National College of Ireland

Speech capable Customer Service Bot with Video-Audio based Emotion Recognition

MSc Research Project
Data Analytics

Karthi Mahendran
Student ID: X20118198

School of Computing
National College of Ireland

Supervisor: Jorge Basilio

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Karthi Mahendran
Student ID:	X20118198
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Jorge Basilio
Submission Due Date:	16/08/2021
Project Title:	Speech capable Customer Service Bot with Video-Audio based Emotion Recognition
Word Count:	5821
Page Count:	19

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	22nd September 2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Speech capable Customer Service Bot with Video-Audio based Emotion Recognition

Karthi Mahendran
X20118198

Abstract

In today's world, feedback from the customer is one of the essential aspects to improve any business. It is an added advantage to understand the customer emotion along with the feedback. This research aims to recognize facial emotion from the user and simultaneously capture the emotion through vocal Text. Two datasets were used in this research, FER2013, a dataset containing human faces classified to different emotions. The other is Dailydialogue, which contains day-day speech of humans and corresponding emotions. Initially, Convolution Neural Network was chosen for facial emotion recognition. The accuracy of the CNN model was comparatively low. Hence, to improve the model's performance, the Capsule network is combined with CNN, and the CNN-CapsNet model is implemented. Naive Bayes algorithm is used for the sentimental analysis of Text, converting from speech to Text, and corresponding emotion is detected. The desired output of both the implemented models is the user's emotion, which is then merged using weighted sum probabilities. For validation of the entire process flow, the model is implemented into a real-time chatter audio bot. The initial accuracy of the CNN model was around 65%, which is then improved to 89.5% using the CNN-CapsNet model for more accurate recognition.

1 Introduction

"Focusing on the customer makes a company more resilient." As quoted by Jeff Bezos, irrespective of the kind of business or medium of business, the essential point to be kept in mind is customer satisfaction. An Emotionally intelligent person should be skilled in four areas: identifying emotions, using emotions, understanding, and regulating emotions. The success of a business can be achieved with happy customers. This research aims to build a customer interactive chatterbot along with a Facial emotion Recognition UI, which will, in turn, lead to understanding the user satisfaction from the audio and face detected.

1.1 Background

In the present scenario, where the world is hit with the pandemic, all significant kinds of business are being converted to the online mode. This has reduced the direct customer interaction to a large extend. When a customer purchases anything online, providing feedback to the product, any product for that sake, is measuring Customer Satisfaction. However, the drawback here can be that from the business aspect, and the company

cannot just go by the customers' feedback since the feedback provided by the user cannot be precisely based on what they feel. In the global market with the pandemic, still holding on to what can be done to increase customer interaction. This research tries to increase customer interaction in a better way using Machine Learning Techniques

1.2 Motivation

So, on saying that, the next thing that comes to one's mind is how do we measure customer satisfaction? Is Customer Feedback well enough to measure Customer Satisfaction? These questions can be the baseline of motivation for this research. Understanding a customer's emotions can largely influence the business in different aspects. There are different existing systems where the customer's emotion is captured and understood. The motive of this research is to improve the accuracy of the emotion recognition of a user. This is done by building two different models that recognize the facial emotion from the customer's face. The other is when an audio text is inputted converted to text, and the customer's emotion is identified. The main objective here is to match both emotions for more accurate results.

1.3 Research Question

How can we improve the magnitude of understanding Customer Satisfaction using Deep Learning models with less customer Interaction?

1.3.1 Research Objective

The Objective of the Research can be split into two parts, one to recognize the customer's emotion from the customer's face, simultaneously capturing audio text from the user and matching both the recognized emotion.

1.4 Outline of the Research Paper

This section explains the whole structure of the report following the introduction section. Section 2 explained all the literature reviewed and analyzed critically to support and choose the appropriate approach for the research. Section 3 gives a brief overview of the proposed methodology followed in the research. Section 4 explains design specifications and brief details about the Machine Learning models implemented. Section 5 provides step by step process of implementation and the different Evaluation Metrics. Against which the models are validated. Section 7 puts an end to the Research concluding by pointing out the key takeaways and the future scope within the domain.

2 Related Work

When conducting research, it is essential to explore the related works within the domain, which will add to our research in many ways. Understanding the existing models within the Research domain, knowing their advantages and limitations, and choosing a suitable approach for the proposed idea can be done only with a proper Literature Review. This section provides details about all the papers, journals reviewed as part of this research.

2.1 Facial Emotional Recognition

A system called EAGR that is emotion, age, gender, recognition, is developed to find out people's emotions, age, gender based on their face. A normalized facial cropping (NFC) technique is used before using a convolution neural network (CNN). NFC is a pre-processing. In one model, facial hairs are removed; on the other hand, NFC extracts the features like age and gender from facial hair. Overall this model could recognize seven feeling, four different age range and two genders with 82.4%, 74.95%, and 96.65%. Lu et al. (2021)

A light weight convolutional neural network is used for multi-task training of face recognition and classification of facial attributes such as age, race, gender on cropped face with no margin. AffectNet dataset is used to find out age, gender, and UTKFace dataset for race recognition. It is demonstrated that features extracted from facial region using neural network. To get a better accuracy in facial recognition a simple pipeline for training the neural network for images and video of several dataset.It has been demonstrated that, when compared to existing models, there is a increased robustness towards the alignment and extraction of face. The region of face were cropped when it was returned by the face detector, also no margin were included. A high precision was able to achieve by the model by improving the speed and scale of mode. Classifiers are needed to be investigated in the future.Savchenko (2021).

Fan et al. (2016) developed a video-based emotion detection system. They used C3D networks and CNN-LSTM to simulate visual interactions and motion at the same time. When coupled with an audio module, this system obtained 59.02 percent identification accuracy without utilizing any additional emotion-labeled video clips in the training sample, compared to 53.8 percent for the EmotiW 2015 database winner. According to the research, combining RNN and C3D may substantially improve the identification of video-based emotions.

Liang et al. (2020) present a novel BiLSTM convolutional fusion network for solving the Facial Emotion Recognition issue in discriminative spatial tasks by learning features and collecting temporal correlations. According to experimental findings on three benchmark datasets, the suggested system outperforms previous approaches, CK+, Oulu-CASIA, and MMI. These studies demonstrate that combining advanced CNN features with long-term bidirectional memory outperforms face emotion recognition models and temporal information quality. The average accuracy of the datasets was 99.6 percent, 91.07 percent, and 80.71 percent, respectively.

