Artificial Intelligence Review (2025) 58:49
https://doi.org/10.1007/s10462-024-11065-x

# Real-time speech emotion recognition using deep learning and data augmentation

**Chawki Barhoumi**[1,2] · **Yassine BenAyed**[2]

Accepted: 30 November 2024 / Published online: 20 December 2024


**Abstract**
In human–human interactions, detecting emotions is often easy as it can be perceived through facial expressions, body gestures, or speech. However, in human–machine interactions, detecting human emotion can be a challenge. To improve this interaction, Speech Emotion Recognition (SER) has emerged, with the goal of recognizing emotions solely through vocal intonation. In this work, we propose a SER system based on deep learning approaches and two efficient data augmentation techniques such as noise addition and spectrogram shifting. To evaluate the proposed system, we used three different datasets: TESS, EmoDB, and RAVDESS. We employe several algorithms such as Mel Frequency Cepstral Coefficients (MFCC), Zero Crossing Rate (ZCR), Mel spectrograms, Root Mean Square Value (RMS), and chroma to select the most appropriate vocal features that represent speech emotions. Three different deep learning models were imployed, including MultiLayer Perceptron (MLP), Convolutional Neural Network (CNN), and a hybrid model that combines CNN with Bidirectional Long-Short Term Memory (Bi-LSTM). By exploring these different approaches, we were able to identify the most effective model for accurately identifying emotional states from speech signals in real-time situation. Overall, our work demonstrates the effectiveness of the proposed deep learning model, specifically based on CNN+BiLSTM enhanced with data augmentation for the proposed real-time speech emotion recognition.

**Keywords** Emotion · Deep learning · Speech emotion recognition · Multilayer perceptron · Convolutional neural network · Recurrent neural network

✉ Chawki Barhoumi
chawki.barhoumi@enetcom.u-sfax.tn

Yassine BenAyed
yassine.benayed@isims.usf.tn

[1] National School of Electronics and Telecommunications of Sfax, 3027 Sfax, Tunisia

[2] Multimedia, InfoRmation systems and Advanced Computing Laboratory, 3027 Sfax, Tunisia

## 1 Introduction

The human–machine relationship is constantly evolving, particularly through the optimization of communication between the two. Much effort is expended in order for us to interact with machines as easily and intuitively as possible. Human–Computer Interaction (HCI) seeks to create not only a more efficient and natural communication interface between humans and computers, but also to improve the overall user experience, assist human development, and enhance e-learning, among other goals. Emotions are an inherent part of human interactions, so they have naturally become an important aspect of developing HCI-based applications Arguel et al. (2019). Due to the complexity of emotional reactions, the term ”emotion” was challenging to define. The issue of emotion is complicated and encompasses mental, physiological, physical, and behavioural factors. So, in robots and HCI, the emotional component has been developed and taken into account Khalil et al. (2019). Emotions can be captured and assessed using technology in a variety of ways, including facial expressions, physiological signals, and speech. One of the most difficult tasks in the field of speech signal analysis is automatic SER. It is a research problem that tries to deduce emotion from speech signals. The emotions conveyed by the signals must be correctly detected and appropriately processed in order to create a more natural and intuitive communication between humans and computers.

SER is used in a wide variety of applications. Anger detection can be used to improve the quality of voice portals or call centers Abbaschian et al. (2021).It enables to adapt the services provided according to the emotional state of the clients. Monitoring the stress of aircraft pilots can also help reduce the rate of a potential aircraft accident in civil aviation. In the field of entertainment, many researchers have integrated an emotion recognition module in their products in order to improve the experience of gamers in video games and maintain their motivation. The goal is to increase player engagement by adjusting the game based on their emotions. In the sector of mental health care, a psychiatric counseling service with a chatbot has been proposed. The basic concept is to analyze the input text, voice, and visual cues to determine the subject’s psychiatric disorder and provide information about diagnosis and treatment Oh et al. (2017). A conversational chatbot is another suggested application for emotion recognition, where identifying vocal emotions can help with a better conversation Yenigalla et al. (2018). Therefore, SER has become not only a challenging topic but also a critical research area.

The performance of SER has improved in various directions, including data collection, data segmentation, data augmentation, feature extraction, and classifier design. A crucial component of SER is the extraction of features. Speech is a continuous, variable-length signal that carries both information and emotion. Therefore, depending on the technique used, global or local information can be extracted. Global features, also known as long-term or supra-segmental features, represent numerical data over longer time intervals. On the other hand, local features, often referred to as segmental or short-term features, capture the temporal dynamics of speech over shorter time intervals. Akçay and Oğuz (2020). In the literature, four types of local and global features have been used in SER systems, which are:

- Prosodic Features Zeng et al. (2009)
- Spectral Features Koolagudi and Rao (2012)
- Voice Quality Features Cowie et al. (2001)

- Teager Energy Operator (TEO) Based Features Teager and Teager (1990) and Kaiser (1990)Prosodic and spectral features are more commonly used in SER systems. Also, TEO-based features are specifically designed to recognise stress and anger Akçay and Oğuz (2020).

After extracting features, they are used as input for the classification step in SER. There are many methods and algorithms that can be used as classifiers. Each method attempts to solve the problem from a specific angle and has its own advantages and disadvantages. Historically, most emotion recognition methods were based on classical machine learning algorithms, including Hidden Markov Models (HMM), Gaussian Mixture Models (GMM), k-Nearest Neighbor (KNN), and Support Vector Machines (SVM). However, in recent years, the trend has shifted towards methods based on deep learning. Several algorithms have been used, such as RNN, Long Short-Term Memory (LSTM), CNN, and Autoencoders (AE). These deep learning algorithms have been shown to outperform classical machine learning algorithms in SER, especially when large amounts of data are available for training. Akçay and Oğuz (2020).

In this paper, we present multiple accurate approaches for SER and evaluate their performance in a real-time environment, which is a highly challenging task in signal processing and speech analysis. We employ advanced spectral feature extraction algorithms such as MFCC, Chroma, Mel Spectrogram, ZCR, and RMS to better capture the emotional content of speech. Additionally, we propose a powerful hybrid deep learning model that combines CNN and Bidirectional Bi-LSTM architectures as a classifier. Moreover, we investigate other classifier architectures, including MLP and CNN, to improve classification accuracy. Our study demonstrates the effectiveness of these approaches, specifically our proposed approaches based on CNN+BiLSTM, providing valuable insights into SER system development. We validate the effectiveness of our methods using three standard speech emotion datasets: TESS (Toronto Emotional Speech Set) Pichora-Fuller and Dupuis (2020), RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song) Livingstone and Russo (2018), and EmoDB (Berlin Emotional Speech Database) Burkhardt et al. (2005). To improve the quality of our datasets, we employ two commonly used data augmentation techniques: noise addition and spectrogram shifting. Noise addition involves adding random noise to the input data, which can help to prevent overfitting and improve generalization performance. Spectrogram shifting, on the other hand, involves randomly shifting the time axis of the spectrogram of an audio signal. This can help to make the model more robust to variations in the timing of different sound events, which is important in many real-world applications such as speech recognition or music classification. By using these techniques, we can create larger and more diverse training datasets that can improve the accuracy and robustness of our proposed machine learning models. The remainder of this paper is organised as follows:

- In Sect. 2, we present a brief review of recent related works.
- The proposed approaches, as well as the system architectures, are then explained in the next Sect. 3.
- Our experiments are highlighted in the Sect. 5. We describe the tools used first, then reveal our proposed method.
- In Sect. 4, we describe the mechanism we used to apply the system in real-time sce-

narios.
- We present and discuss the results in the Sect. 5.3.
- Finally we make a comparative study in the last Sect. 6

## 2 Related works

SER has been the focus of numerous studies in the literature. In this section, we look over a few of these research projects and highlight their successes. Specifically, we discuss studies that have explored various approaches to feature extraction, classification models, and datasets used for SER evaluation.

In SER, feature extraction techniques play a crucial role in achieving high accuracy in classification. Researchers have explored a range of feature extraction techniques and classification models to improve SER performance Chen et al. (2012); El Ayadi et al. (2011). For example, prosodic features such as pitch and intonation can significantly influence classification accuracyBänziger and Scherer (2005). Additionally, time-domain features such as pitch and energy, as well as frequency-domain features such as spectrograms, have been studied extensively. Spectral features are often preferred since they reflect how energy is distributed over the frequency range and are therefore directly linked to emotionsMeng et al. (2019). Schuller et al Schuller et al. (2009). investigated the applicability of MFCCs, along with their first and second derivatives, to SER. They found that the use of these features resulted in improved classification accuracy. In another study, Zheng, Yu and Zou Zheng et al. (2015). proposed a model that employed spectrograms as features for SER classification and reported successful outcomes. The most widely used spectral feature in automatic SER is MFCC. It is a well-established method for analysing emotional content in spoken language Gupta et al. (2018). As an example, the SVM algorithm was used by Pratama and Sihwi Pratama and Sihwi (2022). to create a speech emotion recognition model using the MFCC voice feature obtained from voice processing. This proposed approach achieved an accuracy of 70%. In another work, the contribution described in Bhandari et al. (2022) was to evaluate an approach based on LSTM network models as a classifier using the MFCC features extracted from the RAVDEES datset. Only 4 emotion categories were employed (happy, neutral, sad, and angry). This approach achieved 89% accuracy. To enhance the performance of SER systems, some researchers combined various feature types and used them as input for classification models. As an example, Four different types of characteristics (MFCC, ZCR, HNR, and TEO) were combined by Hadhami Aouani et al. Aouani and Ben Ayed (2020), and the findings were promising. In another work, Lanjewar, Mathurkar and Patel Lanjewar et al. (2015) proposed a SER system using MFCC features, wavelets, and height.They achieved a recognition rate of 66%.

In the work of Aljuhani et al. (2021). a study on emotion recognition based on spoken data in Saudi Arabian dialectal Arabic was conducted. The dataset was created from freely available YouTube videos and consisted of four types of emotions (anger, happiness, sadness, and neutral). Various spectral features, such as MFCCs and Mel spectrograms, were extracted. These features were then used as inputs for the KNN classifier. The experiments showed that the KNN achieved an overall accuracy of 57.14%, demonstrating an improvement in Arabic speech emotion recognition for this classification method.

Kaur and Kumar (2021). compared several methods of recognising emotions from vocal signals using machine learning algorithms such as KNN, MLP, CNN, and random forest (RF). Initially, Short-Term Fourier Transform spectrograms (STFT) and MFCC coefficients were extracted from the EmoDB dataset. The spectrograms were used as input for the CNN, while the MFCC features were used by the KNN, MLP, and random forest classifiers. Each classifier showed satisfactory results in classifying seven emotions (happiness, sadness, anger, neutral, disgust, boredom, and fear), but the MLP classifier was the most successful with an overall accuracy of 90.36%.

Nam and Lee (2021). proposed a new architecture consisting of two steps. The DnCNN (cascaded denoising CNN) was used to denoise the data in the first step. Then, the CNN performed the classification. The results for real datasets showed that the proposed DnCNN-CNN outperformed the basic CNN in terms of overall accuracy for both languages. For the Korean speech dataset, the proposed DnCNN-CNN achieved an accuracy of 95.8%, while the CNN accuracy was slightly lower (93.6%). These results demonstrated the feasibility of applying DnCNN for speech denoising and the effectiveness of the CNN-based approach in recognising speech emotions.

Kwon (2020). proposed a SER system that used hierarchical blocks of convolutional long and short-term memory (ConvLSTM). They designed four ConvLSTM blocks, called local feature learning blocks (LFLBs), to extract emotional features. Then, they tested the proposed system on two standard speech datasets: Interactive emotional dyadic motion capture (IEMOCAP) and RAVDESS. Finally, they obtained a recognition rate of 75% and 80%, respectively.

A nonlinear multi-level feature generation model based on cryptographic structure was created by Tuncer, Dogan and Acharya Tuncer et al. (2021). RAVDESS, Emo-DB, SAVEE, and EMOVO are four speech emotion datasets that were used to validate the model's performance. Using a 10-fold cross-validation technique, the presented model achieved classification accuracy on these datasets of 87.43%, 90.09%, 84.79%, and 79.08%, respectively.

Sowmya et al. (2022). tried to implement a SER system that can recognise four different types of feelings from the RAVDESS dataset, like happy, neutral, sad, and angry. In addition, SVM, MLP, RF, and decision tree calculations were utilised for classification. MFCC, chroma, tonnetz, mel, and contrast were the elements extracted to characterise the feelings. Then the dataset was split into 75% for training and 25% for testing. When compared to others, MLP produced the best results with an accuracy of 85%.

Prabhakar et al. (2023). developed a multichannel Convolution Neural Network-Bidirectional Long Short Term Memory (CNN-BLSTM) architecture with an attention mechanism for speaker-independent SER by considering phase and magnitude spectrum-based features. The phase-based features were extracted using the Modified Group Delay Function (MODGD). The obtained phase features were then combined with the MFCC features. In addition, The IEMOCAP database was employed for performance analysis. The experimental results showed improvements over MFCC features and existing approaches for unimodal SER.

Hama Saeed (2023). proposed a SER approach based on DNN. this approach is divided into three stage,feature extraction, normalization, and emotion recognition. MFCC, Mel-Spectrogram Frequency, Chroma, and Poly Features were extracted from the audio signals. Data augmentation for the minority class using SMOTE and the Min–Max scaler for the normalization process were used. The DNN model was evaluated on three frequently used

languages: German, English, and French, using the EMODB, SAVEE, and CaFE speech datasets. this approach achieve good accuracy result within the three datasets with 95% on EMODB, 90% on SAVEE, and 92% on CaFE.

The study proposed by Alluhaidan et al. (2023) combined MFCCs with time-domain features (MFCCT). By leveraging CNNs, the hybrid MFCCT features demonstrated remarkable efficacy, surpassing both MFCCs and time-domain features across benchmark datasets like Emo-DB, SAVEE, and RAVDESS, achieving accuracies of 97%, 93%, and 92%.

In Table 1, we present a summary of studies on SER. Specifically, we present the aim and proposed solution of each study, as well as the pros and cons. This comprehensive overview allows for a critical analysis of the current state-of-the-art in SER, highlighting the strengths and limitations of various approaches.

Based on the studies summarized in Table 1, our research proposes an efficient SER system based on deep learning methodologies and effective data augmentation techniques. Drawing insights from prior studies such as those by Chen et al. (2012); El Ayadi et al. (2011), Schuller et al. (2009), and Nam and Lee (2021), we address limitations observed in traditional feature extraction methods and single-model architectures. In contrast, our approach integrates multiple deep learning models, including MLP, CNN, and a hybrid CNN with Bi-LSTM, inspired by the successes demonstrated in the works of Kaur and Kumar (2021) and Kwon (2020), to effectively capture and interpret vocal features indicative of speech emotions. Moreover, we employ advanced data augmentation techniques such as noise addition and spectrogram shifting, inspired by methodologies outlined in the studies by Alluhaidan et al. (2023) and Lanjewar et al. (2015), to enhance the robustness and generalizability of our model. By evaluating our system on diverse datasets (including those used in prior studies, such as TESS, EmoDB, and RAVDESS) with diverse emotional categories, we demonstrate its efficacy in accurately identifying emotional states from speech signals in real-time scenarios.

