Vol. 7 No. 2, 2026, pp. 196-206 **Jurnal Pepadun**
© 2026 Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0)

Convolutional Recurrent Neural Network with Bi-LSTM for Speech Emotion Recognition

[*1]M. Raditya Adhirajasa, [2]Favorisen R. Lumbanraja, [3]Yunda Heningtyas, and [4]Akmal Junaidi

[1,2,3,4]Department of Computer Science, University of Lampung, Jl. Prof. Dr. Ir. Sumatri Brojonegoro No.1, Bandar Lampung, Lampung Province, Indonesia, 35145.
e-mail: [*1]**radityaadhirajasaa@gmail.com**, [2]favorisen.lumbanraja@fmipa.unila.ac.id, [3]yunda.heningtyas@fmipa.unila.ac.id, [4]akmal.junaidi@fmipa.unila.ac.id

Abstract *- Understanding the emotion in a human voice is a complex but crucial part of improving human machine interaction. This study explores the potential of a Convolutional Recurrent Neural Network (CRNN), an architecture that combines the spatial feature-finding ability of a CNN with the temporal awareness of a BiLSTM. This model was trained and tested using the CREMA-D dataset, a collection of audio clips covering six basic emotions including anger, happiness, and sadness. To prepare the audio for the model, the signals were converted into log-Mel spectrograms, a format chosen for its relevance to human auditory perception. In an effort to improve the model's ability to generalize, data augmentation techniques like pitch shifting and time stretching were also implemented. The results suggest that this approach was beneficial. Adding data augmentation appeared to lift the validation accuracy to 68.23%, a noticeable improvement from the 66.08% achieved without it. The model showed particularly promising performance when it came to classifying 'anger' and 'neutral' emotions. Ultimately, this work indicates that the CRNN architecture with BiLSTM could be a capable approach for speech emotion recognition, and it strongly underscores the vital role that data augmentation seems to play in achieving better performance. Looking ahead, further gains might be possible by investigating more sophisticated augmentation methods or by incorporating a richer set of acoustic features for the model to learn from.*

Keywords**:** *Speech Emotion Recognition; CRNN; BiLSTM; Log-Mel Spectrogram; Data Augmentation.*

1. INTRODUCTION

The way a person feels often comes through in their voice, fundamentally shaping how they interact and communicate [1]. It is no surprise, then, that as artificial intelligence has advanced, a significant effort has gone into creating systems that might be able to understand these human emotions [2]. This has led to the field of Speech Emotion Recognition (SER), a computational endeavor to identify feelings from audio clips. The potential uses are vast, ranging from building more empathetic digital assistants to providing new tools for psychological research. But as it turns out, teaching a machine to hear emotion is an incredibly difficult task [3].

A key reason SER is so tricky is that people all express emotions in slightly different ways. These personal nuances can be shaped by language, speaking style, and cultural background [4]. On top of this, each emotion appears to leave its own unique acoustic fingerprint in a soundwave; disgust, for example, often shows up in lower frequency bands, while fear might be characterized by higher ones. Trying to build a single model that can navigate all this complexity and pull out the right emotional cues from such a diverse range of sounds is what makes this field so demanding [5][6].

To address complex pattern recognition problems such as this, researchers are increasingly turning to deep learning models. One promising strategy is the Convolutional Recurrent Neural Network (CRNN), an architecture that combines the advantages of two different types of networks. This network uses a Convolutional Neural Network (CNN) to analyze visual representations of sound, such as spectrograms, and automatically identify important features. At the same time, it uses a Recurrent Neural Network (RNN) to

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

track the sequence of these features over time, understanding how they are connected from one moment to the next [7].

Building on that foundation, some interesting refinements have emerged from prior work. For instance, research from Zielonka et al. [8] suggests that using mel-spectrograms, which are tuned to human hearing, might be a better choice for this task. To get a richer understanding of the audio's temporal context, this work also draws on more advanced RNN like Bidirectional Long Short-Term Memory (BiLSTM). These networks have the clever ability to analyze a sequence both forwards and backwards, which appears to help them overcome some of the memory limitations of standard RNN [9][10]. It has also become clear that for any results to be trustworthy, careful attention must be paid to how data is divided between the training and testing phases

Drawing these ideas together, this study sets out to implement and explore a CRNN model that specifically integrates a CNN with a BiLSTM. To give the model a better chance at generalizing to new voices, we also incorporated data augmentation techniques like pitch shifting and time stretching. Ultimately, the goal here is twofold: first, to put this CRNN-BiLSTM architecture to the test using the CREMA-D dataset, and second, to carefully evaluate how effective this approach really is at accurately classifying emotions in speech.

2. RESEARCH METHODOLOGY

This study employs a comprehensive framework comprising several key stages: data collection, preprocessing, augmentation, feature extraction, data preparation, model creation, and evaluation. Figure 1 shows the SER research workflow.

