Ethiopian Journal of Science and Sustainable Development

e-ISSN 2663-3205 *Volume 11 (1), 2024*

Journal Home Page: www.ejssd.astu.edu.et

Research Paper

Detection of Emotions in Afan Oromo Social Media Texts Using Deep Learning Method

Sufian Kedir Abdella, Worku Jifara Sori*

Department of Computer Science and Engineering, Adama Science and Technology University, P.O. Box 1888, Adama, Ethiopia

Article Info

Article History:
Received 04 September 2023
Received in revised form 22January 2024
Accepted 25 January 2024

Keywords:
Afan Oromo,
deep learning,
emotion detection,
hybrid models,
social-media

Abstract

Emotion analysis in foreign languages is common because of its numerous useful applications in commercial activities and decision-making. However, there was a lack of emotion-detection work for Afan Oromo language. Manually identifying and aggregating millions of social media users' emotions into a swift and effective decision-making process is challenging task. Thus, the main objective of this study was to detect emotion in the Afan Oromo social media texts. To achieve this, state-of-the-art deep learning models, namely convolutional neural networks (CNN), long short-term memory (LSTM), bidirectional long short-term memory (BiLSTM), and hybrid of them (CNN-LSTM and CNN-BiLSTM), were designed and investigated to select the best-suited model for emotion detection in Afan Oromo. Data was collected from official Facebook pages, then manually annotated, and preprocessed by using normalization, tokenization and stop-word removal. Word embedding was used for feature encoding, and Keras Python libraries were employed for implementation. The study's results revealed that the proposed models performed well with accuracies of 92, 87, 88, 88 and 90 % for CNN, LSTM, BiLSTM, CNN-LSTM, and CNN-BiLSTM, respectively. Thus, the CNN model outperformed all the other models. It was also found out that the CNN model suited and gave a better result when it was worked on a small dataset and a short sequence of texts. The accuracy of the comparatively less performing models, particularly the performance of the hybrid models, can be increased through the construction of sufficient data, because they leverage the benefits of each of them.

1. Introduction

Recently, social media platforms have become the main medium for people to express their daily activities, reactions and emotions on a wide range of topics that affect their everyday life (Megersa, 2020). People´s emotions have become imperative for making effective decisions, for individuals, government and commercial sectors (AlBalooshi et al., 2019). Emotion is a strong feeling about human's situation or relation with others. Emotion recognition is the task of identifying emotions that are expressed in the text, speech or facial. Recognition of emotion in the text is a technique for detecting emotional text that includes cues expressed by the writer and perceived by the readers (Al-Omari et al., 2020). Detection of emotions expressed in the texts are very useful for expressing human concerns in applications such as customer services, mental health analysis, marketing, workplace, etc. (Acheampong et al., 2020).

Natural language processing (NLP) is used in the area of machine learning that focuses on the generation and understanding of language. Its main objective is to enable machines to understand, communicate and

*Corresponding author, e-mail: worku.jifara@gmail.com

https://doi.org/10.20372/ejssdastu:v11.i1.2024.741

interact with humans in a natural way. Nowadays, people freely express their emotions, opinion, etc. in the form of text on the variety of social media platforms by using their own languages. Afan Oromo is one of the widely spoken languages in Africa, especially in Ethiopia, and over 35 million of people use it for social media and other purposes (Wakshum, 2000). Limited work has been done for Afan Oromo language; however, the amount of Afan Oromo texts available in the internet is growing steadily from time to time (Wegderes et al., 2022).

It is a difficult task for individuals, governments, and commercial sectors to manually understand and recognize a million social media users' emotions and aggregate them towards quick and effective decision-making about their services, products, etc. The sentiment of a post in a social media can be seen in the tone or emotion conveyed in a text. Analyzing text sentiment and community responses can provide insight into people's feelings towards political, social, and religious situations in a country. It is a challenging task to analyze emotions present in a text because of factors such as the complexity of human emotions expressed in the text and the use of implicit and metaphorical language in expressing those (Seyeditabari et al., 2018). Moreover, resources such as training data for emotion analysis have a significant impact on the overall model performance. Afan Oromo, one of the least researched, under resourced and morphologically rich languages, shares some of these challenges (Obsa & Kula, 2023).

A number of techniques were proposed to identify emotions in a text, including keyword-based, lexical affinity-based, learning-based, and hybrid models. According to Sailunaz et al. (2018), the state of the art came into existence; the learning-based approach was more accurate and gave better results. Deep learning is a learning-based method that has emerged as a powerful machine learning technique that detects multiple layers of representations or features of the data and produces advanced prediction results (Bharti et al., 2022). It is an effective method due to the automatic learning capability and generalizability and thus it has found success across a wide range of application domains, including emotion analysis. It is a promising approach and has been extensively applied to learning, semantic parsing, natural language processing, and more (Sailunaz et al., 2018).

Thus, in this study, deep learning method is used with the goal to design and explore the best-suited and most efficient model for detecting emotion in Afan Oromo texts. The study was limited to working with textual data that was collected from different Facebook pages only and it did not encompass other types of documents, such as audio and video as well as other social media, such as Twitter and YouTube. Based on different social media statistics, Facebook tops other social media platforms in terms of the number of active users and textual content (comments and posts) in Ethiopia (Statcounter, 2023). Although emotion detection was mostly researched for resourced languages, such as English, the methods used for English and other languages can’t be directly applied to other languages because of its dissimilarity in structure.

The main motivation of this work came from the popularity of emotion analysis in foreign language due to its various useful applications in commercials and effective decision-making. However, there is lack of emotion detection works for Afan Oromo. Most of the previous studies on Afan Oromo (Megersa, 2020; Ashebir et al., 2021;Wegderes et al., 2022; Obsa & Kula, 2023), and others focused on determining the polarity of the opinion (positive, negative, or neutral sentiment) rather than focusing on the exact feelings of the public or the intensity of their reaction. Emotional detection measures whether the public reacts to different emotional states, such as happiness, sadness, disgust, etc. Thus, developing model to detect emotion in the Afan Oromo social media texts is required. Using the deep learning method, in this study, emotions in tasteless text sentences that lacked any tone or expression were detected.

The findings of the study has many significance if applied as state-of-the-art deep learning models, including hybrid ones, are designed for detecting emotions in the Afan Oromo texts. The performance of the models are evaluated to recognize emotion in the texts with limited dataset, but they gave good results. Kabada and Getachew (2020) conducted research on sentence-level emotion analysis for Afaan Oromo; however, they did not consider hybrid deep learning model. Thus, original Afan Oromo emotional dataset that contained 2020 emotional sentences, with six emotion categories were developed, so as to create a general understanding of the subject matter. Moreover, this work allows a direct comparison of the contemporary state-of-the-art text classification techniques in the context of the Afan Oromo. Besides, removal of the apostrophe "' " which is known as "Hudha" was ignored and helped with word formation in Afan Oromo between either different or similar vowels (e.g., '*ka'ii bahii'* which means stand up and go out, '*baay'een gammade*' for 'I am very happy', etc.) during data pre-processing, and it is one of the special Afan Oromo features that was missed by researchers in the previous works.

