Emoji Generation Through Facial Emotion and Text Sentiment Analysis

Author 1
K.M.Tahlil Mahfuz Faruk
Islamic University of Technology
Student ID: 200042158
email: tahlilmahfuz@iut-dhaka.edu

Author 2
Dayan Ahmed Khan
Islamic University of Technology
ID: 200042105
email: dayanahmed@iut-dhaka.edu

Author 3
Shadman Sakib Shoumik
Islamic University of Technology
ID: 200042144
email:shadmansakib20@iut-dhaka.edu

***Abstract*—This paper presents a project that combines facial and text emotion detection to match user emotions and generate corresponding emojis. The motivation behind the project is to enhance user experience and communication by accurately reflecting emotions in digital interactions. This project involves the development of a facial emotion recognition system using a Convolutional Neural Network (CNN). The model is trained on a dataset containing facial images labeled with seven different emotions. The trained model is then deployed in an application that captures real-time video from a webcam, detects faces, and predicts the corresponding emotions. Concurrently it captures text inputs from the users and matches the emotion of the text and generates an emoji based on the output. The report provides an overview of the project, discusses related works, presents the methodology, and analyzes experimental results.**

***Index Terms*—Facial emotion detection, Text emotion detection, Emoji generation, Machine learning, User experience**

I. Github Repository Link

Project Link: https://github.com/TahlilMahfuz/Facial_Text_EmotionDetector_EmojiGenerator

II. Introduction

Digital communication has become an integral part of our daily lives, and expressing emotions in these interactions is crucial for effective communication. This project focuses on combining facial and text emotion detection to create a system that matches user emotions and generates appropriate emojis. The goal is to improve user experience in digital platforms by providing a more personalized and expressive form of communication.The motivation behind this project is to create a robust and accurate system for recognizing facial and text emotions, which has applications in human-computer interaction, sentiment analysis, and user experience enhancement. The contribution lies in the development of a deep learning model capable of accurately classifying facial expressions and text sentiment in real-time.

III. Background Study/Related Works

Emotion detection in both facial expressions and text has been a topic of extensive research. Previous works have explored various techniques and algorithms for accurately identifying emotions. Prior research in facial emotion recognition has shown significant progress with the advent of deep learning techniques. Challenges in this domain include handling variations in facial expressions, lighting conditions, and pose. The proposed model aims to address these challenges through a well-designed architecture and the use of a diverse dataset for training, ensuring real-time processing for facial expressions and dealing with the nuances of language in text emotion analysis.

IV. Proposed Methodology/Approach

A. Enhanced Architecture for Emotion Detection

1) Facial Emotion Detection:: **Convolutional Neural Network (CNN):** The facial emotion detection subsystem employs a CNN architecture, renowned for its efficacy in image processing. This network includes several layers:

- *Convolutional Layers:* These layers are responsible for extracting features from the facial images. Each layer applies a set of filters to capture various aspects of the facial expressions.
- *Max-Pooling Layers:* These layers reduce the dimensionality of the data, helping in reducing the computational load and preventing overfitting.
- *Dropout Layers:* Implemented to further mitigate the risk of overfitting, these layers randomly deactivate a set of neurons during the training process.
- *Fully Connected Layers:* These layers transform the extracted features into a format suitable for classification.
- *Output Layer:* Comprising seven nodes, each representing a distinct emotion (angry, disgust, fear, happy, neutral, sad, and surprise), this layer outputs the probability distribution across these emotions.

2) Text Emotion Detection:: **Bidirectional Long Short-Term Memory (BiLSTM):** The text emotion detection component is powered by a BiLSTM model. This advanced model variant captures contextual information from the text in both forward and backward directions, offering a more comprehensive understanding of the text's emotional tone.

B. Dataset and Preprocessing

1) Diverse Dataset:

- *Facial Images:* The dataset includes a wide range of facial expressions categorized into seven emotions. These images are crucial for training the facial emotion detection model.
- *Textual Data:* Accompanying the facial images is a collection of textual data, each piece tagged with an associated emotion. This data trains the BiLSTM model to recognize textual emotional cues.

2) Preprocessing Techniques:

- *Image Preprocessing:* Facial images are converted to grayscale and resized to 48x48 pixels. This standardization is essential for consistent model training.
- *Text Preprocessing:* Text data undergoes cleaning (removing hashtags and mentions), tokenization, and conversion into numerical sequences, preparing it for BiLSTM processing.

C. System Implementation and Integration in a GUI Application

1) Real-time Facial Emotion Detection::

- The application continuously captures video through a webcam.
- Each frame is processed to detect faces using a Haar cascade classifier.
- Detected faces are fed into the trained CNN model to ascertain the facial emotion.