## 3 Proposed methodology

The proposed SER methodology is explained in Sect. 3. It includes the following steps: Data collection (3.1), data preparation (3.2), feature extraction (3.4), development of models based on deep learning techniques (3.5), learning and testing of the created models, and finally classification. In our proposed methodology for SER, we employ three models: MLP, CNN, and CNN+BiLSTM. Furthermore, five different feature algorithms: MFCC, ZCR, Mel Spectrogram, RMS, Chroma, are used as input to improve the performance of our proposed models. Figure 1 represents the overall architecture of the proposed SER system.

Each part of the proposed system is covered in detail in the rest of this section, from data collection 3.1 to classification 3.5.

### 3.1 Data collection

In this research, three datasets are included in the experiments conducted. These datasets are introduced in the following.

**Table 1** Summary of Studies on SER

| Aim and proposed solution | Pros | Cons |
|---|---|---|
| Improve SER performance through feature extraction and classification models Chen et al. (2012); El Ayadi et al. (2011): Exploration of various feature extraction techniques (e.g., prosodic, time-domain, frequency-domain) | Comprehensive analysis of multiple feature types for improving SER accuracy | Low recognition rates (68.5% and 50.2%) |
| Enhance classification accuracy in SER using MFCC features Schuller et al. (2009): Use of MFCCs and their first and second derivatives for SER | Improved classification accuracy with detailed feature analysis | Limited to MFCC features; performance in real-world scenarios not discussed |
| Employ spectrograms for SER classification Zheng et al. (2015): Spectrogram-based feature extraction for SER | Successful classification outcomes | Study limited to spectrograms; potential overfitting not addressed |
| Develop SER model using MFCC features Pratama and Sihwi (2022): SVM algorithm with MFCC voice feature | Achieved 70% accuracy | Accuracy could be improved; limited emotion categories |
| Evaluate LSTM network models for SER using MFCC features Bhandari et al. (2022): LSTM classifier with MFCC features extracted from RAVDEES dataset | High accuracy of 89% with limited emotion categories | Limited to four emotion categories |
| Combine various feature types for SER Aouani and Ben Ayed (2020): Combination of MFCC, ZCR, HNR, and TEO features for classification | Promising findings with feature combination approach | The experiment is based on a single RML dataset and one SVM classier |
| Implement SER using MFCC, wavelets, and height features Lanjewar et al. (2015): Combined feature extraction approach | Achieved a recognition rate of 66% | Lower recognition rate compared to other models |
| Compare various ML algorithms for SER Kaur and Kumar (2021): Comparison of KNN, MLP, CNN, and RF classifiers using STFT and MFCC features | MLP achieved highest accuracy of 90.36% among the compared methods | Variation in performance across different classifiers |
| Enhance SER using DnCNN for denoising and CNN for classification Nam and Lee (2021): Two-step architecture with DnCNN for denoising followed by CNN for classification | Achieved high accuracy of 95.8% for Korean dataset | Complex architecture requiring denoising step |
| Develop SER system with hierarchical ConvLSTM blocks Kwon (2020): Use of ConvLSTM blocks for emotional feature extraction | Good recognition rates of 75% (IEMOCAP) and 80% (RAVDESS) | Complexity in model design; Precision needs to be improved |
| Create nonlinear multi-level feature generation model for SER Tuncer et al. (2021): Cryptographic structure-based multi-level feature generation model | High classification accuracy on multiple datasets with 10-fold cross-validation | Model complexity; potential difficulty in real-world implementation |
| Implement SER system for four emotion types using various classifiers Sowmya et al. (2022): SVM, MLP, RF, and decision tree classifiers using MFCC, chroma, tonnetz, mel, and contrast features | MLP achieved highest accuracy of 85% among the compared methods | Limited to four emotion types; varying performance among classifiers |
| SER approach based Hama Saeed (2023) on DNN: MFCC, Mel-Spectrogram Frequency, Chroma, and Poly Features were extracted from the audio signals | High accuracy across multiple datasets and languages | Simple DNN model is used |
| SER apporach based on CNN Alluhaidan et al. (2023): combining MFCCs with time-domain features (MFCCT) | This method creates hybrid MFCCT features and uses them as input for the CNN | The evaluation is conducted using a one widely-used model |

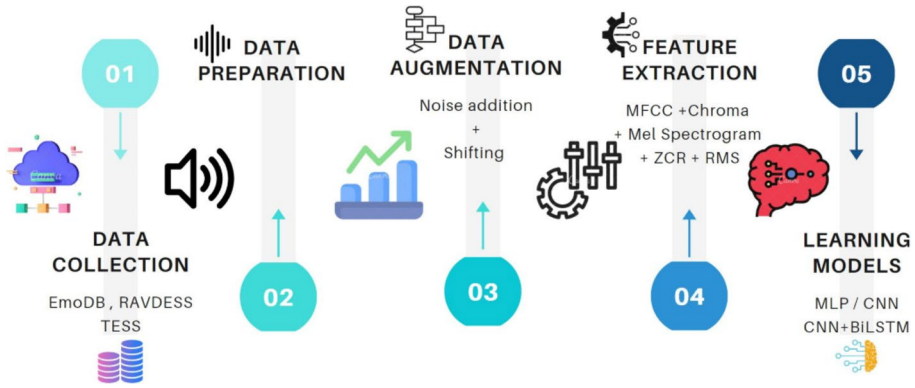



**Fig. 1** Overall architecture of the proposed SER system


**Fig. 2** TESS emotion categories

### 3.1.1 TESS

TESS is a set of 200 target words that were spoken in the carrier sentence ”Say the word a word” by two actresses (ages 26 and 64) and recordings were made of the set describing each of the seven emotions: angry, disgusted, fearful, happy, surprised, sad, and neutral. It contains 2,800 data samples (audio files) in total. The Fig. 2 describes the exact number of samples in each category:

### 3.1.2 RAVDESS

The RAVDESS database is a complete dataset of speech and song, audio and video (24.8 GB). We work with a portion of RAVDESS that contains 1440 files: 60 trials per actor multiplied by 24 actors equals 1440. The RAVDESS contains 24 professional actors (12 female, 12 male), vocalising two lexically matched statements in a neutral North American accent. The speech emotions include the expressions: calm, happy, sad, angry, fearful, surprised,

disgusted, and neutral. Each expression is produced at two levels of emotional intensity (normal and strong), with an additional neutral expression. Each of the 1440 files has a unique file name. In Fig. 3, we present the different categories of emotions in the RAVDESS dataset.

### 3.1.3 EmoDB

The Department of Technical Acoustics, Technical University of Berlin, collected the data for this dataset. An anechoic chamber is used for the recording. In this dataset, 535 utterances are present. The statements that were captured showed the feelings of disgust, boredom, fear, anxiety, anger, happiness, and neutrality. The statements are said by individuals who are in their 20 s to 30 s. Each utterance is named according to the same scheme: Positions 1–2: number of speakers, Positions 3–5: code for the text, Position 6: Emotion (the letter represents the German word for emotion), Position 7: if there are more than two versions, these are numbered a, b, c, etc. Example of a file (.wav): ”03a01Fa.wav” is the audio file of speaker 03 saying the text a01 with the emotion ”Freude” (happiness). The representation of the different categories of emotions is illustrated in Fig. 4):

## 3.2 Data preparation

In this important step, we did:

- Data loading: The datasets consist of samples of data (.wav). We cannot use this format as input for the proposed machine learning models. So we start by loading the data samples and transforming them from an audio file format into a time series representation.

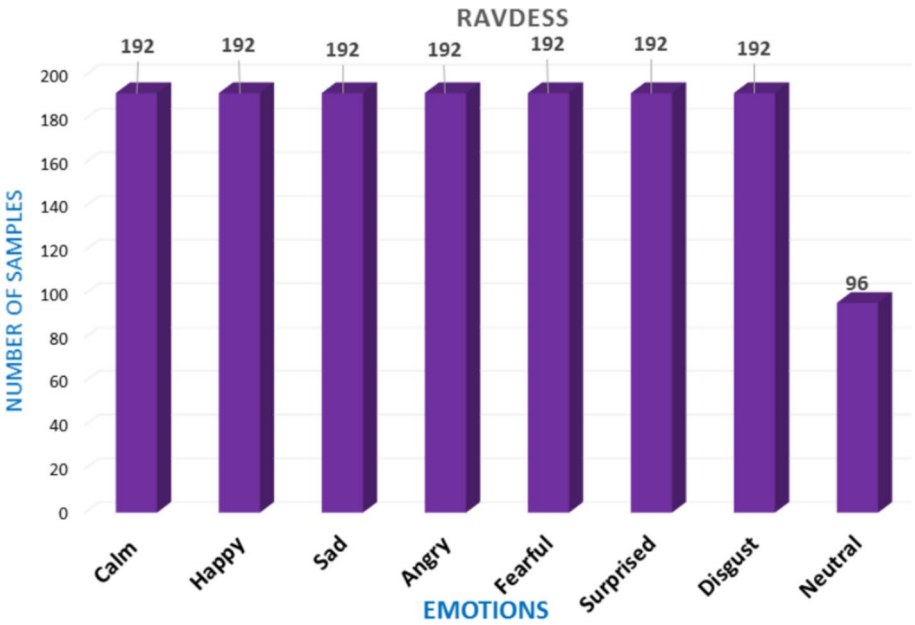



**Fig. 3** RAVDESS emotion categories

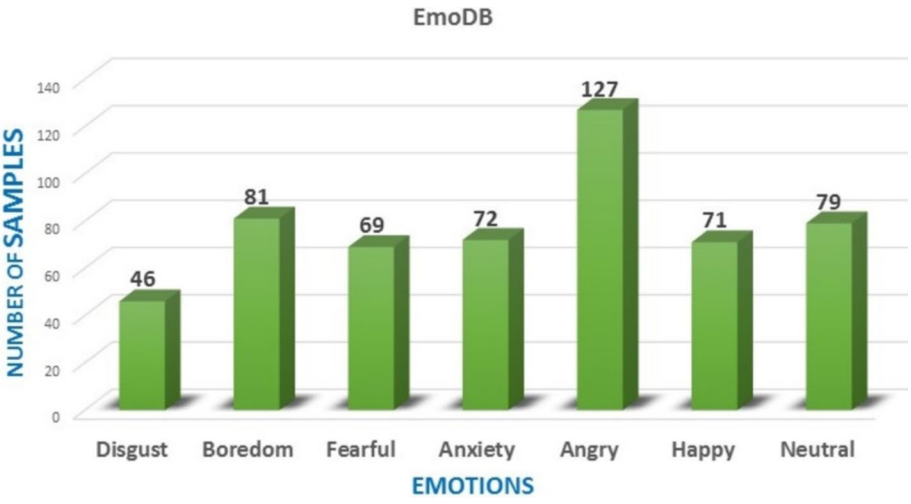



**Fig. 4** EmoDB emotion categories

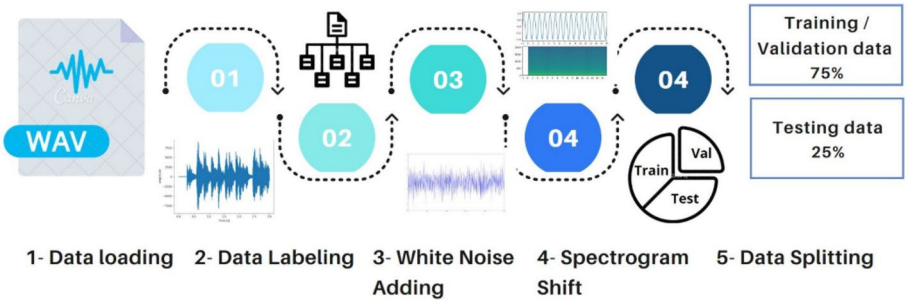



**Fig. 5** Presentation of the data preparation phase

For this, we use librosa, which is a Python package for analysing and processing audio signals, and the librosa.load() function to load the audio files and convert them to a time series representation.

- Data labelling: Also known as data annotation, is crucial for our supervised approach to SER as it helps improve the accuracy and efficiency of the proposed machine learning models. It consists of giving a label for each sample in the datasets. For example, we assign a number (1) to each sample that describes the emotion ”happy,” and so on.
- Data Augmentation: We applied two data augmentation techniques, namely noise addition and spectrogram shift, to enhance the content of the datasets used in this work. This important aspect of our study is explained in detail in Sect. 3.3
- Data splitting: In this last step of data preparation, each dataset is split into 75% for training/validation and 25% for testing.In Fig. 5, we summarize all the steps we performed as a data preparation phase.

### 3.3 Data augmentation

In this section, we present an approach that focuses on leveraging data augmentation techniques to enhance the diversity and robustness of the dataset for SER. Through the introduction of controlled variations such as speed perturbation, noise addition, and spectrogram shifting, our goal is to expose the model to a broader range of speech patterns, thereby improving its generalization performance. The primary objective is to overcome limitations associated with insufficient train- ing data. Data augmentation introduces controlled variations, ensuring that the model encounters a more extensive array of speech patterns during training. This process equips the model to better handle real-world variations in emotional expressions Chen et al. (2020). To diversify the dataset, we applied the following data augmentation techniques:

#### 3.3.1 White noise addition

By adding white noise, we aim to improve the generalization of the proposed model for real-world scenarios. In real-world conditions, speech signals are rarely clear and often contain background noise. Training our deep learning model with augmented, noisy data helps it to accurately detect emotional expressions in such real-world environments Abdelhamid et al. (2022).

By precisely controlling the SNR value to 20, we enhance the generalization capability of the proposed model for real-world scenarios. In real-world conditions, speech signals are frequently marred by background noise, leading to diminished clarity. By training our deep learning model with augmented, noisy data featuring a controlled SNR, we ensure its robustness in accurately detecting emotional expressions in diverse environments. The white noise, as described in Algorithm 1, is tailored to maintain a targeted SNR of 20. This controlled introduction of noise ensures that the model learns to discern emotional cues amidst realistic noise levels, thus bolstering its adaptability.

To calculate white noise, we first need to determine the power of the original signal, which is the mean of the squared signal values. This is calculated using the following equation Huang et al. (2019):

$$P_{\text{signal}} = \frac{1}{N} \sum_{i=1}^{N} s[i]^2 \tag{1}$$

Next, we calculate the power of the noise, which is the signal power divided by a factor that depends on the desired signal-to-noise ratio (SNR). The noise power is given by Huang et al. (2019):

$$P_{\text{noise}} = \frac{P_{\text{signal}}}{10^{\frac{\text{SNR}_{\text{dB}}}{10}}} \tag{2}$$

Finally, the noise is calculated using the following equation Huang et al. (2019):

$$\text{noise} = \sqrt{P_{\text{noise}}} \times \text{randn}(N) \tag{3}$$

where:

- $P_{\text{signal}}$ is the power of the original signal.
- $N$ is the number of samples in the signal.
- $s[i]$ is the value of the signal at the $i$-th sample.
- $P_{\text{noise}}$ is the power of the noise.
- $\text{SNR}_{\text{dB}}$ is the desired signal-to-noise ratio in decibels.
- $\sqrt{P_{\text{noise}}}$ is the standard deviation of the noise.
- $\text{randn}(N)$ is a vector of $N$ samples from a standard normal distribution (mean of 0 and standard deviation of 1).

