Speech Emotion Recognition using MFCC and Hybrid Neural Networks

Youakim Badr, Partha Mukherjee and Sindhu Madhuri Thumati
The Pennsylvania State University, Great Valley, U.S.A.

Keywords: Hybrid Neural Network, Speech Emotion Recognition, MFCC, ConvLSTM, RAVDESS Data.

Abstract: Speech emotion recognition is a challenging task and feature extraction plays an important role in effectively classifying speech into different emotions. In this paper, we apply traditional feature extraction methods like MFCC for feature extraction from audio files. Instead of using traditional machine learning approaches like SVM to classify audio files, we investigate different neural network architectures. Our baseline model implemented as a convolutional neural network results in 60% classification accuracy. We propose a hybrid neural network architecture based on Convolutional and Long Short-Term Memory (ConvLSTM) networks to capture spatial and sequential information of audio files. Our experimental results show that our ComvLSTM model has achieved an accuracy of 59%. We improved our model with data augmentation techniques and re-trained it with augmented dataset. The classification accuracy achieves 91% for multi-class classification of RAVDESS dataset outperforming the accuracy of state-of-the-art multi-class classification models that used the similar data.

1 INTRODUCTION

Speech is the most basic mode of communication between human beings and it is the easiest way to convey emotions. Important information like the mental state of a person and his intent can be determined if we can capture the emotion of a person while he is speaking. This is not only crucial in the case of human conversations but also for human-machine interactions. With the latest advancements in the field of machine learning, the number of human-machine interactions has significantly increased and there is a need to recognize the emotion of a person to make the conversation more natural and real. Detecting the emotion of a person would also make human-machine interaction close to human interaction (Cowie et al. 2001). Interactive chat bots have become prominent in a wide range of industries and speech emotion recognition would allow these conventional chatbots to empathize with the user while being aware of their emotion and intent (Fragopanagos and Taylor 2005). Organizations can enable the chatbots to be more user friendly by customizing their responses based on user emotion. Knowing the emotion of their customers would allow organizations to change their product strategies accordingly. Since speech emotion recognition has a wide impact on many walks of life, it has been a subject of research among many data scientists for about a decade (K.-Y. Huang et al. 2019; Lalitha et al. 2015; Mu et al. 2017). Emotion Recognition is a classification task with input being speech and output being different emotion classes. This task has been challenging because emotions are subjective and can be interpreted in many ways. Another challenge in speech emotion recognition is extracting best features from speech signals to clearly distinguish between different emotions. A significant body of research projects has been done in this area and mainly rely on traditional feature extraction methods like Mel Frequency Cepstral Coefficients (MFCC) (Kerkeni et al. 2019), Short-time Fourier Transformation (STFT) to extract features from audio files before training classification algorithms (Vinola et al. 2015; El Ayadi et al. 2011; Chandrasekar et al. 2014; Koolagudi and Rao 2012). Yet, there is no conclusive evidence about the best features to recognize emotion from speech. Recently, with the advancement in the field of deep learning, context free approaches using auto encoders are implemented and tested to perform the speech emotion classification task (K.-Y. Huang et al. 2019; Kerkeni et al. 2019).

In this paper we propose a deep neural network

Badr, Y., Mukherjee, P. and Thumati, S.
Speech Emotion Recognition using MFCC and Hybrid Neural Networks.
DOI: 10.5220/0010707400003063
In *Proceedings of the 13th International Joint Conference on Computational Intelligence (IJCCI 2021)*, pages 366-373
ISBN: 978-989-758-534-0; ISSN: 2184-2825

model with data augmentation to classify speech into six emotion classes using Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset and compare the performances with that proposed in (A. Huang and Bao 2019).

We propose an enhanced version of the Convolution and Short-Term Memory Network ConvLSTM architecture implemented in (Zhao et al. 2019). Inspired by the work of Kwon (Kwon et al. 2003), we extended our model with vocal feature extraction from raw speech signals and used extracted features to train our ConvLSTM model. The proposed model resulted in higher six class classification accuracy than (A. Huang and Bao 2019; Zhao et al. 2019) when combined with data augmentation techniques.

