Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier xxxxx000

Music Emotion Recognition Using CNN-LSTM Architecture With Hybrid Spectral - Temporal Audio Feature Vector

PRATYUSH KERHALKAR[1], HARSH GUPTA[1], KUSH GUPTA[1], RAMYA S.[1] (SENIOR MEMBER, IEEE), AND KUMARA SHAMA.[1]

[1]Department of Electronics and Communication Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, India

Corresponding author: Ramya S. (e-mail: ramya.lokesh@manipal.edu).

ABSTRACT Music is a multi-layered, sound signal with numerous layers that can be utilized to express emotions. With the advent of machine learning techniques, Music Emotion Recognition has become one of the prominent research areas. This paper discusses the design exploration of the hybrid architecture considering static features, time series features, and image features. The hybrid model is implemented with parallel combination of CNN, LSTM and Deep Neural Network model. The models were trained and evaluated using two datasets namely Deam and Mturks. The result of the proposed architecture has shown improvement over the existing architecture in terms of accuracy metrics and learning capabilities. We successfully implemented a pipeline that clips audio files to a suitable length and extracts relevant features so that emotion(s) in the composition can be recognized by the system. When added to the recommendation pipeline, the system can provide better music recommendations based on the user's mood and can help improve the clustering of music based on emotions.

INDEX TERMS Music emotion classification, music emotion estimation, Music information retrieval, Machine learning.

I. INTRODUCTION

MUSIC is considered a complex combination of sounds. Through music, human emotions can be expressed effectively. Music has become an essential part of human social life, catering to all age spans. Music can be categorized in a variety of ways, such as by genre, artist, emotion, and other factors. Organizing music by emotion, on the other hand, is a useful strategy for allowing listeners to hear similar types of music at the same time. In recent years, the rapid rise of digital music data on the internet has resulted in a surge of consumer demand for search based on various types of meta-data. Modern music streaming has only raised the desire for music classification based on factors other than genre, such as sentiment. In such cases, Music Emotion Recognition (MER) can assist us in classifying music based on the emotion present at a specific point in the composition. The method entails extracting instrumental sections of the music, retrieving relevant attributes, and storing them in a machine-learning-friendly format for the model to use while running the prediction algorithm.

Present-day work in the field of MER includes recognizing temporal variation, and dominant feature selection methods which influence the listener in their perception of the emotions. Music is represented in multiple domains such as time, frequency, wavelet, etc. One of the forms to represent emotion in music is Valence - Arousal space. The Arousal values represent emotions ranging from calm(low) to excited(high). Valence, on the other hand, is the level of pleasantness that an event generates and is defined along a continuum from negative to positive. Valence-Arousal values can be represented in a two-dimensional space using Russell's two-dimensional Valence-Arousal space [1] where emotions are represented by points in the plane.

KR Tan et. al. [2] used Support Vector Machine(SVM) and Naïve Bayes algorithms to classify music emotion using Russell's two-dimensional valence-arousal space. SVM was used for audio features and Naïve Bayes for lyrical features. For the SVM - valence classifier, the model gave an F1 metric (in %) of 57 on a testing set of 26 songs. From the works, a conclusion was drawn that SVM arousal and NB valence models performed well and gave higher accuracies.

JM Brotzer et. al. [3] implemented a feed-forward neural network architecture to predict arousal and valence values on the Deam dataset. Audio features include MFCC, Chromagram of STFT, Mel-based Power Spectrogram, Octave-based Spectral Contrast, and Tonnetz, including another set of features extracted from essentia. The author concluded the experiment by stating recurrent neural networks would be beneficial for music emotion recognition.

In [4], a CNN-BiLSTM architecture was implemented to classify emotions in the dataset. The CNN pipeline consisted of two models with a feature combination of Mel spectrogram and cochleogram. Outputs of both the models are concatenated and fed into biLSTM model with arousal and valence as output. RMSE of 0.07 +- 0.05 for Arousal and 0.06 +- 0.04 for Valence was obtained as a result.