**Ensure:** Data ← ndarray, audio time series
1: **Input:** Signal *signal*, Desired SNR *snr_db*
2: Signal_Power ← `numpy.mean`($signal^2$)
3: Noise_Power ← `Signal_Power` / $10^{snr\_db/10}$
4: Noise ← `numpy.sqrt`(Noise_Power) × `numpy.random.randn(len(signal))`
5: Augmented_Signal ← signal + Noise
6: **return** Augmented_Signal, Noise

**Figure 1**

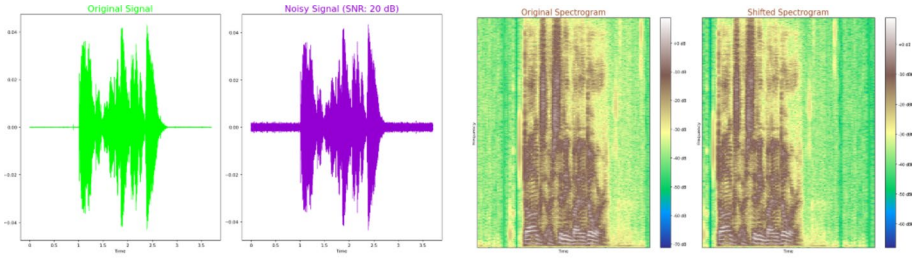
**Algorithm 1** Data augmentation: white noise adding

The suggested deep learning model's generalisation is significantly improved by the addition of this amount of noise to the clean signal. Yet, because there are clean samples in the dataset, the model can recognise the spoken emotions in both a clean signal and a noisy signal.

### 3.3.2 Sprectrogram shift

A shifting spectrogram 2 examines how a spoken signal's frequency content evolves over time. A spectrogram is a graphic representation of a signal's frequency spectrum as it evolves over time. In our work we shift the spectrogram in the right direction Han et al. (2017).

In SER, the timing of the speech signal can vary slightly depending on the emotion being expressed. For example, a speaker may speak more quickly or slowly when they are angry or sad compared to when they are neutral or happy. By shifting the spectrogram along the time axis, we can simulate these variations and create new training examples that are slightly different from the original data. This can help the model to learn more robust features that are invariant to small timing differences, and improve its performance on unseen data. In Fig. 6a and b, we present the results of applying white noise addition and shifting techniques respectively.




(a) Noise addition result (b) Spectrogram shift

**Fig. 6** Results of white noise addition and spectrogram shift

**Table 2** Number of samples before and after the application of data augmentation

| Dataset | Original data size (.wav) | Data augmentation | New data size (.wav) |
|---|---|---|---|
| EmoDB | 535 | 535 + 535 (Noise adding) + 535 (Shifting) | 1605 |
| RAVDESS | 1440 | 1440 + 1400 (Noise adding) + 1400 (Shifting) | 4320 |
| TESS | 2800 | 2800 + 2800 (Noise adding) + 2800 (Shifting) | 8400 |

**Ensure:** Data ← ndarray, audio time series
1: **Input:** Signal *signal*, Desired SNR *snr_db*
2: Signal_Power ← `numpy.mean`($signal^2$)
3: Noise_Power ← `Signal_Power` / $10^{snr\_db/10}$
4: Noise ← `numpy.sqrt`(Noise_Power) × `numpy.random.randn(len(signal))`
5: Augmented_Signal ← signal + Noise
6: **return** Augmented_Signal, Noise

**Ensure:** Data ← ndarray, audio time series
1: Shift-max ← 0.25 : maximum shift rate
2: Shift-direction ← 'right'
3: Shift ← numpy.random.randint(sampling-rate × Shift-max)
4: Shift ← (− Shift)
5: Augmented-data ← data + Shift
6: **return** Augmented-data

**Algorithm 2** Data augmentation: shifting

The results of the application of these two algorithms (Noise Addition and Shifting) are presented in Table 2.

### 3.4 Feature extraction

The goal of feature extraction is to reduce the dimensionality of the input data, while retaining as much information as possible.

#### 3.4.1 Mel frequency cepstral coefcients

The most commonly used speech feature is undoubtedly the Mel Frequency Cepstral Coefficients (MFCC), which are the most popular and robust due to their accurate estimation of speech parameters and their efficient calculation model Selvaraj et al. (2016). The MFCC technique represents speech signals by converting their short-term power spectrum into a linear cosine transform of the logarithmic power spectrum on a nonlinear Mel scale of frequency, as illustrated in Fig. 7.

The Mel scale is a unit of measure that reflects the perceived pitch or frequency of a signal. MFCC provides enhanced frequency resolution in the low frequency range, making it suitable for various types of signals and unaffected by noise. The significant advantage of MFCC is that it is based on human hearing perception, which cannot sense frequencies above 1 kHz Gupta et al. (2018); Koduru et al. (2020). There are seven steps in MFCC. The following details these actions.

1. Pre-emphasis: The low-frequency region of the speech signal contains more energy than the high-frequency region. In order to enhance the energy of the high-frequency region and eliminate noise, a pre-emphasis process is implemented, as follows:

$$A[k] = B[k] \times 0.97 \times B[k-1] \tag{4}$$

where $A[k]$ is output signal and $B[k]$ is input signal.

2. Framing: Speech is an astable signal. To obtain the signal's quasi-stationary state, the signal is framed by dividing it into segments of 25 ms length with a 50% overlap.

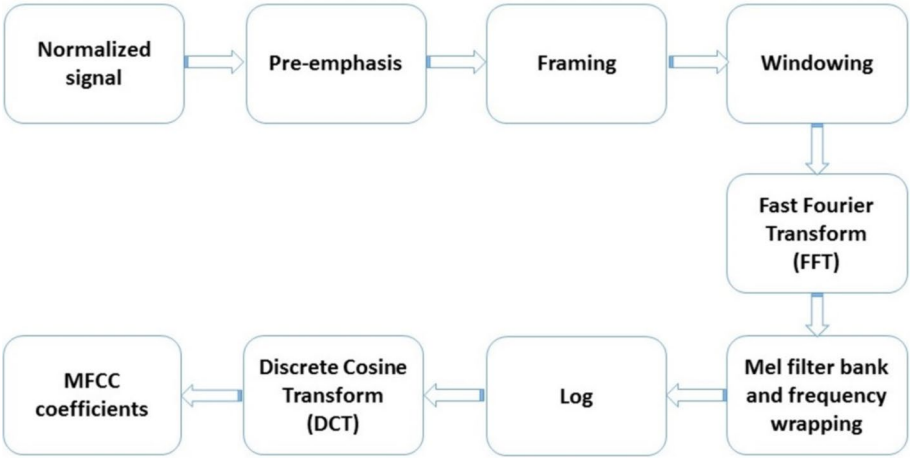



**Fig. 7** Representation of Mel frequency cepstral coefcients

3. Windowing: This process is used to reduce the discontinuities in frames and spectral distortion at the start and end of the signal. Windowing is applied to each individual frame of the signal. The hamming window technique is used since it reduces side lobes and gives an accurate frequency spectrum. And it is given by:

$$X(k) = 0.54 + 0.46\cos\left(\frac{2\pi k}{N-1}\right) \quad 0 \leq k \leq N-1 \tag{5}$$

where N is the number of samples in each frame $X(k)$.

4. Fast Fourier Transform: FFT is used to transform the windowing signal into the frequency domain and obtain the signal's spectrum, and it is provided by:

$$Y(l) = \sum_{m=0}^{M-1} y(m)e^{-i(\frac{2\pi}{M})ml} \quad (l = 0, 1, ..., M-1) \tag{6}$$

where $Y(l)$ represents frequency domain samples, $Y(m)$ represents time domain samples, M represents FFT size, and m is $0, 1, 2, ...(M-1)$.

5. Mel filter bank and frequency wrapping: At frequencies under 1000 Hz, the Mel-frequency scale has linear spacing, whereas for frequencies above, it has logarithmic spacing. A mel is a unit that is used to express the signal's pitch or frequency. The mels is calculated as follows for a given frequency (Hz):

$$F(mel) = 2595 \times \log_{10}(1 + \frac{F}{100}) \tag{7}$$

F is the speech signal frequency in Hz. The Mel filter bank comprises of triangular filters that overlap each other. The cutoff frequency of every filter is determined by the center frequencies of two adjacent filters. The filters have a constant bandwidth on the mel scale and equally spaced center frequencies on a linear scale.

6. Log: The logarithme converts multiplication to addition. This means that it converts magnitude multiplication in Fourier transformation to addition. After the fusion banques, the logarithm is used to calculate the amplitude values. MFCC are the result of a cosine transform of the real logarithm of the short-term energy spectrum expressed on a mel frequency scale. Mel-frequency can be calculated using the formula:

$$Mel(f) = 2595 \times \log 10(1 + \frac{f}{700}) \tag{8}$$

7. Discrete Cosine Transform: DCT is used to obtain the time domain of the signal from the logarithmic Mel spectrum. The DCT outputs the MFCC Coefficients.

### 3.4.2 Mel specgtrogram

A Mel spectrogram, also called a mel-frequency spectrogram, is a graphic representation of an audio signal's frequency content. It is frequently used to analyse the acoustic characteristics of sound in speech and music processing. The Mel spectrogram shows the frequency content of an audio signal over time in a manner similar to a conventional spectrogram, but it does so using a non-linear frequency scale that more accurately corresponds to how people hear sounds. The Mel scale is a set of pitches that listeners perceive to be equally spaced apart from one another. The audio signal is first split into brief time intervals, typically between 10 and 50 ms, in order to construct a Mel spectrogram. A Fourier transform is calculated for each time frame to produce a frequency representation of the signal. An auditory system-inspired filterbank is used to convert the obtained spectrum into the Mel scale Kwak and Kim (2019). The magnitude of each Mel frequency bin is then plotted against time to create the Mel spectrogram. As a result, a 2D representation of the frequency content of the audio signal is produced, with time represented by the x-axis and frequency represented by the y-axis on the Mel scale Jiang et al. (2019).

### 3.4.3 Zero crossing rate

The Zero Crossing Rate (ZCR), is a fundamental property of a signal that refers to the number of times the signal changes direction between positive, negative, and zero values. It is a useful feature for detecting short and loud sounds in a signal and for identifying small changes in the signal's amplitude. ZCR is commonly used in speech processing applications, including speech synthesis, speech enhancement, and speech recognition, as it can help determine the presence of human speech in a given audio sample Bachu et al. (2008).

Furthermore, the zero crossing count can serve as an indicator of the frequency at which energy is concentrated in the signal spectrum. By analyzing the distribution of ZCR values over time, it is possible to evaluate the spectral properties of the speech signal, which can provide insights into the overall frequency content and energy distribution of the signal. Therefore, ZCR is an essential tool for analyzing speech and other types of audio signals in various domains, including speech analysis and audio processing. ZCR is given by:

$$ZCR = \frac{1}{2M} \sum_{k=1}^{M} |sign(s(k)) - sign(s(k-1))| \tag{9}$$

where $M$ is size and $sign(a[k]) = \begin{cases} 1 & \text{k}>0 \\ -1 & \text{k}<0 \end{cases}$

### 3.4.4 Root mean square

The Root Mean Square (RMS) value is a significant method used to calculate the average of values over a period of time. In the case of audio, the signal value (amplitude) is squared, averaged over a period of time, and then the square root of the result is calculated. The result is a value that, when squared, is proportional to the effective power of the signal Tariq et al. (2019). For a set of numbers or values from a discrete distribution $x_1, \ldots, x_n$, RMS is the square root of the mean of the values. its is given by:

$$RMS = \frac{1}{n}(\sum_{i=1}^{n} x_i^2)^2 \tag{10}$$

RMS is a commonly used feature in SER because it provides information about the energy or loudness of the speech signal. The RMS value can be calculated for short-time windows of the speech signal, typically in the range of 20–50 ms. The RMS values for these short-time windows can be used to characterize the changes in loudness or energy over time, which can be indicative of changes in emotional content Bhangale and Kothandaraman (2023).

#### 3.4.5 Chroma

The Chroma features, also called "pitch class profiles", features are an interesting and powerful representation of musical sound where the spectrums are projected into 12 different boxes representing the 12 different halftones of the musical octave Alnuaim (2022). This feature extraction module is used to generate a chromagram, which is constructed using the twelve pitch classes. The variation of these pitch classes during the audio allows us to produce 12 additional features belonging to the different pitch classes. The pitch classes in the specific order are: C, C#, D, D#, E, F, F#, G, G#, A, A#, B Garg et al. (2020).

### 3.5 Proposed models

We're currently focused on exploring learning approaches that utilize both classical and deep learning techniques. After completing data preparation and feature extraction, our goal is to create models capable of automatically predicting the sentiment of future audio. Three models are proposed in this section, which are: MLP (3.5.1), CNN (3.5.2), and CNN+BiLSTM (3.5.3).

#### 3.5.1 Multi-layer Perceptron

A Multi-Layer Perceptron (MLP) is a neural network that connects multiple layers in a directed graph, which means that the signal path through the nodes (or neurons) can only go in one direction. Each node, except for the input nodes, has a non-linear activation function. An MLP uses backpropagation as a supervised learning technique and is composed of three types of layers as showen in Fig. 9: an input layer, a hidden layer, and an output layer Krishna et al. (2022).

The input layer receives the data as input and multiplies it by weights to generate the second layer, known as the hidden layer. Within the hidden layer, there are activation functions that enable the model to learn nonlinear data. Following the activation function, the calculation is passed through each of the hidden layers until it reaches the final output layer. The output of a node in a MLP can be computed using the following equation Alnuaim (2022):

$$y = f(Wx + b) \tag{11}$$

where: $x$: is a vector of inputs to the node. $W$: is a matrix of weights connecting the node to the previous layer. $b$: is a vector of bias terms. $f$: is the activation function applied element-

wise to the weighted sum.$(Wx + b)$ . $y$: is the output of the node.The logistic activation function, also known as the sigmoid function, is used in each neuron of the hidden layers and output layer of an MLP. The function takes the weighted sum of the inputs to the neuron, and maps the result to a value between 0 and 1. This means that the output of the neuron will be close to 0 if the weighted sum is negative, and close to 1 if the weighted sum is positive. The sigmoid function is defined as follows:

$$sigmoid(x) = \frac{1}{(1 + e^{(-x)})} \tag{12}$$

We use Adam (Adaptive Moment Estimation) as our optimization algorithm, with a batch size of 128 and a hidden layer size of 256. The architecture of this first model is presented in Fig. 8:

### 3.5.2 Convolutional neural network

To understand speech emotions, researchers have one key challenge: the extraction of the most distinctive features that accurately represent emotions. Based on the existing methods of feature extraction, speech features can be categorised as either studied features or hand-crafted features. Deep neural networks, such as CNNs, offer a simple way to extract features that can achieve exceptional performance Kim et al. (2013).

