# Audio and Text Based Emotion Recognition System using Deep Learning

Palash Thakur
and
Ronit Shahu
and
Vikas Gupta
Shri Ramdeobaba college of Engineering and Management, Ngapur, Maharashtra

Emotion is inherent in people, and hence emotion comprehension is a critical component of human-like artificial intelligence (AI). Because of its ability to mine opinions from a plethora of publicly available conversational data on platforms such as Facebook, YouTube, Reddit, Twitter, and others, emotion recognition in conversation (ERC) is becoming increasingly popular as a new research frontier in natural language processing (NLP). Meeting these demands necessitates the development of conversational emotion-recognition algorithms that are both effective and scalable. However, due to various research hurdles, it is a challenging subject to address. In this paper, we have demonstrated the implementation of audio and text-based emotion recognition algorithms. MELD dataset is used in this paper to train models. This paper demonstrates the working of both audio-based and text-based models individually as well as their respective results and the result of the combined model i.e. Audio + Text model. The text model performed worst with a weighted average F1 score of 52% whereas the Audio + Text-based model has performed best with a weighted average F1 score of 70%.

Keywords: Software maintenance, automated regression analysis, application baselining.

## 1. INTRODUCTION

In this modern age of technological advancement, people want everything to be smart. whether be it their phone, TV, AC or any other electronic device. While making such lifeless machine smart, a key factor is emotion recognition. Machines have to be smart enough to understand when the user is happy or sad, These critical issues can be achieved by human-computer interaction. Such HCI systems should not ignore any of the emotions as it would lead to a failure of the said system. The said system should be able to classify the human's emotion based on the conversation it is having and the text input it may sometimes receive from the user. The system should then classify the user's emotion and should give an appropriate response for the predicted emotion. The appropriate response may differ from system to system. For example, a chat box system may give an praising text or a apologetic text based on the user emotion. A virtual companion (google Assistant, SIRI, Alexa) may actually give the same output as the chat box system but since its voice based, user may find their voice more calming.

Audio and Text processing via deep learning is already an established domain of research in machine learning. A speech based emotion recognition system works in 2 parts. The first one extracts features from the given audio input. The second part is the classification of the emotion based on the extracted features. Classification process requires a substantial amount of knowledge on machine learning. This is because using different classifier based on our working condition will yield better accuracy than other algorithms. As a result our so called smart HCI system will better understand humans and it will in return will be a better system than others.

The first proposed model takes an audio file (.WAV) as input. Then the Algorithm used for feature extraction for audio input is OpenSMILE Eyben et al. [2010]. The features extracted from the above model is then passed to fully connected layer for classification. After the features

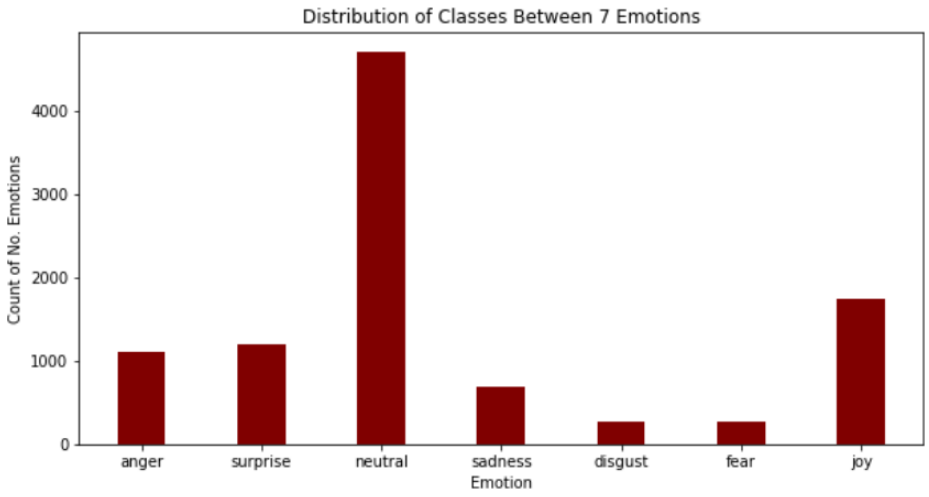


Figure 1. Bar Graph representation of dataset Among 7 Classes

are passed through the fully connected layer, the model classified the given audio input into 5 classes namely joy, sad, neutral, angry and surprised.

The second proposed model takes a text file (.TXT) as input. The algorithm used for feature extraction for this text file is word2vec Mikolov et al. [2013]. The features extracted from the above mentioned algorithm is then passed through fully connected layer. The model classified the given text input into the above mentioned 5 classes.