Richard Orjese et. al. [5] proposes a CNN-RNN architecture for music emotion recognition. The valence and arousal vary between -1 to 1 and the dataset consisted of 431 songs of 45 seconds each. An average RMSE of 0.217 +- 0.003 was obtained for the proposed model.

A regression approach is proposed in [6], which aims to categorize the emotion based on the dominant emotion in a music clip. The dataset consisted of 385 songs of 30-second each. Support Vector Regression (SVR) was used as the regression algorithm which predicted the two most dominant emotions in the music clip. The model gave the best accuracy of 84.5% on the exciting class of emotion while the accuracy for sad/calm emotion went as low as 47.6%.

The approaches mentioned in the previous research either use feed-forward networks or CNN methods for the classification problem, disregarding time series and static information or having a limited data set. We implemented a hybrid architecture to take into consideration the static features, time series features, and image features for the larger datasets.

II. METHODOLOGY

A. DATASET

The models were trained and evaluated using two datasets: Deam [7] and Mturks [8]. Deam dataset consists of 1802 45-second songs, whereas Mturks is made up of 1000 45-second songs. The datasets consist of valence and arousal values for each audio file. These values were then mapped to quadrants based on Russell's 2-D matrix for valence and arousal - (+VA, +AR) lies in quadrant 1, (+VA, -AR) lies in quadrant 2, (-VA, -AR) lies in quadrant 3, and (-VA, +AR) lies in quadrant 4. Each clip was then divided into 8 segments, each lasting 5 seconds. The last segment was discarded in order to maintain uniformity across all songs, owing to a disparity in the duration of the final clip of the songs across the datasets.

The segments were later filtered on the basis of the threshold technique where the clips having valence and arousal data points below a certain threshold (0.025) were filtered out. This eliminates the ambiguity in segments that lie very close to the axes. In order to reduce the imbalance between the emotion classes, the classes for quadrants 2, 3 and 4 were over-sampled to match the count of quadrant 1 (the highest number of samples). This process yields us a total of 25718 segments in the Deam dataset and 8043 segments in Mturks dataset. A split of 70, 10, and 10 was done for training, testing, and validation respectively using scikit learn [9]. Figures 1 and 2 show scatterplots for Deam and Mturks datasets respectively, and Table 1 outlines the data points across all the emotion classes for Deam and Mturks datasets.

TABLE 1. Data points across all classes.

Quadrant	Deam			MTurks		
	Train	Test	Validation	Train	Test	Validation
Q1	5221	1477	582	1573	454	194
Q2	4238	1194	496	1396	380	153
Q3	4983	1302	506	1562	423	166
Q4	4081	1173	475	1290	360	134

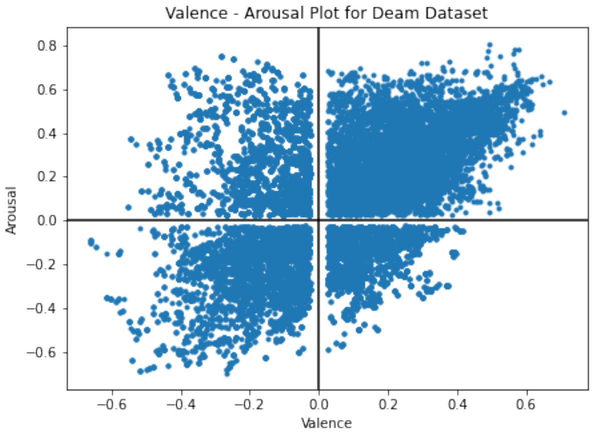

FIGURE 1. Valence – Arousal plot for Deam dataset.