2. Materials and Methods

2.1. Data Sources and Their Description

Afan Oromo language has no sufficient standard datasets for natural language processing (NLP) tasks, including emotion analysis, so far. It is one of the least researched and under-resources languages (Workineh & Duresa, 2017). Hence, textural emotional posts and comments between December 2021 and 2022 on Afan Oromo official facebook pages of Oromia Broadcasting Network (OBN), Oromia Media Network (OMN), British Broadcasting Corporation (BBC), and Fana Broadcasting Corporation (FBC) using the Face Pager software tool were collected for the experiments. The data were collected randomly, with six emotion categories from the identified public pages. Table 1 shows the distribution of the data used in the study corresponding to the emotion category. The prepared data encompassed the Afan Oromo short text sequence, and it also contained links, numbers, and punctuation marks. Then all unnecessary characters, links, and different punctuation, except apostrophe and hyphen, were cleaned, and they were made of only text sentences with their corresponding emotions.

This data included over 2020 emotional text sentences, and each sentence was labeled with six categories; namely, hapiness ('gammachuu'), love ('jaalala'), anger ('aarii'), disgust ('jibba'), fear ('sodaa'), and sadness ('gadda'), according to its syntactic and semantic polarities. The number of data corresponding to OBN, OMN, BBC and FBC were 525, 489, 512 and 494, respectively. All data were combined into a single file with a comma-delimited value (CSV).

Table 1: Dataset size with respect to emotion type

Emotion type	№ of data
Happiness ('Gammachuu')	315
Love ('Jaalala')	470
Anger ('Aarii')	312
Disgust ('Jibba')	312
Fear ('Sodaa')	309
Sadness ('Gadda')	302
Total	2020

The study dataset was not large because of the associated challenges such as the complexity of human emotions, language features (spelling correction issues), resources (time and budget), and difficulties in the data preparation activities like data annotation due to the dataset being multiclass. However, the data used for this study meaningfully contributed in terms of dataset size, data preprocessing, and emotion classes compared to the earlier related research in Afan Oromo sentiment, especially emotion detection by deep learning methods.

The data used for the study was annotated manually by experts, depending on the annotation procedures set. Data annotators were limited to three because of a resource shortage for each master degree holder. The selection of experts was based on their interests, Afan Oromo language knowledge, and computer skills. At the time of the annotation, when there was any disagreement between the annotators, the major vote was selected; otherwise, it was ignored from the data list.

2.2. Data Preprocessing

Data preprocessing increases algorithm efficiency while reducing the computational load (Motwani et al., 2023). The collected data needed preprocessing because it contained extraneous text, symbols, and unnecessary characters. Various preprocessing techniques, including cleaning, spelling correction and normalizations, tokenization, stop-word removal, case conversation (upper case to lower case), punctuations mark removal with the exception of an apostrophe **" ' "** and hyphen **"-"** that helped for word formation in Afan Oromo, were applied throughout the data preprocessing. For the preprocessing activity, Natural Language Toolkit (NLTK) was employed.

Cleaning involved removing the extraneous text contents which were not significant to represent the study dataset, including removal of numbers, links, none Afan Oromo texts, emails, html tags, Uniform Resource Locator (URL) special symbol and punctuation marks with the exception of the apostrophe "Hudha"**,** and hyphen "-" that helped in the Afan Oromo word formation.

The second preprocessing was about spelling correction and normalization. Several users write Afan Oromo words with the correct spelling, while others type Afan Oromo words incorrectly. Incorrect spelling of the Afan Oromo word may alter the meaning of a sentence, paragraph, or entire document. To handle the issue of misspelled words in the language, words with misspelling were identified and replaced with correctly spelled words by writing using Python scripting. In addition, uppercase, lowercase, or a mix of case formats are used to write words. To consider this, case normalization (lowercase conversion) was performed to have the same case and correct spell words in the entire dataset.

Tokenization is the process of splitting a text into individual tokens or words. In this study, the emotional sentences were put into tokens to deal with the structure of the text in the dataset. Splitting a text document into discrete token sequences was done without considering the meaning or relationship of the tokens. There are a variety of methods of tokenization, including white space, semicolons, commas, quotations, and periods, that are used to separate words or sentences in the text document. In the Afan Oromo language, a whitespace character was used to separate words from each other in a text document (Gezehagn, 2012).

The other preprocessing activity involved stop-word removal. Stop words are insignificant words in a language that generate noise when utilized as text classification features and increase processing time. In this study, Afan Oromo stop words were removed from the text sentences in the dataset in order to prepare the terms that accurately represent the document. For instance, stop words like I '*ani', outside 'alaa', now 'amma', after 'booda', she 'ishee'*, are some of Afan Oromo stop words which were removed from the sentences in the dataset.

After cleaned and preprocessed, the dataset was categorized into training and testing using the scikit-learn train-test split library. In this study, a 90:10 splitting ratio, which means 90% of the total dataset was used for training and 10% for the test.

2.3. Feature Extraction

This is the technique of mapping textual data into real-valued vectors and the dimension reduction technique that reduces a large amount of raw data into smaller categories to facilitate processing. In the text classification, word embedding is a technique that represent words or sentences in a text with real-number vectors, as described in a vector model (Singh et al., 2022). In the study, word vector representation was used as a text feature that works with a random initialization for the word vectors, and it was set in such a way that it was trainable and updated during the model training phase. In addition, in the CNN, the convolutional layer played a key role as a feature extractor through the filters, such as semantic and context features, from the sequence of words.

2.4. Deep Learning Model for Afan Oromo Emotion Detection

Machine learning models are an artificial intelligence (AI) technique that aid a machine's ability to learn and develop without being explicitly programmed (Navarrete et al., 2021). Deep learning method is one of the types of machine learning that has shown significant improvements in various natural language processing (NLP) tasks, including emotion detection in the text. Thus, convolutional neural networks (CNN), long short-term memory (LSTM), bidirectional long short-term memory (BiLSTM), and a hybrid of them, CNN-LSTM and CNN-BiLSTM, were designed and studied to recognize emotions in the Afan Oromo texts. The preprocessed training dataset was given as input into the randomly initialized embedding layer and passed to deep learning models to train models and predict the emotions in the test dataset. There was no a single prior study that used CNN, CNN-LSTM, and CNN-BiLSTM models in the Afan Oromo text emotion analysis domain.

2.4.1 Convolutional Neural Network (CNN)

CNN has been widely used for emotion classification tasks due to its ability to capture spatial features in data. The CNN architecture proposed in this study (Figure 1) was induced by the model presented by Kim (2014). Feature extraction was done in the model for the detection of emotion in the texts. The model takes the sequence of texts as input, and the processed input is passed through a set of layers. The set of layers extracts the features from the sequence of words. The set of words from the text is converted into high-dimensional vectors using the randomly initialized word embedding; it is learned and updated during training the model. In CNN model, filters (kernels) also play a great role as feature extractors, such as semantic and contextual features from the sequence of words (Megersa, 2020). The CNN model used was a single layer and it had a 1D convolution layer with a kernel size of three, 100 number of filters and ReLU nonlinear activation function used at the hidden layer to introduce non-linearity.

A max-pooling layer was followed to pool the features and to select the maximum value within a pool window, and a fully connected layer with a softmax classifier to predict the emotions in the dataset. The features were extracted by convoluting the filters over the sequence of words. The convolution layer extracted the local and contextual features from the vector representation of the words and fed them to the fully connected layer via the max-pooling layer to perform classification. Dropout regularization was also used to prevent co-adoption and overfitting. The model was trained and tested in this manner, and its parameters and coefficients were tuned through trial and error to achieve the best performance result. Thus, the CNN architectures for emotion classification typically consisted of convolutional layers followed by pooling layers, fully connected layers, and an output layer.