2.2 Multimodal in video, audio and text Emotion Recognition

A hybrid network was proposed for an audio-video community emotion recognition. This model includes the environmental object statistics stream(EOS), audio, facial expression, and video streaming. A tool called EmotiW was used in the 8th Emotion Recognition of a Wild challenge. The reviews for the test database were 76.8%, which had a higher 26% compared to the average. The use of a hybrid network improved the results. The model won first place in solving the AudioVisual community emotion recognition problem. Liu et al. (2020)

A novel multimodal dual recurrent encoder type of model that uses both audio and text signals was proposed. The dual RNNs were used to encode information from both text data and audio, and then a feed-forward neural model was used to integrate this information to predict the type of emotion. The dataset used here is IEMOCAP, and

the accuracy is satisfactory and ranges from 68.8% to 71.8%. This model solved existing models that focused only on audio features and incorrectly assigned the neutral class due to predictions. In the future, they are planning to implement audio, video, and text as input. This method shows some uncovered new learning schemes that would help emotion detection to have great success. Yoon et al. (2018)

A multimodal emotion recognition using deep learning was proposed. Emerging research into human-computer interactions, thus this makes the interaction between computer and human. Many attributes define human emotion, such as behavior, facial expression, how they talk, etc. This paper focuses on multimodal emotion recognition using deep learning and also compares it with the existing models. Abdullah et al. (2021)

Multimodal Sentimental analysis is a new research area where it teaches machines how to recognize, identify and express emotions. Bidirectional Encoder Representations from Transformers (BERT) is a pre-trained language representation model that is efficient. The existing works implemented using fine-based BERT are only based on text, but here they propose a Cross-Modal BERT. It uses both text and audio for interaction. As the core unit of the CM-BERT, masked multimodal attention is designed to dynamically adjust the weight of words by combining the information of text and audio modality. As the CM-core BERT's unit, Masked multimodal attention is intended to dynamically adjust the word's weight by combining information from audio and text modality. Yang et al. (2020)

Multi-modal emotion recognition on the IEMOCAP dataset using a neural network was proposed. Using this dataset, emotions from the face, speech, also hand rotation movements. This approach can identify the best architecture based on performance at the last layer. The first model is Speech-based emotion recognition, and it has got an accuracy of 50.6 %. When it was compared with all the existing models, it was less. The second model is based on Text emotion recognition, and the model is built with two stacks of LSTM, and accuracy is around 64%. LSTM and COvolution Neural Network-based emotion recognition was the third model with the hand rotation-based data and acquired an accuracy of 51.1 percent. This Mocap model and the Speech and Text-based model were combined and got an accuracy of 71 percent. The fusion of all final layers gave the best result. This model can have much better accuracy by replacing any one of the models. Tripathi et al. (2018)

A Faster R-CNN was proposed for the identification of facial expressions. The facial expression was presented in a standardized manner using the maximum of trainable convolution kernel, and the implied features were removed. A Region Proposal Network of higher quality were created, and Faster R-CNN used the created regional proposal for identification purpose. At last, for classifying the feeling in the face, a regression layer and softmax layer were used. All these were performed on a Chinese Linguistic Data Consortium that has video and audio data for multi-modal emotions. Li, Zhang, Zhang, Zhang, Li, Xia, Yan and Xun (2017)

Eight basic emotions were Chinese Natural Audio-Visual Emotion Database (CHEAVD) 2.0 were selected. Racial features such as mouth, eyebrow causes racial expressions. In capsule networks, these are like capsule features. The capsule network not only captures the static racial emotions from the video but also records the parameters. The accuracy was initially low. Reinforcement learning was proposed for the fusion method of audio and video to improve the accuracy where the previous accuracy is 32%, which 52.3% then gains. Ouyang et al. (2018)

2.3 Text Sentiment Analysis

A RECCON (Recognizing Emotion Cause in CONversations) system is created for emotion recognition. The transformer-based baselines are used to address two different subtasks on this RECCON dataset. The sub-tasks are Casual Emotion Entailment and Causal Period Extraction. This dataset at the conversion stage contains over 1,126 dialogues and 10,600 utterance causal span pairs. They have identified various emotional forms and key obstacles that determine the source of emotion in conversations extremely difficult. In the proposed dataset, only dyadic conversations are used. A new task highly relevant for emotion-aware is introduced to artificial intelligence that recognizes the emotion caused in conversations. Poria et al. (2020)

For emotion identification in conversations, a COSMIC is proposed. That is COmmonSense knowledge for eMotion Identification in Conversations(COSMIC); during the conversation, different commonsense analyzes mental state, behavior, activities, etc. Since there are multiple groups, there can be a collapse in identifying emotions. However, this model eliminated that problem, and there is no misclassification in any similar type of emotion. The model is build using RNN and GCN with the use of commonsense. Ghosal et al. (2020)

2.4 Sentiment Analysis in ChatBot

Sentiments from chat and email are recognized. In social media platforms and business applications, people share their feeling, and it is straightforward to identify their moods from the content they have shared on the internet. Several questions are asked to that person before correctly understanding their mood. The initial attempt is to refresh the person's mind on this basis of the response that they give while offering refreshments. The whole idea is that consumers express their feelings, and the chatbot responds to them accordingly. Sekhar et al. (2021)

A chatbot is trained to influence interlocutors. The main goal of this chatbot is to a given response like how humans give. The chatbot is created with three variables that can be controlled. The controllable factors are sentence length, specific words, and emotion. The whole system contains a chatbot and an interlocutor that plays the role of a human. The chatbot is made to respond same as that of interlocutors, like a happy response, joy, particular words, etc. Trails are conducted with the human interlocutors to demonstrate the guiding the effectiveness of chatbot influencing responses of humans.Su et al. (2021)

2.5 Weighted Ensemble

This Shen and Kong (2004) offers an ensemble regression approach that outperforms basic weighted or weighted average combining methods. The output of an ensemble is dynamically weighted, with the weights determined by the predicting accuracy of the trained networks using training dataset; the more accurate a network appears to predict, the higher the weight. The new collection of empirical findings improves prediction accuracy.