The final model uses both audio as well as the corresponding text file to the audio as an input. The feature extraction algorithms used are the same as mentioned above that is OpenSMILE for audio files and word2vec for text files. The feature extracted from these are concatenated using feature fusion. The fused feature from both the audio and text model are then passed through fully connected layer for classification. The model then classifies the given input of audio and text into the above mentioned 5 classes.

In Kaya et al. [2017], the authors decided that the speech based emotion recognition system should have less computing power as well as fast processing and high accuracy for it to be used in real time applications. The proposed model is divided into 5 parts namely face alignment, Pre-processing/CNN-training, feature extraction, modelling, and prediction. The data set used was EmotiW 2015/16 corpora. The data set included movie clips which represented close to real world conditions. The data set is divided into 3 parts namely training, development and test sets. The authors concluded that the recall value of fear and disgust is very low. The class imbalance made the learning issue of such automatic system very painful.

In PZhou et al. [2019], the authors used several techniques for providing a strong benchmark. These techniques include feature extraction using pre-trained 300 dimensional GloVe vectors and feeding it to 1D-CNN for textual features, OpenSMILE for audio features, textcnn, bclstm, Dialogue RNN. The data set used is MELD. It included 9,989 videos clips from a series to meet close to real life conditions. The Result is obtained by using a combination of different model. The best result is obtained by using DialogueRNN and text+audio model with a weighted average of 60.25%.

In ¡]c3, the authors divided the complete system into 4 parts namely video+audio processing, feature extraction, feature fusion and classifier. There are 4 datasets used to train the model namely AffectNet, RAF-DB, FER+, AFEW. The model has a testing accuracy of 62.328% making it second in EmotiW2019 challenge.

## 2. LITERATURE SURVEY

Many papers are referred in order to select the best as well as the strategy building a model. This paper mainly focuses on audio and text features of a speaker and hence In this paper we only account the audio and text parts of the referred papers and the facial parts are omitted.

In Kaya et al. [2017], The proposed model is divided into 5 parts namely face alignment, Pre-processing/CNN-training, feature extraction, modelling, and prediction. The facial alignment is done using PCA (principal component analysis). It basically dealt with false positives and rotated faces. After the PCA the images with high mean reconstruction error per image is discarded since the image is probably poorly detected or misaligned. For the CNN training part the authors used a pre-trained model via transfer learning and fine tuning it with the FER 2013 data set. For the pre-trained model the authors used VGG-Face model that is used for face recognition. For feature extraction from the already fine-tuned CNN model, the authors re scaled the images in 224 x 224 pixels and then normalized them by subtracting the average image of the VGGFace network. The authors concluded that the multi-modal system diminishes the results for natural/semi-natural data.

In Poria et al. [2018a], the authors initialized each token with pre-trained 300 dimensional GloVe vectors and then fed them to 1D-CNN to extract 100 dimensional textual features. For the feature extraction of audio input OpenSMILE toolkit is used. It extracted 6373 dimensional features. As the audio representation is high dimensional, the authors employed L2-based feature selection with sparse estimators like SVMs (support vector machines), to get a dense representation of the overall audio segment. Later, the audio and textual features are concatenated to obtain a bi-modal feature. Strong benchmark was to be given to MELD. In order to do that different models such as textcnn, bclstm and Dialogue RNN was used. Among these, the DialogueRNN was suited the best as it can handle multi-party conversations .The results obtained from such experiments are that the DialogueRNN with multi-modal variant gave the best performance i.e. 67.59% F-Score. It surpassed the bcLSTM which has a F-Score of 66.68%. The authors also concluded that the textual modality (57.03% F-Score) is better than the audio modality(41.79% F-Score) by 17%. For the positive sentiments, the audio model performed poorly. The multi-modal fusion did increase the emotion recognition by 3%. However the multi-modal classifier performed worse than the textual classifier while classifying sadness. The performance of some particular classes like sadness, disgust and fear are very poor. The main cause of this is the imbalance of the data points in these classes in the MELD dataset. The authors mentioned that the future works should be done on enhanced audio feature extraction so that the classification performance increases.