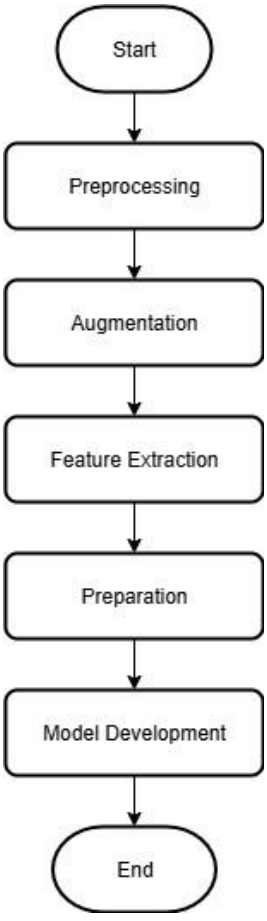


Figure 1. SER research workflow.

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

2.1. Dataset

The research utilizes the Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D). This dataset consists of 7,442 audio clips in English, presented in .wav format, from 91 different actors, with each clip having an approximate duration of 3 seconds. The actors uttered one of twelve predefined sentences. Each sentence was delivered with one of six distinct emotions (Anger, Disgust, Fear, Happiness, Neutral, and Sad) at one of four intensity levels (Low, Medium, High, and Unspecified) [11]. Figure 2 shown class distribution of dataset, while Figure 3 displays sample data from each class as a waveform.

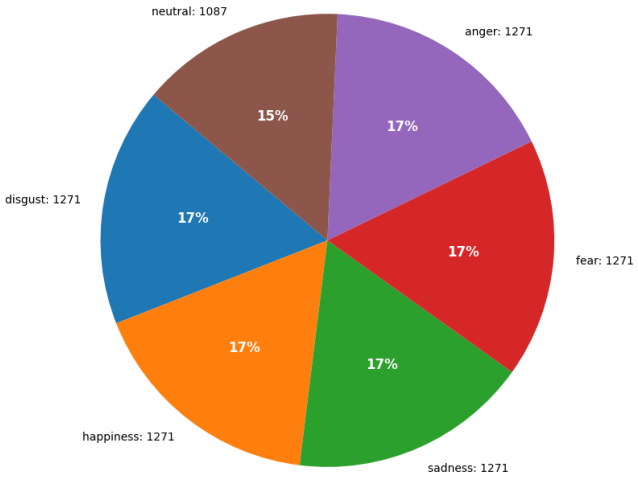


Figure 2. Class distribution.

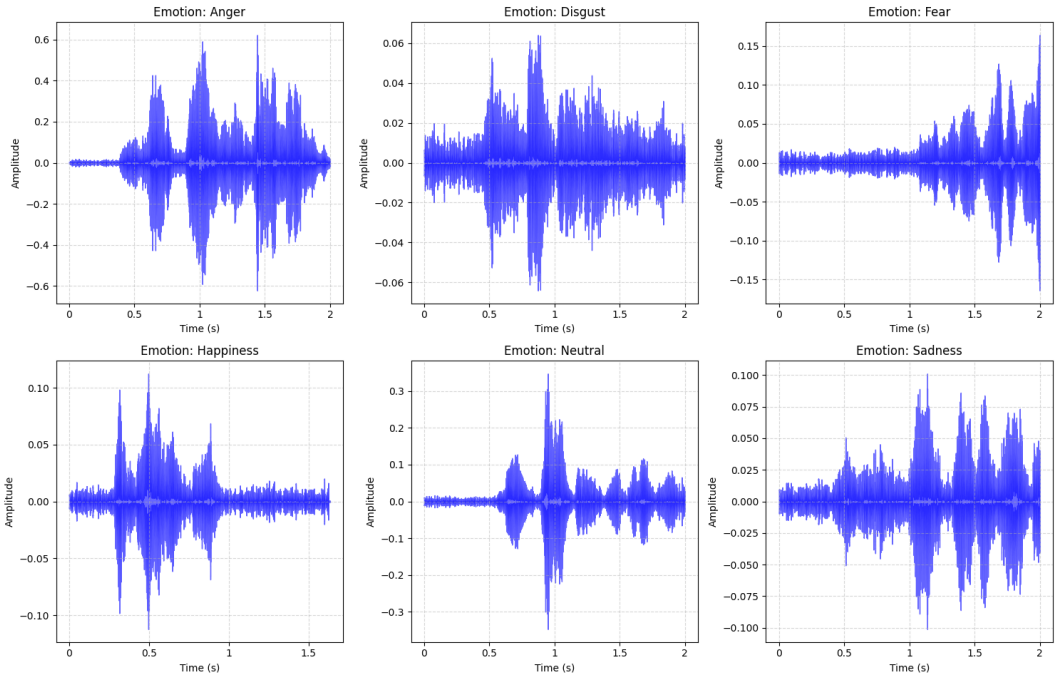


Figure 3. Sample data from each class.