2.6 Summary

The purpose of the Literature Review is to understand all of the criteria for the idea we're proposing for our research. Following a critical examination of existing works in

the domain, the application of machine learning algorithms in this domain, and the benefits and limitations of each model, it is clear that CNN, Capsule Networks can be used effectively in emotion recognition and for text analysis, some of Machine learning algorithm can be used too. We have chosen to build the CNN and CNN-CapsNet models, compare the results, and select the best functioning model to develop the facial and vocal text emotion recognition system for the research.

3 Methodology

This research project proposed the model to predict the facial and vocal text emotional recognition to be implemented in the audio chatter-bot. The project's main objective is to implement this face and vocal text recognition model in any support assisting platform to analyze the customer behavior and feedback about any products. This model would bring a new method for marketing strategies in the current market. We have implemented two models CNN and CNNCapsNet, for facial recognition and Multinomial Naive Bayes, for the sentimental text analysis. Crisp-DM (Cross-Industry Standard Process for Data Mining) methodology is followed throughout the research project.

3.1 Data Acquisition

3.1.1 FER 2013 Dataset

The "Facial Expression Recognition(FER)Challenge" dataset was downloaded from the Kaggle repository [1]. The dataset includes 35,887 pixels of images, which is adequate for creating a useful model. As seen in Figure 1 are the sample images of the FER2013 Dataset. "Emotion" and "pixels" are the two columns in the dataset. The 'feeling' column includes a numeric code ranging from 0 to 6 that represents the emotion shown in the picture. Figure 2 shows the plot of number data present in the training, testing and validation. The following is the string in the 'Usage' column: Training, public examination, and private examination are all part of the process. The files include 48x48 pixel grayscale portraits of people. The data categorizes each face into one of seven groups based on the emotions expressed in the facial expressions: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral.

Figure 1: Sample grayscale image of the FER2013 Dataset

[1] https://www.kaggle.com/ashishpatel26/facial-expression-recognitionferchallenge

3.1.2 DailyDialog Dataset

Li, Su, Shen, Li, Cao and Niu (2017) DailyDialog is a high-quality multi-turn discussion dataset [2] with tremendous potential created from human language, which is less noisy. The talks in the dataset reflect how we interact daily and cover a wide variety of topics. It has been meticulously labeled with information on communication and emotion. The developed DailyDialog dataset contains 13,118 multi-turn dialogues where these sentences are saved in several text files. The dialogues text file contains 11,318 recorded chats. Each line in emotion file corresponds to the emotion annotations in the dialogues text file. The emotion number represents: 0: no emotion, 1: anger, 2: disgust, 3: fear, 4: happiness, 5: sadness, 6: surprise

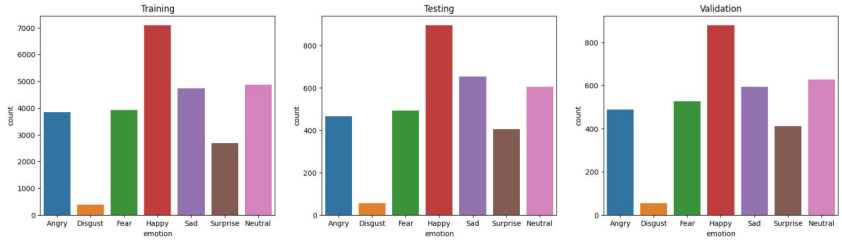

Figure 2: A plot of data count corresponding to Usage

3.2 Data Pre-Processing

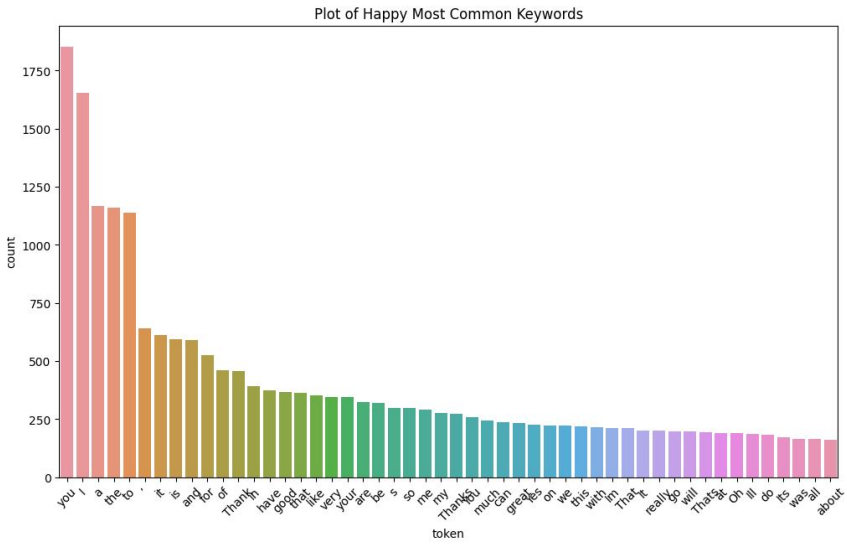

Figure 3: A plot of Happy Most Common Keywords

The FER 2013 CSV was loaded into the pandas' data frame, and using the values from the "pixels" and "Emotions" columns of the data frame; the images were generated and stored corresponding to the emotions folder. These images are in RGB then grayscaled and reshaped into 48x48 for improved representation and computational practicality. The

[2] `http://yanran.li/dailydialog.html`

image of the dataset was augmented by changing the rotation and width shift range; the remaining pixel was filled by nearest fill mode. Data augmentation would help add more training data into the models, preventing data scarcity for better models. Moreover, it reduces data overfitting and creating variability in data. The dataset was split into training, testing, and validation data. The training sample has 28,709 cases, whereas the test sample contains 3,589 cases. Another 3,589 cases were included in the validation samples that can be used to assess the prediction's performance.Figure 3 shows the plot of 'Happy' Emotion Most Common Keywords.