In Zhou et al. [2019], the authors divided the complete system into 4 parts namely video+audio pre-processing, feature extraction, feature fusion and classifier. For the video pre-processing part the authors used the Dlib toolbox and extended the face bounding box with a ratio of 30%. The cropped faces where then resized to 224 x 224 pixels. If no face was detected in an image the entire frame was passed in the network. For audio feature extraction, the speech spectrogram was passed through a hamming window with a 40 msec window size and 10 msec window shift. The 200 dimensional low frequency parts of the spectrogram was used for audio modality. The authors used 3 different CNN models to extract the facial features namely VGGFace, ResNet18, IR50. For the feature extraction of audio files they extract the feature maps of the audio from the last pooling layer of AlexNet.The size of a 3-dimensional feature map is H$\times$W $\times$C, where H,W are the height and width of the feature map, and C is the number of the channel of the feature map. The feature maps are then split into n vectors(n = H $\times$W ). Each vector is C-dimensional. The authors applied attention based fusion strategies for intramodel feature fusion. There were 3 attention methods namely Self-attention, Relation-attention and Transformer-attention. The authors used 4 different emotion dataset namely AffectNet, RAF-DB, FER+, AFEW. The AffectNet dataset had emotion labels. The RAF-DB dataset had both 7-class basic and 12-class compound emotion label in them. Only the basic emotion label were used. The FER+ and AFEW did not have

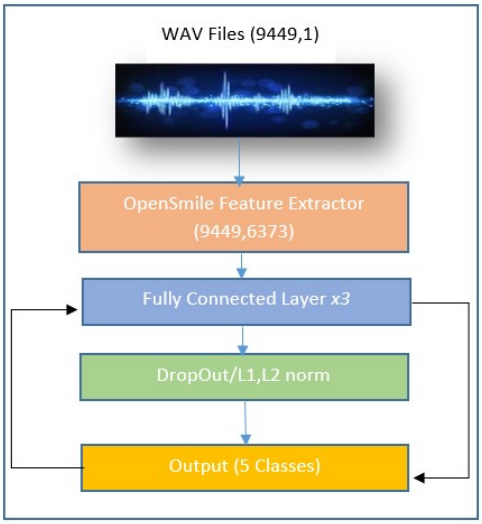


Figure 2. Block Diagram Representation of Audio Model

such labels. When all the 3 trained model i.e. the ResNet18, the IR50 and the VGGFace were compared, the authors found that the IR50 was superior to the rest. The well trained IR50 was then used on all the 4 above mentioned dataset and it was found that the IR50 when pre-trained on AffectNet gave the highest result of 53.78%. Hence it was used as the visual feature in the fusion experiments. The authors used speech spectrogram for Audio CNN which gave 38% on AFEW validation set. Log Mel-Spectrogram was also used which gave a slight better accuracy. The authors concluded that the 3 main intra-modal fusion techniques namely self-attention, Relation-attention and Transformer-attention was used which gave better accuracy. For fusion of Audio and Video feature, feature concatenation and factorized bi-linear pooling was used. The result obtained was 62.48% and ranked 2nd in EmotiW 2019 challenge.

All the paper that was referred had tremendous amount of work done in the field of emotion recognition using facial feature and almost 10% work done in audio and text based emotion recognition when compared to the facial one. This showed that the main focus was emotion recognition via facial features and not via audio/text features.

## 3. DATASET

### 3.1 Selection of Dataset

Many available datasets in multimodal sentiment analysis and emotion recognition is non-conversational. But IEMOCAP, SEMAINE and MELD are one of the most popular conversational datasets where each utterance in a dialogue is labeled by emotion.

The McKeown et al. [2011] dataset is an audio-video based database which was created for building agents that can engage a person in an emotional and sustained conversation. It involves conversation between a human and an operator. The operator can be a machine or a human simulating a machine. The dataset contains 150 participants, 959 conversations. Each of such session last around 5 minutes.

The Interactive Emotional Dyadic Motion Capture Database (IEMOCAP)Busso et al. [2008] consist videos of conversation among a pair of 10 people lasting for 10 hours of various dialogues scenarios. The videos are divided into utterances with fine-grained emotion categories such as anger, happiness, sadness, neutral, excitement, and frustration. IEMOCAP also has continuous attributes namely activation, valence and dominance. The labels in IEMOCAP dataset was annotated by more than 2 annotators per utterance.

The Multimodal EmotionLines Dataset (MELD) Poria et al. [2018b] was created by extending

EmotionLines dataset and enhancing it further more. It consists audio and visual modality along with text. MELD has more than 1400 dialogues and 13000 utterances from a famous T.V series named "F.R.I.E.N.D.S". Multiple speakers participated in some dialogues. Each utterances in any dialogue is labeled as any of the 7 emotion viz anger, disgust, sadness, fear, happy, surprised and neutral. MELD also has coarse-grained sentiments such as positive, negative and neutral for each utterances.