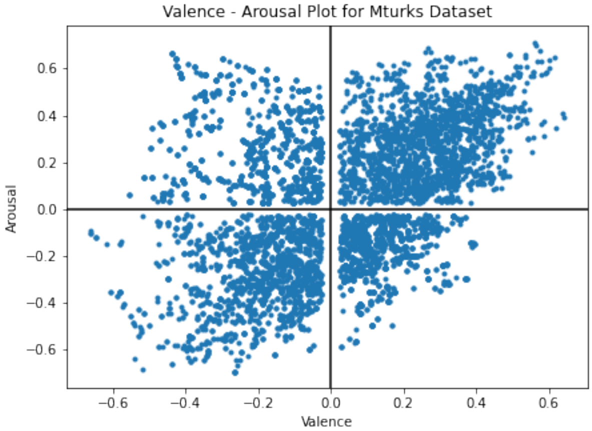

FIGURE 2. Valence – Arousal plot for MoodSwings Turk dataset.

TABLE 2. Features used and their shapes

Feature	Vector Shape	Description
Amplitude Envelope	(216, 1)	Change in the amplitude of a sound wave over time
Root Mean Square Energy	(216, 1)	Represents the intensity of an audio clip.Higher the RMS energy, the higher in tempo musical composition tends to be
Zero Crossing Rate	(216, 1)	The rate at which audio signal changes from positive to negative or vice versa
Spectral Centroid	(216, 1)	The centre of mass of the spectrum of audio signal
Spectral Spread	(216, 1)	Deviation around the centroid of the audio signal
MFCC	(13, 216, 1)	Representation of non-linear function of sound perception by the human ear
Delta MFCC	(13, 216, 1)	First order derivative of MFCC
Delta 2 MFCC	(13, 216, 1)	Second order derivative of MFCC
Static features	(10, 1)	Collation of mean and standard deviation of amplitude envelope, root mean squared energy, zero crossing rate, spectral centroid, spectral spread

B. FEATURES

The feature vector used for training the models is extracted and stored using LibROSA [10] and Pandas [11] respectively. and is a collation of temporal and spectral features. Temporal features include Amplitude Envelope(AE), Root Mean Squared Energy(RMSE), and Zero Crossing Rate(ZCR) while the spectral features consist of spectral centroid, spectral spread along with Mel Frequency Cepstral Coefficients (MFCC), Delta MFCC(a first-order derivative of MFCC), and Delta2 MFCC(second order derivative of MFCC). Additional derived static features have been generated by calculation of mean and standard deviation of amplitude envelope, root mean squared energy, zero crossing rate, spectral centroid, and spectral spread, giving a vector size of (10,1). Table 2 highlights the various vector sizes of input features. Training on these features has been done on Tensorflow [12].

C. MODELS

1) Convolutional Neural Network Model (Model 1)

A 3-layer, 3-channel convolutional neural network model was built for MER. The inputs consist of MFCC, Delta MFCC and Delta2 MFCC. The features were horizontally stacked with a shape of (13,216) where each MFCC was 13*216. The output layer consisted of 4 neurons and ReLU activation was used in the hidden layers and softmax was used in the output layer. Also, Adam optimizer was used for this model with a learning rate of 0.0001. The loss function chosen was the cross-entropy loss function or log loss function as shown in equation 1, where t_i is the true label and p_i is the softmax probability of i^{th} class of the 4 classes.

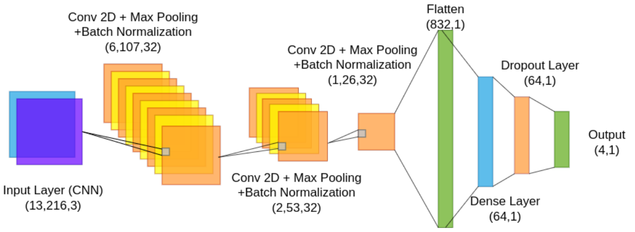

FIGURE 3. Model 1 Architecture.