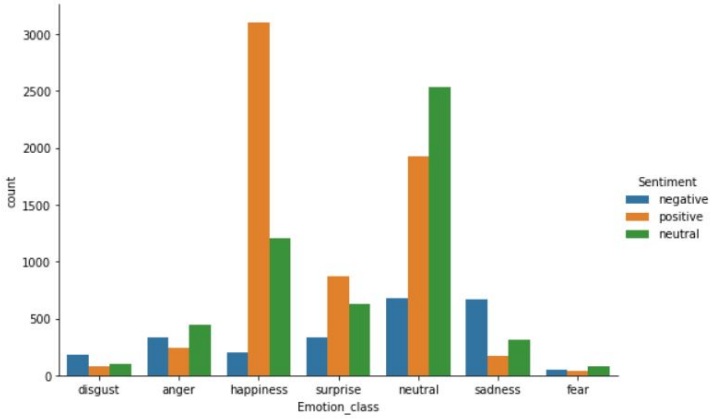

Figure 4: A Sentiment analysis of Text Emotions

The dialogue text file is read line by line, and it is split with the token '__eou__' and stored the split line into text list variable. Then the emotion file is mapped with the corresponding text list variable and stored in the new CSV file with "Text" and "Emotion" columns named as dailydialog.csv. Then dialydialog.csv loaded into a panda's data frame. TextBlob, a python library for processing textual data, checks the text polarity score for text data. If the text data polarity score is greater than 0, it is positive; lesser than 0, it is negative and equals 0, it is neutral. NFC (Near Field Communication) is a robust python framework used for text cleaning, such as removing stopwords, user handles, punctuation, and emojis. Using NFC, the text data is cleaned and stored with the "Clean Text" column in the data frame used in the naive Bayes model as input. Figure 4 shows the most common words used for Happy emotion. WordCloud is plotted for each emotion by extracting the most common keywords from the clean text data. This gives a better understanding of the data. Figure 5 shows the WordCloud of Surprise Emotions.

A matrix of tokens count will be the input for the Naive Bayes, so the collection of the cleaned text data is passed to CountVectorzier (Scikit-learn Package) that converts into a matrix of token counts that get stored as the X matrix input. Then the X is converted to an array, and the Emotion data from the data frame is split into training and testing using the train_test_split function from Scikit-learn. The splitting is done with 80% for training and the remaining 20% for the testing phase.

4 Design Specification

The research project recognizes facial emotion from the user and simultaneously captures the emotion through vocal Text. The facial emotion recognition is implemented

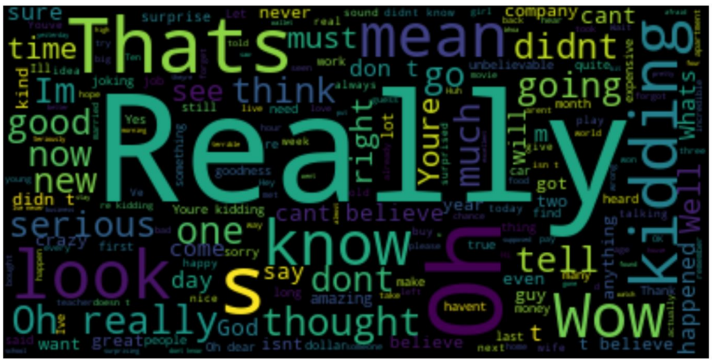

Figure 5: A WordCloud for surprise Emotinonal Words

using CNN and CNN-CapsNet. The text sentiment analysis is implemented using Naive Bayes. Training and testing are done with these models and selected the best model for implementing it in the real-time audio chatter bot.

4.0.1 Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a deep artificial neural network consisting of several convolution layers followed by fully connected layers. The CNN takes the 2D structure as an input, where the dataset image size is 48 x 48 pixels. The input will be in the format of h x w x r where the where h,w is the height and width of the image and r is the number of channels. Since our input image is grayscale, thus r is 1. The models' final output will be the seven emotion classes.

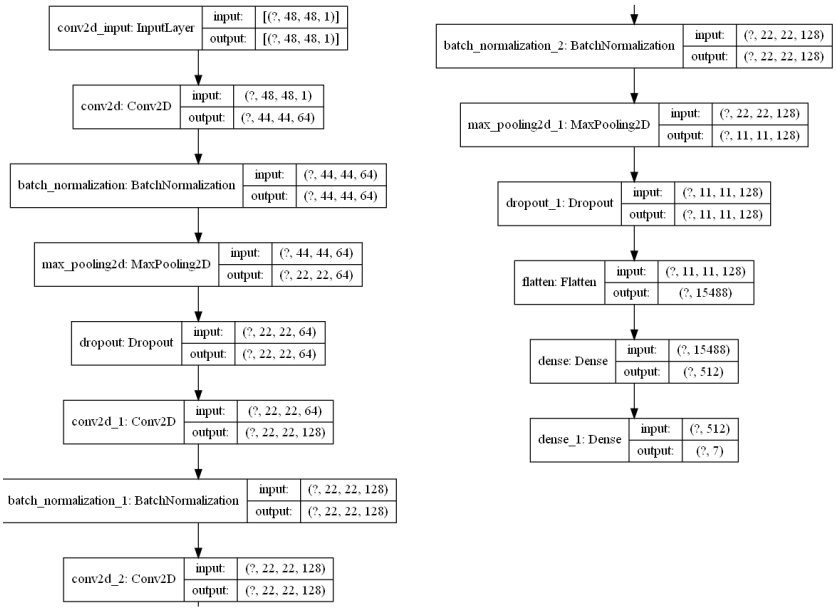

Figure 6: CNN Model Plot

The proposed CNN architecture consists of two convolution layers and two fully connected layer. Figure 6 shows the CNN model plot where we can see the outputs of each

layers. The convolution layer is the building block of our network, and these compute the dot product between their weights and the small regions to which they are linked. The first convolution layer consists of 64 output filters, and the kernel will be 5 x 5. The second convolution layer consists of two 128 output filters and two kernels of 3 x 3. Both convolution layers have the same padding values; activation function Relu is applied to the outputs of all layers in the network and with a pool size of 2 x 2. Each layer has its Batch Normalization: normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation. To prevent overfitting, randomly 0.25% of neurons are turned off in both dropout layers. Before introducing the dense layer, both of the convolution layers are flattened. The end layer consists of two dense layers of units 512 and 7 dimensionalities of the output shape. The whole network is compiled using Adam optimizer, and Since we are dealing with a classification problem that involves seven emotional categories, the categorical_crossentropy loss function has been used. For the FER, this model has been learned and updated using this dataset.