A typical CNN model comprises multiple convolutional layers, pooling layers, fully connected layers, and a Softmax classification layer. The fundamental components of a CNN are the convolution layers, which use filters to perform convolution operations on the input data. The output of the convolution layers is then passed to the pooling layers, whose primary goal is to reduce the output resolution and computational load. The resulting output is then fed to the fully connected layer, where it is flattened and classified by the Softmax unit.

Our proposed CNN-based model is composed of four Local Feature-Learning Blocks (LFLBs). Each LFLB includes a convolutional layer followed by an activation layer (relu). Only in the last LFLB do we add a max pooling layer with a kernel size of 8. In the first LFLB, we adjust a number of filters of 16 for the convolutional layer, and we multiply it by two each time, as illustrated in Table 3. finally, after flattening the data from the max pooling layer, we apply a dense layer using the Softmax activation function to classify the processed features.

- Convolution layer: The convolution layer is the most important operation in CNN, but

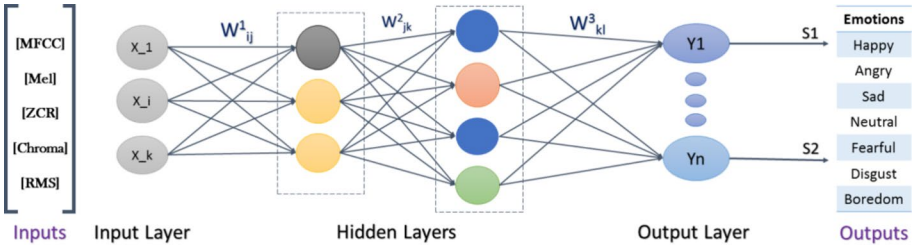



**Fig. 8** MLP model architecture

**Table 3** Summary of the model architecture

| Layer type | Output shape | Number of parameters | Activation |
|---|---|---|---|
| Conv1D | (182, 16) | 96 | ReLU |
| Conv1D | (182, 32) | 2592 | ReLU |
| Conv1D | (182, 64) | 10,304 | ReLU |
| Conv1D | (182, 128) | 41,088 | ReLU |
| MaxPooling1D | (22, 128) | 0 | N/A |
| Flatten | (2816) | 0 | N/A |
| Dense | (7) | 19,719 | Softmax |
| Total parameters | 76616 | | |
| Trainable parameters | 76616 | | |
| Non-trainable parameters | 0 | | |

it also takes longer than other operations. Its main role is to extract the most relevant features from the input extracted features. This is achieved by using filters or kernels, which are small squares of size (e.g., $3 \times 3$, $5 \times 5$, $7 \times 7$), that are slid over the input features to compute the scalar product and generate a "features map". This map is also known as the "activation map" or "convoluted features". In this layer, it's important to choose the filter size and stride. The filter size determines the spatial extent of the kernel, while the stride determines the step size of the kernel as it moves over the input shape. If the stride is not chosen correctly, the neurons do not fit the input in a clean and symmetrical way, which can lead to suboptimal results. To address this issue, padding can be used on the entire feature map to adjust the neurons in a clean and symmetrical way. Overall, the convolution layer is critical for CNNs because it allows them to extract important features that are useful for making accurate predictions. However, choosing the right filter size, stride, and padding is essential for achieving optimal performance Sawardekar and Naik (2018).

- Relu layer: Activation functions can be applied to a neural network through an activation layer or via the activation argument supported by a convolution layer. In our proposed model, we choose to use the "relu" activation function as an activation layer. This function replaces all negative values in the filtered features with zero, ensuring that the sum of the values does not equal zero. By doing so, the activation function enhances the network's ability to learn and generalize, leading to improved performance. Abdullah and Abdulazeez (2021). The output of the Relu layer is given by:

$$f_r = Relu(x_i) = max(0, x_i) \tag{13}$$

- Pooling layer: The pooling layer plays a crucial role in CNNs by reducing the spatial dimension of the feature map. This operation results in a more compact representation of the features, which can lead to better performance and faster training. The output of the pooling layer is a pooled feature map that contains more relevant features and is smaller in size compared to the input feature map. There are several pooling techniques available, but two of the most widely used are max pooling and mean pooling. In our proposed model, we use max pooling, which selects the maximum value from each pool, as it has been shown to be effective in preserving the most important features of the input data. Wang et al. (2017).
- Classification layer: Softmax is an activation function commonly used in deep learning

to classify the feature vectors generated by a fully connected layer into different classes. After the fully connected layer, the Softmax function converts the output vector of numerical values into a probability distribution, where each element in the vector represents the probability of the input belonging to a certain class. By doing so, the Softmax function helps to normalize the output vector and ensure that the sum of all probabilities is equal to 1. This makes it easier to interpret the output and make decisions based on the predicted class probabilities. It is defined as follows:

$$\sigma(\overrightarrow{z}_i) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{x_j}} \tag{14}$$

The Softmax function is represented by $\sigma$. $\overrightarrow{z}$ represent the input vector,$e^{x_i}$ exponential function for the input vector; K is the number of classes; and $e^{x_j}$ is exponential function for the output vector. The proposed CNN model is more explained in Fig. 9.

### 3.5.3 Hybrid model (CNN+BiLSTM)

Speech is a complex signal with two main aspects that make up its structure: the textual sequential aspect, which refers to the numerical values of the signal, and the temporal aspect, which describes how content is distributed over time. To take both aspects into account and achieve accurate speech analysis, we develop a model that combines two powerful techniques: a CNN and a Bi-LSTM network (Table 4).

The CNN is well-known for its ability to extract the best features of the signal through its convolution layers and max pooling, while the Bi-LSTM network focuses on the chronological aspect of speech and is specifically designed for processing long sequences. By combining these two techniques, our hybrid model can effectively process and analyze speech data, taking into account both the textual and temporal aspects of speech.

Our proposed model consists of three parts: the convolutional part, the BiLSTM part, and the classification part.

The convolutional part is designed to extract useful features from the input data. It employs four LFLBs, each consisting of a convolutional layer followed by a relu activation layer. The first LFLB has 16 filters, while the subsequent LFLBs have twice as many filters

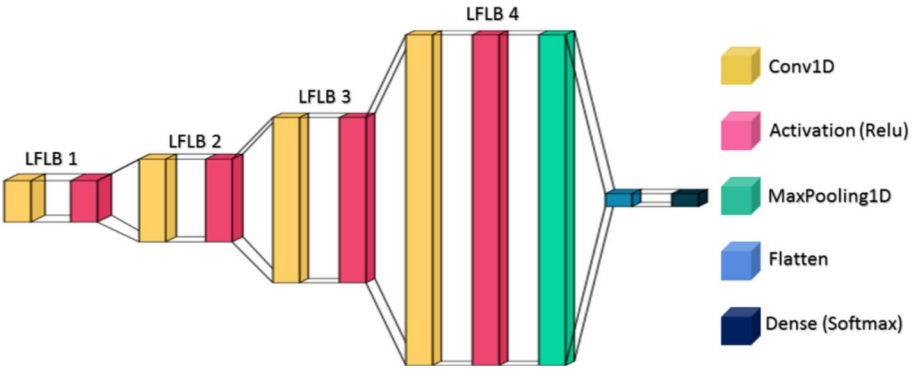



**Fig. 9** CNN model architecture

**Table 4** Summary of the Model Architecture

| Layer Type | Output shape | Number of parameters | Activation |
|---|---|---|---|
| Conv1D | (182, 16) | 96 | ReLU |
| Conv1D | (182, 32) | 2592 | ReLU |
| Conv1D | (182, 64) | 10,304 | ReLU |
| Conv1D | (182, 128) | 41,088 | ReLU |
| MaxPooling1D | (26, 128) | 0 | N/A |
| TimeDistributed(Flatten) | (26, 128) | 0 | N/A |
| Bidirectional(LSTM) | (512) | 788,480 | N/A |
| Dense | (128) | 65,664 | ReLU |
| Dense | (64) | 8256 | ReLU |
| Dense | (7) | 455 | Softmax |
| Total parameters | 916,935 | | |
| Trainable parameters | 916,935 | | |
| Non-Trainable parameters | 0 | | |

as the previous one. The last LFLB includes a max pooling layer with a kernel size of 8, which helps to reduce the spatial dimension of the feature maps.

The BiLSTM part is used to capture the temporal dynamics of the input data. It consists of a single layer with 256 units, followed by two dense layers of sizes 128 and 64, respectively. The output of the BiLSTM part is a feature vector that summarizes the temporal information of the input data.

The classification part is responsible for predicting the emotion category of the input data. It uses a dense layer with a size equal to the number of emotion categories, followed by a Softmax activation function, which normalizes the output scores and provides a probability distribution over the emotion categories.

Overall, our proposed model is capable of extracting both spatial and temporal features from the input data, and using them to make accurate predictions about the emotional content of the data. The proposed architecture is more detailed in Fig. 10.

## 4 Real-time SER system

Developing a deep learning model for recognition, includion SER, is a complex task that requires more than just achieving high recognition rates and good evaluation metrics. To truly validate a proposed SER system, it must be tested in a real-time environment. This involves testing the system using live audio streams, which provides a more accurate representation of how the system would perform in the real world. The results of this real-time test can determine whether the proposed system operates correctly and effectively in real time. Although achieving a high learning rate (100%) is impressive, it's essential that the proposed SER system works correctly in real-time scenarios. Therefore, the success of the proposed system depends on its ability to operate accurately and effectively in a real-time environment.

As shown in Fig. 11, the first step is to input an audio stream. Next, we load the pretrained deep learning model based on CNN+BiLSTM architecture, which has demonstrated superior performance in terms of accuracy, precision, recall, and other relevant metrics (which we present in the next section). After loading the model, we apply the same process

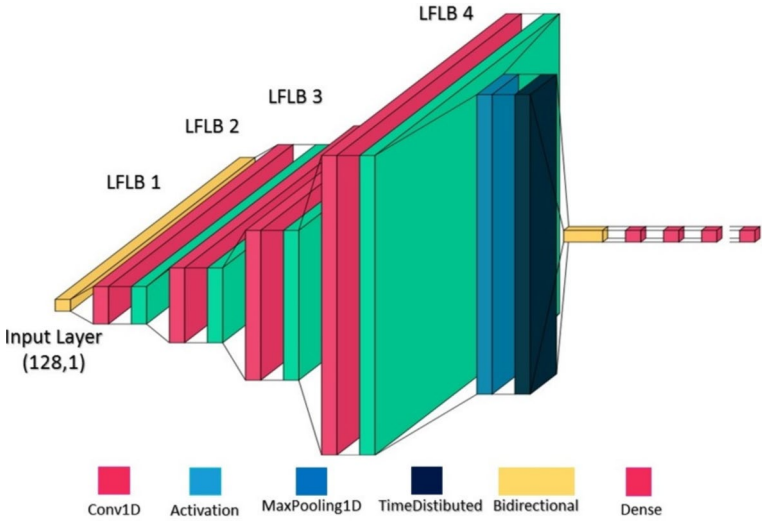



**Fig. 10** CNN+BiLSTM model Architecture

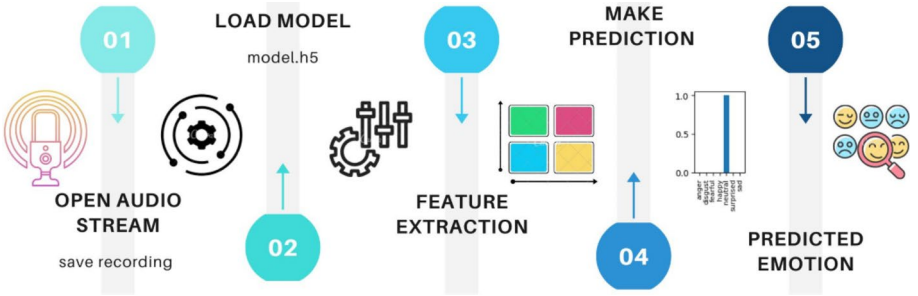



**Fig. 11** Real-time speech emotion recognition process

of feature extraction to the saved recordings. Finally, we use the model to make predictions based on the extracted features, which allows us to obtain the predicted emotion.

**Ensure:** Data ← ndarray, audio time series
Shift-max ← 0.25 : maximum shift rate
Shift-direction ← 'right'
Shift ← numpy.random.randint(sampling-rate × Shift-max)
Shift ← (− Shift)
Augmented-data ← data + Shift
**return** Augmented-data

**Ensure:** label map = [0: 'anger', 1: 'boredom', 2: 'disgust', 3: 'fear', 4: 'happiness', 5: 'sadness', 6: 'neutral']
**Inputs:** audio recording
**Outputs:** predicted emotion category
**Open recording**: audio, sr = get-audio()
**Save the recording**: wavfile.write('recording.wav', sr, audio)
**Load pre-trained model**: model = load-model('path/to/model.h5')
**Read the recording**: data, sr = librosa.load('recording.wav')
**Extract features**: features = 'extract-feature'(data)
**Make a prediction**: prediction = model.predict(features)
**Get the predicted class**: predicted-class = argmax(prediction)
**Get the predicted emotion category**: predicted-category = label-map[predicted-class]
**return** predicted emotion category

**Algorithm 3** Real-time speech emotion recognition

In Algorithm 3, we present more details for the different steps of the real-time SER process. This algorithm takes an audio recording as input that describes an emotional state and returns the predicted class of the recording.

## 5 Experimental results

### 5.1 Tools and packages

To implement the proposed approaches in this paper, we used the following tools and packages:

- Python: a programming language widely used in machine learning and data science.
- Google Colab: an online platform that provides a Jupyter notebook environment for running code and sharing notebooks.
- Scikit-learn, TensorFlow, and Keras: machine learning libraries that provide high-level

APIs for building and training deep neural networks.
- NumPy: a powerful mathematical library for working with arrays, matrices, and mathematical functions.
- Librosa: a signal processing library specifically designed for music and audio analysis.By using these tools and packages, we were able to develop and test our proposed approaches efficiently and effectively.

### 5.2 Evaluation metrics

We must evaluate the performance of our deep learning model after it has been constructed and trained. Precision, recall, accuracy, and F-score are four separate criteria used to categorise the SER evaluation metrics Ko (2018). For this, we use these metrics to evaluate our proposed model. The precision is defined as:

$$Precision = \frac{TP}{(TP + FP)} \tag{15}$$

And the recall is defined as Alnuaim (2022):

$$Recall = \frac{TP}{(TP + FN)} \tag{16}$$

where *TP* is the number of true positives in the dataset, *FN* is the number of false negatives, and *FP* is the number of false positives.
The F-score is the harmonic mean of these two indicators (recall and precision), both of which must be high Ding et al. (2013). It is defined as:

$$F - score = \frac{2 * recall * precision}{recall + precision} \tag{17}$$

The accuracy is defined as the ratio of true outcomes (both true positive and true negative) to the total number of cases examined Fabian Benitez-Quiroz et al. (2016). It is defined by:

$$Accuracy = \frac{TP + TN}{Total\ population} \tag{18}$$

We also use the confusion matrix as an evaluation metric to properly evaluate our model. The confusion matrix, as the name implies, produces an output matrix that describes the model's overall performance.