2.2. Implementation Details

This research was conducted using Python programming language, which has strong accessibility and ecosystem, making it suitable for machine intelligence development [12]. Several major libraries were used and played the following roles:

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

a. TensorFlow: designed for high-performance numerical calculations and used to build, compile, and train CRNN-BiLSTM model architectures [13].

b. Librosa: Employed for all audio processing tasks, including loading files, resampling, trimming, and extracting Log-Mel Spectrogram features [14].

c. Scikit-learn: Used to divide the dataset into several parts, validation, and testing sets, as well as for feature normalization and label encoding [15].

d. Matplotlib: Used for data visualization, including displaying audio waveforms, training curves (accuracy and loss), and confusion matrix [15].

e. Pandas & Numpy: Provides a basis for data manipulation. Pandas was used to manage dataset paths and labels, while Numpy was used for numerical operations on audio arrays and feature matrices [15][16].

2.3. Preprocessing

For the model to begin its learning process, the raw audio files first required a fair bit of cleaning and preparation to ensure consistency across the entire dataset. The first step was getting all the audio clips on the same page, so to speak, by resampling every file to a standard sample rate of 44,100 kHz. Following that, any dead air was trimmed away from the recordings. This involved removing the moments of silence at the beginning and end of the files which contain little useful emotional information. A noise reduction process was run to help quiet down some of the unwanted background hiss and hum from the original signals. Final step was duration standardization, where the length of each audio clip was unified to a target of 3 seconds. Clips shorter than the target were zero-padded by appending zeros to the end, while longer clips were truncated to meet the required length. Respectively dataset was divided into three subsets using a 70:20:10 ratio: 70% training, 20% test, and 10% validation. This splitting strategy aims to ensure that the model not only memorizes training patterns but can also generalize effectively to new unseen data. figure 4 displays data before and after preprocessing.

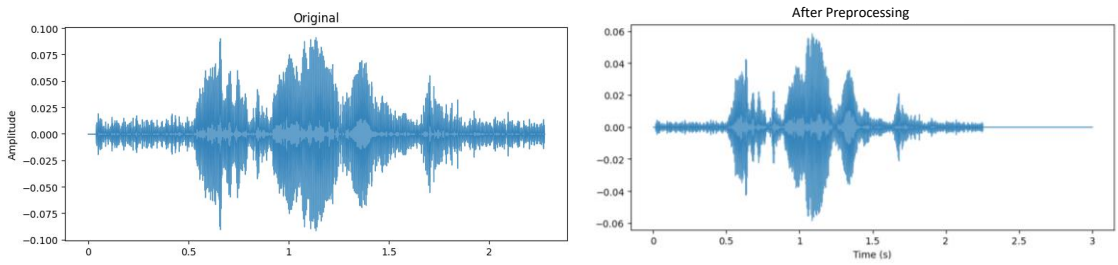


Figure 4. Data before and after preprocessing.

2.4. Data Augmentation

The data augmentation process is applied to increase the quantity and variety of samples in the training data. Its primary goal is to mitigate potential overfitting, a condition where the model excessively memorizes the training data and fails to generalize well to new data. This process is applied only to the training set to maintain the integrity of the validation and test sets [17]. In this study, two data augmentation techniques were used:

a. Pitch Shifting: This technique modifies fundamental frequency (pitch) of audio signal while keeping its duration constant. For this study, pitch of each audio file was raised by 50%. This step aims to simulate natural variations in human vocal characteristics that may occur due to changes in emotion or mood.

b. Time Stretching: This technique gently alters speed of an audio clip without making voice sound higher or lower in pitch. For this particular study, duration of audio files was extended by 10%. An example of this process is shown in Figure 5.

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

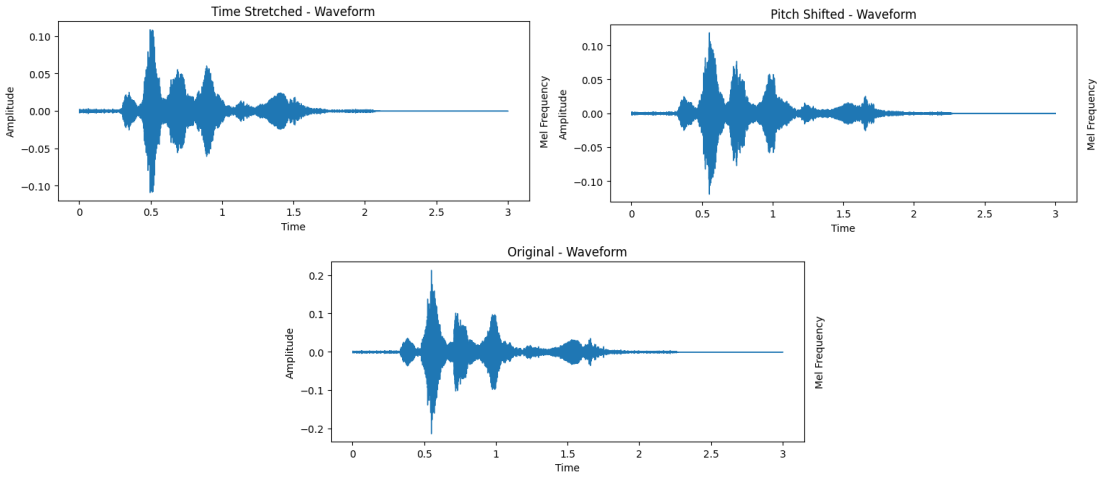


Figure 5. Data augmentation.