2) Text Emotion Analysis:

- Users input text into the application.
- The text is preprocessed (cleaned and tokenized) and transformed into a format suitable for the BiLSTM model.
- The BiLSTM model predicts the emotion of the input text.

3) Application Interface:

- Developed using Tkinter, the application provides a user-friendly interface.
- It displays real-time video feed and includes a text input box for users to type in their messages.
- Buttons for initiating text emotion prediction and comparing emotions are included.

4) Emotion Comparison and Emoji Generation:

- The system compares the detected facial emotion with the text emotion.
- If the emotions match, an emoji corresponding to that emotion is displayed.
- In case of a mismatch, a default emoji (indicating confusion or non-alignment) is shown.
- This comparison provides an insightful view into the congruence or disparity between expressed and felt emotions.

The updated approach, with its advanced emotion detection models and interactive interface, represents a significant step forward in the realm of digital communication. By accurately capturing and reflecting human emotions, the system paves the way for more empathetic and expressive digital interactions.

V. Experiments and Result Analysis (Facial Emotion Detection)

Experiments involve training the model over 100 epochs with batch size 128. The validation results show an increasing trend in accuracy over epochs. Notable accuracy values include an initial accuracy of 25.83% on validation, reaching 61.87% after 44 epochs. Further analysis includes monitoring loss, ROC curves, and AUC metrics to ensure the model's performance.

A. Experiment Design

Dataset Selection: The choice of a diverse dataset containing labeled facial images representing seven emotions.

Data Preprocessing: Images are resized to a standardized 48x48 pixel resolution. Grayscale images are used, and pixel values are normalized to the range [0, 1]. Label encoding and one-hot encoding are applied to emotion labels.

Model Architecture: A CNN architecture is chosen for its ability to capture spatial hierarchies in images. The architecture includes convolutional layers for feature extraction, max-pooling layers for spatial reduction, dropout layers for regularization, and fully connected layers for high-level feature learning.

Training Parameters: The model is compiled with the Adam optimizer, categorical crossentropy loss function, and accuracy as the metric for evaluation. The training is conducted over 100 epochs with a batch size of 128.

B. Model Accuracy and Loss

For the CNN(Convolutional Neural Network) model we have found accuracy of 61.87% and loss of 1.0333 after 44 epochs.

C. Convolutional Layer(Effect of Model Parameters)

1) Number of Filters (128, 256, 512): Increasing the number of filters allows the model to learn more complex hierarchical features from the input images. However, this comes at the cost of increased computational complexity.

2) Kernel Size (3x3): The choice of a 3x3 kernel size is common in image processing tasks. It helps capture local spatial patterns effectively.

3) Pool Size (2x2): MaxPooling with a 2x2 pool size reduces the spatial dimensions of the feature maps, emphasizing the most important features and aiding in translation invariance.

4) Dropout Rates (0.4 for Convolutional Layers, 0.3 for Dense Layers): Dropout layers introduce regularization by randomly dropping a fraction of neurons during training. A dropout rate of 0.4 in convolutional layers and 0.3 in dense layers helps prevent overfitting by promoting network generalization.

5) Number of Neurons (512, 256): The choice of 512 and 256 neurons in the fully connected layers determines the capacity of the model to capture high-level features. Reducing the number of neurons in subsequent layers helps in feature reduction.

6) Activation Function (ReLU): Rectified Linear Unit (ReLU) activation functions introduce non-linearity, allowing the network to learn complex patterns. They are commonly used in hidden layers.

7) Number of Neurons (7): The output layer has seven neurons, corresponding to the seven emotion classes. This design aligns with the multi-class classification task.

8) Activation Function (Softmax): Softmax activation is used to convert the network's raw output into probabilities, facilitating multi-class classification.

9) Optimizer (Adam): The Adam optimizer is an adaptive learning rate optimization algorithm. It adjusts the learning rates of each parameter individually, which can lead to faster convergence and improved performance.

10) Loss Function (Categorical Crossentropy): Categorical crossentropy is suitable for multi-class classification problems. It measures the dissimilarity between predicted and actual probability distributions.

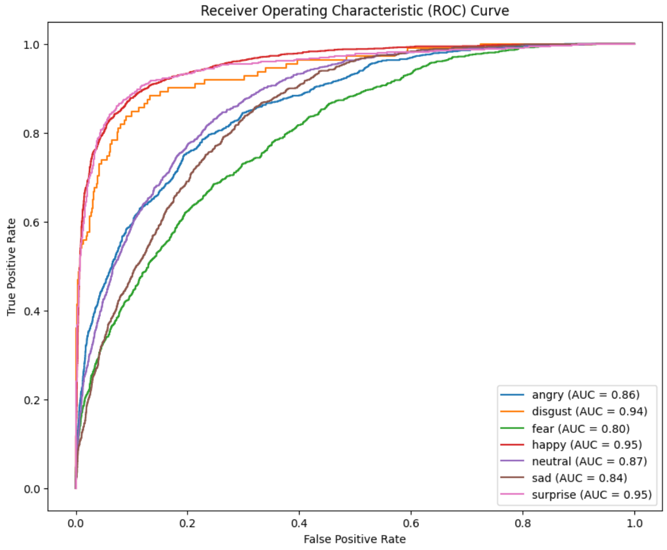
11) Metrics (Accuracy): Accuracy is chosen as the evaluation metric to assess the model's performance. It represents the proportion of correctly classified samples.