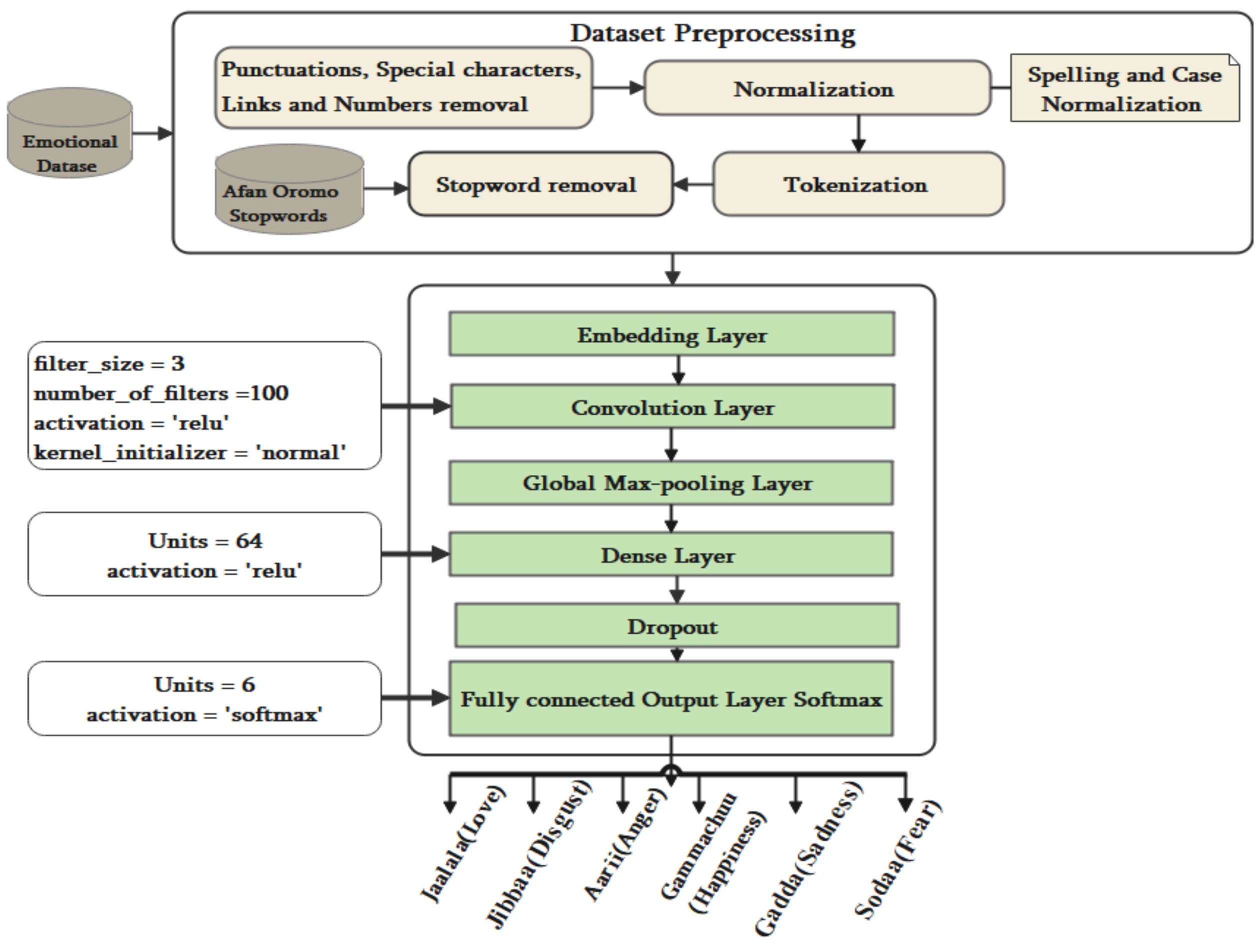


Figure 1: Architecture of the CNN model to detect emotions in the Afan Oromo Text

The 1D Convolution Layer applies filters (kernels) to the input data, extracting relevant features. These filters learn to detect features by convoluting the kernels over the sequence of words. The number of filters and their sizes determine the number of feature maps produced. Nonlinear function is commonly used to introduce non-linearity (Ghourabi et al., 2020). The Rectified Linear Unit (ReLU) was used to introduce nonlinearity in this study, as it increases the accuracy of the results as well as speeds up the training (Nair & Hinton, 2010). ReLU is defined as**:**

$$f(x) = max\ (0, x) \tag{1}$$

where f as nonlinear function returns x if the value is positive, elsewhere returns zero.

On the other hand, max pooling is a commonly used technique to select the maximum value within a pool window and to help reduce computational complexity, making the model more robust. The extracted feature maps by the convolution layer were passed to the Max Pooling layer as input. These features are characterized by a high-level vector representation. To reduce this representation without losing the most important features, a Max Pooling layer was applied to the result of the convolution layer and to pick out the more informative features.

To prevent overfitting, dropout regularization technique was used. It randomly sets a fraction of the output features of a layer to zero during training. This encourages the network to learn more robust features by reducing interdependencies between neurons. Then, fully connected (dense layers) take the flattened output from the convolutional layers and transform it into a vector representation. These layers help in learning higher-level representations by combining lower-level features. Activation functions like ReLU or sigmoid are used. A fully connected layer with a ReLU activation function that takes the maximum features from the pooling layer as input, performs computations, and feeds its result to the output layer that works with Softmax to perform classification was used in the study. As a loss function, categorical cross-entropy was used because of its ability to deal with multiclass classification.

2.4.2 Long-Short Term Memory (LSTM)

LSTM was designed to overcome the recurrent neural network (RNN) problem of gradient vanishing and exploding. In LSTM, the RNN hidden vectors are replaced by memory blocks equipped with gates. The LSTM gate mechanism implements three layers: (1) input gate, (2) forget gate, and (3) output gate (Kumar & Rastogi, 2019). Each LSTM unit, has a memory cell, and the states at time t are represented as c_t. Reading and modifying are controlled by the sigmoid gate and it affects the input gate i_t, forget gate f_t, and output gate o_t. In LSTM, the model receives input from two external sources (h_{t-1} and x_t). The x_t input vector, network received at time t, and the previous hidden state h_{t-1} determine the hidden state h_t. When calculating the hidden layer, node states, input gate, output gate, forget gate, and x_t will simultaneously affect the state of the node. The transition functions that exist between the LSTM units are given below (Zhou et al., 2015):

Input Gate determines how much new information is stored in the memory cell at each time step. It is computed by equations (2) and (3).

$$i_t = \sigma(W_i.\ [h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{C}_t = \tanh\ (W_C.\ [h_{t-1}, x_t] + b_C \tag{3}$$

The candidate vector $\tilde{C}_t$ is created by neural network hyperbolic tangent (*tanh*) and is added to the internal state. Now the old cell state C_{t-1} is updated into new cell state C_t (memory state) via the following rule:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{4}$$

Where $*$ denotes the elementwise multiplication

Forget Gate controls information that is discarded from the memory cell. Works as equation (5).

$$f_t = \sigma(W_f.\ [h_{t-1}, x_t] + b_f) \tag{5}$$

Output Gate determines how much information from the memory cell is exposed to the next hidden state and the output. Equation (6) and (7) indicate how it works.

$$o_t = \sigma\ (W_o\ [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = o_t * \tanh\ (C_t) \tag{7}$$

The LSTM model used in this study consisted of a single layer with 100 memory units, followed by a dense layer with a ReLU activation function. The word embedding facilitates learning word representation. After text preprocessing, pad-sequence was performed by filling in zeros to make the texts in a dataset have the same length, which is equivalent to the maximum length of the text in the dataset. The maximum length of the text in the present study dataset was 200, so the text less than this was filled with zero, made into matrices of the same length, and then fed as input to the LSTM layer. Then LSTM performed all necessary computations, and the output was fed into the fully connected layer. With this strategy, the model learns the required feature representation by itself. Like the CNN model, the softmax function was used in the output layer since the work is multiclass classification, and dropout regularization was utilized to prevent overfitting. The structure of LSTM model is summarized in Figure 2.