The remaining sections of the paper are organized as follow. In section 2, we present the literature survey while section 3 deals with data description and data preparation. Section 4 discusses on the extraction of features from the audio signals. Section 5 and 6 point out the experimental set up for CNN and ConvLSTM models respectively while section 7 delineates the findings of model performances. Nest two sections deal with the discussion of result and conclusion and future work respectively.

2 RELATED WORK

Speech emotion recognition has been an active field of research for about a decade now. Traditional speech emotion recognition models were built using feature extraction and machine learning techniques (K.-Y. Huang et al. 2019). Researchers proposed different approaches to extract frame level and utterance level features from audio files and implemented machine learning frameworks to classify emotions using the extracted features, SVM being a very widely proposed model among them. (Mannepalli et al. 2018) proposed a thresholding fusion mechanism for integrating a set of SVM classifiers for human emotion classification. The classification accuracy was improved by combining a group of SVM classifiers. But, noise resilient features of the audio have not been analyzed by the model. Cao et al. (Cao et al. 2015) implemented a ranking based model for recognizing the emotions in the speech based on the multi-class prediction strategy. Noroozi et al. (Noroozi et al. 2017) introduced a vocal-based emotion recognition approach using Random Forests. Some other classifiers, such as Decision Trees (Lee et al. 2011) and K-Nearest Neighbor (KNN) (Kim and Provost 2013), have also been used in speech emotion recognition. These classifiers require empirically chosen very high-dimensional handcrafted features. Deep Learning is an emerging field in machine learning in recent years. A very promising characteristic of Deep Neural Networks (DNN) is that they can learn high-level invariant features from raw data (Vinola et al. 2015; Koolagudi and Rao 2012), which is potentially appropriate for emotion recognition classification. Zheng et al. (Zheng et al. 2015) constructed a Convolutional Neural Network (CNN) architecture to implement emotion recognition, the ultimate experimental results showed that their proposed approach outperformed the SVM classification. Zhao (Zhao et al. 2019) used a Long Short-Term Memory Network and CNN into one-dimensional CNN-LSTM network to recognize speech emotion from audio clips. The two-dimensional CNN-LSTM model focuses on learning global contextual information from handcrafted features, and achieved recognition accuracy of 52.14%. Perez-Rosas et al. (Pérez-Rosas et al. 2013) have shown that features such as prosody, voice, MFCC, and spectral, prove promising in identifying sentiment. While most of these works have extensively used the IEMOCAP dataset (Busso et al. 2008), Huang and Bao (A. Huang and Bao 2019) explored conventional feature extraction techniques like MFCC and STFT, and implemented CNN-based classifier which yielded the accuracy of 85% on the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset (Livingstone and Russo 2018). Kwon et al. (Kwon et al. 2003) implemented SVM and Hidden Markov Model (HMM) to process pitch, log energy, mel-band energies, MFCCs and velocity/acceleration of pitch. Their model received 96.3% accuracy for 2-class classification, and 70.1% for 4-class classification, proving that pitch and energy are the most contributing features. However, the result for 5-class classification is much lower. The lower performance of the models explored in the previous studies motivates us to formulate our research using deep learning method on augmented RAVDESS data to evaluate whether our method could achieve higher accuracy compared to the existing research.

3 DATA PREPARATION

Many of the state of the art speech emotion recognition models were mostly trained and evaluated on the IEMOCAP database (Busso et al. 2008) and the Berlin database (Burkhardt et al. 2005). We choose to train and evaluate our proposed speech

emotion recognition model using the RAVDESS dataset (Livingstone and Russo 2018). Firstly, this dataset has high levels of emotional validity and reliability of audio. Secondly, it has a large number of samples when compared to common speech databases (Busso et al. 2008; Burkhardt et al. 2005). RAVDESS is a multimodal database, including files in three modalities (audio-video, video-only and audio-only) and two vocal channels (speech and song). Each audio file is rated on emotional validity, intensity, and genuineness. In our study, we only extracted audio files as our purpose is the audio emotion recognition task. It is worth noting that the dataset is gender balanced, consisting of 24 professional actors, vocalizing lexically matched statements in neutral North American accent. The dataset contains 1440 audio files recorded in the ".wav" format and annotated into 8 classes of human emotions as shown in Table 1. The classes "Calm" and "Neutral" were selected as baseline conditions, while the remaining classes constitute the set of six basic or fundamental emotions that are thought to be culturally universal.