2.5. Feature Extraction

A deep learning model cannot directly listen to raw audio, so the first challenge is to translate the processed soundwaves into a more informative and digestible format. For this study, the chosen method was the Log-Mel Spectrogram, a technique selected for its ability to capture acoustic details in a way that appears to be relevant to human hearing. Essentially, this process turns a sound clip into a kind of picture, a visual map of the audio's frequency spectrum as it unfolds over time. Turning a soundwave into one of these spectrograms involves a few key computational steps. The audio signal is first broken down into tiny, overlapping snippets to analyze how its frequency content shifts from one moment to the next using a Short-Time Fourier Transform (STFT). These frequencies are then organized according to the Mel scale, a perceptual scale designed to mimic the human ear by giving more resolution to lower-pitched sounds [18]. Once this Mel spectrogram is formed, its amplitude values are converted to a logarithmic scale. This transformation seems to be quite important, as it helps to stabilize the dynamic range of the sound's energy, resulting in features that are seemingly easier for the model to analyze. What comes out at the end is a two-dimensional array for each audio clip, which then serves as the direct input for the CRNN model [19].

2.6. Data Preparation

With the features extracted, a couple of final steps were needed to get the data ready for the model. The Log-Mel spectrograms first went through a normalization process using a Z-Score technique. This involved calculating the mean and standard deviation across all the feature values and then using them to standardize the data. The idea behind this is to adjust all features to a similar scale to help the model learn more effectively [20]. At the same time, emotion labels that are still in the form of text or objects are converted to a numerical format. To do this, one-hot encoding is used, which converts each emotion object into a simple binary vector that can be understood by the model.

2.7. Model Architecture

The CRNN model architecture proposed in this study is carefully designed to optimize the process of recognizing emotions from voice signals. This model fundamentally combines two main strengths in deep learning, namely the spatial feature extraction capability of CNN and the temporal dependency analysis capability of BiLSTM. The details of the presented architecture are as follows:

a. Convolutional Block (CNN): The first part of the model consists of four convolution blocks that extract features from the log-Mel spectrogram input. The initial layer uses larger kernels to capture general patterns, while the subsequent layers use smaller kernels to detect more detailed features [21]. Each block is also equipped with a MaxPooling layer to reduce dimensions and a Dropout layer for regularization to prevent overfitting [22].

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

b. Recurrent Block (BiLSTM): After the features are extracted by CNN, the data format is changed from spatial to sequence through the Reshape layer so that it can be processed by the BilSTM layer. This data is then processed by two BiLSTM layers with 128 units. The ability of BiLSTM to analyze data from two directions (forward and backward) allows it to capture the temporal context of the entire audio signal more comprehensively [23].

c. Classification Block (Dense Layer): The final part is the classification layer, which makes the final decision. Features from the BiLSTM are processed by a dense layer before being passed to the output layer. This output layer has 6 neurons, corresponding to the number of emotion classes, and uses a Softmax activation function to generate a probability distribution as the final prediction output.

To ensure an optimal and efficient training process, two callbacks were applied. EarlyStopping was used to prevent overfitting by monitoring the validation loss; if no improvement occurred for 15 epochs, training was automatically halted, and the best model weights were restored. ReduceLROnPlateau adjusted the learning rate during training, reducing it by 50% if the validation loss stagnated for 5 epochs. The detailed CRNN model architecture is presented in Table 1.

Table 1. CRNN model architecture.

Layer	Name	Filter	Kernel	Dropout
Conv 1	Conv2D	64	(9, 9)	0.3
	Pool		(2, 2)	
Conv 1	Conv2D	64	(7, 7)	0.3
	Pool		(4, 4)	
Conv 1	Conv2D	128	(5, 5)	0.3
	Pool		(4, 4)	
Conv 1	Conv2D	128	(3, 3)	0.3
	Pool		(4, 4)	
BiLSTM	BiLSTM		128	0.3
BiLSTM	BiLSTM		128	0.3
Dense	Softmax		6	0.4

2.8. Evaluation Metrics

The model's performance evaluation in this study was conducted using a Confusion Matrix. This is a table that presents a comparison between the predicted values generated by the model and the actual values from the data, which effectively measures performance on classification problems. From this confusion matrix, metrics such as Accuracy, Precision, Recall, and F1-score are calculated to provide a comprehensive analysis [24].

Accuracy is the proportion of all correct predictions compared to the total number of samples. Accuracy provides an overview of how often the model predicts the correct class, whether it is positive or negative [24]. Accuracy can be calculated using Equation (1).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision is an evaluation metric that measures how well a model makes correct predictions for positive classes out of the total positive predictions made. Precision helps calculate how often a model correctly predicts positive classes, among all positive predictions made by the model [19]. Precision can be calculated using Equation (2).

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

Recall is an evaluation metric that describes how well a model correctly identifies positive classes [19]. Recall can be calculated using Equation (3).