2.4.3 Bidirectional Long-Short Term Memory (Bi-LSTM)

In a Bi-LSTM, the input sequence is processed both in the forward (past to future) and in the reverse (future to past) direction. The outputs from both directions are concatenated in some way to provide a comprehensive representation of the input sequence. This allows much richer sequential patterns to be captured and helps in learning a much relevant feature representation for the input text sequence. The architecture of the Bi-LSTM model used in the study is depicted in Figure 3. By considering both past and future contexts, it can capture dependencies and patterns in a sequence more effectively, which lead to improved performance.

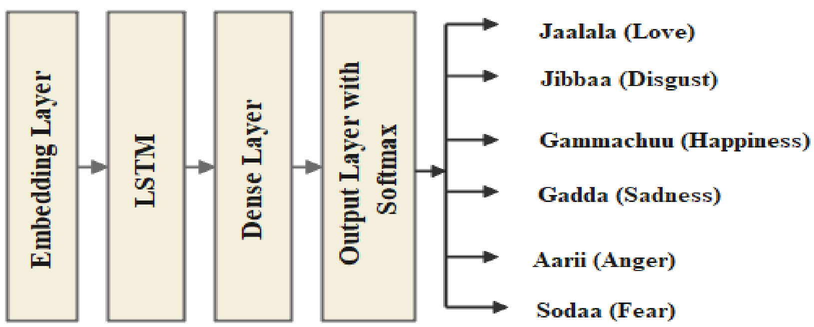


Figure 2: LSTM model to detect emotions from text

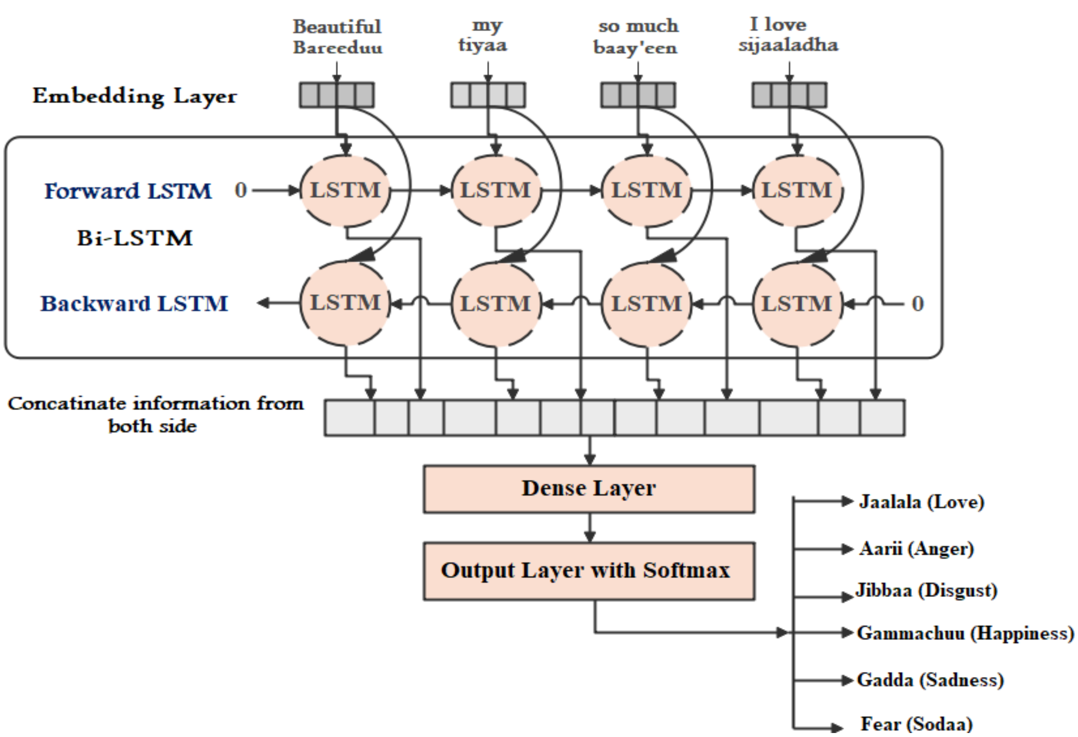


Figure 3: Bi-LSTM model architecture for detecting emotions in Afan Oromo text

2.4.4 Combined CNN-LSTM

The reason to hybrid CNN and LSTM models was to leverage the strengths of both models to capture both local and global dependencies in the text data. The CNN layers extract local features and n-gram combinations, effectively capturing important information within short spans of text; whereas, the LSTM layers capture the contextual information and long-range dependencies between words, allowing the model to understand the overall semantics and meaning of the text.

The CNN and LSTM were combined in sequential ways. The inputs to the model are a sequence of words representing a text document. Each word is typically encoded as a fixed-length vector, such as in word embedding. The CNN model used has one convolutional layer and applies filters to the input text sequence, sliding them across the sequence to extract local features. The outputs are a set of feature maps that encode the local features. These feature maps are reshaped to convert into a 1D sequential format via max pooling. The reshaped features are then fed into single LSTM layers. The LSTM layers process the sequential features. The output of the LSTM layers can be further processed by additional layers, such as fully connected layers, to generate the final predictions. These layers map the learned representations from the LSTM to the target classes or labels of emotion.

2.4.5 Combined CNN-Bi-LSTM

CNNs and Bi-LSTM were also combined to leverage the strengths of both models. The combination of the CNN-Bi-LSTM was also done in sequential form, following the same combing procedure as the CNN-LSTM.

2.5. Evaluation Metrics

This is used to evaluate the performance of a trained model in terms of accuracy, precision, recall, and f1-score, which are obtained from the confusion matrix (Megersa, 2020). Accuracy is a measure of the percentage value for correct prediction of data. It is determined using equation (8).

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \tag{8}$$

where *TP* (True Positive) is the sentence that is actually positive and will be predicted as positive; *TN* (True Negative) is the sentence that is actually negative and will be predicted as negative; *FP* (False Positive) is the sentence that is actually negative but predicted as positive; *FN* (False Negative) is the sentence that is actually positive but predicted as negative.

Precision is the total estimate of the class labels accurately predicted for each class. The precision measure is calculated using equation (9).

$$Precision = \frac{TP}{TP+FP} \tag{9}$$

Recall is the weighted average of the correct labels, correctly classified for each class. It demonstrates the classifier's capacity to predict as many correct answers (reviews with correct labels) as possible from a set of expected answers. It is determined using equation (10).

$$Recall = \frac{TP}{TP+FN} \tag{10}$$

F1 score is a harmonic mean of precision and recall. It is used to find the right balance between the two metrics and it is determined using equation (11).

$$F1 = \frac{2*recall*precision}{recall+precision} \tag{11}$$

2.6. Experimental Parameters Setting

The study was implemented in Python using Tensor Flow and the Keras library. All the optimal parameter values and coefficients were settled through trial and error and fine-tuning. The Adam optimization algorithm with a default learning rate of 0.001, which is an extension of stochastic gradient descent, was used in all the models to optimize the training mechanism. It was chosen because it is a popular algorithm in the field of deep learning and it produces good and fast results. As a loss function, categorical cross-entropy was used due to its ability to deal with multi-class classification, to match this work. In the study, 30 epochs and 32 batches were used for all models. Because all the models gave the best performance within these selected epochs and batch sizes, it was decided to fix these and others coefficient for the models through fine-tuned, trial and error by increasing and decreasing values of the parameters at the time of training models, and finally the optimal results were chosen. Global max poling for selecting informative features and dropout regularization used to prevent overfitting.