Table 1: Initial distribution of audio files.

Emotion Class	Number of Files
Neutral	96
Calm	192
Happy	192
Sad	192
Angry	192
Fearful	192
Disgust	192
Surprised	192
Total Files	1440

In Table 1, we observe two major issues in the RAVDESS dataset – 1) class imbalance, where the classes are not represented equally and 2) relatively small size for training and evaluation. To address this issue, there are a few techniques we can apply to deal with class imbalance such as over-sampling or adjusting the weight of cost function to balance the classes, etc. In our case, we choose to group similar class labels together to obtain better classification accuracy. To this end, we analyze the intensities of a random sample of audio files from each emotion class and group the classes with similar overall intensity into one class. The overall intensity of speech signal for male audio files is significantly different from those of female audio files. Also, emotions like happy and surprised have similar amplitudes. We grouped emotions with similar amplitude together to a total of 6 classes.

4 FEATURE EXTRACTION

Feature extraction is the first step in the process of implementing a speech emotion recognition model. A speech signal comprises spectral features and prosody features. Prosody features represent the pattern of speech signal like pitch, intensity and energy etc. Sometimes prosody features alone are enough to distinguish between emotions, but few emotions might be too close to be distinguished using prosody features alone. Feature extraction is accomplished by changing the speech signal to a form of parametric representation at a relatively lesser data rate for subsequent processing and analysis. Feature extraction approaches usually yield a multi-dimensional feature vector for every speech signal and there are different algorithms to parametrically represent the speech signal for the emotion recognition process, such as Perceptual Linear Prediction (PLP) (Kim and Provost 2013), Linear Prediction Coding (LPC) (Kim and Provost 2013) and Mel-Frequency Cepstral Coefficients (MFCC).

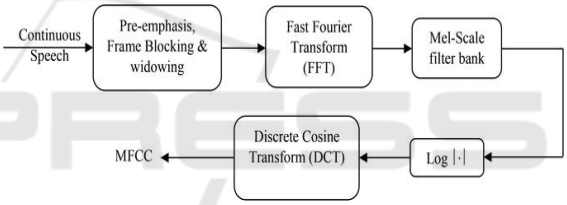

Figure 1: MFCC Process.

In this paper we use MFCC to extract features as it is the best representation of spectral properties of speech signal (Muda et al. 2010). In the computation of MFCC, the first step is **Pre-emphasis** where the signal is passed through a filter to emphasize higher frequencies. Next the speech signal is split into individual frames and this process is called **Framing**. The next step is **Windowing** to integrate all the closest frequency lines. To convert each frame from time domain to frequency domain **Fast Fourier Transform (FFT)** is applied to find the power spectrum of each frame. Subsequently, the filter bank processing is carried out on the power spectrum, using Mel-scale. Mel-scale was developed to overcome the linear interpretation of pitch by the human auditory system. It scales the frequency to match closely to what the human ear can perceive as humans are better at identifying small changes in speech at lower frequencies. With the Mel-scale applied, the coefficients will be concentrated only around the area perceived by humans as the pitch, which may result in a more precise description of a

signal, seen from the perception of the human auditory system.

$$F(Mel) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (1)$$

where F(Mel) is the resulting frequency on the mel-scale measured in mels and FHz is the normal frequency measured in Hz (see Figure 1). Next, the log Mel spectrum is converted into time domain using Discrete Cosine Transform (DCT). The result of the conversion is called Mel Frequency Cepstrum Coefficient. The speech signal after translating the power spectrum to log domain to calculate MFCC coefficients as shown in Figure 2.

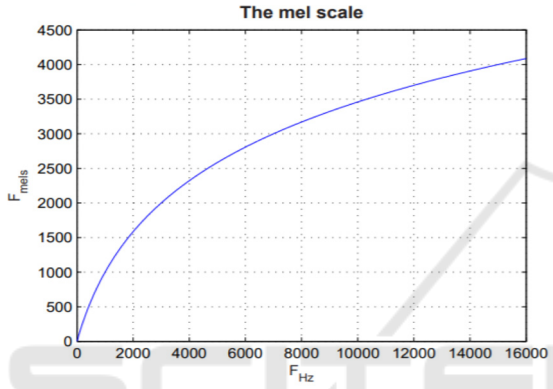

Figure 2: MFCC Frequency plot.

