2022) 66, https://doi.org/10.1186/s40537-022-00619-x. Dec.

[37] L. Flight, S.A. Julious, The disagreeable behaviour of the kappa statistic, Pharm. Stat. 14 (1) (2015) 74–78, https://doi.org/10.1002/pst.1659. Jan.

[38] Degu Mequanent, Amharic text dataset extracted from memes for hate speech detection or classification, Mendeley Data (2023).

[39] B.K. Poornima, D. Deenadayalan, and A. Kangaiammal, "Text preprocessing on extracted Text from audio/video using R," 2017.

[40] M.D. Belete, A.O. Salau, G.K. Alitasb, T. Bezabh, Contextual word disambiguates of Ge'ez language with homophonic using machine learning, Ampersand 12 (Jun. 2024) 100169, https://doi.org/10.1016/j.amper.2024.100169.

[41] S. Hirpassa, Information extraction system for amharic text, Int. J. Comput. Sci. Trends Technol. (IJCST) 5 (2013) [Online]. Available, www.ijcstjournal.org.

[42] M. TR, V. K.V, D. K.V, O. Geman, M. Margala, M. Guduri, The stratified K-folds cross-validation and class-balancing methods with high-performance ensemble classifiers for breast cancer classification, Healthcare Analyt. 4 (2023) 100247, https://doi.org/10.1016/j.health.2023.100247. Dec.