### 3.2 Analysis of MELD Dataset

MELD has 7 emotions classes namely anger, surprise, neutral, disgust, fear, joy and sadness. The class data distribution of each of the above mentioned class is given in fig. xxyxyyxyyxyxyx. The dataset include video clips of a famous sitcom named "F.R.I.E.N.D.S". These said clip ranges from 2 to 13 sec having a resolution of 1280 x 720 pixels. The total number of video files is 15,907. The dataset was clearly very imbalanced as the majority of data-points was belonging to neutral class. These fine-grained emotion labels were converted to more coarse-grained sentiments. The classes anger, fear, disgust and sadness was considered negative. The class of joy was considered as positive and class of neutral as neutral. Surprise is a complex emotion since it can be considered as both positive and negative sentiment. The entire task of sentiment annotation reaches a Fleiss' kappa score of 0.91. On average, 3 emotion was present in each of the dialogues of dataset. The average utterances duration is 3.59 sec. The emotion shift of the speaker in any dialogues makes emotion recognition task in speech very difficult.

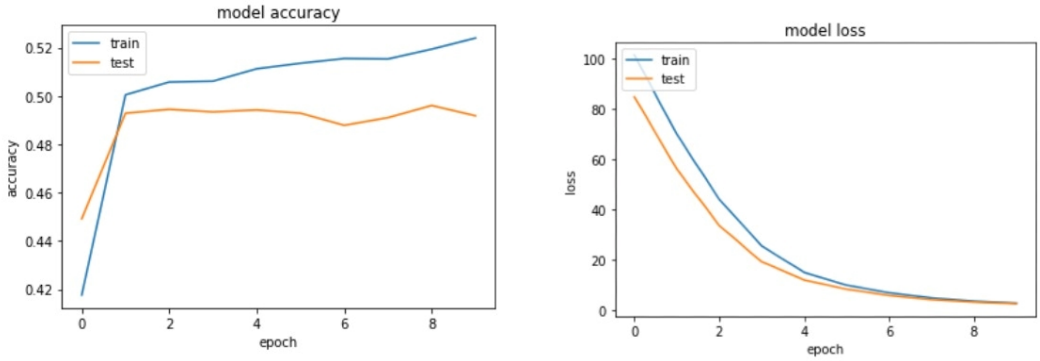


Figure 3. Accuracy and Loss of Audio Model

### 3.3 Preparation of MELD Data set for Training

MELD dataset is basically a conversational dataset. It has visual, audio and textual information provided for each speaker. Since this paper mainly focuses on audio and textual features of speaker, the visual data is omitted and only audio is extracted from video. Every video file has a dialogue associated with it. These dialogues are used for preparation of text dataset. The dataset is highly imbalanced. Disgust and fear are two emotions which contribute to less than 3% of total data. Training the model with entire dataset showed that these two classes are resulting in zero recall values. These two emotions are removed and training and testing is done on only 5 classes namely anger, surprise, neutral, sadness and joy. 9,449 samples are used for training and validation . Testing is done on 2,000 samples. The distribution of dataset for training, validation and testing is explained in table 1. 20% data is used for validation.

| **Training** | **Validation** | **Testing** |
|---|---|---|
| 7559 | 1890 | 2000 |

Table I: Distribution of dataset for Training, Validation and Testing

For the purpose of testing every model that is trained follows a same sample distribution of dataset across different classes. The distribution of data used for testing follows same proportion of data that is kept for training. The distribution of samples across every class is shown in table 2.

| Emotions | Count |
|---|---|
| Anger | 226 |
| Joy | 390 |
| Neutral | 1005 |
| Sadness | 115 |
| Surprise | 264 |

Table II: Distribution of class samples for Testing

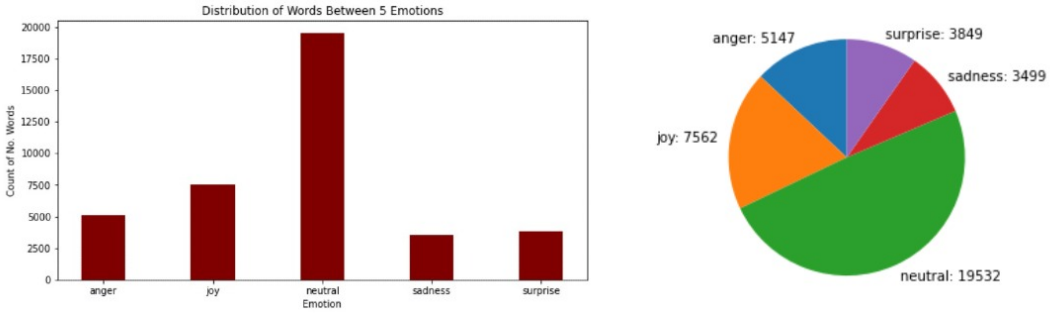


Figure 4. Bar Graph and Pie Chart Representation of Distribution of Words Among 5 Classes

| Utterances | Emotions |
|---|---|
| also I was the point person on my company's transition from the KL-5 to GR-6 system. | Neutral |
| My duties? All right. | Surprise |
| What?! What is with everybody? It's Thanksgiving, not...Truth-Day! | Anger |
| Do I ever. | Joy |
| Oh, totally. Oh, God, oh, she seemed so happy too. | Sadness |

Table III: Tabular Representation of Raw Text Data

## 4. REGULARIZATION TECHNIQUES

The multi-model approach used in this paper uses audio and text features combined. For better results and better accuracy, transfer learning has been used. The feature extractor outputs the data in very high dimensional vector space which results in model overfitting and high variance while training the model. To solve this problem without loosing important features, this paper has used some of the well known regularization techniques.