In our experiment, all audio files have been timed for a window of 3 seconds to extract length invariant feature sets using MFCC. We extracted 13 MFCCs per frame which is the default window sufficient to capture meaningful information for the model. The sampling rate of each file is doubled keeping sampling frequency constant to get more features from a small audio file. We select the 80:20 random split, which implies 80% of the samples are used for training our proposed models and 20% of samples for model validation.

5 TRAINING CNN MODELS

We build a CNN model with convolutional layer which is mainly used for image classification and object detection tasks, as a baseline model for speech emotion recognition. Convolution is a specialized type of linear operation used for feature extraction. The convolution generates a feature map with the help of kernel and tensor. Kernel is defined by a small array of numbers, while the tensor is the input array of numeric values. The kernel is applied on the input tensor where the element wise multiplication between each item of the kernel and the input tensor is determined. The product at each location of the tensor is summed up to yield a feature map which is the value in the corresponding position of the output tensor. The feature maps represent different characteristics of the input tensors; different kernels can, thus, be considered as different feature extractors. Two key hyper-parameters that define the convolution operation are size and number of kernels. As CNN captures spatial data and MFCC has coefficient information along the x-axis, and its value on the y-axis, capturing correlation with convolution will provide a reasonable and interesting approach.

Figure 3 depicts the architecture of our CNN model architecture. We implemented a trial and error approach to select hyper parameters for our model. We train the first CNN model (referred as CNN-1 model) on training dataset with Stochastic Gradient Descent algorithm to optimize the classification errors. Stochastic Gradient Descent (SGD) is the common optimizer that updates the parameters of learning such as kernel and weights to minimize the loss, i.e., the cost function that measures the difference between the prediction and the ground truth. It can be represented as the partial derivative of the cost function w.r.t the learnable parameters (i.e. the gradient) as shown in the following equation:

$$\omega = \omega - \lambda \frac{dL}{d\omega} \quad (2)$$

ω represents parameters, L represents the cost function (loss) and λ denotes learning rate.

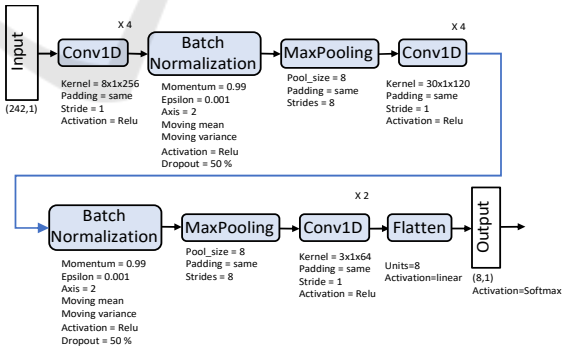

Figure 3: CNN Model Architecture.

We implement a trial and error strategy to select best hyper parameters. We re-trained the model with Adam optimizer (CNN-2) (Yamashita et al. 2018) keeping all other parameters same as CNN-1. Adam optimization requires first-order gradients with memory requirement. The method computes first and second moment of the gradients. The update of

learnable parameters is based on the estimates of first and second moments. The estimate of first and second moments and update of the learnable parameters are given by the following equations (Kingma and Ba 2014):

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_t) \cdot g_t \quad (3)$$

$$\gamma_t = \beta_2 \cdot \gamma_{t-1} + (1 - \beta_2) \cdot g_t^2 \quad (4)$$

$$\widehat{m}_t = m_t / (1 - \beta_1^t) \quad (5)$$

$$\hat{\gamma}_t = \gamma_t / (1 - \beta_2^t) \quad (6)$$

$$\omega_t = \omega_{t-1} - \alpha \cdot \widehat{m}_t / \left(\sqrt{\hat{\gamma}_t} + \partial\right) \quad (7)$$

$$g_t = \nabla_\omega \cdot L_t(\omega_{t-1}) \quad (8)$$

Where β_1 and β_2 represent hyper-parameters $\in [0, 1]$ which monitors the exponential decays of $\boldsymbol{m_t}$ and $\boldsymbol{\gamma_t}$ while $\boldsymbol{\alpha}$ is the learning rate.