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1-Score is an evaluation metric that reflects the balance between Precision and Recall. The F1-Score value provides information about how well our model combines Precision and Recall capabilities, allowing to understand how effective our model is in performing classification. [19]. The F1-score can be calculated using Equation (4).

$$F1\text{-}Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (4)$$

3. RESULTS AND DISCUSSION

This section describes the results of the implementation and evaluation of the CRNN-BiLSTM model, which was designed in accordance with the methodology described above. The discussion will focus on analyzing the model's performance during the training process to understand the impact of data augmentation, as well as a comprehensive evaluation of the model's final performance on the test data to measure its effectiveness in classifying speech emotions.

3.1. Model Training

Finding the right set of hyperparameters was a key step in guiding the model's learning process toward a good balance of quality, efficiency, and stable convergence. To give the model enough time to see and learn from the complex patterns in the data, the training was set to run for 100 full cycles, or epochs. During each cycle, the data was processed in small batches of 32 samples at a time, a choice that helps find a middle ground between computational speed and making steady, stable updates to the model's internal weights. For the optimization algorithm itself, the Adam optimizer seemed like a fitting choice given its ability to adapt the learning rate on the fly, which often leads to a smoother and faster convergence. The starting learning rate was kept quite low at 0.0001. This conservative approach encourages the model to make very fine and careful adjustments, reducing the risk of it jumping past the best possible solution. Furthermore, the training was dynamically regulated by two callbacks: EarlyStopping was configured to halt the process if the validation loss did not improve for 15 consecutive epochs to prevent overfitting, and ReduceLROnPlateau was set to decrease the learning rate by 50% if performance stagnated for 5 epochs, allowing the model to find a more refined solution. The complete hyperparameter configuration is detailed in Table 2.

Table 2. Hyperparameter CRNN.

Parameter	Value
Epoch	100
Learning Rate	0.0001
Batch-size	32
Dropout	0.3, 0.4
Optimizer	Adam
Callback	Reduce Learning Rate, Early Stop

For the baseline model trained without augmentation, a significant gap emerged between the training and validation performance. The training accuracy continuously rose to approximately 80%, while the validation accuracy plateaued at around 66.08%. This divergence, coupled with a validation loss that remained high and stagnant while the training loss steadily decreased, is a clear indication of overfitting, a condition where the model memorized the training data but struggled to generalize to new, unseen samples [25]. In contrast, the model trained with augmentation demonstrated markedly improved generalization.

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

The validation accuracy achieved a higher peak of 68.23%, representing a significant improvement over the baseline. More importantly, the gap between the training and validation curves was visibly reduced. While a slight degree of overfitting was still present, the augmented model's ability to generalize was far superior to its non-augmented counterpart. This improvement can be attributed to the success of the augmentation techniques; by exposing the model to variations in pitch and tempo, it was compelled to learn the more robust and fundamental acoustic features of each emotion rather than superficial characteristics of the original audio files. Figure 6 shows the training accuracy and loss graphs.