Windows environment computer, which has a Core™ processor Intel® CPU i5-4200M CPU @ 2.50 GHz, 8.00 GB RAM, 500GB storage, and Python 3.7 with Jupiter notebook and its necessary libraries was used because Python offers the best support for working with deep learning algorithms. Keras with the Tensor Flow backend library in Python for implementation that includes efficient numerical computation, Pandas, Numpy, and Matplotlib were also used for data analysis, providing fast mathematical computation on arrays and matrices, and visualizing data in a pictorial or graphical representation, respectively.

The flow diagram showing the overall processes of this study is given in Figure 4.

3. Results and Discussions

3.1. Convolutional Neural Network (CNN)

In the study, the CNN model achieved the best performance with an accuracy of 92%. The strengths of the CNN model are that it's powerful in terms of learning and extracts features such as local, semantic, and contextual features from the sequence of words by using filters in the convolution layer. In addition, as experiments indicated that it provides better results when working with short sequences of the text. Table 2 gives the performance of the model for each emotion class and Figure 5 shows the training and validation of CNN.

Table 2: CNN Model performance values

Emotion type	Precision	recall	F1	support
Happiness (Gammachuu)	0.88	0.90	0.89	39
Fear (Sodaa)	1.00	0.93	0.96	29
Anger (Aarii)	0.95	0.90	0.93	21
Sadness (Gadda)	0.91	0.98	0.94	49
Love (Jaalala)	0.84	0.93	0.89	29
Disgust (Jibba)	0.97	0.93	0.89	35
accuracy	-	-	0.92	202
macro avg.	0.92	0.91	0.92	202
weighted avg.	0.92	0.92	0.92	202

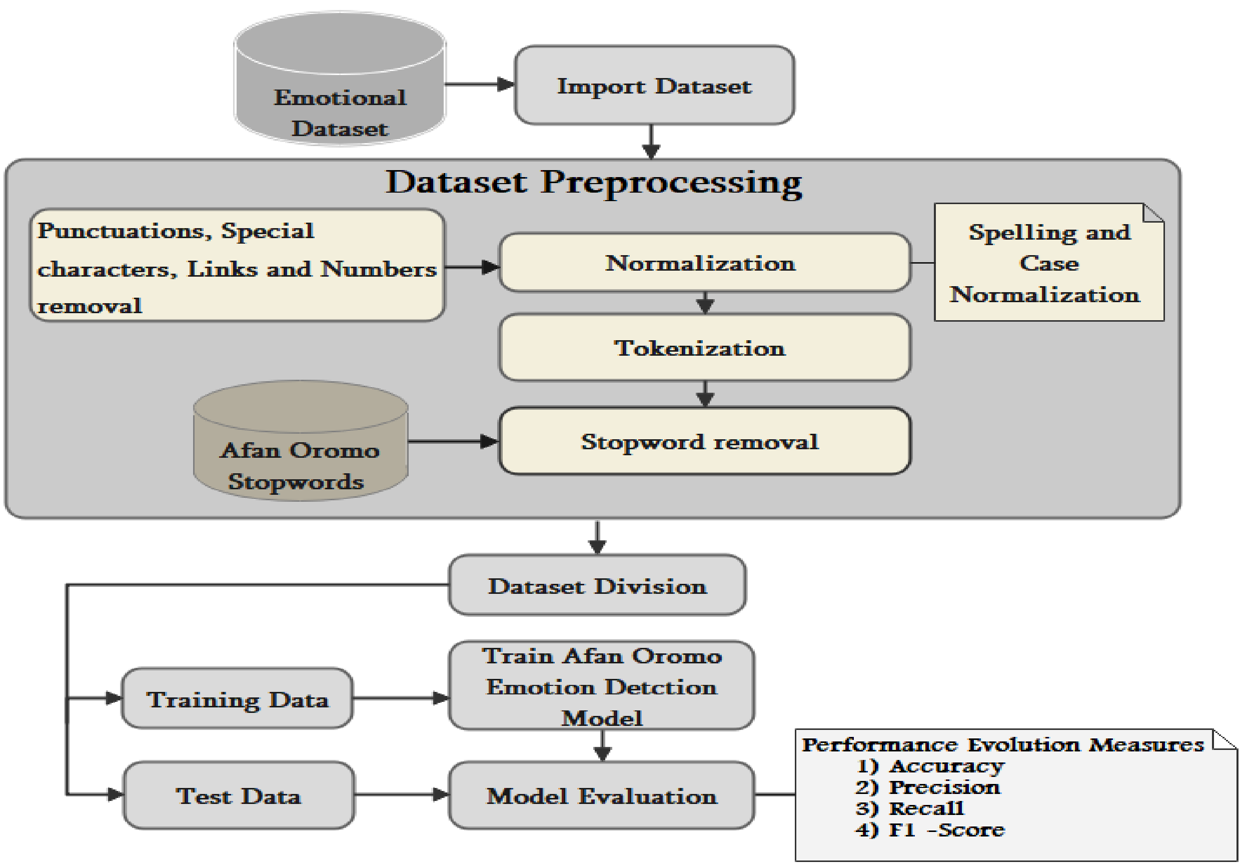
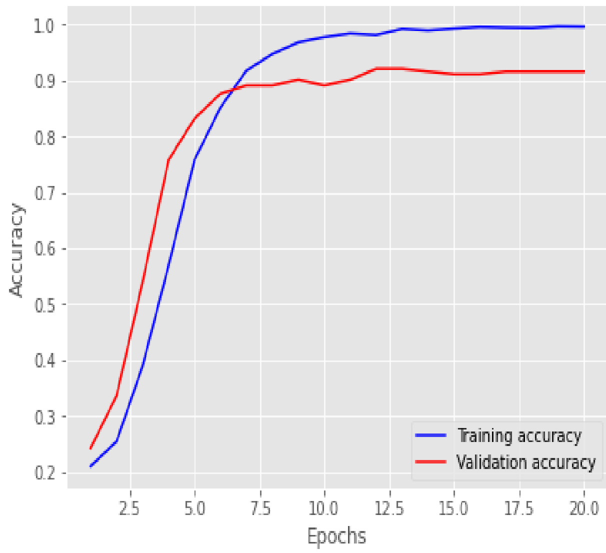
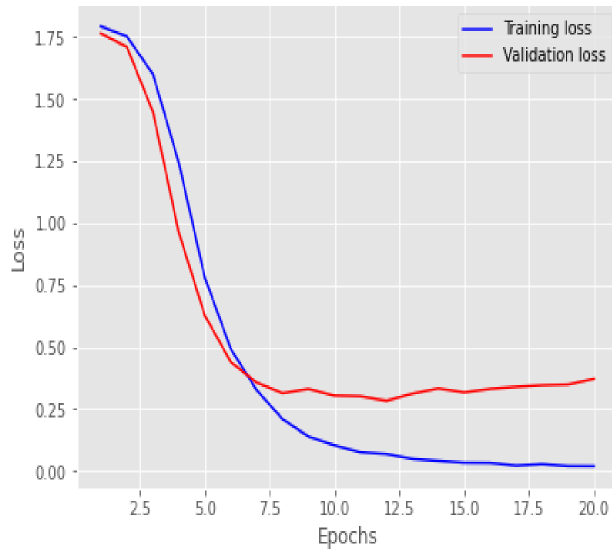


Figure 4: Framework of Afan Oromo Emotion Detection Development Process



(a)



(b)

Figure 5: CNN model validation training and validation (a) accuracy and (b) loss

3.2. Long Short-term Memory (LSTM)

The LSTM model achieved the lowest performance when compared with other models because of its computational complicity and lack of a large enough training dataset. The training and validation graph results shown in Figure 6 are different from the CNN model, which are less smooth and consistent. Table 3 gives the model's performance.