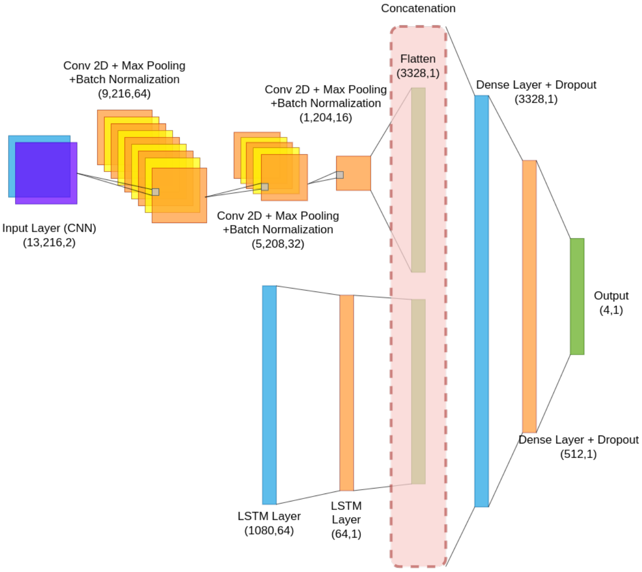

FIGURE 4. Model 2 Architecture.

The overall architecture is shown in figure 3.

$$L_{CE} = -\sum_{i=1}^{4}(t_i log(p_i)) \tag{1}$$

2) CNN-LSTM Model (Model 2)

The model consists of CNN and LSTM models running in parallel. The CNN model is trained on MFCC and Delta MFCC of 5-second sub-segments of songs. The input shape of two-channel training data is (13,216,2). The first 13 MFCC coefficients were selected for this task. The CNN architecture consists of 3 convolutional layers with max-pooling in hidden layers.

The LSTM model is fed with other time series features extracted – amplitude envelope, zero crossing, rmse, spectral centroid, and spectral spread. The model consisted of 2 LSTM layers and similar hyper-parameters were chosen for the model. The outputs of both the models were concatenated and fed into a 3-layer neural network with 1 hidden layer. Softmax activation was used in the output layer with 4 neurons. The architecture is shown in figure 4.

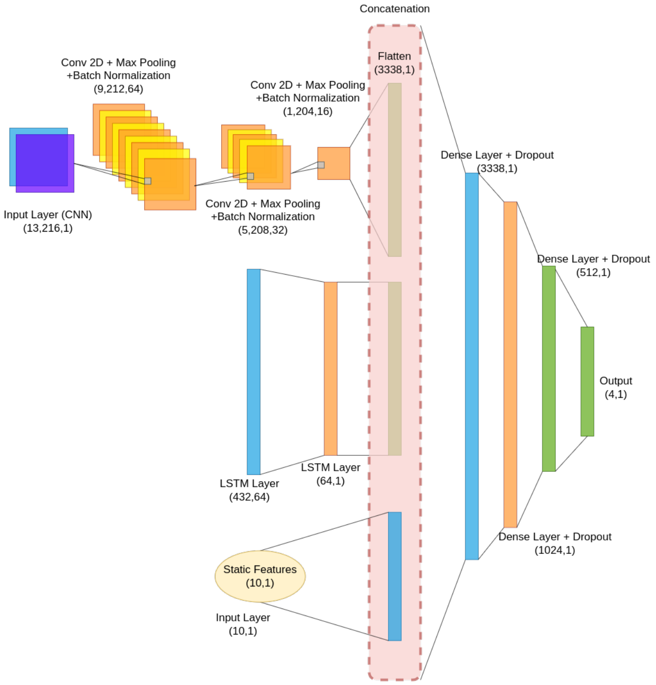

FIGURE 5. Model 3 Architecture.