4.0.2 CNN with Dynamic Routing Between Capsules (CNN-CapsNet)

The researcher proposed a CapsNet model with shallow architecture with only two convolutional layers and one fully connected layer. The first layer (Conv1) has 256 filters with 9 x 9 kernels with ReLU Activation. The second layer is a primary capsule layer where the converted pixel from the Conv1 layer takes as input and forms an inverse graphics perspective. The primary capsule layer has 32 channels with 8D capsules (i.e., 8 Convolution units with 9 x 9 kernel). The final layer DigitCaps has a 16D capsule per digit class. The routing happens between only two consecutive capsule layers, which are Primary capsules and DigitCaps. Initially, all the output from the capsules is sent to all parent capsules with equal probability. We have proposed the new model by combining CNN and Capsule Layer from CapNets to make the better prediction and increase the accuracy.

Layer (type)	Output Shape	Param #
input_3 (InputLayer)	(None, 48, 48, 1)	0
conv2d_9 (Conv2D)	(None, 46, 46, 64)	640
conv2d_10 (Conv2D)	(None, 44, 44, 64)	36928
average_pooling2d_3 (Average	(None, 22, 22, 64)	0
conv2d_11 (Conv2D)	(None, 20, 20, 128)	73856
conv2d_12 (Conv2D)	(None, 18, 18, 128)	147584
reshape_3 (Reshape)	(None, 324, 128)	0
capsule_3 (Capsule)	(None, 7, 16)	14336
lambda_3 (Lambda)	(None, 7)	0

Total params: 273,344
Trainable params: 273,344
Non-trainable params: 0

Figure 7: An summary of the CNN-CapsNet Model

The proposed CNN-CapsNet model uses the two standard conv2d layers and adds a capsule layer (DigitCaps Layer) as a fully connected layer. Figure 7 shows the CNN-CapsNet model summary where we can see the outputs shape of each layers. Our image

data of shape 48 X 48 X 1 is passed to the first conv2d layers with 64 filters and 3 x 3 kernels with the ReLU activation. Then the input data get downsampled along with spatial dimensions by taking average value over an input window of size 2 X 2 for each input channel done through AveragePooling2D. The second conv2D layer is constructed the same as the first by changing the filter by 128 alone. We reshape the batch size of 128, the input number of the capsule (7), and input dimensions (3) to connect the capsule layer. The capsule layer has three routings, and weights among the capsule are share equally. The squash of value less than 0.5 is used as activation in this layer because the norms will be zoomed in when the value is higher than 0.5. The output of the final model is the lengths of 7 Capsule, whose dimension is 16. Thus, the problem becomes a seven two-classification problem. Then the model complies with the defined loss function (margin loss) and with adam optimizer. Then the model has been learned and updated using this FER2013 dataset.

4.0.3 Multinomial Naive Bayes Classifer

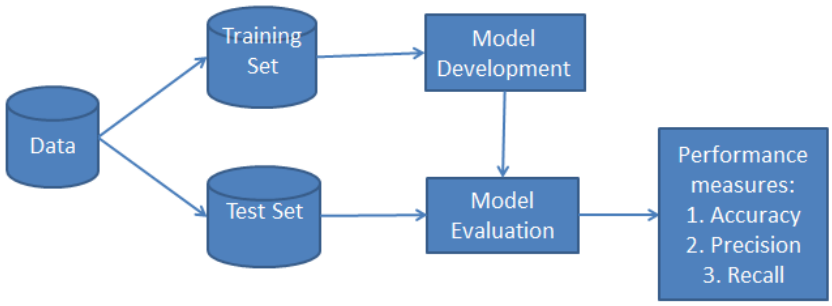

Figure 8: Architecture for Multinomial Naive Bayes

MultinomialNB from scikit-learn packages are implemented the naive Bayes algorithm for multinomially distributed data and is one of the two classic naive Bayes variants used in text classification. Figure 8 show the architecture of the Naive Bayes. The multinomial Naive Bayes classifier is suitable for classifying discrete features; for example, text classification uses words counts. The multinomial distribution usually requires integer feature counts. We already transformed the data to a sparse matrix with n sample and n features for the training vectors, which will be the X input, and the Emotion data will be out Y input, and the model is trained. We can predict the model's accuracy using the score function available in the same package. To predict the probability of the given model, the predict_proba function is used.

4.1 Weighted Average Ensemble

The weighted average or weighted sum ensemble is a machine learning technique that combines predictions from many models, with the contribution of each model weighted according to its capacity or quality. The first step in creating a weighted average prediction is to give each ensemble member a specific weight coefficient. This may be a floating-point value between 0 and 1 that represents a percentage of the weight. It may also be an integer starting with one that indicates the number of votes each model will get. The probability output of both models CNN-CapsNet and Multinomial Naive Bayes

is collected. since both model Y output will be same seven Emotions classes with probability. Let take the CNN-CapsNet model weightage as 0.60 and the MultinominalNB model weightage as 0.40. These weights can be used to calculate the weighted average by multiplying each predicted probability output by the model's weight to give a weighted sum, then dividing the value by the sum of the weights. The scores have the same scale and weights. In turn, the weighted average is also sensible, meaning the outcome scale matches the scale of the scores. As seen in Figure 9 shows the preditected output of each model is weighted and gives the result.

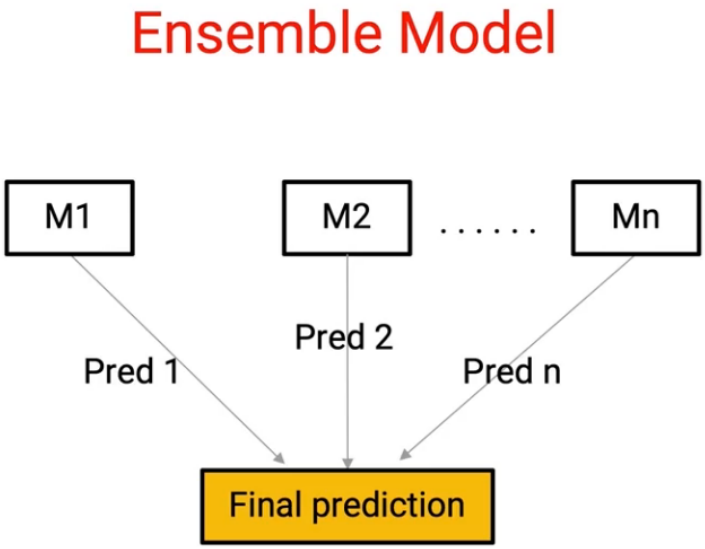

Figure 9: Weighted Average Ensemble Model

4.2 Evaluation Metrics for Models

The Confusion matrix is used to assess a classifier's efficiency by detecting the types and amounts of classification errors. The actual class is represented by a row in the confusion matrix, whereas each column represents the predicted class. The number of words correctly understood for each class is shown by the diagonal elements of the confusion matrix. Unrecognized phrases or recognition failures are examples of off-diagonal components. Given the confusion matrix, Equation 1 is used to determine accuracy. Equation 2 uses the F1 score to calculate accuracy by combining precision and recall (F measure).