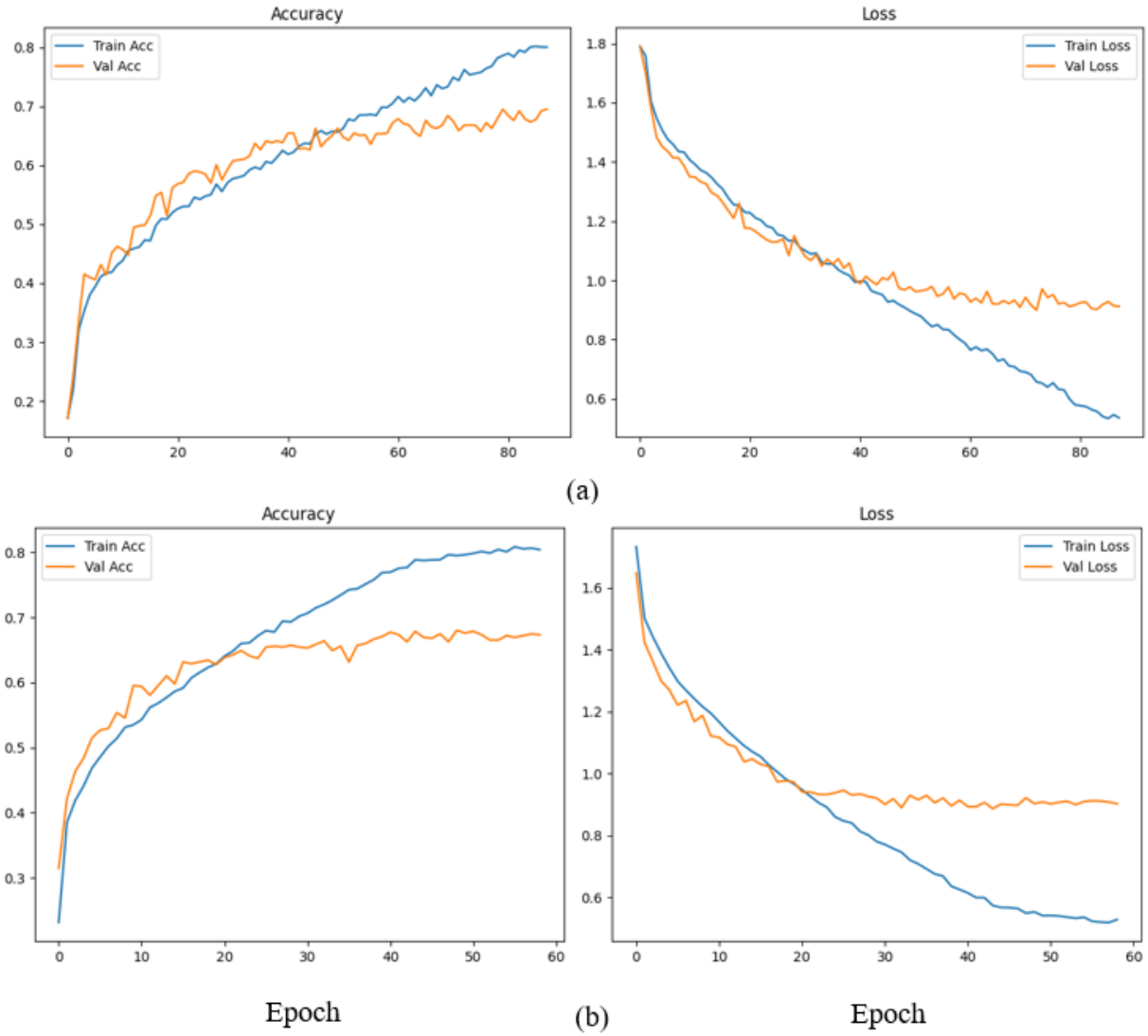


Figure 6. Training accuracy and loss graphs without augmentation (a) and with augmentation (b).

3.2. Model Evaluation

Based on the confusion matrix for the model without augmentation, several classes experienced significant misclassifications. The 'fear' and 'disgust' classes showed considerable classification errors, with many predictions incorrectly classifying these emotions as other classes. Meanwhile, 'neutral' and 'happiness' had more accurate predictions. For the model with augmentation, the confusion matrix shows a considerable improvement in the classification of 'anger' and 'neutral', with fewer errors. The 'anger' class had more correct predictions, and 'neutral' also showed improvement. However, 'fear' and 'sadness' still exhibited some misclassifications, although the number of errors was reduced with augmentation. The model succeeded in reducing classification errors for 'disgust' and 'sadness'. Figure 7 shows confusion matrix for CRNN model.

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

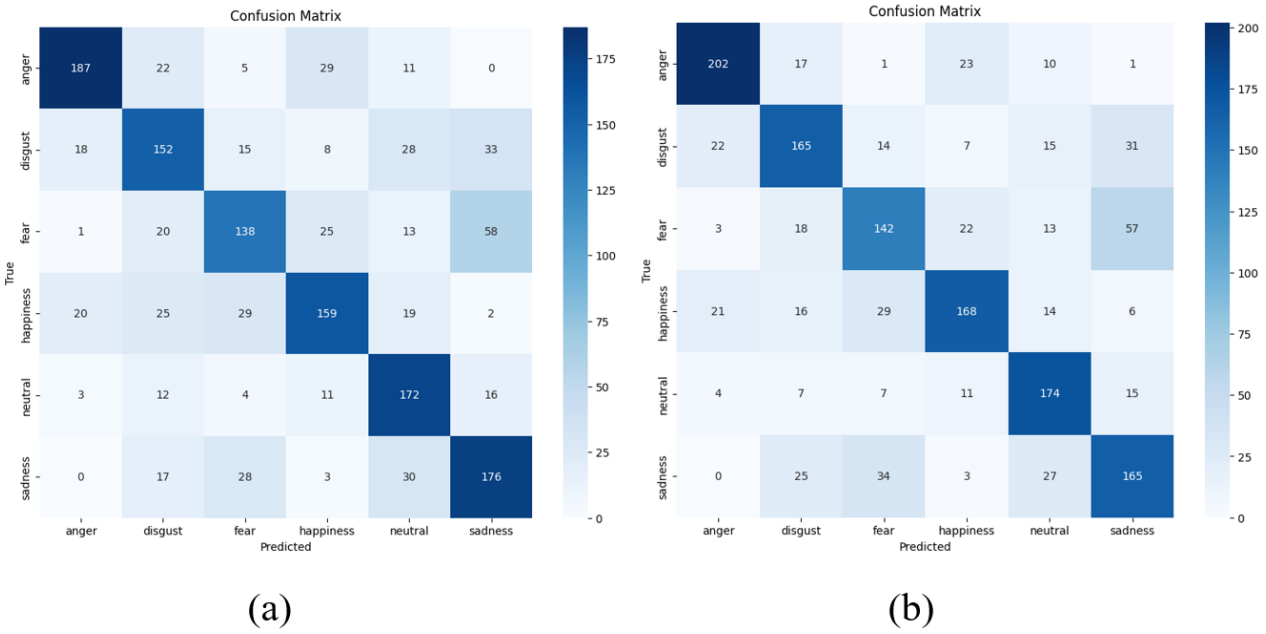


Figure 7. Confusion matrix of CRNN model without augmentation (a) and with augmentation (b).