Table 3: LSTM Model performance values

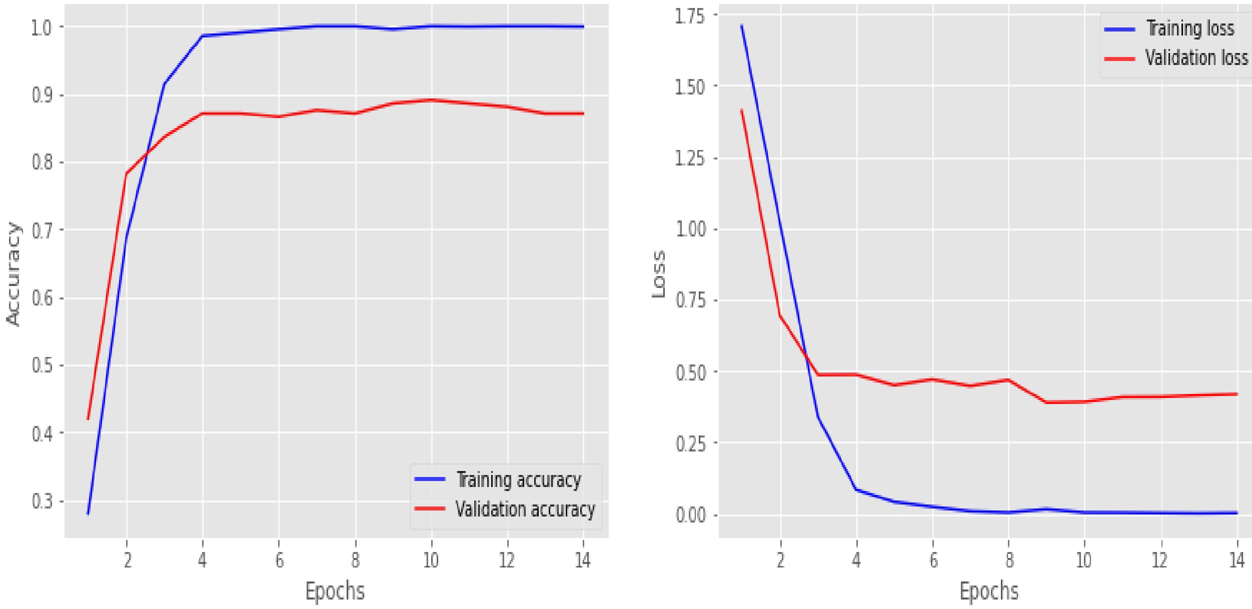
Emotion type	Precision	recall	F1	support
Happiness (Gammachuu)	0.89	0.87	0.88	39
Fear (Sodaa)	0.88	0.72	0.79	29
Anger (Aarii)	0.83	0.90	0.86	21
Sadness (Gadda)	0.89	0.98	0.93	49
Love (Jaalala)	0.86	0.83	0.84	29
Disgust (Jibba)	0.86	0.86	0.86	35
accuracy	-	-	0.87	202
macro avg.	0.87	0.86	0.86	202
weighted avg.	0.87	0.87	0.87	202

3.3. Bidirectional Long-Short Term Memory (Bi-LSTM)

The distinctive feature of Bi-LSTM, which is performing computations in both forward and backward and then combining the informative features from both ends to derive a single output, has resulted in good detection results than the LSTM. Figures 7 and Table 4 show all the computational results of the Bi-LSTM model.

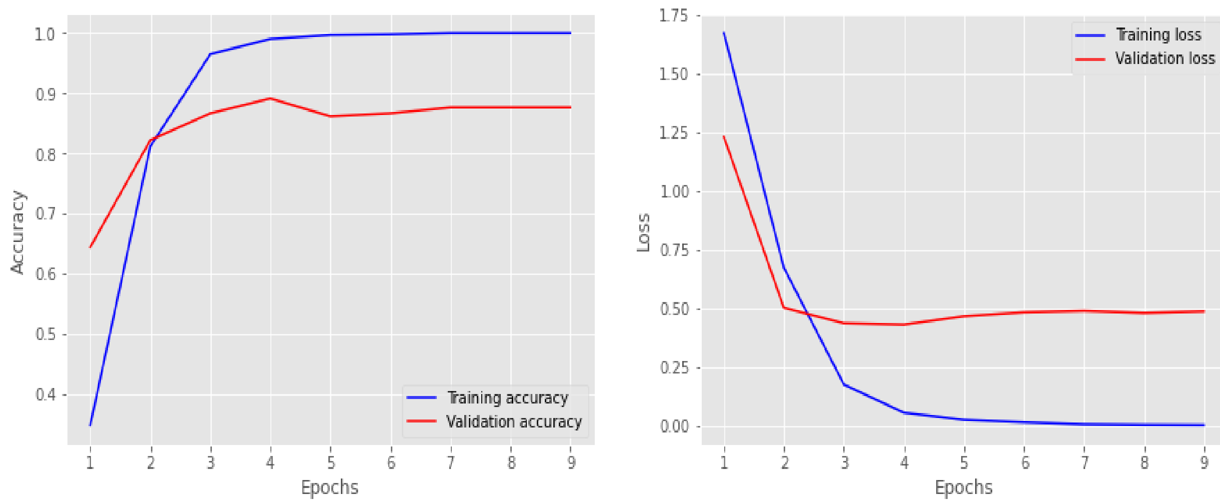
Table 4: *Bi-LSTM* Model performance values

Emotion type	Precision	recall	F1	support
Happiness (Gammachuu)	0.83	0.87	0.85	39
Fear (Sodaa)	0.89	0.86	0.88	29
Anger (Aarii)	0.89	0.81	0.85	21
Sadness (Gadda)	0.91	0.98	0.94	49
Love (Jaalala)	0.83	0.86	0.85	29
Disgust (Jibba)	0.90	0.80	0.85	35
accuracy	-	-	0.88	202
macro avg.	0.88	0.86	0.87	202
weighted avg.	0.88	0.88	0.88	202



(a) (b)

Figure 6: LSTM model validation training and validation (a) accuracy and (b) loss



(a) (b)

Figure 7: Bi-LSTM model validation training and validation (a) accuracy and (b) loss

3.4. Combined CNN-LSTM

Figures 8 and Table 5 indicate the training and validation results, and performance report for the CNN-LSTM model, respectively. The combined CNN-LSTM model has better performance than LSTM, likely because the model benefited from CNN's ability to extract local features and leverage the unique advantages of both of them.