$$Accurcy = \frac{\sum(DiagonalElements)}{\sum(AllElements)} \tag{1}$$

$$F1 = 2 \times \frac{precesion \times recall}{precesion + recall} \tag{2}$$

4.3 Facial & Vocal Text emotion Recognition in a Chatter-Bot

We have implemented an audio chatterbot to validate the proposed facial and vocal text emotion recognition model. We train the CNN-CapsNet model to identify the human face

and predict facial expressions from the observed face. To recognize the human faces from the image, Haar Cascade Classifier has been used. It is an XML file created by OpenCV (design library to solve real-time computer vision problems) to detect the frontal face by drawing a rectangular box around the face. Simultaneously, the user audio is converted into text using Python's Speech Recognition Library. Then the MutinomialNB Classifiers train the text to classify and predict the emotions in text. ChatterBot Python library helps to create a simple chatbot and generate automated responses from the user's input. The bot's response is converted into audio for a better experience that makes an audio chatterbot. The whole process renders into a UI using the pyglet package. This module creates an application window to display OpenGL context and set various border styles or full screen. MultiThreading is used when the camera detects the facial expression, and the user audio converted to vocalized text is analyzed, and the chatterbot responds through audio. Along with the threading process, facial emotion, vocal text emotion, and weighted average emotion results are displayed in the pyglet UI.

5 Implementation and Evaluation

The essential element of the research study is implementation and evaluation, which we can strive to accomplish by implementing Section 4. Input as image and text, pre-processing the images and text, creating models, preparing models, and deploying the real-time recognition of facial and vocal text emotion in audio chatterbot is part of the implementation process. This research aims to recognize the facial emotion from the user and simultaneously capture the emotion through vocal text. To achieve this, we have used CNN, CNN-CapsNet, and Multinomial Naive Bayes. Figure 10 shows the workflow of audio chatter-Bot.

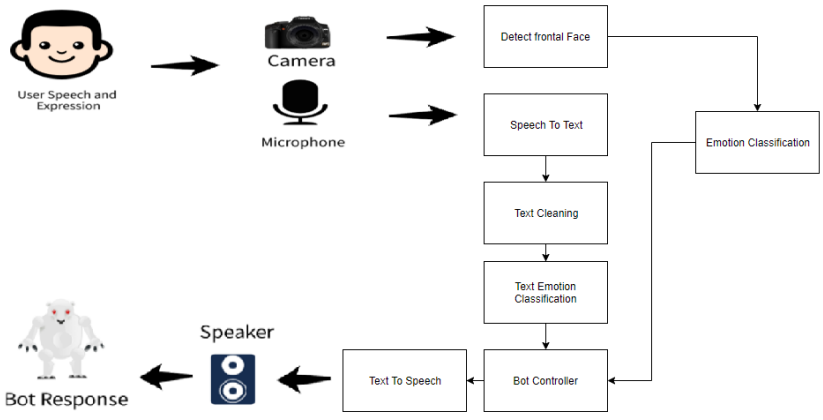

Figure 10: WorkFlow of the Simple Audio Chatter-Bot

5.1 CNN Model

This model is implemented using six-layered neural networks and a batch size of 128, trained with 25 epochs on the FER 2013 Dataset. The model is compiled with adam optimizer, and metrics are calculated with accuracy, and the loss is calculated from categorical cross-entropy. The CNN model has achieved 52% accuracy.

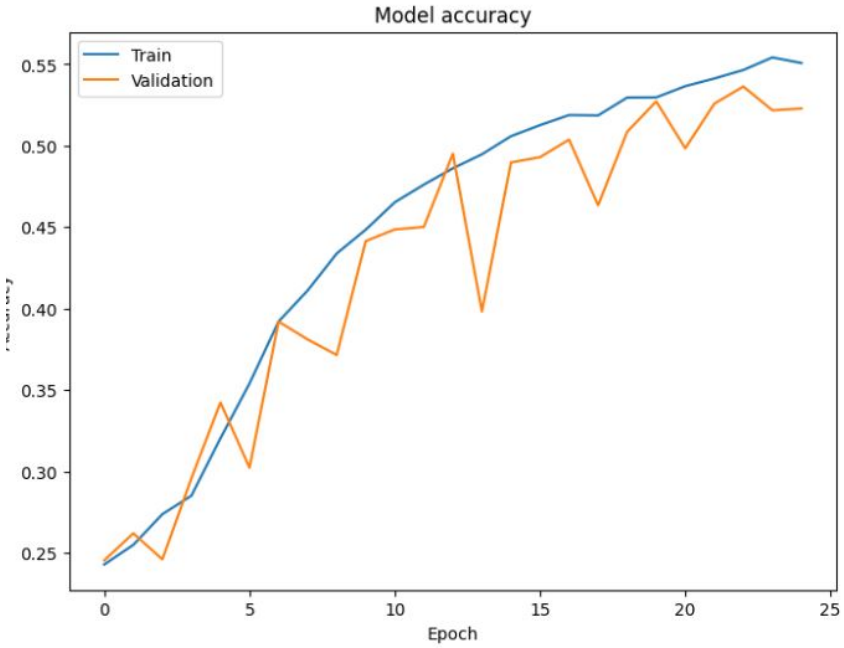

Figure 11: Accuracy Plot using CNN Model

Figure 11 denotes the accuracy plot where the train and validation curve increase dependently along the epoch while training.