Table 2 exhibits the hyper parameter details for both the gradient descent and Adam optimizer where the loss function is taken as categorical cross entropy for both CNN-1 and CNN-2.

Table 2: Hyperparameters for CNN Models.

CNN Model 1	CNN Model 2
Loss Function: Categorical Cross Entropy	Loss Function: Categorical Cross Entropy
Optimizer: SGD with α = 0.0001, momentum=0.0, decay= 0.0, nesterov = False	Optimizer: Adam optimizer with α = 0.0001, β_1=0.9, β_2= 0.999, amsgrad = False
Batch Size = 32 Number of Epochs = 500	Batch Size = 32 Number of Epochs = 700

6 TRAINING ConvLSTM MODELS

Since audio files are sequence data, we use a hybrid model which combines Convolutional Layers and Long Short-Term Memory (LSTM) layers to preserve the sequential relationship between audio frames. In fact, the ConvLSTM architecture consists of using Convolutional layers for feature extraction on input data combined with LSTM layers to support sequence prediction. Figure 6 illustrates the architecture of the ConvLSTM Model. Convolution layers in the beginning of the network help preserve the spatial patterns in the spectrogram followed by LSTM layers which capture the temporal information. The ConvLSTM estimates the future state of a certain cell in the tensor grid by the values of the input cells and previous states of the local neighboring cells of it. We use a convolution operator in the inter-state and input-to-state transitions to achieve this (Shi et al. 2015). The ConvLSTM inner structure is represented by the following equations (Shi et al. 2015).

$$i_t = \sigma(\omega_{xi} * x_t + \omega_{hi} * h_{t-1} + \omega_{cell_i} * \text{ cell }_{t-1} + \epsilon_i)$$
$$r_t = \sigma(\omega_{xr} * x_t + \omega_{hr} * h_{t-1} + \omega_{cellr} * \text{cell}_{t-1} + \epsilon_r)$$
$$\text{cell }_t = r_t^\circ \text{ cell }_{t-1} + i_t^\circ \sigma\begin{pmatrix}\omega_{x\_cell} * x_t + \omega_{h\_cell} * h_{t-1} \\ +\epsilon_{cell}\end{pmatrix}$$
$$c_t = \sigma(\omega_{xc} * x_t + \omega_{hc} * h_{t-1} + \omega_{cellc} * \text{ cell }_t + \epsilon_c)$$
$$h_t = c_t^\circ \sigma(\text{cell}_t)$$

Here x_t's are the inputs while $cell_t$'s are outputs of cells, h_t's are hidden states, i_t, r_t and c_t's are the tensors while r and c are the spatial dimensions of the cells. ω_x , ω_h , and ω_{cell} are the weights of the input data, the inputs to the hidden states and the cell outputs respectively. '*' and '°' are denoted as convolutional and Hadamard operator. The σ function is the sigmoid activation function. We used the hyper parameters from our baseline CNN-2 model to train our ConvLSTM network. Table 3 lists the hyper-parameters we used to train the ConvLSTM Model. We explored data augmentation techniques to overcome this issue in the next section.

Table 3: ConvLSTM Models Hyperparameters.

ConvLSTM Model
Loss Function: Categorical Cross Entropy
Optimizer: Adam optimizer with β_1= 0.5, β_2= = 0.99
Batch Size = 32, Number of Epochs = 250

Data augmentation is a key strategy adopted in scenarios where we have less training data. It also acts as a regularizer to prevent overfitting and reduce the effect of class imbalance making the entire model more robust. To resolve the problem of overfitting we used two data augmentation techniques which are pitching and noise injection. Random audio files are selected from the dataset to which noise is injected and pitch is varied. These random samples are augmented to the original dataset creating a larger and more randomized corpus. These techniques increased the sample size and sample variance thereby addressing the overfitting issue. We also used stratified shuffle split instead of random split to maintain similar distribution of records over classes in both train and test data. The audio corpus has increased from 1440 audio files to 4320 audio files, giving us more samples to train the model.