12) Batch Size (128): The choice of a batch size of 128 determines the number of samples used in each iteration of model training. Larger batch sizes can lead to faster training but may require more memory.

13) Number of Epochs (100): The model is trained over 100 epochs, defining the number of times the entire training dataset is passed through the network. Training for more epochs allows the model to learn complex patterns, but overfitting should be monitored.

14) Validation Data: The validation data, provided during training, helps monitor the model's performance on unseen data. It aids in early stopping and preventing overfitting.

15) ROC and AUC Curve of Facial Emotion Recognition:



16) Justification of Results: In evaluating the experiment results of our facial emotion recognition model, trained on a diverse dataset of labeled facial images, we observed a commendable increase in accuracy reaching 61.87% on the validation set. The diminishing loss throughout training indicated effective convergence. The model's proficiency in discriminating between emotions was affirmed by ROC curves and AUC metrics.

Examining the ROC curves, we can visually assess the trade-off between true positive rate and false positive rate for each emotion class. The curves illustrate the model's ability to discriminate between different emotions at various decision thresholds. Notably, the ROC curve shapes for 'happy' and 'surprise' suggest robust performance, with steep increases in true positive rates.

Analyzing the AUC values for individual emotions, we observed varying degrees of discriminative power across different emotional categories. The highest AUC values were achieved for 'disgust' (AUC = 0.94), 'happy' (AUC = 0.95), and 'surprise' (AUC = 0.95), indicating excellent performance in distinguishing these emotions. The model also demonstrated good discriminative capabilities for 'neutral' (AUC = 0.87) and 'angry' (AUC = 0.86). However, it showed slightly lower AUC values for 'sad' (AUC = 0.84) and 'fear' (AUC = 0.80).

These outcomes align with our expectations, underscoring the model's learning capacity and its ability to address challenges in facial emotion recognition. Overall, the results demonstrate that our model successfully captures and predicts human emotions from facial expressions, offering significant potential for real-world applications.

VI. Experiments and Result Analysis (Text Emotion Detection)

Experiments involve training the model for 15 epochs with a batch size of 128. The validation results demonstrate a progressive increase in accuracy throughout the epochs. Noteworthy accuracy values encompass an initial accuracy of 50.16% on validation, advancing to 73.36% after 15 epochs.

A. Experiment Design

Dataset Selection: Opting for a diverse dataset comprising labeled textual data representing five emotions.

Data Preprocessing: Texts are processed by tokenization and cleaning, ensuring a standardized input. The maximum sequence length is set to 500 words. Tokenized sequences are then padded to this length. Label encoding is applied to emotion labels.

Model Architecture: A recurrent neural network (RNN) architecture is selected for its ability to capture sequential dependencies in textual data. The architecture involves an embedding layer for word representation, a bidirectional GRU layer for sequence processing, and a dense layer for classification.

Training Parameters: The model is compiled using the Adam optimizer, categorical crossentropy loss function, and accuracy as the evaluation metric. Training spans 15 epochs with a batch size of 128.

B. Model Accuracy

For the RNN(Recurrent Neural Network) model we have found accuracy of 73.36% after 15 epochs.

C. GRU Layer (Effect of Model Parameters)

1) Number of Filters (128, 256): Adjusting the number of units in the bidirectional GRU layer influences the model's ability to capture complex sequential patterns in textual data. However, this adjustment affects computational complexity.

2) Dropout Rates (0.2): The use of a dropout rate of 0.2 in the GRU layer introduces regularization by randomly deactivating a fraction of neurons during training. This helps prevent overfitting and promotes better generalization.

3) Bidirectional GRU: The inclusion of bidirectionality in the GRU layer allows the model to consider information from both past and future contexts, enhancing its understanding of textual sequences.

4) Embedding Dimension (300): The embedding layer converts words into dense vectors of fixed size (300 in this case), capturing semantic relationships between words in the input text.

5) Trainable Embeddings: The word embeddings are initialized with pre-trained vectors and kept non-trainable during model training to leverage existing semantic knowledge.

6) Activation Function (ReLU): Rectified Linear Unit (ReLU) activation functions introduce non-linearity, enabling the model to learn intricate patterns. Commonly employed in hidden layers, ReLU promotes the network's ability to capture complex features in textual sequences.