3) CNN-LSTM Model with Static Features (Model 3)

To accommodate the static features of the music clip, a Deep Neural Network(DNN) layer was added to the previous CNN-LSTM model. Static feature is stacked with the outputs of CNN and LSTM to form a combined input to feed-forward neural network. Figure 7 shows the architecture of the described model. The model consists of CNN, LSTM and DNN layers. Visual representation of the first 13 coefficients of MFCC with size 13*216*1 was fed to CNN layers. The extracted Root Mean Square Energy (RMSE) and Zero Crossing Rate (ZCR) of each song were fed to the LSTM layer and static features such as standard deviation and mean of RMSE, ZCR, spectral centroid, spectral spread, amplitude envelope were given as the input for DNN layer. Zero crossing and RMSE were chosen for the LSTM model as they provide the amplitude and tempo of the song. The internal architecture for LSTM and CNN models was the same as model 2. The outputs of both the models were concatenated with the output of the DNN model which was further fed into a 3-layer neural network with 1 hidden layer. Softmax activation was used in the output layer with 4 neurons. Figure 5 shows the entire architecture of model 3.

III. RESULT ANALYSIS

Table 3 highlights the performance of each model on both datasets. An accuracy delta of 4% was observed between model 1 and model 2 when considering the performance on the Deam dataset. Model 3 generated improved results when compared to model 2 with a 3% improvement in both Deam and MTurks datasets in terms of accuracy.

There is a significant improvement in accuracy and F1 scores for models 2 and 3 as compared to model 1. Accuracy higher than 87% was achieved for testing and validation sets. This confirms that time-series data provided additional information and features which helped the model to improve its accuracy.

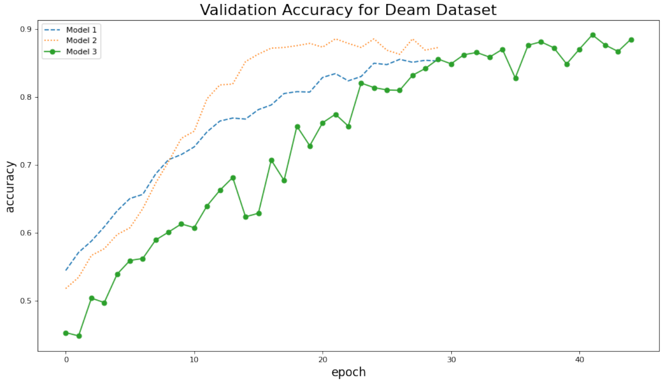

FIGURE 6. Training - Validation plot for Deam dataset.

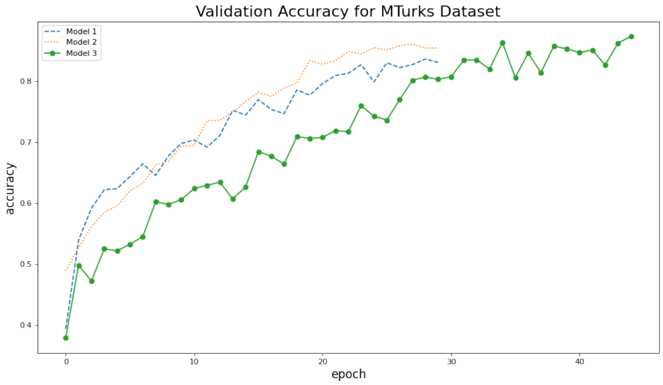

FIGURE 7. Training - Validation plot for Mturks dataset.

As seen from figure 6 and figure 7, the model taking the image, time series, and static features into account outperform model 2 by 3%, and model 1 by 10% on the Mturks dataset. These results confirm the initial hypothesis of the requirement of various images, time series, and static features for giving a better prediction. Figure 8 and figure 9 show the confusion matrix for model 3 on the Mturks dataset and the Deam dataset respectively. Based on the data, quadrant 1 had the most ambiguous values with 363 misclassifications for the Deam dataset.

TABLE 3. Performance summary for all the models.