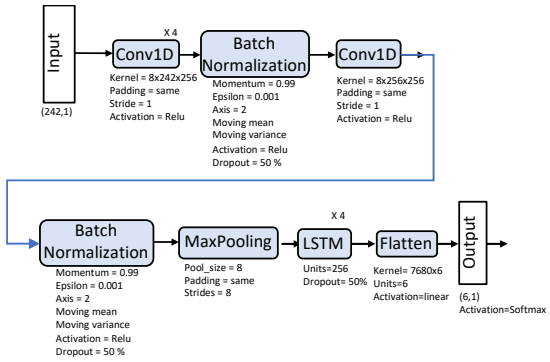

Figure 4: ConvLSTM Architecture.

Distribution of audio files after data augmentation is shown in Table 4. Retraining the ConvLSTM model using the augmented dataset significantly improved the performance of the model.

Table 4: Number of audio files after data augmentation.

Target Classes	Number of Audio Files
male positive	576
female positive	576
male negative	1152
female negative	1152
male neutral	432
female neutral	432

7 MODEL TUNING

Our initial convolutional neural network (CNN-1) with stochastic gradient descent optimizer resulted in a low validation accuracy of 55%.

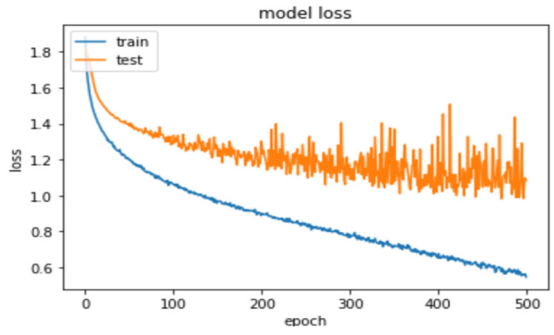

Figure 5: CNN-1 model training.

Figure 5 shows the training of CNN-1 model with stochastic gradient descent optimizer to exhibit the loss. But the same experimental setup with Adam optimizer (CNN-2) worked slightly better as a baseline model with a 60% validation accuracy.

Figure 6 shows the training of our baseline CNN-2 model with Adam optimizer. However, the model was overfitting with 98% training accuracy and 60% validation accuracy.

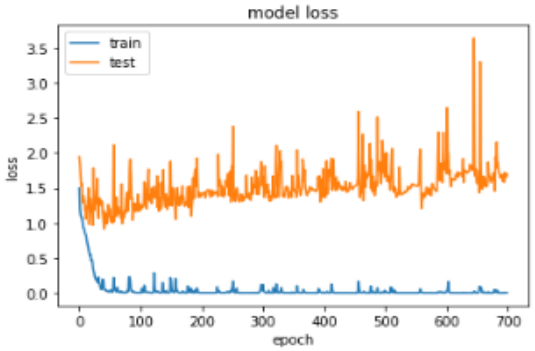

Figure 6: CNN-2 Baseline model training with Adam.

Similar overfitting issue occurred when we trained our ConvLSTM model which resulted in 99% accuracy on training dataset and 59% accuracy on test dataset (see Figure 7) before data augmentation.

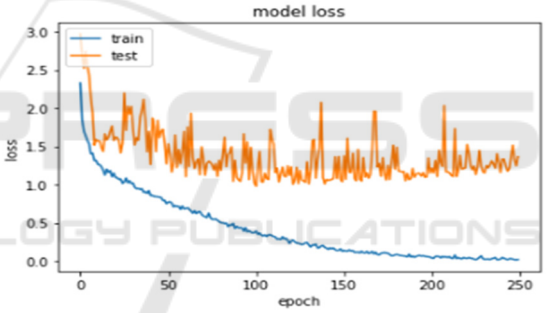

Figure 7: ConvLSTM training before data augmentation.

We addressed this overfitting by re-training our ConvLSTM Model with augmented datasets.