5.2 CNN-CapsNet Model

```
score = model.evaluate(X_crossval, Y_crossval, verbose=0)
model.save_weights('data/cnn-capsnet_w.h5')
print('Trained model saved to \'cnn-capsnet_w.h5\'')
print('Test loss:', score[0])
print('Test accuracy:', score[1])
```

```
Trained model saved to 'cnn-capsnet_w.h5'
Test loss: 0.3692532868604474
Test accuracy: 0.8573816263908132
```

Figure 12: Test Accuracy for CNN-CapsNet Model

This model is implemented using seven-layered neural networks and a batch size of 256, trained with 5 epochs on the FER 2013 Dataset. The model is compiled with adam optimizer, and metrics are calculated with accuracy. We have used two callbacks for the model. First, each checkpoint is saved for every epoch if the accuracy is higher than the previous epoch accuracy, and others are the CSVLogger, where the logs of the model training are stored in the log.csv. The model trained with the validation dataset got an accuracy of 79%. The model accuracy is improved by passing the image augmented data to the model and achieving an accuracy of 85.73% as seen in Figure 12. Figure 13 shows the accuracy plot of the train and test data using CNN-CapsNet Model.

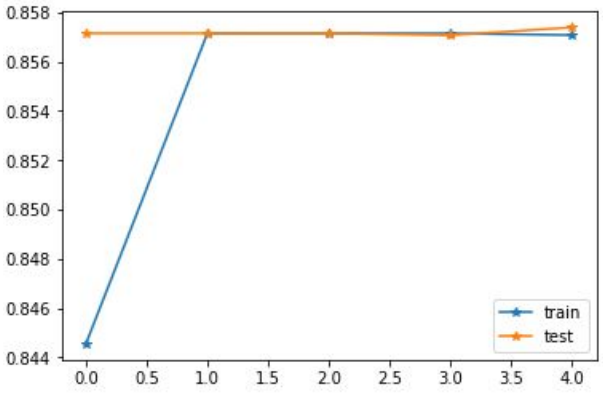

Figure 13: Accuracy Plot of Train and Test data using CNN-CapsNet Model

5.3 MultiNomialNB Classifiers

Once the clean text data converted to sparse matrix given as input to the model and trained. The accuracy score predicted from the Multinomial Naive Bayes is 0.59%. The probablity output of the mode is predicted for the average weigthed essembles as a input. Figure 14 shows the classification report for the Naive Bayes model where each Emotion class precision, recall, f1-score are shown.

	precision	recall	f1-score	support
0	0.55	0.70	0.62	1031
1	0.47	0.17	0.25	191
2	0.25	0.02	0.03	59
3	0.00	0.00	0.00	32
4	0.65	0.74	0.69	953
5	0.63	0.34	0.44	224
6	0.63	0.46	0.53	344
accuracy			0.60	2834
macro avg	0.45	0.35	0.37	2834
weighted avg	0.58	0.60	0.57	2834

Figure 14: Classification report for Multinomial Naive Bayes

5.4 Finiding

We have compared the models of CNN and CNN-CapsNet and their accuracy. CNN-CapsNet model has gained 63 percent comparatively higher accuracy than the other models. Then we merged the BinomialNB, and CNN-CapsNet predicted output by weighted average Ensemble to produce the average results.

5.5 Real-Time Emotion Recognition in a Chatter-Bot

Audio Chatter-bot is deployed using Python 3.8. We have used OpenCV to live stream the camera and capture the frontal face using HaarCascade Classifiers. This captured image is cropped, reshaped to 48 X 48, and converted to grayscale. These grayscale image pixels are sent as input to the CNNCapsNet model to predict their corresponding emotion. These all operations occur in the synchronous thread, so the facial and vocal text prediction would proceed simultaneously. By pressing Enter Key as a triggering point, the user can speak through the mic. The voice is converted to text using python's library (Speech Recognition). The vocal text is cleaned and converted into a sparse matrix that can be passed to model and predict the emotions. Meanwhile, the exact vocal text is passed to the chatterbot, and that gives the response text. We have used chatterbot python library and trained the bot with basic sentences, so the bot would respond to the fundamental question on what we trained on. Both of the predicted output is weighted averaged and all of the emotional response is shown in the UI of the chatbot. Figure 15 shows a demo of the audio bot with UI.

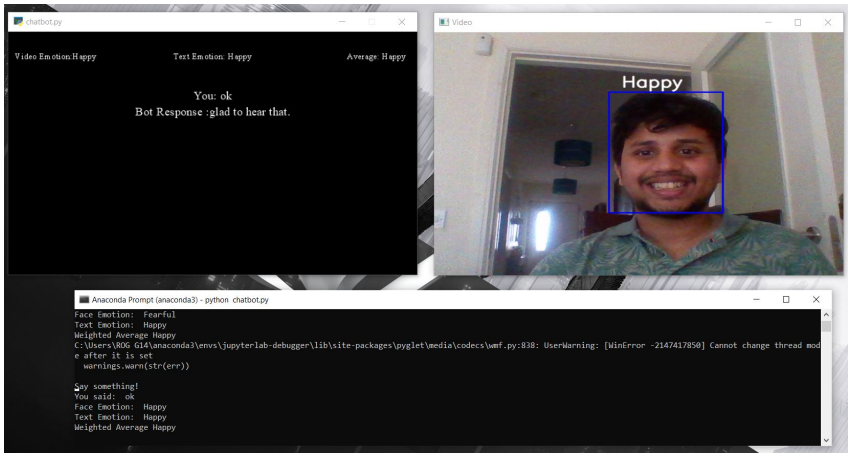

Figure 15: A demo of the Audio Chatter Bot

6 Discussion

Table 1: Summary of the evaluation metrics of models

Models	**Accuracy**
CNN-CapsNet	**85**
CNN	52
Naive Bayes	59

This research project proposed the model to predict the facial and vocal text emotional recognition to be implemented in the audio chatter-bot. The project's main objective is

to implement this face and vocal text recognition model in any support assisting platform to analyze the customer behavior and feedback about any products. This model would bring a new method for marketing strategies in the current market. We have implemented two models CNN and CNNCapsNet, for facial recognition and Multinomial Naive Bayes, for the sentimental text analysis. When we compare the models of the CNN and CNNCapsNet model, the higher accuracy of CNNCapsNet is better. Table 1 show the accuracy of all these models. So we have chosen CNNCapsNet to identify the facial emotion. Both CNNCapsNet and Naive Bayes predicts the same emotion classes, and both models predicted output could be merged using the Weighted Average Ensemble.