### 5.3 Results and discussion

In this section, we present the results of different proposed approaches for SER. Various evaluation metrics, including accuracy score, loss, recall, precision, F-score, and confusion matrix, are employed throughout this research to assess the performance of the models. We

**Table 5** Comparative Study of Average Accuracy, Training Time, and GPU Mode

| Dataset | MLP | CNN | CNN+BiLSTM |
|---|---|---|---|
| Average accuracy | | | |
| TESS | 99.90% | 99.95% (GPU) | 100% (GPU) |
| EmoDB | 98.76% | 98.51% (GPU) | 99.50% (GPU) |
| RAVDESS | 90.09% | 86.85% (GPU) | 90.12% (GPU) |
| Training time (seconds) | | | |
| TESS | 19.361 | 43.499 | 82.055 |
| EmoDB | 5.473 | 42.804 | 30.967 |
| RAVDESS | 49.551 | 43.68 | 56.18 |

**Table 6** Classification report for the EmoDB

| Emotion | MLP | | | CNN | | | CNN+BiLSTM | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| Anger | 0.99 | 0.97 | 0.98 | 0.98 | 0.98 | 0.98 | 0.99 | 0.99 | 0.99 |
| Boredom | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Disgust | 0.96 | 1.00 | 0.98 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Fearful | 1.00 | 1.00 | 1.00 | 0.95 | 1.00 | 0.97 | 1.00 | 1.00 | 1.00 |
| Happy | 0.95 | 0.96 | 0.95 | 0.96 | 0.91 | 0.93 | 0.98 | 0.98 | 0.98 |
| Neutral | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Sad | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |

provide a detailed analysis of the previously described models performance on three datasets: EmoDB, TESS, and RAVDESS, across multiple evaluation metrics. Certainly! Here is the combined table with both datasets:

Table 5 presents a comparative study of the average accuracy and training time results obtained by the proposed models on various datasets using data augmentation techniques such as noise addition and spectrogram shift.

For the RAVDESS dataset, the CNN model seems to have the shortest training time but also the lowest accuracy among the three models. The CNN+BiLSTM model achieves a marginally higher accuracy than the MLP model but at the cost of a longer training time. Overall, we can observe that in general, more complex models like the CNN and CNN+BiLSTM tend to achieve higher accuracies than the simpler MLP model, but this comes at the expense of longer training times. However, the relationship between training time and accuracy is not always linear, as seen in the case of the EmoDB dataset, where the CNN+BiLSTM model achieves the highest accuracy with a shorter training time than the CNN model.

Tables 6, 7, and 8 (where P: Precision, R: Recall, and F: F-score) present the classification reports, which include precision, recall, and F-score for all the proposed models across the different datasets. The proposed CNN + BiLSTM model achieves high precision, recall, and f1 score values of 100% for all emotion categories in the TESS and EmoDB datasets, except for Happy, which produces 98% on the same measures. For the RAVDESS dataset, the classification support values ranged from 80 to 100%, which is still considered good. We also observe that the proposed MLP model performs well in classifying emotions. Specifically, the precision, recall, and f1 score range from 95 to 100%. Furthermore, the proposed model based on CNN shows promising results, but we observe that its classification report for the RAVDESS dataset is not as high as the other models. For instance, the precision

**Table 7** Classification report for the TESS

| Emotion | MLP | | | CNN | | | CNN+BiLSTM | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| Anger | 0.99 | 0.96 | 0.97 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Calm | 0.90 | 0.89 | 0.89 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Disgust | 0.89 | 0.90 | 0.90 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Fearful | 0.91 | 0.92 | 0.91 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Happy | 0.88 | 0.92 | 0.90 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Neutral | 0.76 | 0.80 | 0.78 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Sad | 0.93 | 0.87 | 0.90 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| Surprised | 0.90 | 0.87 | 0.90 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |

**Table 8** Classification report for the RAVDESS

| Emotion | MLP | | | CNN | | | CNN+BiLSTM | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F |
| Anger | 0.99 | 0.96 | 0.97 | 0.95 | 0.96 | 0.96 | 0.98 | 0.91 | 0.94 |
| Calm | 0.90 | 0.89 | 0.89 | 0.91 | 0.82 | 0.86 | 0.91 | 0.84 | 0.88 |
| Disgust | 0.89 | 0.90 | 0.90 | 0.94 | 0.79 | 0.86 | 0.83 | 0.96 | 0.89 |
| Fearful | 0.91 | 0.92 | 0.91 | 0.83 | 0.94 | 0.88 | 0.92 | 0.96 | 0.94 |
| Happy | 0.88 | 0.92 | 0.90 | 0.85 | 0.89 | 0.87 | 0.91 | 0.86 | 0.88 |
| Neutral | 0.76 | 0.80 | 0.78 | 0.70 | 0.84 | 0.76 | 0.80 | 0.85 | 0.82 |
| Sad | 0.93 | 0.87 | 0.90 | 0.81 | 0.87 | 0.84 | 0.87 | 0.90 | 0.88 |
| Surprised | 0.90 | 0.87 | 0.90 | 0.95 | 0.86 | 0.90 | 0.95 | 0.91 | 0.93 |

and F-score values for the neutral category are 70% and 76%, respectively. However, we also notice that this model performs better with the TESS dataset compared to the MLP model. This suggests that different models may perform better or worse depending on the dataset. Overall, the results suggest that the proposed model based on CNN combined with BiLSTM network can be effective in classifying emotions from speech signal. However, the performance may vary depending on the dataset and the specific emotion categories being classified.

As depicted in Figs. 18, 19, 20, 21, 22, 23, 24, 25, 26, the precision of different emotion categories is noteworthy, and we detect minimal instances of confusion that could cause problems. For example, when using the RAVDESS dataset, the CNN model achieves an impressive 80% precision for emotions such as surprise, fear, and happiness. Similarly, the CNN+BiLSTM model produces the best precision results with the least confusion across all datasets.

Figures 12, 13, 14, 15, 16, 17, provides an informative visualization of the learning performance of the MLP, CNN, and CNN-BiLSTM models across 100 epochs and a batch size of 64 for the different datasets. This visualization enables a comprehensive understanding of the models convergence and training behavior during the training process. The curves illustrates the training accuracy, validation accuracy, and loss of the models over the epochs, highlighting the trends and fluctuations in the performance of each model. This visualization can be useful for identifying potential over fitting or under fitting issues and can guide the fine-tuning of the models to improve their performance.