Table 5: Combined CNN-LSTM Model performance

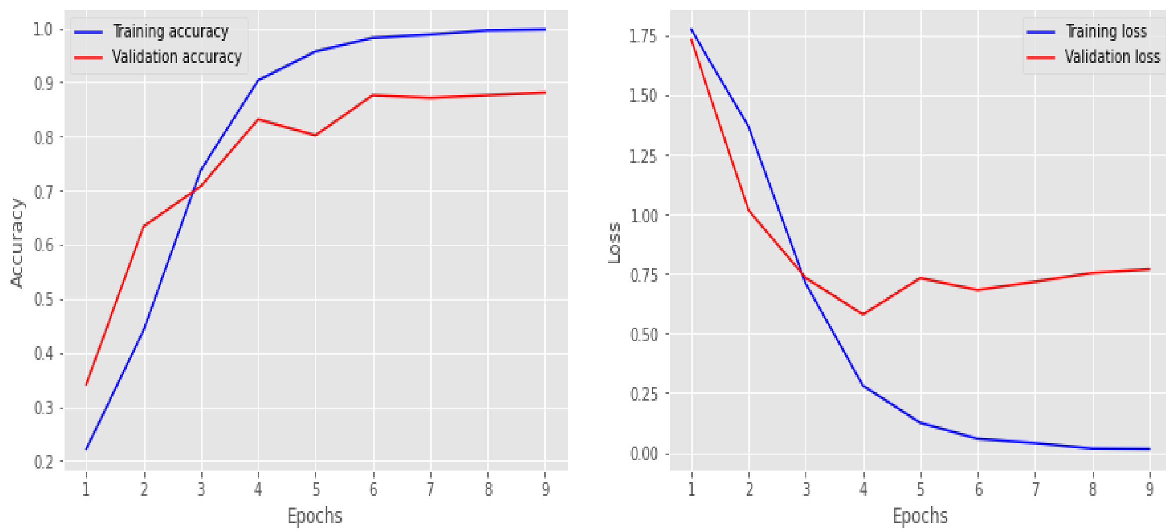
Emotion type	Precision	recall	F1	support
Happiness (Gammachuu)	0.92	0.90	0.91	39
Fear (Sodaa)	0.96	0.79	0.87	29
Anger (Aarii)	0.81	0.81	0.81	21
Sadness (Gadda)	0.92	0.96	0.94	49
Love (Jaalala)	0.76	0.90	0.83	29
Disgust (Jibba)	0.88	0.86	0.87	35
accuracy	-	-	0.88	202
macro avg.	0.88	0.87	0.87	202
weighted avg.	0.89	0.88	0.88	202

3.5. Combined CNN-Bi-LSTM

The hybrid of CNN-BiLSTM gave results higher than that of CNN-LSTM, mainly because BiLSTM is an advanced form of LSTM and it processes data in both directions and then combines the informative features from both ends to derive an output. This allows the network to learn relevant features from the text data. Thus, combined CNN-BiLSTM leveraged the unique benefits of the individuals and achieved the performance value just after the CNN model. Figures 9 and Table 6 illustrates the training and validation, and performance report for the CNN–BiLSTM model.

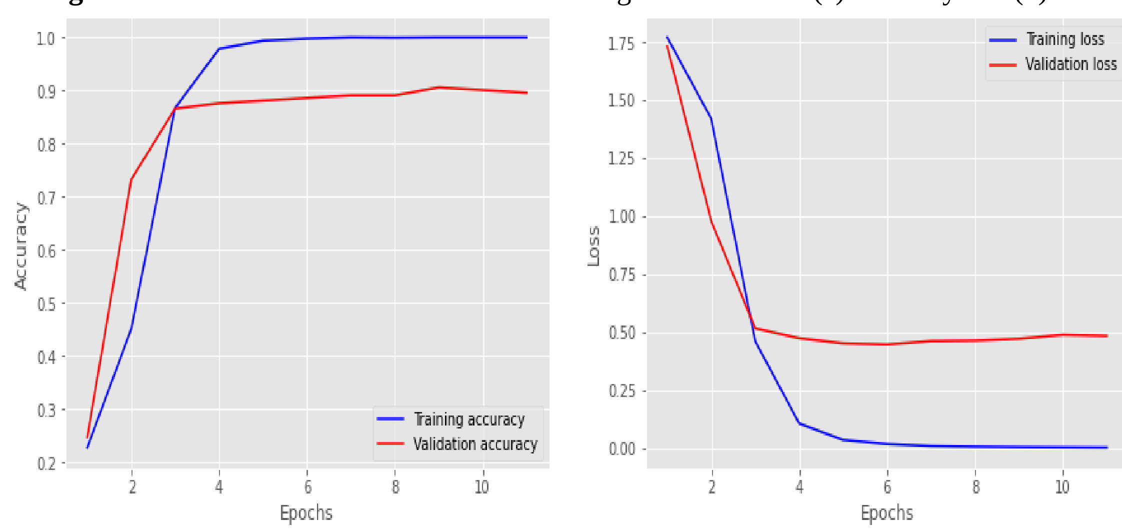
Table 6: Combined CNN-BiLSTM Model performance values

Emotion type	Precision	recall	F1	support
Happiness (Gammachuu)	0.94	0.82	0.88	39
Fear (Sodaa)	0.96	0.90	0.93	29
Anger (Aarii)	0.95	0.90	0.93	21
Sadness (Gadda)	0.89	0.98	0.93	49
Love (Jaalala)	0.82	0.93	0.87	29
Disgust (Jibba)	0.85	0.83	0.84	35
accuracy	-	-	0.90	202
macro avg.	0.90	0.89	0.90	202
weighted avg.	0.90	0.90	0.90	202



(a) (b)

Figure 8: CNN-LSTM model validation training and validation (a) accuracy and (b) loss



(a) (b)

Figure 9: CNN-BiLSTM model validation training and validation (a) accuracy and (b) loss

In general, the results of the proposed deep learning models showed accuracy of 92, 90, 88, 88 and 87 % for CNN, CNN-BiLSTM, BiLSTM, CNN-LSTM and LSTM, respectively. Thus, the LSTM performed least compared to other models. The low performance is caused by features of LSTM: it processes information sequentially, its computational complexity, and its worst feature extractor, which has information loss and long-term dependency (Jang et al., 2020). In addition, LSTMs have a number of parameters, which can make the training process slow and computationally expensive. The computational complexity LSTM needs v large training data; however, in the present study training data was limited that the process was likely susceptible to overfitting. Both LSTM and Bi-LSTM had slower computational times due to their architectural computational complexity when compared with CNN.

The CNN model gave superior accuracy compared to the other models because it extracts low-level semantic, local, and contextual features from the sequence of words using convolutional filters, reducing the number of dimensions without losing the information. Besides, CNN model performed better than LSTM and BiLSTM while dealing with a limited number of dataset and a short sequence of texts. The consequence of not having a large enough dataset restricted the performance of the other models.

Thus, the CNN model is suitable to detect emotion in the Afan Oromo short sequence of texts and to achieve better results when dealing with limited data size. However, all the models used in the study are state-of-the-art, and in the existing challenge of the dataset, they generally performed well in training and testing. Thus, the study showed the possibility of emotion detection by using the available data size and by considering effective preprocessing methods (Kabada & Getachew, 2020).