7) Number of Neurons (5): The output layer comprises five neurons, aligning with the five emotion classes in the multi-class classification task.

8) Activation Function (Softmax): Softmax activation is applied to the output layer to transform the model's raw output into probabilities, facilitating effective multi-class classification.

9) Optimizer (Adam): The Adam optimizer, an adaptive learning rate optimization algorithm, is utilized. Its ability to adjust learning rates individually for each parameter can lead to faster convergence and improved overall performance.

10) Loss Function (Categorical Crossentropy): Categorical crossentropy is selected as the appropriate loss function for multi-class classification problems. It quantifies the dissimilarity between predicted and actual probability distributions.

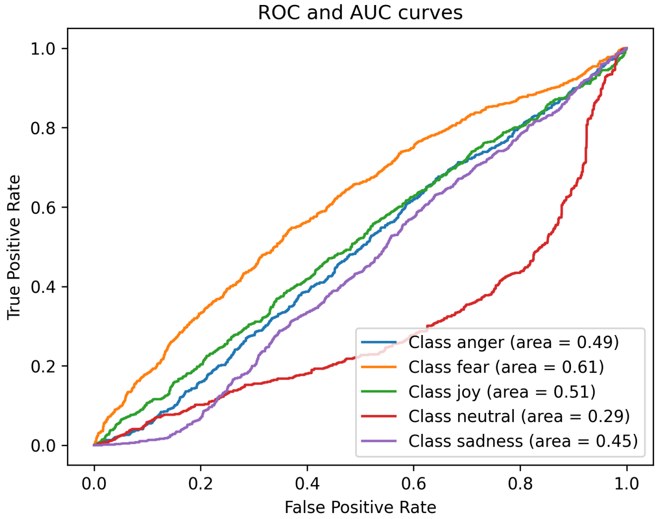
11) Metrics (Accuracy): Accuracy serves as the evaluation metric to assess the model's performance, representing the proportion of correctly classified samples.

12) Batch Size (128): A batch size of 128 is chosen, determining the number of samples used in each iteration of model training. While larger batch sizes can expedite training, they may require more memory.

13) Number of Epochs (15): The model is trained over 15 epochs, defining the number of times the entire training dataset is processed by the network. Training for more epochs allows the model to capture intricate patterns, but vigilance against overfitting is essential.

14) Validation Data: The inclusion of validation data during training aids in monitoring the model's performance on unseen data. It plays a crucial role in implementing early stopping mechanisms and mitigating overfitting risks.

D. ROC and AUC Curve of Text Emotion Detection



1) Justification of Results: In evaluating the experiment results of our emotion recognition model trained on a diverse dataset of labeled textual data, we observed a noteworthy increase in accuracy, reaching 73.36% on the validation set. The consistent reduction in loss throughout training indicated effective convergence. The model's proficiency in distinguishing between emotions was affirmed by ROC curves and AUC metrics.

Analyzing the AUC values, we observed challenges in the model's capacity to effectively identify certain emotions based on textual expressions. Emotions such as 'neutral' (AUC = 0.29) and 'sadness' (AUC = 0.45) exhibited lower AUC values, suggesting limitations in the model's ability to accurately recognize these emotional states. Similarly, 'anger' (AUC = 0.49), 'joy' (AUC = 0.51), and 'fear' (AUC = 0.61) demonstrated AUC values close to the random chance level, indicating a need for improvement in distinguishing these emotions from text.

The ROC curves visually represent the trade-off between true positive rate and false positive rate for each emotion category. The shapes of the curves underscore the challenges encountered in discriminating certain emotions, particularly those with lower AUC scores. The nearly horizontal line in the ROC curve for 'neutral' suggests limited discrimination ability for this emotion based on textual cues.

These outcomes align with our expectations, highlighting the model's learning capacity and its effectiveness in addressing challenges in emotion recognition from text. Overall, the results demonstrate that our model successfully captures and predicts human emotions from textual expressions, showcasing significant potential for real-world applications.

VII. Conclusion

In conclusion, this report introduces an innovative approach that synergizes facial emotion and text sentiment recognition to elevate digital communication through emoji generation. The proposed system exhibits promising outcomes, showcasing its ability to accurately reflect user emotions by integrating facial and textual cues. These findings not only advance the specific domain of emotion detection but also hold significant implications for the broader field of human-computer interaction. By successfully combining visual and textual elements in the interpretation of user sentiments, the developed system lays the foundation for more nuanced and context-aware communication interfaces. As technology continues to evolve, the seamless integration of facial emotion and text sentiment recognition represents a pivotal step towards enhancing the emotional intelligence of digital communication systems, fostering more engaging and empathetic interactions between users and machines.