		Model 1		Model 2		Model 3	
		Deam	Mturks	Deam	Mturks	Deam	Mturks
Test Accuracy (%)		82	81	86	84	**89**	**87**
F1 Score	Q1	0.79	0.81	0.82	0.80	0.83	0.83
	Q2	0.88	0.87	0.90	0.88	0.91	0.89
	Q3	0.89	0.81	0.90	0.86	0.92	0.89
	Q4	0.88	0.84	0.90	0.84	0.92	0.85

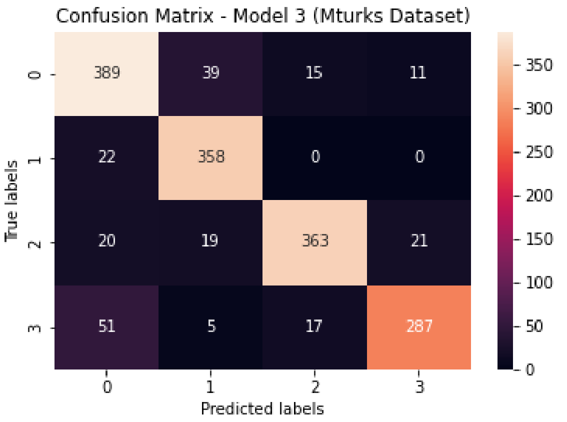

FIGURE 8. Confusion Matrix for Model 3 - Mturks Dataset

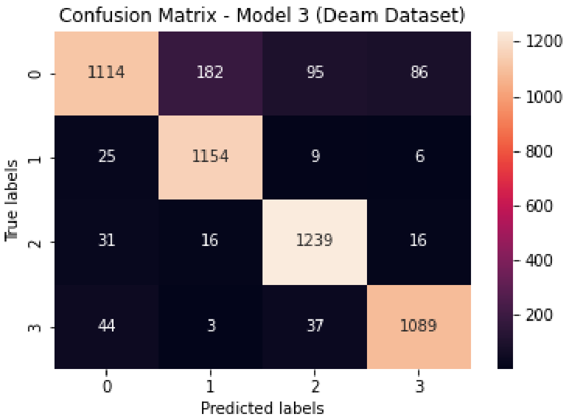

FIGURE 9. Confusion Matrix for Model 3 - Deam Dataset

IV. CONCLUSION AND FUTURE SCOPE OF WORK

MER is a part of Music Information Retrieval (MIR) which aims to determine the emotional characteristics of the music by applying machine learning and signal processing techniques. MER systems enable us to better music discoverability across streaming services. The results highlight that the image data itself was not enough to recognize the patterns and distinguish the emotions properly. Since music is a time series data, we added features involving time series information such as zero-crossing rate to identify the tempo, and amplitude envelope to identify the energy of each frame. To accommodate these time-series data, LSTM layers were added in parallel to the CNN layers to process both the image and time-series data. Our work proposed an architecture to allow static, time-series, and image data to be processed simultaneously. The result of the proposed architecture has shown improvement over the existing architecture in terms of accuracy metrics and learning capabilities. When added to the recommendation pipeline, can provide better music recommendations based on the user's mood and can help improve the clustering of music based on emotions.

Although our work has shown improvement over other traditional models and algorithms, there are certain improvements that can be considered in the future scope of the project. The inclusion of metadata and textual features such as lyrics can be beneficial for determining emotion. Using natural language processing techniques on the metadata, a rough estimation of mood can be determined. Additional data such as genre and artist might provide useful information and improve the efficiency and accuracy of the model.

References

[1] Yi-Hsuan Yang and Homer H Chen. "Machine recognition of music emotion: A review". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 3.3 (2012), pp. 1–30.

[2] KR Tan, ML Villarino, and Christian Maderazo. "Automatic music mood recognition using Russell's twodimensional valence-arousal space from audio and lyrical data as classified using SVM and Naıve Bayes". In: *IOP Conference Series: Materials Science and Engineering*. Vol. 482. 1. IOP Publishing. 2019, p. 012019.

[3] JM Brotzer, ER Mosqueda, and K Gorro. "Predicting emotion in music through audio pattern analysis". In: *IOP Conference Series: Materials Science and Engineering*. Vol. 482. 1. IOP Publishing. 2019, p. 012021.