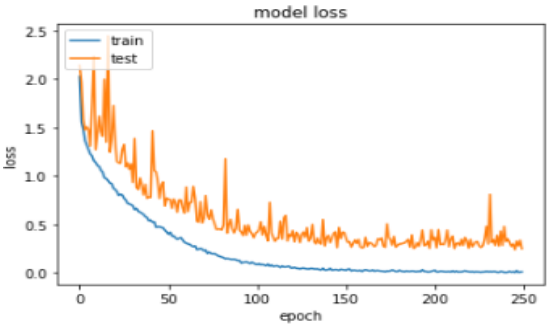

Figure 8: ConvLSTM training after data augmentation.

Thus, the classification accuracy was significantly improved to 91% and resolved the overfitting issue with a training accuracy of 99% as shown in Figure

8. For clarity Table 4 summarizes the findings from the experiment.

Table 5: Training / validation accuracy metrics.

Models	Training Accuracy	Validation Accuracy
CNN-1	93%	55%
CNN-2 with Adam optimizer	98%	60%
ConvLSTM without data augmentation	99%	59%
ConvLSTM with data augmentation	99%	91%

8 DISCUSSION OF RESULTS

Convolutional Neural networks are traditionally used to solve image related problems. We demonstrated the possibility of using convolutional neural networks in speech classification. It is also interesting to see how using a traditional feature extraction approach like MFCC with a combination of CNN and LSTM resulted in a far better accuracy than individual convolutional/recurrent neural networks. We have used four types of deep learning techniques i.e., CNN with stochastic gradient descent optimizer, CNN with Adam optimizer, Convolutional LSTM and Convolutional LSTM with data augmentation. Though the first three algorithms exhibit high training frequency, the validation accuracy remains poor. This leads to overfitting in the result. We resort to the fourth technique as we have less training data and data augmentation helps to resolves the overfitting issues observed in first three techniques.

Our ConvLSTM with data augmentation model performed significantly better with 91% accuracy for six class classification than the research in (A. Huang and Bao 2019) which achieved 85% accuracy using CNN based network for seven class classification for speech emotion using the same dataset. Hybrid model architecture proposed in this paper also performed better than that showed in (Satt et al. 2017) which implemented a hybrid model approach for emotion classification on the IEMOCAP database. Overall, this study explores the possibility of using CNN and ConvLSTM neural networks for speech emotion classification. Data augmentation techniques played an important role in overcoming the overfitting issues and resulted in high accuracy of ConvLSTM model for speech emotion recognition. Our work can be easily adapted to any speech emotion classification problems with length invariant audio files that will eventually facilitates the machine to identify the state of emotion of the speaker and hence generate a quality human computer interaction.

9 CONCLUSIONS

In the present study we proposed a deep learning framework to classify the emotions from the speech by segregating the data into six classes. We quantify the emotions extracted from male and female voices into positive, negative and neutral classes after regrouping the original eight emotion classes shown in Table 1. We introduced four deep learning models and found that convolutional LSTM with data augmentation achieved the best validation accuracy with minimum overfitting which is significantly better performance compared to the existing research on same RAVDESS dataset. For future work we will explore bi-directional LSTMs to classify the audio files. In addition, we will extend and test our models by collecting audio files recorded by random individuals and evaluate how our models recognize emotions. We will also build an end-to-end deep neural network using raw spectrograms to eliminate the feature extraction step. Additionally, availability of more training data and using length variant audio files to train models can lead to generalized findings.

REFERENCES

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W. F., & Weiss, B (2005). A database of German emotional speech. In 9th European Conference on Speech Communication and Technology pp.1517 - 1520,

Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., et al. (2008). IEMOCAP: Interactive emotional dyadic motion capture database. Language Resources and Evaluation, 42(4), 335-339.

Cao, H., Verma, R., Nenkova, A. (2015). Speaker-sensitive emotion recognition via ranking: Studies on acted and spontaneous speech. Computer Speech & Language, 29(1), 186-202.

Chandrasekar, P., Chapaneri, S., & Jayaswal (2014), D. Automatic speech emotion recognition: A survey. Int. Conference on Circuits, Systems, Communication & Information Technology Applications, pp. 341-346

Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W. Taylor J.G. (2001). Emotion recognition in human-computer interaction. IEEE Signal Processing Magazine, 18(1), pp. 32-80.

El Ayadi, M., Kamel, M. S., & Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. Pattern Recognition, 44(3), 572-587.