Haque and Valles (2018) presented research utilizing a deep CNN in facial expression detection on the FER2013 dataset. They received an accuracy score of 67%. Ul and Valles (2018) utilized the FER2013 and KDD datasets to apply the same technique and obtained a 78% accuracy score. We attempted to outperform the accuracy. Using FER2013, the proposed new model, which combines CNN and Capsule Networks to form CNN-CapsNet, can get the maximum accuracy of 85%. However, Haque and Valles (2019) found that utilizing 200 epochs to construct a DCNN model for facial emotion recognition, they achieved 86.44% accuracy. However, our algorithm achieved 85% accuracy with just five epochs. The unique approach of implementing two deep learning models trained with two different datasets (face and text), followed by merging the predicted output, cannot be seen in any existing research work. The photos in our collection are of poor quality. The faces are not all in the exact location, some images have written text on them, and some individuals hide part of their faces with their hands, which becomes some of the dataset's disadvantages. This system can be implemented in any customer support service area to analyze the customer's behaviors.

7 Conclusion and Future Work

The main objective of this research was to implement an audio chatterbot that builds from recognizing facial and vocal text emotions. The idea of building this system is that it can be deployed in any support service system of any business to improve customer satisfaction and understand customer behavior. After critically analyzing the relevant research works in the same domain, CNN was a perfect choice for implementing the facial emotion recognition system. After implementing CNN, it was seen that model showed an accuracy of 52%. To improve the overall accuracy of the model CNN-CapsnNet model was implemented, and the model showed accurate results with an overall accuracy of 85%. Simultaneously Multinomial Naive-Bayes classifier is used for text classification, and text classification is seen to have an accuracy of 59 %. The novel method of applying two deep learning models trained on two distinct datasets (face and text), followed by combining the predicted output, is not seen in any other research paper. The whole model can be implemented in any support service system. A point to be noted as a limitation is that data used for text classification is inaccurate, and text classification accuracy is comparatively lesser.

As part of Future work, the implementation of audio emotion classification can be considered rather than converting audio to text for classification. This may tend to produce more accurate results. The implementation of different Classifiers can be explored for the more stable model. The whole system can be made more user interactive, and implementation of this system can be done by other GUI python tools.

References

Abdullah, S. M. S. A., Ameen, S. Y. A., Sadeeq, M. A. and Zeebaree, S. (2021). Multimodal emotion recognition using deep learning, *Journal of Applied Science and Technology Trends* **2**(02): 52–58.

Fan, Y., Lu, X., Li, D. and Liu, Y. (2016). Video-based emotion recognition using cnn-rnn and c3d hybrid networks, *Proceedings of the 18th ACM international conference on multimodal interaction*, pp. 445–450.

Ghosal, D., Majumder, N., Gelbukh, A., Mihalcea, R. and Poria, S. (2020). Cosmic: Commonsense knowledge for emotion identification in conversations, *arXiv preprint arXiv:2010.02795* .

Haque, M. I. U. and Valles, D. (2018). A facial expression recognition approach using dcnn for autistic children to identify emotions, *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, IEEE, pp. 546–551.

Haque, M. I. U. and Valles, D. (2019). Facial expression recognition using dcnn and development of an ios app for children with asd to enhance communication abilities, *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, IEEE, pp. 0476–0482.

Li, J., Zhang, D., Zhang, J., Zhang, J., Li, T., Xia, Y., Yan, Q. and Xun, L. (2017). Facial expression recognition with faster r-cnn, *Procedia Computer Science* **107**: 135–140.

Li, Y., Su, H., Shen, X., Li, W., Cao, Z. and Niu, S. (2017). Dailydialog: A manually labelled multi-turn dialogue dataset, *Proceedings of The 8th International Joint Conference on Natural Language Processing (IJCNLP 2017)*.

Liang, D., Liang, H., Yu, Z. and Zhang, Y. (2020). Deep convolutional bilstm fusion network for facial expression recognition, *The Visual Computer* **36**(3): 499–508.

Liu, C., Jiang, W., Wang, M. and Tang, T. (2020). Group level audio-video emotion recognition using hybrid networks, *Proceedings of the 2020 International Conference on Multimodal Interaction*, pp. 807–812.

Lu, T.-T., Yeh, S.-C., Wang, C.-H. and Wei, M.-R. (2021). Cost-effective real-time recognition for human emotion-age-gender using deep learning with normalized facial cropping preprocess, *Multimedia Tools and Applications* **80**(13): 19845–19866.

Ouyang, X., Nagisetty, S., Goh, E. G. H., Shen, S., Ding, W., Ming, H. and Huang, D.-Y. (2018). Audio-visual emotion recognition with capsule-like feature representation and model-based reinforcement learning, *2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia)*, pp. 1–6.

Poria, S., Majumder, N., Hazarika, D., Ghosal, D., Bhardwaj, R., Jian, S. Y. B., Ghosh, R., Chhaya, N., Gelbukh, A. and Mihalcea, R. (2020). Recognizing emotion cause in conversations, *arXiv preprint arXiv:2012.11820* .

Savchenko, A. V. (2021). Facial expression and attributes recognition based on multi-task learning of lightweight neural networks, *arXiv preprint arXiv:2103.17107* .

Sekhar, C., Rao, M. S., Nayani, A. K. and Bhattacharyya, D. (2021). Emotion recognition through human conversation using machine learning techniques, *Machine Intelligence and Soft Computing*, Springer, pp. 113–122.

Shen, Z.-Q. and Kong, F.-S. (2004). Dynamically weighted ensemble neural networks for regression problems, *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 04EX826)*, Vol. 6, IEEE, pp. 3492–3496.

Su, H., Jhan, J.-H., Sun, F.-y., Sahay, S. and Lee, H.-y. (2021). Put chatbot into its interlocutor's shoes: New framework to learn chatbot responding with intention, *arXiv preprint arXiv:2103.16429* .

Tripathi, S., Tripathi, S. and Beigi, H. (2018). Multi-modal emotion recognition on iemocap dataset using deep learning, *arXiv preprint arXiv:1804.05788* .

Ul, M. I. and Valles, D. (2018). Facial expression recognition from different angles using dcnn for children with asd to identify emotions, *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, IEEE, pp. 446–449.

Yang, K., Xu, H. and Gao, K. (2020). Cm-bert: Cross-modal bert for text-audio sentiment analysis, *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 521–528.

Yoon, S., Byun, S. and Jung, K. (2018). Multimodal speech emotion recognition using audio and text, *2018 IEEE Spoken Language Technology Workshop (SLT)*, IEEE, pp. 112–118.