Overall, the model with augmentation demonstrated better performance compared to the model without augmentation, with an increase in accuracy to 68.23%. Furthermore, augmentation helped reduce classification errors for several emotions, particularly 'anger', 'disgust', 'fear', and 'neutral'. The classification report can be used to calculate accuracy. Accuracy is the proportion of all correct predictions relative to the total number of samples. The classification report can be seen in Table 3.

Table 3. Classification report.

	With Augmented Data			**Without Augmented Data**			**Support**
Class	**Precision**	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**	
Anger	0.82	0.74	0.77	0.80	0.80	0.80	254
Disgust	0.61	0.60	0.61	0.67	0.65	0.66	254
Fear	0.63	0.54	0.58	0.63	0.56	0.59	255
Happiness	0.68	0.63	0.65	0.72	0.66	0.69	254
Neutral	0.63	0.79	0.70	0.69	0.80	0.74	218
Sad	0.62	0.69	0.65	0.60	0.65	0.62	254
Accuracy			**0.6608**			**0.6823**	1489

3.3. Error Analysis

Further analysis of the prediction errors reveals two main issues causing misclassifications. The first issue is that the model sometimes focuses too much on minor details and loses the broader context. For example, an audio clip that was actually neutral and calm was misidentified as 'fear'. This occurred because there was a single, very brief and sharp spike in the audio amidst an otherwise calm signal. The second issue is that some emotions can produce very similar acoustic patterns. For instance, a 'happiness' clip was incorrectly guessed as 'disgust'. Upon inspection, the pattern for laughter ('happiness') produced a waveform with rapid and strong fluctuations, which turned out to be very similar to the pattern of a sharp vocal burst associated with 'disgust'. Figure 8 shows a comparison of correct and incorrect predictions by the CRNN model.

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

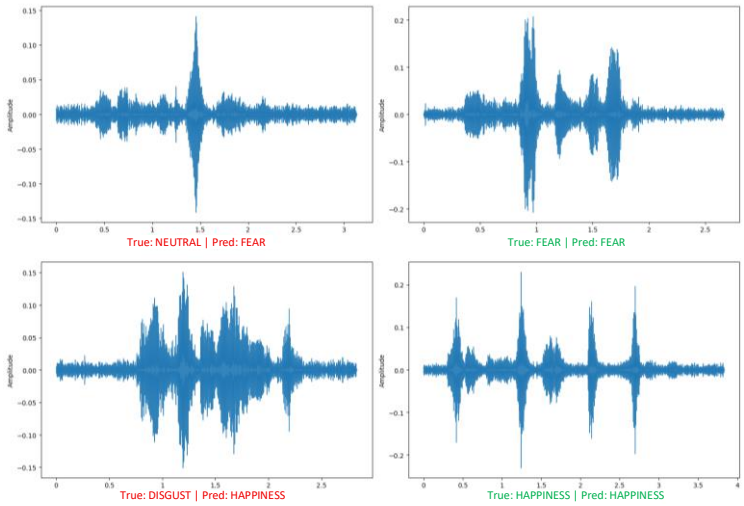


Figure 8. Comparison of correct and incorrect predictions.

Feature extraction using log-Mel spectrograms proved to be effective in capturing essential information from the audio signals, closely mirroring human auditory perception. This representation allows the model to extract spatially relevant patterns for emotions. Additionally, log-Mel spectrograms offer a more compact dimensionality, which reduces computational complexity compared to other techniques. The CRNN model, combining CNN and BiLSTM, was capable of identifying local patterns within the log-Mel spectrograms and understanding temporal context by capturing relationships in sequential data from both directions, delivering a reasonably good performance of 68.23%. However, the model still exhibited overfitting, identified by the significant gap between the training and validation accuracies.

4. CONCLUSIONS

This study successfully implemented an effective CRNN-BiLSTM model for speech emotion recognition, leveraging CNN for feature extraction from log-mel spectrograms and BiLSTM for temporal analysis. The model achieved a validation accuracy of 68.23%, with data augmentation proving essential in enhancing performance, particularly in the classification of 'anger' and 'neutral' emotions. The primary contributions of this study are the empirical validation of the CRNN-BiLSTM architecture's effectiveness and the confirmation of data augmentation's critical role in SER tasks. Future research could focus on exploring more advanced augmentation methods and incorporating additional acoustic features for further improvement.

LITERATURE

[1] M. A. Qamhan, A. H. Meftah, S. Selouani, Y. A. Alotaibi, and Y. M. Seddiq, "Speech emotion recognition using convolutional recurrent neural networks and spectrograms," *in Proc. IEEE Canadian Conf. Electr. Comput. Eng.*, 2020, pp. 1-6.

[2] M. Gokilavani, "Ravdness, Crema-D, Tess based algorithm for emotional recognition using speech," *in Proc. 4th Int. Conf. Smart Syst. Inventive Technol.*, 2023, pp. 1658-1664.

[3] K. A. Mukarram, M. A. Mukhlas, and A. Zahra, "Enhancing speech emotion recognition with deep learning using multi-feature stacking and data augmentation," *Bull. Electr. Eng. Informatics*, vol. 13, no. 3, pp. 1920-1926, 2024.