### 4.1 DropOut

While training a large neural network, one major issue in learning is co-adaption.

(1) As the network is trained iteratively, powerful connections are learned more while weaker ones are ignored.
(2) After many iterations, only a fraction of node connections participate.
(3) DropOut is a regularization method to address this issue.

Training Phase: Ignore (zero out) a random percentage p of nodes for each hidden layer, each training sample, and each iteration (and corresponding activations).

Test Phase: Use all activations, but scale them down by a factor of p. (to account for missing activations) The p value used in this paper is 0.3.

### 4.2 $L_1$ and $L_2$ Norm

The L1 norm will cause certain weights to be zero, resulting in weight sparsity. This can be useful for memory efficiency or feature selection (i.e. we want to select only certain weights). The L2 norm, on the other hand, will drop all weights but not all the way to zero. This is less memory efficient, but it may be advantageous if we want or need to keep all arguments.

## 5. AUDIO MODEL

MELD has audio infused with video.There are no separate audio files provided.We used python script in order to extract audio from every single video. The extracted audio files are in WAV(Waveform Audio File) format.By doing so,we created a entirely new audio dataset with corresponding emotion to it.There are total 9,449 audio files in the dataset.The sole task of this model is to detect emotion entirely from audio of a person.MELD provides us with conversational dataset where more than two people are talking with each other but one at a time.While training audio model,the base assumption is to predict emotion of a single person while he is speaking to other person.OpenSmile is used as feature extractor for the purpose of transfer learning.openSMILE (open-source Speech and Music Interpretation by Large-space Extraction) is an open-source toolkit for audio feature extraction and classification of speech and music signals.

### 5.1 Training of Audio Model

The audio files are passed through OpenSmile for feature extraction. OpenSmile expects input to be a WAV file. Dimension of feature vector after feature extraction is (9449,6373).Since the feature vector is high in dimension,we normalized data for faster training and to improve accuracy.The block diagram representation of audio model is given in figure 1.

The input to the model is 6,373 dimensional vector space and output is probability distribution among 5 classes namely sadness,fear,neutral,surprise and joy. Due to high dimensionality of feature vector, model was overfitting resulting in high variance between training and validation data. To prevent model from overfitting, Drop out and $L_1$ and $L_2$ norm like regularization techniques are used. The activation function used is ReLU activation and Adam optimizer is used for optimization of model with learning rate of 0.0001 and reducing learning-rate by a factor of 0.2 once learning stagnates and there is no improvement even after few epoch. The audio model is trained for 10 epochs. The training accuracy of audio model is 52.22% and validation accuracy is 50.40% . The graph of accuracy and loss is shown in figure 2.

$$new\_learning - rate = 0.2 * old\_learning - rate \tag{1}$$

MELD contains multiple group discussions. Binary variants found in previous databases are more difficult to distinguish. There are more than 13,000 expressions in MELD, making our dataset nearly twice the size of datasets for most current conversations. MELD provides multidisciplinary resources and can be used in a multidisciplinary discussion program to enhance core learning. There are 260 different speakers in the MELD utterance data. The aim of this model is to solely predict emotion simply from text irrespective of speaker's name or expression. While making this as base assumption for building data pipeline, we considered only utterance and emotion corresponding to that person. There are 9,449 utterance's samples just for training and validation while 2,000 samples are kept aside for testing. There are total 39,589 words in all utterances. The distribution of words across every emotion is given in bar chart as well as pie chart in figure 1. It is clear from this graphical representation that the neutral class emotion

