Emotion Prediction Based on Facial Expression and Speech

Author(s): -

Kavita Virk ( [KV28@myscc.ca](mailto:KV28@myscc.ca) )

Kunal Ghaware ( [KG88@myscc.ca](mailto:KG88@myscc.ca) )

Harsh Trivedi ( [HT35@myscc.ca](mailto:HT35@myscc.ca) )

Pranav Jadhav ( [PJ25@myscc.ca](mailto:PJ25@myscc.ca) )

**ABSTRACT**

The Abstract should be a stand-alone summary of the contents of the report, equaling 250 words or less. It should present the primary objectives and scope of the study, techniques, methods or approaches briefly described and a concise summary of findings and/or conclusions reached.

The aim of this project is to build a machine learning model which can predict the emotions of a person based on the facial expression and speech of that person and then deploying this model on the webpage of CogniXR health. CogniXR health is start-up which provides telehealth services. In this project we have build 2 separate models which can predict the emotion of the person. One model will used to predict the emotions from the image while the other one will be used to predict the emotion from the speech of the person.

**Description of dataset**

There are 2 datasets used in this project. One for the emotion recognition from image and other for the emotion recognition form the audio.

The dataset used for the image recognition is downloaded from the Kaggle website. The name of the dataset is FER-2013. There are total 35887 images in this dataset which is broadly divided into training and testing dataset. The training dataset has 28,709 images while the testing dataset has 7,178 number of images. However, we have used only 50% of the training data for training our model due to computing constraints. These datasets are further divided into 7 different categories as follows: Angry (13.86%), Disgust (1.55%), Fear (14.57%), Happy (25.11%), Neutral (17.12%), Sad (16.78%), Surprise (11.02%). All the images are in grayscale of size 48 X 48 pixels.

The datasets used for Audio emotion recognition is downloaded from Kaggle website. To make the dataset multilingual, we have mixed different datasets with specific languages together. There are total 5 different languages used for our project and that are as follows: German, Italic, Urdu, Arabic and English. Expect English, audio files of other languages are spoken by both genders. The files are in WAV format. There are total 4975 Audio files in our dataset from which 20% of data we used or testing. This dataset is further divided into 7 different classes which are as follows: Angry (16.42%), Disgust (11.45%), Fear (14.23%), Happy (13.70%, Neutral (16.46%), Sad (14.59%), Surprise (13.12%).

**Analytical**

We will broadly classify this section into 3 parts.

**Image recognition model:**

For building the image recognition model, we tried multiple pretrained models. Using the transfer learning technique were successful in achieving substantial accuracy level. In the very beginning of the project, we started with CNN model which gave us the accuracy of 45% on training dataset. Then we switched to AlexNet. This is another predefined model which gave us accuracy of around 84% on training set. We also tried to use MobileNet where the accuracy for training was around 94%. But unfortunately, all the above models gave less than 50% accuracy on testing dataset. At last, we tried another pre-defined model, VGG16. It is a convolution neural net (CNN ) architecture which was used to win ILSVR(Imagenet) competition in 2014. With VGG16, our training accuracy was around 95%, while the testing accuracy was around 82%. We have used different metrics to evaluate the working of our model. The same has been depicted below.

Graphical user interface

Description automatically generated

**Audio recognition model:**

For building the audio recognition model, we started with LSTM. The accuracy for the training model was around 34% while the accuracy for testing was around 28%. We tried adjusting hyper parameters to increase the accuracy by changing the LR and dropout rate. This increased the accuracy rates for both training and testing by 4-5%. Hence, we switched to different model. We build the new model using CNN. Here, we got the training accuracy of about 87% and testing accuracy was about 79%. Here also tuned the hyper parameters to increase the accuracy level. The training accuracy we got now was around 90% and testing accuracy was around 82%. Below are the metrics we used to evaluate our model.

A picture containing chart

Description automatically generated

Table, calendar

Description automatically generated

**API:**

After building these 2 models, now we needed something that can be used to deploy this model on the website. Here comes API in action. The API will take the uploaded video and break it down into image frames. The frequency for frames is 3 seconds which means the image will be captured from the video after every 3 seconds. After an image is captured, the API will send this image to our image recognition model for emotion prediction. After predicting the emotion of the image, it would save into an array. Then the next image would be captured, and this cycle will continue until the API gets to the end of the video. Once the processing on video is finished, the API will automatically switch to another function where now it will start prediction emotions from the audio. Similar process will be seen here, where the audio is first divided into chunks of 3 seconds. Then these chunks will be sent to the audio model for emotion prediction and the output of this model will be saved into an array. At the end of the audio, the saved arrays for image and audio will be used to produce a graph which will show the count of each emotion on the webpage. The API will also generate a slider which will help the user to move the video/audio to specific moment and check the emotion at that specific time.