[4] Pengfei Du, Xiaoyong Li, and Yali Gao. "Dynamic Music emotion recognition based on CNN-BiLSTM". In: *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*. IEEE. 2020, pp. 1372–1376.

[5] Richard Orjesek et al. "DNN based music emotion recognition from raw audio signal". In: *2019 29th International Conference Radioelektronika (RADIOELEKTRONIKA)*. IEEE. 2019, pp. 1–4.

[6] Yi-Hsuan Yang et al. "A regression approach to music emotion recognition". In: *IEEE Transactions on audio, speech, and language processing* 16.2 (2008), pp. 448–457.

[7] Anna Alajanki, Yi-Hsuan Yang, and Mohammad Soleymani. "Benchmarking music emotion recognition systems". In: *PloS one* (2016), pp. 835–838.

[8] Mohammad Soleymani et al. "1000 songs for emotional analysis of music". In: *Proceedings of the 2nd ACM international workshop on Crowdsourcing for multimedia*. 2013, pp. 1–6.

[9] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[10] Brian McFee et al. "librosa: Audio and music signal analysis in python". In: *Proceedings of the 14th python in science conference*. Vol. 8. Citeseer. 2015, pp. 18–25.

[11] Wes McKinney et al. "Data structures for statistical computing in python". In: *Proceedings of the 9th Python in Science Conference*. Vol. 445. 1. Austin, TX. 2010, pp. 51–56.

[12] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

PRATYUSH KERHALKAR received his Bachelor's degree in Electronics and Communication Engineering with a minor in Computational Mathematics from the Manipal Institute of Technology. He has a strong background in natural language processing, having previously worked with the Samsung Research Institute, Bangalore where he built NLP systems at scale. In addition to his expertise in NLP, Pratyush also has experience with modeling time series signals. He is currently working in the industry, leveraging his skills and knowledge to solve real-world problems.

HARSH GUPTA was born in India in 1999. He received his B.Tech. in Electronics and Communication Engineering with minors in Data Science from Manipal Institute of Technology, Manipal, India in 2021. In the same year, he worked at Samsung India as a Research and Development student, where he focused on computer vision and the enhancement of night sky images. In 2021, he joined Deloitte USI as an analyst, where he worked on project related to price elasticity modeling. He also worked on a project to design and develop ETL pipelines. His main areas of interest include computer vision and price elasticity models.

KUSH GUPTA completed his Bachelor's degree in Electronics and Communication Engineering with a minor in Data Science from Manipal Institute of Technology, Manipal. He has significant experience in Web development, having previously worked with multiple companies as an intern during his tenure as a student. In addition to his expertise in Web development, Kush also has a keen interest in Data Science, specifically Data Cleaning. Currently, he is working as a software engineer at Oracle, India

DR. RAMYA S. is Senior Member, IEEE ,working as Associate Professor in the Dept. of Electronics Communication, Manipal Institute of Technology, MAHE, Manipal, India. She received her B.E Degree in Electronics and Communication from Karnataka University, Belgaum, Karnataka and Master Degree in Microelectronics from Manipal Institute of Technology, MAHE, Manipal. She obtained her PhD from MAHE, Manipal in the field of online Kannada handwriting recognition. She has published several research papers in international journals and conferences. Her research areas of interest include Music , image and signal processing using Neuro-fuzzy techniques, Machine learning and Deep learning concepts. She had also worked as a software engineer in various companies in Bangalore, Karnataka

DR KUMARA SHAMA is a Professor in the Dept. of Electronics Communication Engineering at MIT, Manipal, with a teaching experience of 34 years. He pursued his Master's degree in Digital Electronics Advanced Communication from Karnataka Regional Engineering College and PhD in Speech Processing Applications in Medicine from MAHE, Manipal. He has published several articles in reputed international journals and also guided many research students. His research interest includes Digital signal processing, speech processing, Optical communication

• • •