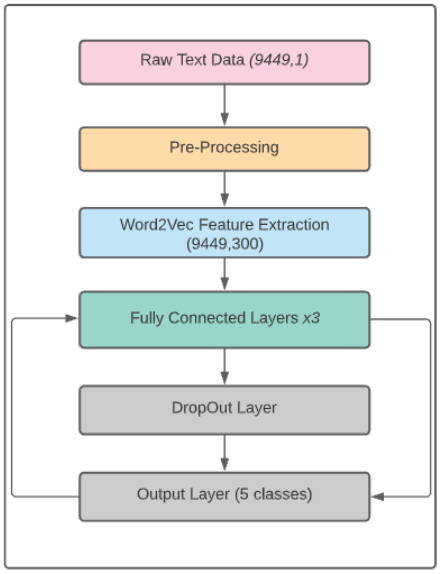


Figure 3. Block Diagram Representation of Text Model

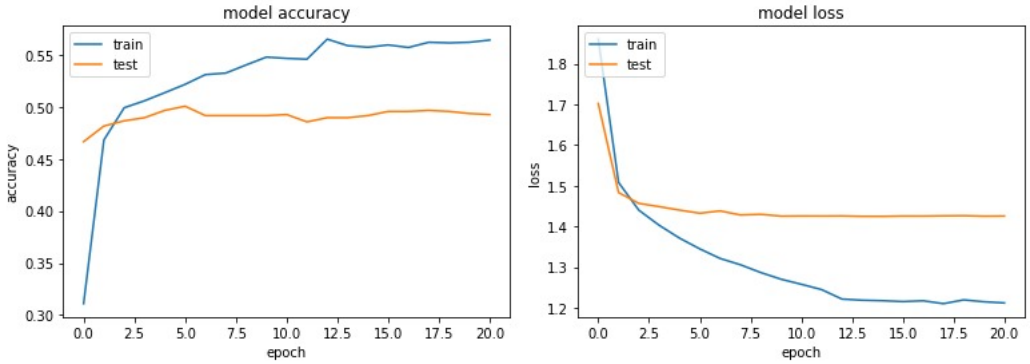


Figure 3. Accuracy and Loss of Text Model

is dominant and is occupying almost 50 percent of total data. Sample utterances from training data across every emotion is given in table 1.

## 5.2 Training of Text Model

We used the word2vec model for transfer learning for training text model and feature extraction. The word2vec algorithm uses a neural network model to learn word associations in a large collection of text. Once such a model is trained, it can find words with the same meaning or suggest additional words for a piece of sentence. As the name suggests, word2vec represents each unique word with a specific list of numbers called vectors. The vectors are chosen so carefully that the simple mathematical function (cosine similarity between vectors) indicates the degree of semantic similarity between the words represented by these vectors. Word2vec is a collection of related models used to create embeddings. These models are shallow, two-layer neural networks trained to reconstruct grammatical forms of words. Word2vec takes this as the inclusion of a large chorus of text and produces a vector space, usually several hundred in size, where each unique word in the chorus is given a corresponding vector in space. The words are placed in the vector position so that the words that share common formations on the sentence are side by side in space.

The word2vec model gives a 300-dimensional vector space. Word2Vec is a state-of-the-art model and requires preprocessing of text before being fed into the word2vec model. This preprocessing includes removing stop words and punctuation, tokenizing, and then averaging.

The extracted features are then fed into fully connected layers and tested with several newly introduced activation functions. The output of this model is the probability distribution among the 5 classes obtained using the softmax activation function. Due to high dimensionality of feature vector, model was prone to overfitting and was overfitting quickly. Overfitting causes reduced generalization ability of model and results in high variance of test data. To avoid overfitting and the problem of high variance, some of the regularization techniques such as dropout as well as $L_1$ and $L_2$ norm are introduced in model to converge model's training accuracy and validation accuracy.

The block diagram of text model is shown in figure 2.The dimension of feature vector before preprocessing was (9449,1) while the dimension after preprocessing became (9449,300).The text model follows the same data distribution as explained in table 1. The text model is trained for 20 epochs. The training accuracy of audio model is 57% and validation accuracy is 49.65% . The graph of accuracy and loss is shown in figure 5.