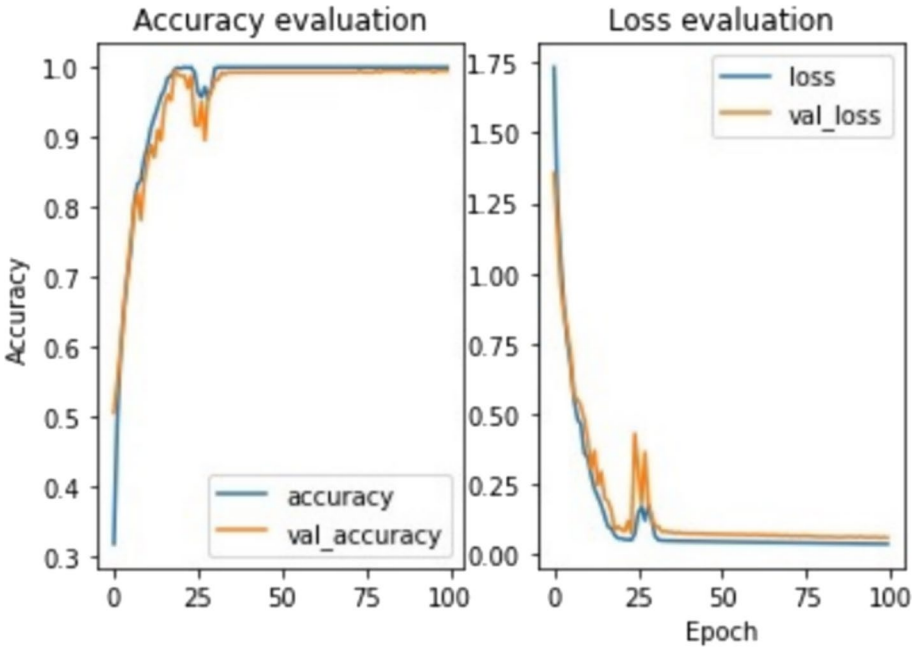



**Fig. 12** Loss and accuracy variation on EmoDB using CNN-BiLSTM model

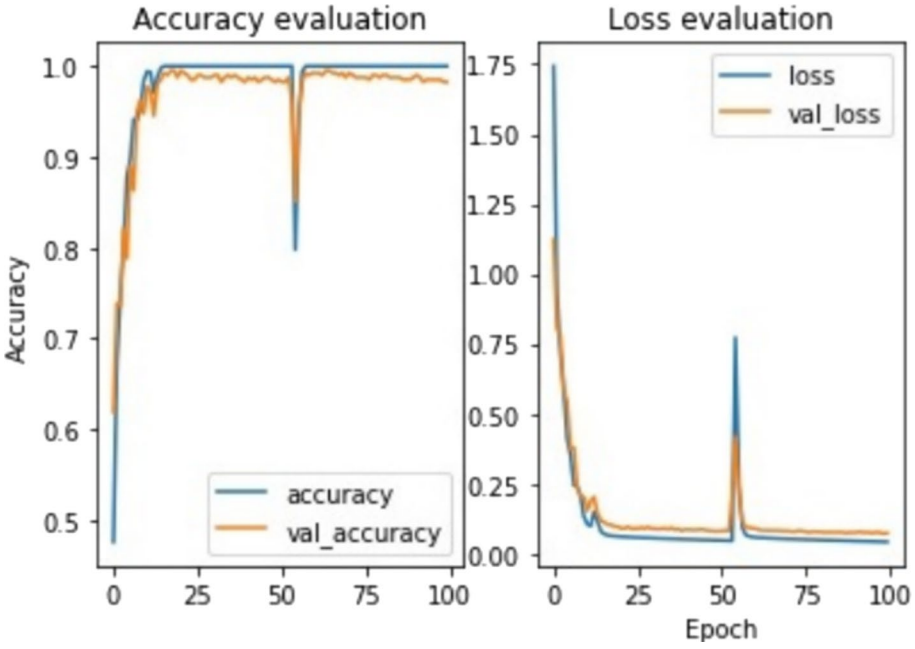



**Fig. 13** Loss and accuracy variation on EmoDB using CNN model

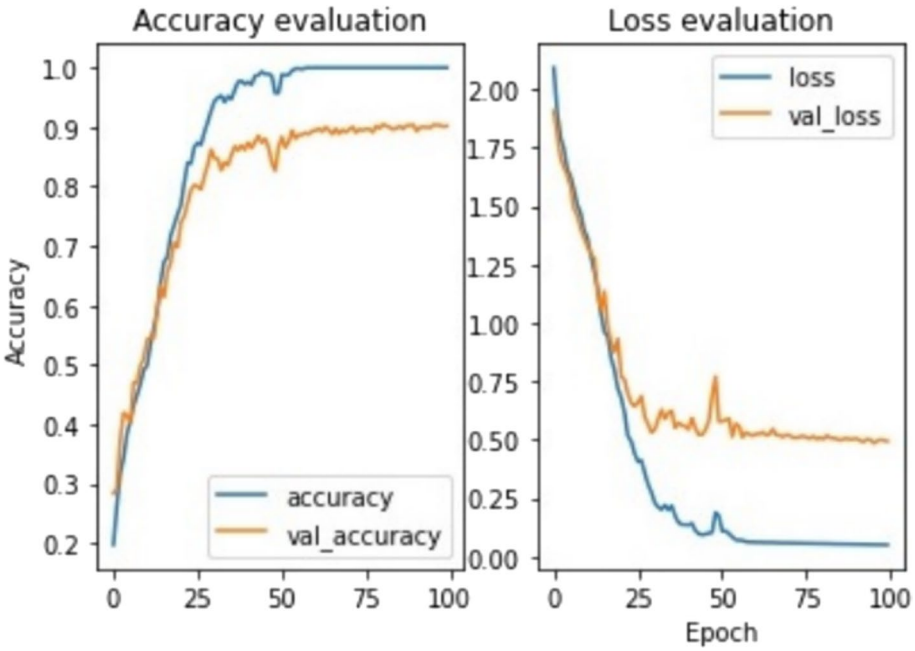



**Fig. 14** Loss and accuracy variation on RAVDESS using CNN-BiLSTM model

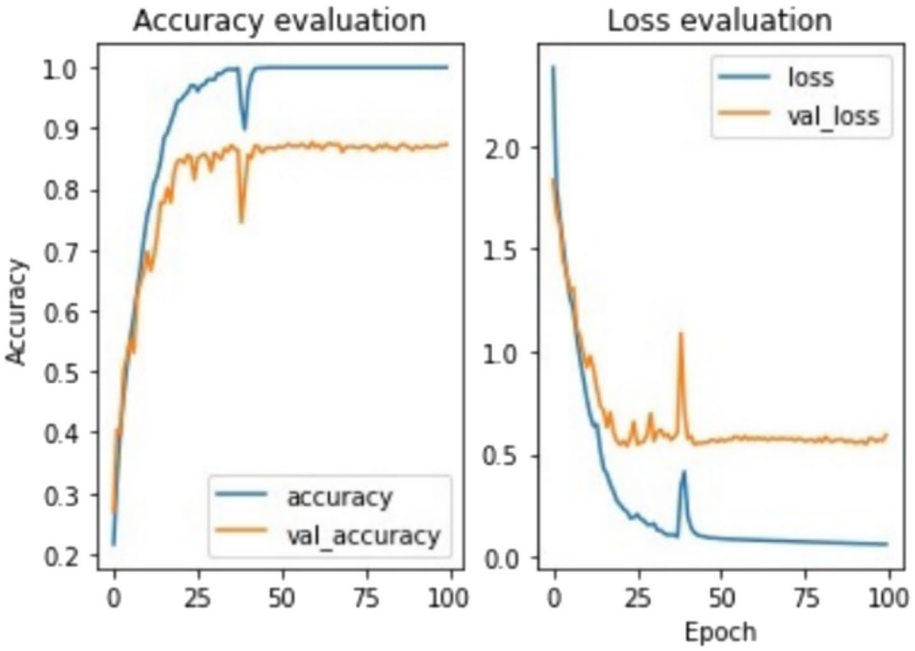



**Fig. 15** Loss and accuracy variation on RAVDESS using CNN model

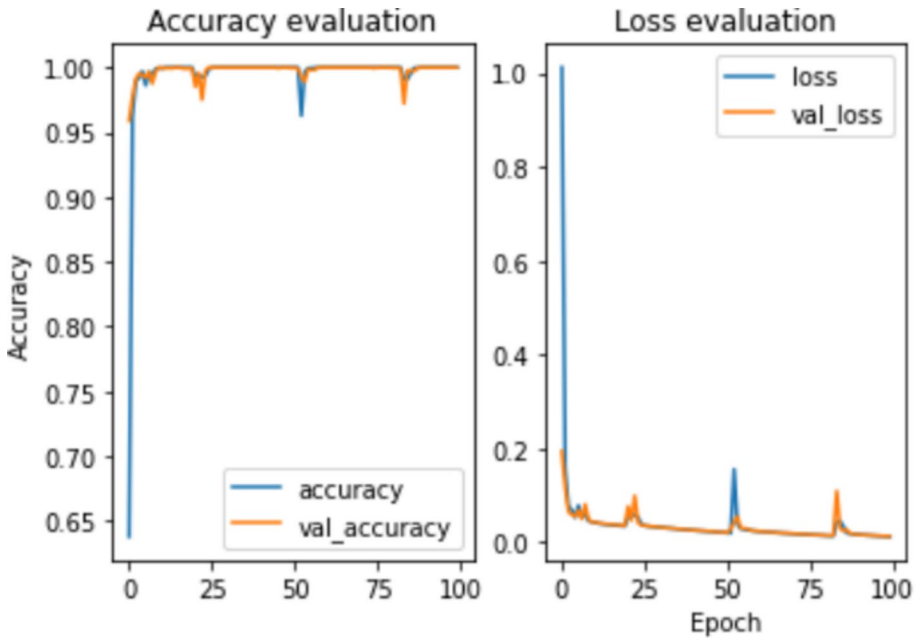



**Fig. 16** Loss and accuracy variation on TESS using CNN-BiLSTM model

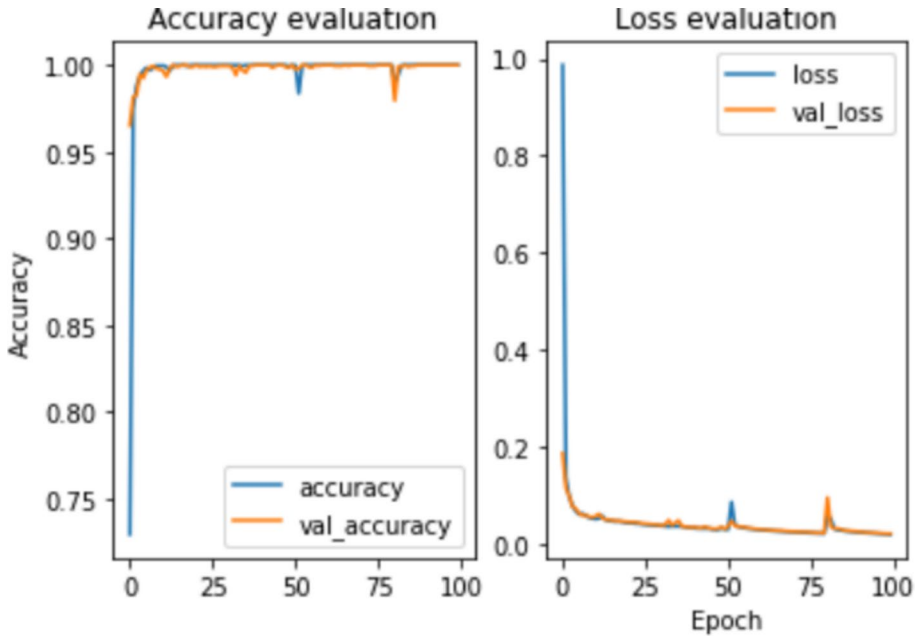



**Fig. 17** Loss and accuracy variation on TESS using CNN model

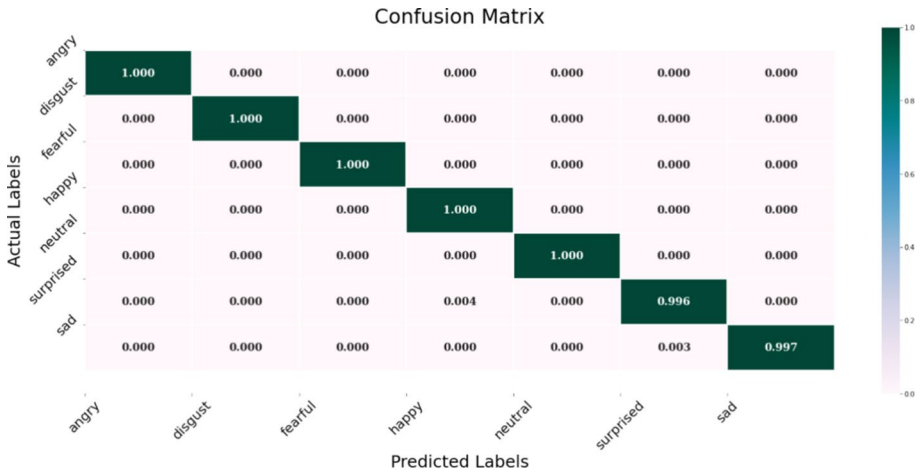



**Fig. 18** Confusion matrix for TESS dataset using MLP model

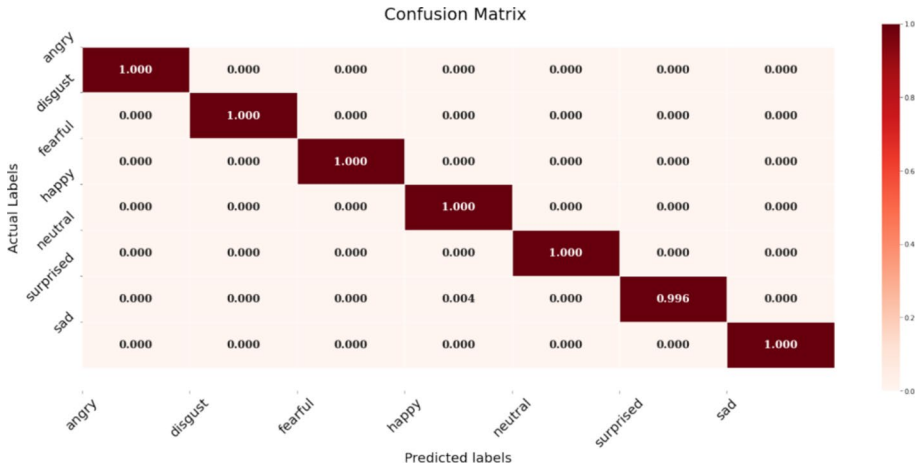



**Fig. 19** Confusion matrix for TESS dataset using CNN model

## 5.4 Real-time evaluation of the proposed SER system

Figure 27 present some examples to validate our approach to speech emotion recognition, with four recordings correctly predicted with high accuracy. These predictions serve to validate our approach and demonstrate its effectiveness in accurately identifying emotional states from speech signals in a real-time scenario. In this figure, we present the time domain series for each record saved by random choice and plot below each example the probability prediction for each emotion class.

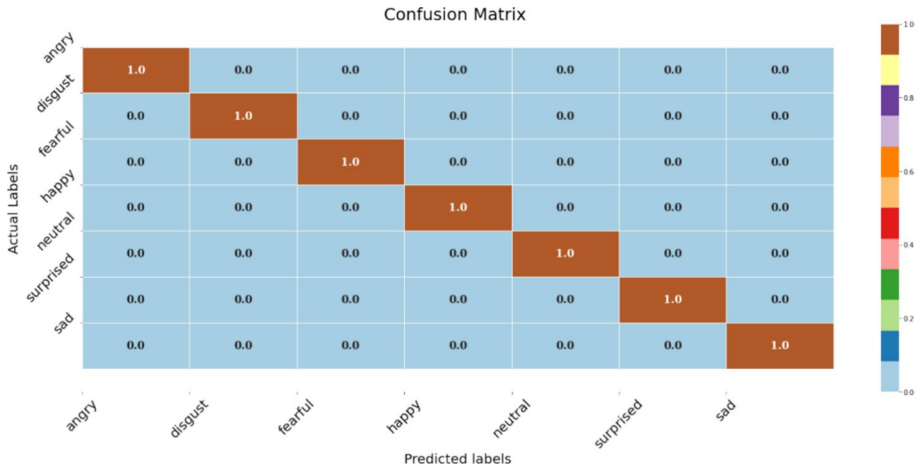



**Fig. 20** Confusion matrix for TESS dataset using CNN+BiLSTM model

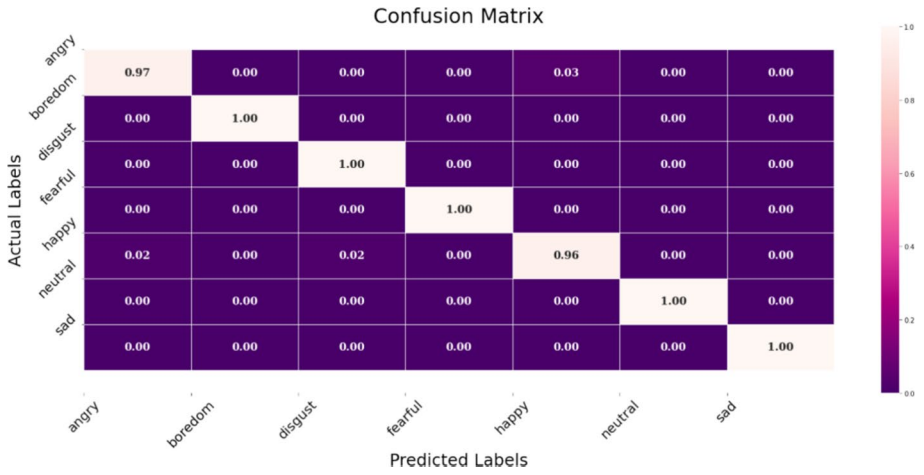



**Fig. 21** Confusion matrix for EmoDB dataset using MLP model

## 6 Comparative study

The Multilayer Perceptron (MLP) neural network represents a simple learning technique in terms of implementation and adjustment. It does not require a GPU-enabled environment for training, and can quickly learn even from large databases. The results achieved by the MLP are powerful, as evidenced by the good overall recognition rates of our system: 99.90%, 98.76%, and 90.09% respectively for the TESS, EMO-DB, and RAVDESS databases. This demonstrates the efficacy of the MLP in speech emotion recognition tasks and highlights its potential as a valuable tool for a range of applications in this field.

When it comes to the static and spatial aspects of learning data, such as images, audio sequences, and text, we certainly think of the CNN, which is widely used for analyzing spe-

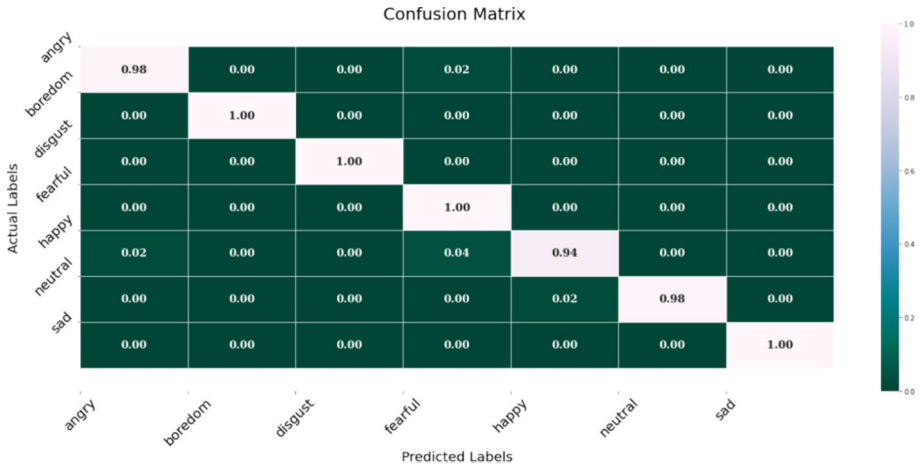



**Fig. 22** Confusion matrix for EmoDB dataset using CNN model

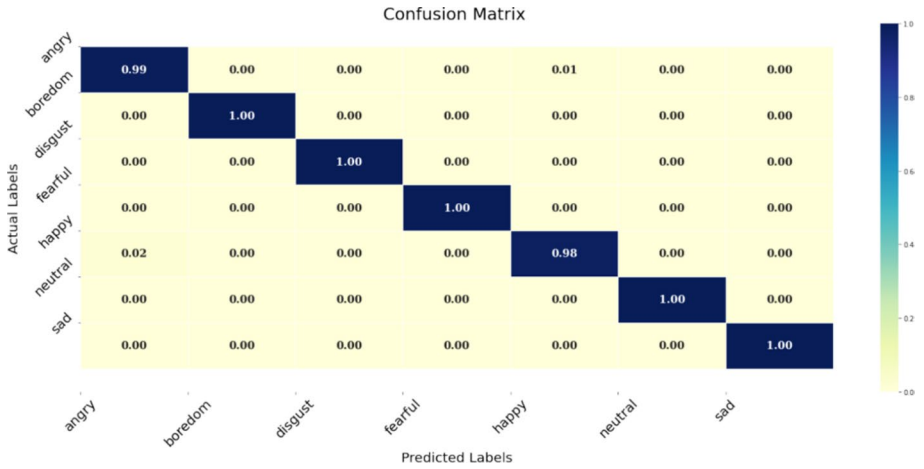



**Fig. 23** Confusion matrix for EmoDB dataset using CNN+BiLSTM model

cial hierarchical features through convolution operations. An environment with GPU execution mode is required to train models based on CNN. Thanks to this free service provided by Google Colab, we reduce execution time. Thus, the CNN model gives us the following global recognition rates: 99.95%, 98.51%, and 86.85%, respectively, for the TESS, EMO-DB, and RAVDESS databases. These results indicate that the CNN is a powerful model for learning special features in static and spatial data.