Fragopanagos, N., & Taylor, J. G. (2005). Emotion recognition in human–computer interaction. Neural Networks, 18(4), 389-405.

Huang, A., & Bao, P. (2019). Human Vocal Sentiment Analysis. NYU Shanghai CS Symposium, arXiv:1905.08632.

Huang, K.-Y., Wu, C.-H., Hong, Q.-B., Su, M.-H., & Chen, Y.-H (2019). Speech emotion recognition using deep neural network considering verbal and nonverbal speech sounds. IEEE International Conference on Acoustics, Speech & Signal Processing, pp.5866-5870)

Kerkeni, L., Serrestou, Y., Mbarki, M., Raoof, K., Mahjoub, M. A., & Cleder, C. (2019). Automatic Speech Emotion Recognition Using Machine Learning. In Social Media and Machine Learning: IntechOpen. DOI: 10.5772/intechopen.84856

Kim, Y., & Provost, E. M. Emotion classification via utterance-level dynamics: A pattern-based approach to characterizing affective expressions. In 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 3677-3681

Kingma, D. P., & Ba, J. Adam (2015): A method for stochastic optimization. International Conference for Learning Representations.

Koolagudi, S. G., & Rao, K. S. (2012). Emotion recognition from speech: a review. International Journal of Speech Technology, 15(2), 99-117.

Kwon, O.-W., Chan, K., Hao, J., & Lee, T.-W (2003). Emotion recognition by speech signals. European Conference on Speech Communication and Technology

Lalitha, S., Geyasruti, D., Narayanan, R., & Shravani, M. (2015). Emotion detection using MFCC and cepstrum features. Procedia Computer Science, 70, 29-35.

Lee, C.-C., Mower, E., Busso, C., Lee, S., & Narayanan, S. (2011). Emotion recognition using a hierarchical binary decision tree approach. Speech Communication, 53(9-10), 1162-1171.

Livingstone, S. R., & Russo, F. A. J. P. o. (2018). The Ryerson Audio-Visual Database of Emotional Speech and Song: A dynamic, multimodal set of facial and vocal expressions in North American English. PLoS ONE, 13(5):e0196391.

Mannepalli, K., Sastry, P. N., Suman, M. J. (2018). Emotion recognition in speech signals using optimization based multi-SVNN classifier. Journal of King Saud University-Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2018.11.012.

Mu, Y., Gómez, L. A. H., Montes, A. C., Martínez, C. A., Wang, X., Gao, H. et al. (2017). Speech emotion recognition using convolutional-recurrent neural networks with attention model. DEStech Transactions on Computer Science and Engineering. pp. 341-350

Muda, L., Begam, M., & Elamvazuthi, I. J. a. p. a. (2010). Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping techniques. Journal of Computing, 2(3), 138-143.

Noroozi, F., Kaminska, D., Sapinski, T., & Anbarjafari, G. (2017). Supervised vocal-based emotion recognition using multiclass support vector machine, random forests, and adaboost. Journal of Audio Engineering and Society, 65(7/8), 562-572.

Pérez-Rosas, V., Mihalcea, R., & Morency, L.-P. Utterance-level multimodal sentiment analysis. The 51st Annual Meeting of the Association for Computational Linguistics, Volume 1, 2013 (pp. 973-982)

Satt, A., Rozenberg, S., & Hoory (2017), R. Efficient Emotion Recognition from Speech Using Deep Learning on Spectrograms. Interspeech, pp. 1089-1093

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., & Woo, W.-c (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. International Conference on Neural Information Processing Systems, pp. 802-810.

Vinola, C., Vimaladevi, K. (2015). A survey on human emotion recognition approaches, databases and applications. ELCVIA: electronic letters on computer vision and image analysis, 14(2), 24-44.

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. Insights into Imaging, 9(4), 611-629.

Zhao, J., Mao, X., Chen, L. (2019). Speech emotion recognition using deep 1D & 2D CNN LSTM networks. Biomedical Signal Processing & Control, 47, 312-323.

Zheng, W., Yu, J., & Zou, Y (2015). An experimental study of speech emotion recognition based on deep convolutional neural networks. International conference on affective computing and intelligent interaction, pp. 827-831)