## 6. AUDIO + TEXT MODEL

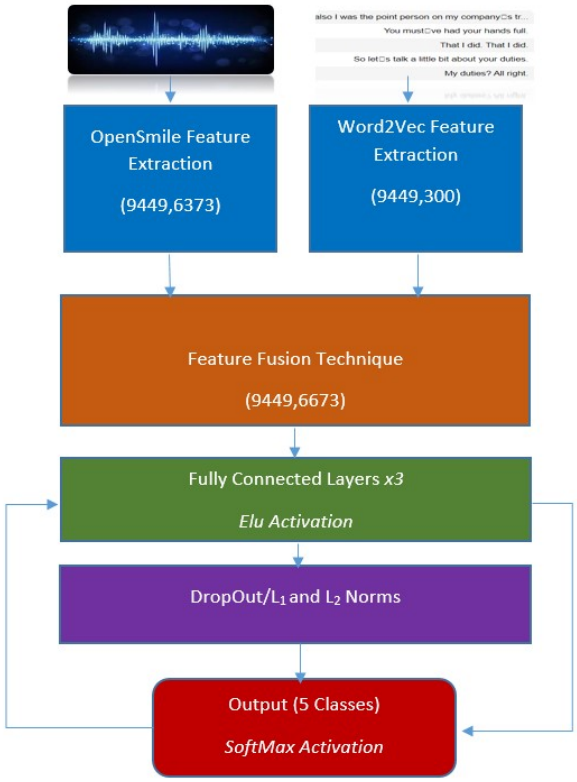


Figure 3. Block Diagram Representation of Audio+Text Model

The main aim of this paper is to demonstrate the multimodal approach used in building this model. This paper mainly focuses on two major approaches of detection of emotions i.e. detection of emotion from audio of the speaker and the words spoken by same speaker.So far both approaches are explained individually. In the multimodal approach, we aim to combine

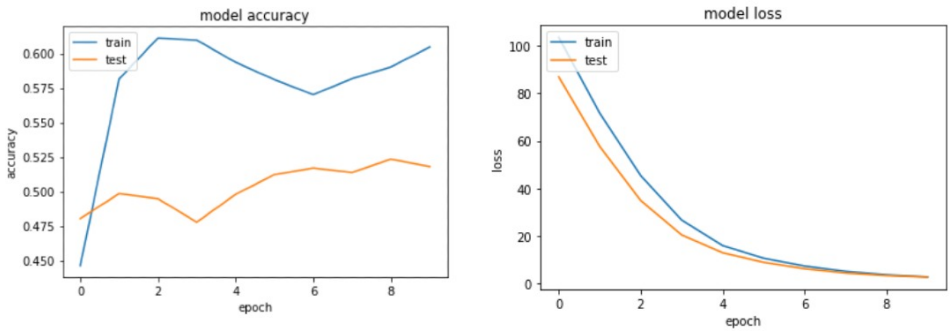


Figure 3. Accuracy and Loss of Audio + Text Model

| Model | Emotions | | | | | | |
|---|---|---|---|---|---|---|---|
| | Anger | Joy | Neutral | Sadness | Surprise | Weighted f1 | Correct Predictions |
| Audio | 72% | 45% | 84% | 8% | 62% | 68% | 1328 |
| Text | 19% | 38% | 71% | 23% | 38% | 52% | 1040 |
| Audio + Text | 63% | 64% | 80% | 18% | 67% | **70%** | 1070 |

Table IV: RESULT SUMMARY OF AUDIO, TEXT AND AUDIO + TEXT MODEL

both the models and create a robust and more accurate emotion detection system which can identify emotions into five classes with more confidence.The combination of models can be done on the basis of the their output. Using ensemble based techniques are other option to average the output probability of both the model and result the output with highest probability. But the problem with this approach is the model will not be able to learn the relationship between audio and text features as well as with the corresponding emotions. To avoid this problem, this paper uses feature fusion strategies. Feature fusion strategies are some mathematical equations used to merge different linear features from different models.There are some strategies predefined for feature fusion. Based on performance of different strategies feature concatenation is used to build the model. Some of the feature fusion strategies are :

(1) Feature Concatenation Concatenating two features by their columns.

$$X := X_{Audio}(9449, 6373) + X_{Text}(9449, 300) \tag{2}$$

$$Dimension(X) = (9449, 6673)$$

(2) Maximum value extraction

$$X[i][j] = MAX(X_{Audio}[i][j], X_{Text}[i][j]) \tag{3}$$

$$Dimension(X) = (9449, 6373)$$

(3) Product of two features

$$X[i][j] = X_{Audio}[i][j] * X_{Text}[i][j] \tag{4}$$

$$Dimension(X) = (9449, 6373)$$

After testing above strategies while building this model, feature concatenation is outperforming all the other strategies and it is used to as a final feature fusion method.

### 6.1 Training of Audio + Text Model

The block diagram representation of audio + text model is given is figure 4. Features are extracted at individual stages from both audio and text files. Extracted features are then stacked by column extending entire feature vector length to 6,673 containing both audio and text features. The

features that are extracted comes from both audio and text and hence they have different normal distribution. Since the features are combined, a new normal distribution must be defined. Hence entire feature space containing 6,673 features are normalized again making normal distribution over 6,673 features. After normalizing data, the next step is training the model. Fully connected layers are used to simply train the model. The activation function used in fully connected layers is "ELU" activation . ELU stands for "Exponential Linear Unit" .