The speech data we use for this work includes both spectral and temporal aspects. We combine the Bi-LSTM technique, which processes the temporal aspect, with the CNN technique to analyze both aspects simultaneously. This approach allow us to increase the accuracy rate of the proposed model, resulting in a recognition rate of 100% with the TESS database, 99.50% with the EmoDB database, and 90.12% with the RAVDESS database. As

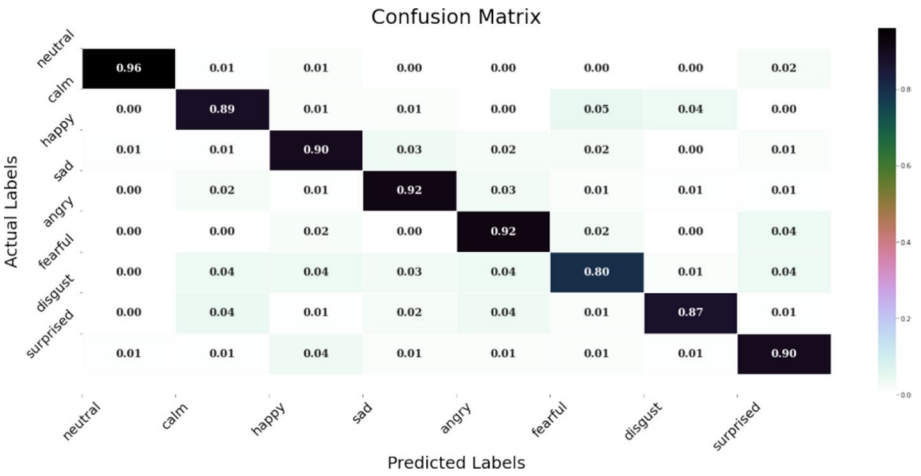



**Fig. 24** Confusion matrix for RAVDESS dataset using MLP model

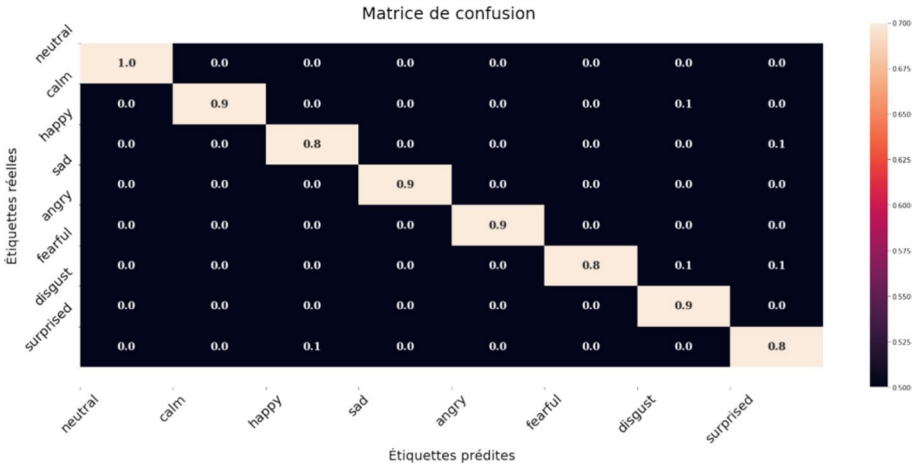



**Fig. 25** Confusion matrix for RAVDESS dataset using CNN model

shown in the Fig. 28, the last proposed model achieves the highest recognition rate values across the different databases.

In the three Figs. 29, 30, and 31 we display the accuracies of the different categories of the TESS, EMO-DB, and RAVDESS datasets using the following classification models: MLP, CNN, and CNN+Bi-LSTM.

With all datasets, the CNN + Bi-LSTM models always give us -the best results either in terms of average accuracy rate or in terms of precision per category.

The TESS dataset contains a total of 2800 original samples in.wav format that describe emotional situations. Each category in the TESS dataset contains 200 samples, and we notice that this database is balanced. We apply a data augmentation process to enhance the

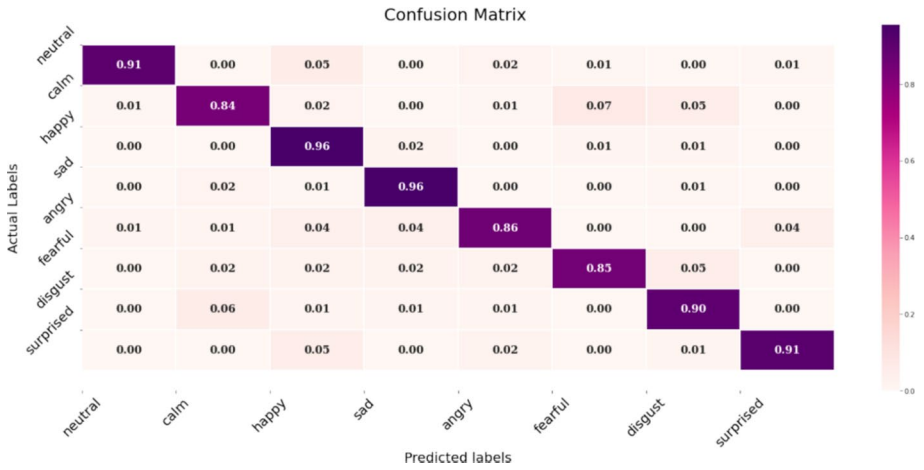



**Fig. 26** Confusion matrix for RAVDESS dataset using CNN+BiLSTM model

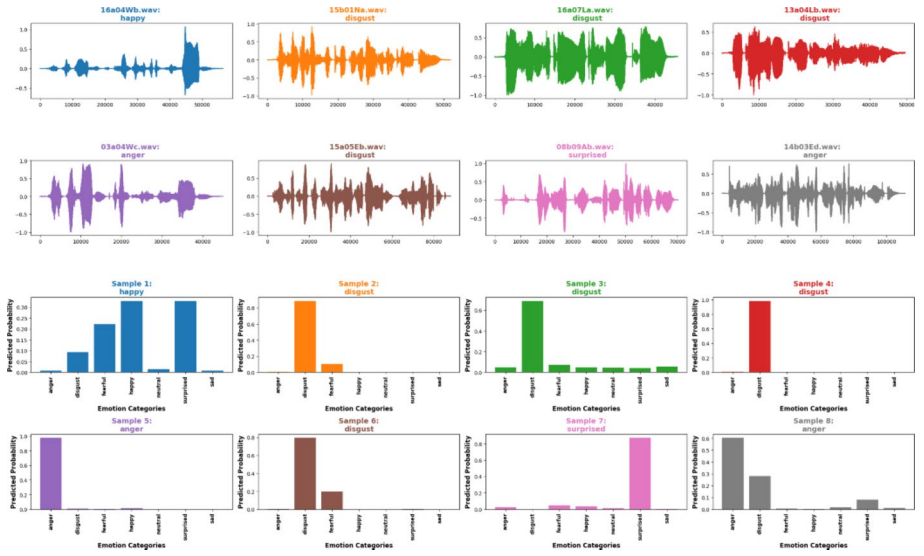



**Fig. 27** Result of the prediction of some records in real time

content of this dataset, which explains the obtained result of high average accuracy and precision as shown in Fig. 29.

The EmoDB dataset contains a total of 535 original.wav samples. To increase the dataset size and improve the performance of deep learning models such as CNN and CNN+BiLSTM, we apply data augmentation techniques. The results obtained are good, as described in Fig. 30.

Having a powerful model is important for speech emotion recognition, but the quality of the database used and the features extracted are equally critical. Even with an increase in data volume, if the database lacks high-quality samples or the features used are inap-

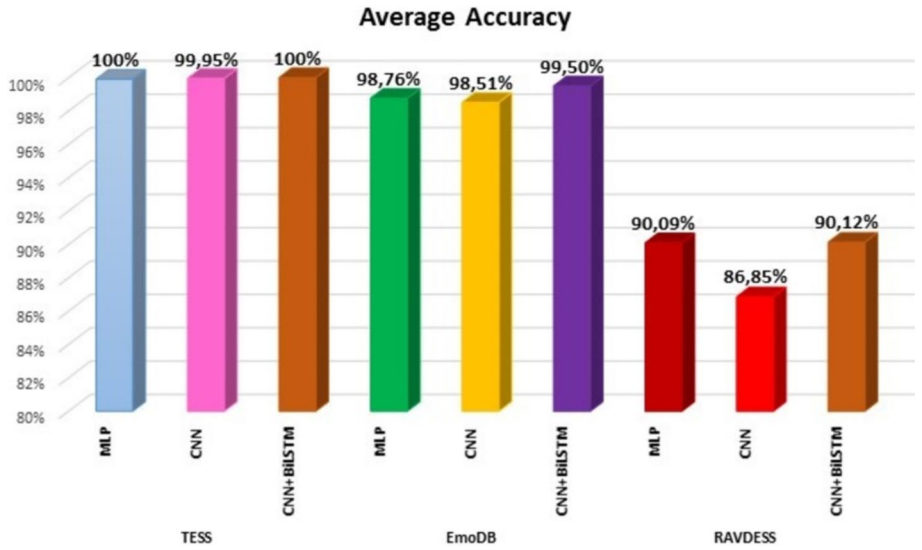



**Fig. 28** Summary of average accuracy

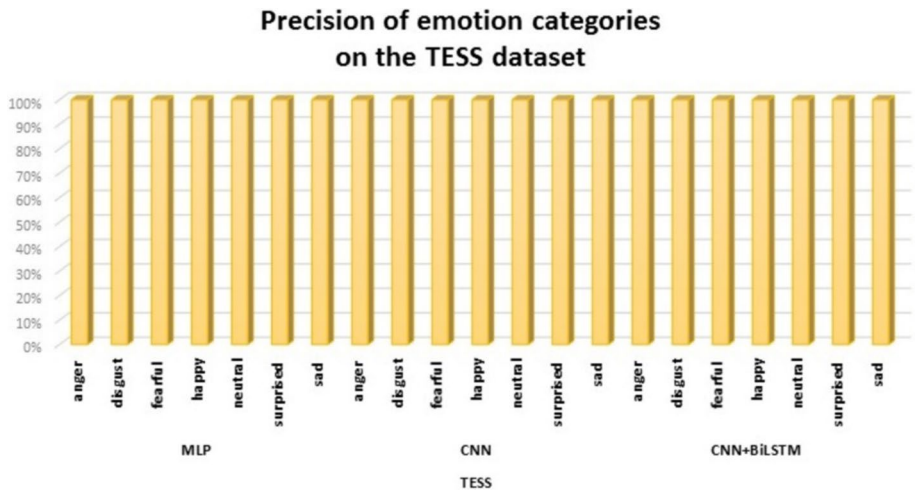



**Fig. 29** Precision of emotion categories on the TESS dataset

propriate, it can affect the model's performance. The RAVDESS database, although comprehensive, may not yield perfect recognition rates due to the nature of the samples and the variations in emotional expressions. In contrast, the TESS database, being well-balanced and augmented, may provide more accurate recognition results. Therefore, it is crucial to choose the right database with high-quality samples and to use appropriate features to improve the performance of speech emotion recognition models.

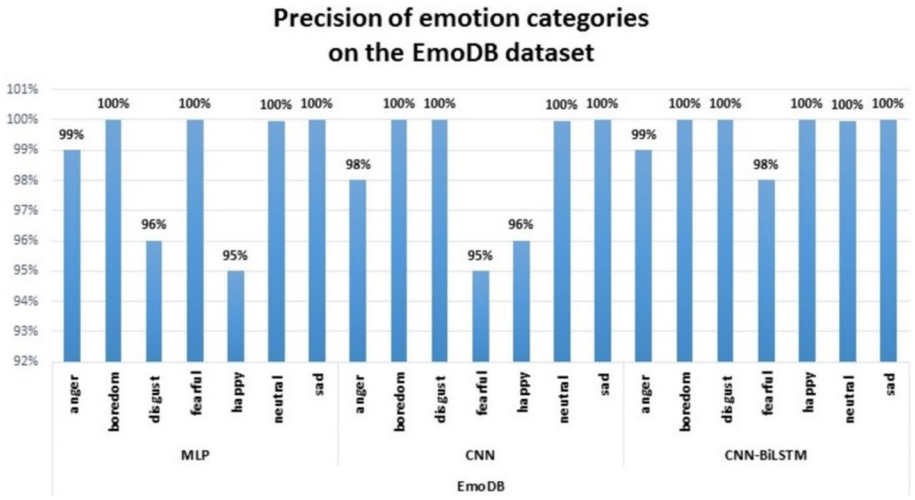



**Fig. 30** Precision of emotion categories on the EmoDB dataset

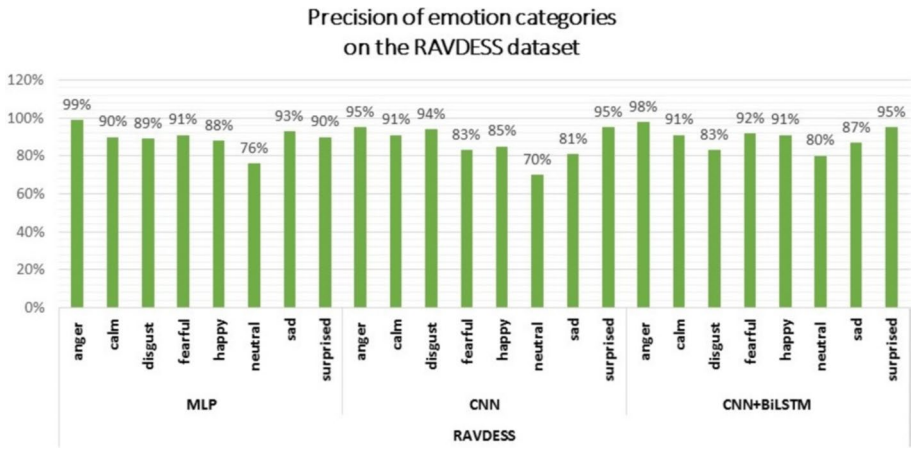



**Fig. 31** Precision of emotion categories on the RAVDESS dataset

In Tables 10, 9, and 11, we present a comparison of the average accuracy of our proposed models on the TESS, EmoDB, and Ravdess datasets, respectively, with other recent works to demonstrates the performance of our proposed SER systems.

## 7 Conclusion

In this paper, we present a comprehensive study of a SER system that employs multiple acoustic features and three neural network models, namely MLP, CNN, and CNN+BiLSTM. The system's architecture is designed to leverage the power of multiple acoustic features, including MFCCs, ZCR, Chroma, Mel spectrogram, and RMS, to accurately classify speech

**Table 9** Comparison between the performance of the proposed approaches and the other competing approaches on the EmoDB dataset

| Reference | Methodology | Feature | Accuracy (%) |
|---|---|---|---|
| Kaur and Kumar (2021) | CNN | STFT+ MFCC | 90.36 |
| Garg et al. (2020) | SVM | Spectral, Time domain, Voice quality | 93.81 |
| Venkata Subbarao et al. (2021) | KNN | 39 MFCC | 90.01 |
| Tuncer et al. (2021) | SVM | Twine-shuf-pat | 90.36 |
| Proposed approach based on: MLP | MLP | MFCC+ Mel spectrogram + ZCR + Chroma + RMS | 98.76 |
| approach based on: CNN | CNN | // | 98.51 |
| Proposed approach based on: CNN+BiLSTM | CNN+BiLSTM | // | 99.50 |

**Table 10** Comparison between the performance of the proposed approaches and the other competing approaches on the TESS dataset

| Reference | Methodology | Feature | Accuracy (%) |
|---|---|---|---|
| Aggarwal et al. (2022) | VGG-16 | 2D features extraction | 97.15 |
| Aggarwal et al. (2022) | DNN | Log Mel-Spectrogram | 99.99 |
| Proposed approach based on: MLP | MLP | MFCC+ Mel spectrogram + ZCR + Chroma + RMS | 99.90 |
| approach based on: CNN | CNN | // | 99.95 |
| Proposed approach based on: CNN+BiLSTM | CNN+BiLSTM | // | 100 |

emotions. Additionally, we use two data augmentation techniques, namely noise addition and shifting spectrogram, to improve the quality and diversity of the dataset samples.