4. Conclusions and Recommendations

The detection of emotions is a challenging task due to the complexity of human emotion in texts and the various ways of emotions expression. Previous studies mainly focused on Afan Oromo sentiment analysis and the objective of this study was detecting using a deep learning method. The state-of-the-art deep learning models were deigned and explored to leverage the benefits of each of them. The models were trained and evaluated on Afan Oromo text data which were collected from Facebook. The findings showed that all the models performed well in detecting emotions in the Afan Oromo text, with accuracies ranging from 87 to 92%. The CNN model gave the highest accuracy of 92%, closely followed by the CNN-BiLSTM model at 90%. The superior performance of the CNN model can be attributed to its ability to capture local patterns in the text using convolutional filters. By scanning the input text, CNNs are adept at identifying relevant patterns and features that contribute to emotion detection. In addition, the CNN model was found to be suitable for detecting emotion in a short text sequence and that it tolerated working with limited annotated data. Additionally, it is expected that the CNN model had a profound impact on the performance results of the hybrid models. The CNN-BiLSTM model combined the strengths of CNN and Bi-LSTM, leveraging both local and contextual information in the text, resulting in its high accuracy. The complexity of the LSTM and BiLSTM architectures and the limited training dataset used in experiments lead the impact on the results of the hybrid models; however, the hybrid had greater performance than individual LSTM and BiLSTM.

Overall, to the best of our knowledge, there are no previous research works on Afan Oromo emotion detection in the texts with similar classification algorithms and achieved satisfactory results in all the models. The current study has important implications for real-world applications such as sentiment analysis, market research, and customer feedback analysis. These applications can help businesses and organizations understand their customers' emotions and needs to make data-driven decisions to improve their products and services. Further research can involve other deep learning architectures to improve the accuracy and robustness of emotion detection in Afan Oromo text. Moreover, developing a large size of dataset that will be used for emotion detection and other related studies can be worked on. Future works may also use other approaches like transfer learning methods (BERT) and traditional machine learning techniques and compare the results with those of the present study. Moreover, documents such as video, audio, and images, and combinations of them, and collection of more data from different social media platforms like Twitter and YouTube are recommended.

Acknowledgments: The authors would like to thank all participants in this study for their collaborative support during the study.

Reference

Acheampong, F. A., Wenyu, C., & Nunoo-Mensah, H. (2020). Text-based emotion detection: Advances, challenges, and opportunities. *Engineering Reports*, *2*(7), 1–24. https://doi.org/10.1002/eng2.12189

Al-Omari, H., Abdullah, M. A., & Shaikh, S. (2020). EmoDet2: Emotion Detection in English Textual Dialogue using BERT and BiLSTM Models. In Proceedings of the *11th International Conference on Information and Communication Systems, ICICS 07-09 April* 2020, Irbid, Jordan. https://doi.org/10.1109/ICICS49469.2020.239539

AlBalooshi, H., Rahmanian, S., & Venkatesh Kumar, R. (2018). EmotionX-SmartDubai_NLP: Detecting User Emotions In Social Media Text. In Proceedings of *the Sixth International Workshop on Natural Language Processing for Social Media*, pp. 45–49, Melbourne, Australia. https://aclanthology.org/W18-3508

Ashebir Hunegnaw, Million Mesheshab, EndalewAlamirc & Bahiru Shiferaw. (2021). Sentiment Analysis Model for Afaan Oromoo Short Message Service Text: A Machine Learning Approach. Turkish Journal of Computer and Mathematics Education, 12(13), 332-342

Bharti, S. K., Varadhaganapathy, S., Gupta, R. K., Shukla, P. K., Bouye, M., Hingaa, S. K., & Mahmoud, A. (2022). Text-Based Emotion Recognition Using Deep Learning Approach. *Computational Intelligence and Neuroscience*, *2645381*. https://doi.org/10.1155/2022/2645381

Gezehagn Gutema Eggi (2012). Afaan Oromo Text Retrieval System. MSc Thesis, Addis Ababa University, Addis ababa, Ethiopia.

Ghourabi, A., Mahmood, M. A., & Alzubi, Q. M. (2020). A hybrid CNN-LSTM model for SMS spam detection in arabic and english messages. *Future Internet*, *12*(9), 156. https://doi.org/10.3390/FI12090156

Jang, B., Kim, M., Harerimana, G., Kang, S., & Kim, J. W. (2020). Bi-LSTM Model to Increase Accuracy in Text Classification: Combining Word2vec CNN and Attention Mechanism. *Applied sciences.* 10(17), 5841; https://doi.org/10.3390/app10175841

Kabada Sori & Getachew Mamo (2020). Emotion Detection for Afaan Oromo Using Deep Learning. New Media and Mass Communication, 92, 1–14. https://doi.org/10.7176/NMMC/92-01

Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pp. 1746–1751, Doha, Qatar. https://doi.org/10.3115/v1/d14-1181

Kumar, A. & Rastogi, R. (2019). Attentional recurrent neural networks for sentence classification. In: Deb, D., Balas, V., Dey, R. (eds) Innovations in Infrastructure. Advances in Intelligent Systems and Computing, vol 757. Springer, Singapore. https://doi.org/10.1007/978-981-13-1966-2_49.

Megersa Oljira, Rase (2020). Sentiment Analysis of Afaan Oromoo Facebook Media Using Deep Learning Approach. New Media and Mass Communication, 90, 7–22. https://doi.org/10.7176/nmmc/90-02

Motwani, A., Shukla, P. K., & Pawar, M. (2023). Novel framework based on deep learning and cloud analytics for smart patient monitoring and recommendation (SPMR). J Ambient Intell Human Comput 14, 5565–5580. https://doi.org/10.1007/s12652-020-02790-6.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. *In Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel.

Navarrete, A. S., Martinez-Araneda, C., Vidal-Castro, C., & Rubio-Manzano, C. (2021). A novel approach to the creation of a labelling lexicon for improving emotion analysis in text. The Electronic Library, 39(1), 118–136. https://doi.org/10.1108/EL-04-2020-0110

Obsa Gelchu Horsa & Kula Kekeba Tune. (2023). Aspect-Based Sentiment Analysis for Afaan Oromoo Movie Reviews Using Machine Learning Techniques. Applied Computational Intelligence and Soft Computing, 3462691. https://doi.org/10.1155/2023/3462691

Sailunaz, K., Dhaliwal, M., Rokne, J., & Alhajj, R. (2018). Emotion detection from text and speech: a survey. *Social Network Analysis and Mining, 8*, 28. https://doi.org/10.1007/s13278-018-0505-2

Seyeditabari, A., Tabari, N., & Zadrozny, W. (2018). *Emotion Detection in Text: a Review*. Cornell University. https://doi.org/10.48550/arXiv.1806.00674

Singh, D., Kaur, M., Jabarulla, M. Y., Kumar, V., & Lee, H.-N. (2022). Evolving fusion-based visibility restoration model for hazy remote sensing images using dynamic differential evolution. *IEEE Transactions on Geoscience and Remote Sensing, 60(4),* 1–14. https://doi.org/10.1109/TGRS.2022.3155765

Statcounter (2023). Social Media Status in Ethiopia, Dec 2022 - Dec 2023. Available at https://gs.statcounter.com/social-media-stats/all/ethiopia

Wakshum, M. (2000). Development of Stemming Algorithm for Afaan Oromo Text. M. Sc. Theses, Addis Ababa University, Ethiopia.

Wegderes Tariku, Million Meshesha, Ashebir Hunegnaw & Kedir Lemma. (2022). Sentiment Mining and Aspect Based Summarization of Opinionated Afaan Oromoo News Text. *American Journal of Embedded Systems and Applications,* 9(2), 66-72 https://doi.org/10.11648/j.ajesa.20220902.12

Workineh Tesema & Duresa Tamirat. (2017). Investigating Afan Oromo Language Structure and Developing Effective File Editing Tool as Plug-in into Ms Word to Support Text Entry and Input Methods. American Journal of Computer Science and Engineering Survey, 1–8.

Zhou, C., Sun, C., Liu, Z., & Lau, F. (2015). A C-LSTM neural network for text classification. Cornell university. https://doi.org/10.48550/arXiv.1511.08630