$$f(x) = x, \text{if } x \geq 0 \alpha(e^x - 1), \text{if } x < 0 \tag{5}$$

The ELU hyperparameter $\alpha$ determines how much an ELU saturates for negative net inputs. The vanishing gradient effect is mitigated by ELUs. ELUs have negative values, which brings the activation mean closer to zero. Mean activations closer to zero provide for quicker learning by bringing the gradient closer to the natural gradient. When the argument becomes smaller, ELUs saturate to a negative value. Saturation denotes a tiny derivative that reduces the variance and information conveyed to the following layer. The $\alpha$ value used here is 1.0. The optimizer used is Adam optimizer with learning rate of 0.0001, reducing learning-rate by a factor of 0.2 once learning stagnates and there is no improvement even after few epoch. The dimension of feature vector is very high resulting in model overfitting and high variance between training and validation data. To avoid this problem, DropOut as well as $L_1$ and $L_2$ norm have be used for regularization. The final layer of model outputs probability distribution among 5 classes using softmax activation function. The audio+text model is trained for 10 epochs. The training accuracy of audio model is 60% and validation accuracy is 53.60% . The graph of accuracy and loss is shown in figure 5.

## 7. COMPARISON OF RESULTS

Every model is tested on 2,000 samples having different number of test samples across each class. The accuracy metric used in this paper to compare model's performance is weighted F1-score. The formulae of calculating F1-score is given in equation 6.

$$F_1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{6}$$

where,

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TN}{TP + FN} \tag{8}$$

Precision is the proportion of true positive cases among those labelled as positive by the model. Recall, also known as sensitivity, is the proportion of positive instances among the total number of positive examples. The weighted average of each model is calculated by multiplying F1 score of each class with count of samples across that class then adding them and finally dividing them by total number of samples which is 2,000. F1 score gives us the balance between precision and recall. The audio model has highest prediction accuracy with correct prediction of 1328 out of 2000. But the model is not confident while making prediction. The audio+text model has predicted 1070 samples out of 2000 samples correctly and has highest F1 score of 70% as shown in table 2. The model is most confident when making a decision in this case. The detailed result summary of all the models is given in table 2. The two emotions that were dropped before preparing dataset has zero recall value. The reason is they are very few in number as compared to others.

## 8. CONCLUSIONS

The MELD dataset is a multiparty conversational dataset which gives data samples across all the 7 major emotions. The major issue with the dataset is that it has highly imbalanced distribution of samples across every class as shown in figure 1. Neutral class is consuming more than 50% of dataset resulting in every model being more biased towards neutral class. Balancing such type of dataset is complicated. With dataset being biased towards neutral class, audio model has highest F1-score of 84% towards neutral class. Apart from neutral class, disgust and fear are two emotions having less than 3% of samples and when trained along with other emotions, none of the model is able to pick them as positive samples resulting in zero recall and precision values. The model is hence trained only on 5 emotions. The audio model performed best while making a prediction i.e. it made total of 1328 correct predictions and text model performed worst on making prediction as well as low F1-score. Overall combined model i.e. audio + text model is providing a good balanced result, both in weighted F1-score and accuracy. The weighted F1 score of the audio+text model is 70% . Combining audio and text model increased the audio+text model's confidence by 2% over audio model and 18% over text model without sacrificing accuracy. Audio+Text model is able to pick samples across every class with moderate confidence. One of the reason for low accuracy is imbalanced class dataset. If dataset can be balanced, then accuracy will be improved. Good feature extractor's output is higher dimensional vector space resulting in model's overfitting over a few epoch. The controlled training is done in this paper along with regularizing the model, hence avoiding overfitting even with high dimensional input. Text model is resulting in lowest F1 score of 52%. It can be improved with some advanced and enhanced feature extractors. If text model's accuracy is improved, then the overall audio+text model's accuracy will be improved.

## References


Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., and Narayanan, S. S. 2008. Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation Vol.42,* No.4.

Eyben, F., Wöllmer, M., and Schuller, B. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. *Language resources and evaluation*.

Kaya, H., Gürpinar, F., and Salah, A. A. 2017. Video-based emotion recognition in the wild using deep transfer learning and score fusion. *Image and Vision Computing Vol.65,* No.4.

McKeown, G., Valstar, M., Cowie, R., Pantic, M., and Schroder, M. 2011. The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent. *IEEE transactions on affective computing Vol.3,* No.1.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., and Mihalcea, R. 2018a. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv preprint arXiv:1810.02508*.

Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., and Mihalcea, R. 2018b. Meld: A multimodal multi-party dataset for emotion recognition in conversations. *arXiv preprint arXiv:1810.02508*.

Zhou, H., Meng, D., Zhang, Y., Peng, X., Du, J., Wang, K., and Qiao, Y. 2019. Exploring emotion features and fusion strategies for audio-video emotion recognition. *2019 International Conference on Multimodal Interaction*.