Our experiments are conducted on three datasets, namely TESS, EmoDB, and RAVDESS, and the results demonstrate the effectiveness of our proposed SER system. Specifically, the MLP model achieve the highest accuracy rates, with an average of 99.90%, 98.76%, and 90.09% for TESS, EmoDB, and RAVDESS, respectively. The CNN model also performs well, achieving an average accuracy of 99.95%, 98.51%, and 86.85%, respectively. Finally, the CNN+BiLSTM model outperforms the other models with an average accuracy of 100%, 99.50%, and 90.12%, respectively. The findings of our study indicate that the proposed SER system, with its use of multiple acoustic features and neural network models, can accurately classify speech emotions. Moreover, the system's performance is improved with the use of

**Table 11** Comparison between the performance of the proposed approaches and the other competing approaches on the RAVDESS dataset

| Reference | Methodology | Feature | Accuracy |
|---|---|---|---|
| Alnuaim (2022) | MLP | MFCC+ Mel frequency cepstrum +Chroma | 81 |
| Bhandari et al. (2022) | LSTM | MFCC | 89 |
| Tuncer et al. (2021) | SVM | Twine-shuf-pat | 87.43 |
| Xu et al. (2021) | CNN | Attention Networks | 77.4 |
| Proposed approach based on: MLP | MLP | MFCC+ Mel spectrogram + ZCR + Chroma + RMS | 90.09 |
| approach based on: CNN | CNN | // | 86.85 |
| Proposed approach based on: CNN+BiLSTM | CNN+BiLSTM | // | 90.12 |

data augmentation techniques, demonstrating the importance of diverse and high-quality datasets in training machine learning models. Overall, this study provides a valuable contribution to the field of speech emotion recognition and could potentially have practical applications in fields such as human–computer interaction, psychology, and entertainment.

Although the acoustic features used in this study are effective in recognizing speech emotions, there may be other features that could be useful in improving the accuracy of the SER system. For instance, additional acoustic features such as pitch, spectral centroid, and spectral flux could be investigated to determine their impact on the classification performance. Furthermore, it would be interesting to explore the use of physiological signals, such as heart rate and skin conductance, as supplementary features to augment the existing acoustic features.

While the datasets used in this study were diverse and representative, they are relatively small in size. To evaluate the generalizability of the proposed SER system, future studies could evaluate its performance on larger datasets with more diverse speakers, speech styles, and emotional expressions. Additionally, the effect of different sampling rates, bit resolutions, and compression techniques on the performance of the SER system could be studied to determine the optimal configuration for different application scenarios.

Although this study focuses on recognizing speech emotions in English, the proposed SER system could potentially be extended to other languages. Future research could explore the feasibility of extending the system to other languages and evaluate its performance in cross-lingual scenarios. Furthermore, it would be interesting to investigate the use of transfer learning and domain adaptation techniques to enhance the performance of the SER system when applied to new languages and cultures.

In summary, by exploring additional acoustic features, evaluating the system on larger datasets, and extending it to other languages, future research could improve the accuracy and applicability of the proposed SER system.

**Author contributions** All authors contributed equally to this work. Chawki Barhoumi conceived the study, designed and implemented the survey, and prepared the manuscript. Yassine BenAyed contributed to the

design and implementation of the survey, and the preparation of the manuscript. All authors read and approved the final manuscript.

**Funding** The authors did not receive support from any organization for the submitted work.

**Data availability** The Python notebooks used in this study are publicly available on Google Drive at the following link: ● Notebook folder: https://drive.google.com/drive/folders/1bVzqTY9KxlTwaVpuPIRXngWgF4SFE2EB?usp=share_link The datasets used in this study are publicly available on Kaggle at the following links: ● TESS Dataset: https://www.kaggle.com/datasets/ejlok1/toronto-emotional-speech-set-tess● EmoDB Dataset: https://www.kaggle.com/datasets/piyushagni5/berlin-database-of-emotional-speech-emodb● RAVDESS Dataset: https://www.kaggle.com/datasets/uwrfkaggler/ravdess-emotional-speech-audio Please contact the corresponding author for access to any additional materials related to this study.

## Declarations

**Conflict of interest** The authors declare that they have no financial or non-financial Conflict of interest.



## References

Abbaschian BJ, Sierra-Sosa D, Elmaghraby A (2021) Deep learning techniques for speech emotion recognition, from databases to models. Sensors 21(4):1249

Abdelhamid AA, El-Kenawy E-SM, Alotaibi B, Amer GM, Abdelkader MY, Ibrahim A, Eid MM (2022) Robust speech emotion recognition using cnn+ lstm based on stochastic fractal search optimization algorithm. IEEE Access 10:49265–49284

Abdullah SMS, Abdulazeez AM (2021) Facial expression recognition based on deep learning convolution neural network: a review. J Soft Comput Data Mining 2(1):53–65

Aggarwal A, Srivastava N, Singh D (2022) Alnuaim: two-way feature extraction for speech emotion recognition using deep learning. Sensors 22(6):2378

Akçay MB, Oğuz K (2020) Speech emotion recognition: emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers. Speech Commun 116:56–76

Aljuhani RH, Alshutayri A, Alahdal S (2021) Arabic speech emotion recognition from Saudi dialect corpus. IEEE Access 9:127081–127085

Alluhaidan AS, Saidani O, Jahangir R, Nauman MA, Neffati OS (2023) Speech emotion recognition through hybrid features and convolutional neural network. Appl Sci 13(8):4750

Alnuaim, Hatamleh (2022) Human–computer interaction for recognizing speech emotions using multilayer perceptron classifier, vol. 2022. Hindawi

Aouani H, Ben Ayed Y (2020) Speech emotion recognition with deep learning. Proc Comput Sci 176:251–260

Arguel A, Lockyer L, Kennedy G, Lodge JM, Pachman M (2019) Seeking optimal confusion: a review on epistemic emotion management in interactive digital learning environments. Interact Learn Environ 27(2):200–210

Bachu R, Kopparthi S, Adapa B, Barkana B (2008) Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal. In: American society for engineering education (ASEE) zone conference proceedings, pp. 1–7. American Society for Engineering Education

Bänziger T, Scherer KR (2005) The role of intonation in emotional expressions. Speech Commun 46(3–4):252–267

Bhandari SU, Kumbhar HS, Harpale VK, Dhamale TD (2022) On the evaluation and implementation of lstm model for speech emotion recognition using mfcc. In: Proceedings of international conference on computational intelligence and data Engineering: ICCIDE 2021, pp. 421–434. Springer
Bhangale K, Kothandaraman M (2023) Speech emotion recognition based on multiple acoustic features and deep convolutional neural network. Electronics 12(4):839
Burkhardt F, Paeschke A, Rolfes M, Sendlmeier WF, Weiss B et al (2005) A database of German emotional speech. Interspeech 5:1517–1520
Chen L, Mao X, Xue Y, Cheng LL (2012) Speech emotion recognition: features and classification models. Digital Signal Proc 22(6):1154–1160. https://doi.org/10.1016/j.dsp.2012.05.007
Chen S, Dobriban E, Lee JH (2020) A group-theoretic framework for data augmentation. J Mach Learn Res 21(1):9885–9955
Cowie R, Douglas-Cowie E, Tsapatsoulis N, Votsis G, Kollias S, Fellenz W, Taylor JG (2001) Emotion recognition in human–computer interaction. IEEE Signal Process Mag 18(1):32–80
Ding X, Chu W-S, Torre F, Cohn JF, Wang Q (2013) Facial action unit event detection by cascade of tasks. In: Proceedings of the IEEE international conference on computer vision, pp. 2400–2407
El Ayadi M, Kamel MS, Karray F (2011) Survey on speech emotion recognition: features, classification schemes, and databases. Pattern Recogn 44(3):572–587. https://doi.org/10.1016/j.patcog.2010.09.020
Fabian Benitez-Quiroz C, Srinivasan R, Martinez AM (2016) Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5562–5570
Garg U, Agarwal S, Gupta S, Dutt R, Singh D (2020) Prediction of emotions from the audio speech signals using mfcc, mel and chroma. In: 2020 12th international conference on computational intelligence and communication networks (CICN), pp. 87–91. IEEE
Gupta D, Bansal P, Choudhary K (2018) The state of the art of feature extraction techniques in speech recognition. Speech and language processing for human–machine communications: proceedings of CSI 2015:195–207
Hama Saeed M (2023) Improved speech emotion classification using deep neural network. Circuits Syst Signal Proc 42(12):7357–7376
Han L, Mao X, Zhao G, Xu B (2017) Emotion recognition from speech using shifting short-time Fourier transform and convolutional neural networks. In: Proceedings of the international conference on computer vision and pattern recognition workshops, pp. 2436–2444. IEEE
Huang Y, Tian K, Wu A, Zhang G (2019) Feature fusion methods research based on deep belief networks for speech emotion recognition under noise condition. J Ambient Intell Humaniz Comput 10:1787–1798
Jiang P, Fu H, Tao H, Lei P, Zhao L (2019) Parallelized convolutional recurrent neural network with spectral features for speech emotion recognition. IEEE Access 7:90368–90377
Kaiser JF (1990) On a simple algorithm to calculate the 'energy' of a signal. In: International conference on acoustics, speech, and signal processing, pp. 381–3841. https://doi.org/10.1109/ICASSP.1990.115702
Kaur J, Kumar A (2021) Speech emotion recognition using cnn, k-nn, mlp and random forest. In: Computer networks and inventive communication technologies: proceedings of Third ICCNCT 2020, pp. 499–509. Springer
Khalil RA, Jones E, Babar MI, Jan T, Zafar MH, Alhussain T (2019) Speech emotion recognition using deep learning techniques: a review. IEEE Access 7:117327–117345
Kim Y, Lee H, Provost EM (2013) Deep learning for robust feature generation in audiovisual emotion recognition. In: 2013 IEEE international conference on acoustics, speech and signal processing, pp. 3687–3691. IEEE
Ko BC (2018) A brief review of facial emotion recognition based on visual information. Sensors 18(2):401
Koduru A, Valiveti HB, Budati AK (2020) Feature extraction algorithms to improve the speech emotion recognition rate. Int J Speech Technol 23(1):45–55
Koolagudi SG, Rao KS (2012) Emotion recognition from speech: a review. Int J Speech Technol 15:99–117
Krishna KV, Sainath N, Posonia AM (2022) Speech emotion recognition using machine learning. In: 2022 6th international conference on computing methodologies and communication (ICCMC), pp. 1014–1018. IEEE
Kwak K, Kim J-H (2019) A convolutional neural network for speech emotion recognition using a mel spectrogram. Appl Sci 9(13):2697
Kwon S (2020) Clstm: deep feature-based speech emotion recognition using the hierarchical ConvLSTM network. Mathematics 8(12):2133
Lanjewar RB, Mathurkar S, Patel N (2015) Implementation and comparison of speech emotion recognition system using gaussian mixture model (gmm) and k-nearest neighbor (k-nn) techniques. Procedia Comput Sci 49:50–57

Livingstone SR, Russo FA (2018) The Ryerson audio-visual database of emotional speech and song (RAVDESS): a dynamic, multimodal set of facial and vocal expressions in north American english. PLoS ONE 13(5):0196391

Meng H, Yan T, Yuan F, Wei H (2019) Speech emotion recognition from 3d log-mel spectrograms with deep learning network. IEEE Access 7:125868–125881. https://doi.org/10.1109/ACCESS.2019.2938007

Nam Y, Lee C (2021) Cascaded convolutional neural network architecture for speech emotion recognition in noisy conditions. Sensors 21(13):4399

Oh K-J, Lee D, Ko B, Choi H-J (2017) A chatbot for psychiatric counseling in mental healthcare service based on emotional dialogue analysis and sentence generation. In: 2017 18th IEEE international conference on mobile data management (MDM), pp. 371–375. IEEE

Pichora-Fuller MK, Dupuis K (2020). Toronto emotional speech set (TESS). https://doi.org/10.5683/SP2/E8H2MF

Prabhakar GA, Basel B, Dutta A, Rao CVR (2023) Multichannel cnn-blstm architecture for speech emotion recognition system by fusion of magnitude and phase spectral features using DCCA for consumer applications. IEEE Transactions on consumer electronics

Pratama A, Sihwi SW (2022) Speech emotion recognition model using support vector machine through mfcc audio feature. In: 2022 14th International conference on information technology and electrical Engineering (ICITEE), pp. 303–307. IEEE

Sawardekar S, Naik SR (2018) Facial expression recognition using efficient LBP and CNN. Int Res J Eng Technol (IRJET) 5(6):2273–2277

Schuller B, Vlasenko B, Eyben F, Rigoll G, Wendemuth A (2009) Acoustic emotion recognition: a benchmark comparison of performances. In: 2009 IEEE workshop on automatic speech recognition & understanding, pp. 552–557. IEEE

Selvaraj M, Bhuvana R, Padmaja S (2016) Human speech emotion recognition. Int J Eng Technol 8:311–323

Sowmya G, Naresh K, Sri JD, Sai KP, Indira DV (2022) Speech2emotion: intensifying emotion detection using mlp through ravdess dataset. In: 2022 International conference on electronics and renewable systems (ICEARS), pp. 1–3. IEEE

Tariq Z, Shah SK, Lee Y (2019) Speech emotion detection using iot based deep learning for health care. In: 2019 IEEE international conference on big data (Big Data), pp. 4191–4196. IEEE

Teager H, Teager S (1990) Evidence for nonlinear sound production mechanisms in the vocal tract. Speech production and speech modelling, pp. 241–261

Tuncer T, Dogan S, Acharya UR (2021) Automated accurate speech emotion recognition system using twine shuffle pattern and iterative neighborhood component analysis techniques. Knowl Based Syst 211:106547. https://doi.org/10.1016/j.knosys.2020.106547

Venkata Subbarao M, Terlapu SK, Geethika N, Harika KD (2021) Speech emotion recognition using k-nearest neighbor classifiers. In: Recent advances in artificial intelligence and data engineering: select proceedings of AIDE 2020, pp. 123–131. Springer

Wang M, Wang Z, Li J (2017) Deep convolutional neural network applies to face recognition in small and medium databases. In: 2017 4th international conference on systems and informatics (ICSAI), pp. 1368–1372. IEEE

Xu M, Zhang F, Zhang W (2021) Head fusion: improving the accuracy and robustness of speech emotion recognition on the IEMOCAP and RAVDESS dataset. IEEE Access 9:74539–74549. https://doi.org/10.1109/ACCESS.2021.3067460

Yenigalla P, Kumar A, Tripathi S, Singh C, Kar S, Vepa J (2018) Speech emotion recognition using spectrogram & phoneme embedding. Interspeech 2018:3688–3692

Zeng Z, Pantic M, Roisman GI, Huang TS (2009) A survey of affect recognition methods: audio, visual, and spontaneous expressions. IEEE Trans Pattern Anal Mach Intell 31(1):39–58 (**Cited By: 2111**)

Zheng WQ, Yu JS, Zou YX (2015) An experimental study of speech emotion recognition based on deep convolutional neural networks. In: 2015 International conference on affective computing and intelligent interaction (ACII), pp. 827–831. https://doi.org/10.1109/ACII.2015.7344669