[4] S. Premjeet, Md. Sahidullah, and S. Goutam, "Modulation spectral features for speech emotion recognition using deep neural networks," *Speech Commun.*, pp. 53-59, 2022.

[5] S. H. Hawley, "Fourier transforms, audio engineering, and the quantum nature of reality," *in Proc. AES 135th Convention*, 2013, pp. 1-4.

© 2026 The Authors | doi: 10.23960/pepadun.v7i2.287

[6] D. A. Sauter, F. Eisner, A. J. Calder, and S. K. Scott, "Perceptual cues in non-verbal vocal expressions of emotion," *Q. J. Exp. Psychol.*, vol. 63, no. 11, pp. 2251–2272, 2010.

[7] K. Markou, L. Tsochatzidis, K. Zagoris, A. Papazoglou, X. Karagiannis, S. Symeonidis, and I. Pratikakis, "A convolutional recurrent neural network for the handwritten text recognition of historical Greek manuscripts," *Lect. Notes Comput. Sci.*, pp. 227-239, 2021.

[8] M. Zielonka, A. Piastowski, A. Czyzewski, P. Nadachowski, M. Operlejn, and K. Kaczor, "Recognition of emotions in speech using convolutional neural networks on different datasets," *Electronics*, vol. 11, no. 3831, pp. 1-12, 2022.

[9] K. S. Nugroho, I. Akbar, A. N. Suksmawati, and Istiadi, "Deteksi depresi dan kecemasan pengguna Twitter menggunakan bidirectional LSTM," *in Proc. 4th Conf. Innov. Appl. Sci. Technol. (CIASTECH 2021)*, 2021, pp. 287-296.

[10] N. T. Pham, D. N. M. Dang, N. D. Nguyen, T. T. Nguyen, H. Nguyen, B. Manavalan, C. P. Lim, S. D. Nguyen, "Hybrid data augmentation and deep attention-based dilated convolutional-recurrent neural networks for speech emotion recognition," *Expert Syst. Appl.*, vol. 230, pp. 1-13, 2023.

[11] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "CREMA-D: Crowd-sourced emotional multimodal actors dataset," *IEEE Trans. Affect. Comput.*, vol. 5, no. 4, pp. 377-390, 2014.

[12] S. S. Mantrala, "Python's pivotal role in AI and data science," *Int. J. Inf. Technol. Manag. Inf. Syst.*, vol. 16, no. 2, pp. 1676-1686, 2025.

[13] A. Pavithra and S. M. Prakash, "TensorFlow in deep learning," *Int. J. Innov. Res. Technol.*, vol. 5, no. 9, pp. 142-147, 2019.

[14] T. G. Patil and A. V. Zade, "To design and develop advanced speech emotion recognition using MLP classifier with evolutionary LIBROSA library," *Int. J. Multidiscip. Res.*, vol. 5, no. 3, pp. 1-6, 2023.

[15] J. M. E. Diana and A. Mathivanan, "Exploring the paradigm shift: Harnessing data analytics for real-world applications," *Int. J. Sci. Res.*, vol. 12, no. 6, pp. 1467-1480, 2023.

[16] F. Reiss, B. Cutler, and Z. Eichenberger, "Natural language processing with Pandas DataFrames," *in Proc. 20th Python Sci. Conf.*, 2021, pp. 49-57.

[17] F. Makhmudov, A. Kutlimuratov, and Y. -I. Cho, "Hybrid LSTM–attention and CNN model for enhanced speech emotion recognition," *Appl. Sci.*, vol. 14, no. 23, pp. 1-19, 2024.

[18] D. Vazhenina and K. Markov, "End-to-end noisy speech recognition using fourier and hilbert spectrum features," *Electronics*, vol. 9, no. 1157, pp. 1-18, 2020.

[19] O. P. Yadav, L. P. Bastola, and J. Sharma, "Speech emotion recognition using convolutional recurrent neural network," *in Proc. 10th IOE Graduate Conf.*, 2021, pp. 1164-1172.

[20] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput. J.*, pp. 1-23, 2019.

[21] Y. Lee, "The CNN: The architecture behind artificial intelligence development," *J. Stud. Res.*, vol. 12, no. 4, 2023.

[22] T. Bezdan and N. Bacanin, "Convolutional neural network layers and architectures," *in Proc. Sinteza 2019 - Int. Sci. Conf. Inf. Technol. Data Relat. Res.*, vol. 4, 2019, pp. 445-451.

[23] P. Meyer, Z. Xu, and T. Fingscheidt, "Improving convolutional recurrent neural networks for speech emotion recognition," *in Proc. IEEE Spoken Lang. Technol. Workshop*, 2021, pp. 1-9.

[24] B. J. Hutagaol and T. Mauritsius, "Risk level prediction of life insurance applicant using machine learning," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, no. 2, pp. 2213-2220, 2020.

[25] S. Pothuganti, "Review on over-fitting and under-fitting problems in machine learning and solutions," *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.*, vol. 7, no. 9, pp. 3692-3695, 2018.