Name	Saad Obaid ul Islam
University	NUST School of Electrical Engineering and Computer Sciences (SEECS), Islamabad Pakistan
Year	3rd
Email	sislam.bscs15seecs@seecs.edu.pk
Skype Name	saadobaidd

Project Title:

Emotion detection and Characterization

Description:

This project extends the project initiated during GSoC2017. Aim of this project is to develop and deploy emotion-detection tools in language, voice qualities, gestures, and/or facial expressions to achieve a more complex, nuanced, and integrated characterization of emotions.

Abstract:

Emotions play an extremely important role in human communication. It is a medium of expression of a person's mental state. Emotions influence both the voice characteristics as well as linguistic content of speech. For this project, I propose a solution for extracting emotions from text and audio by using machine learning techniques. The universal emotions according to Paul Eckman model[1] are; anger, fear, happiness, surprise, sadness and disgust. For text, I will apply sentiment analysis using Recurrent Neural Networks. For Audio data, I will apply Convolution Neural Networks and Recurrent Neural Networks. Idea is to bring sentiment analysis on text together with sound cues, so that researchers could look at places where these sentiments line up or diverge in interesting ways.

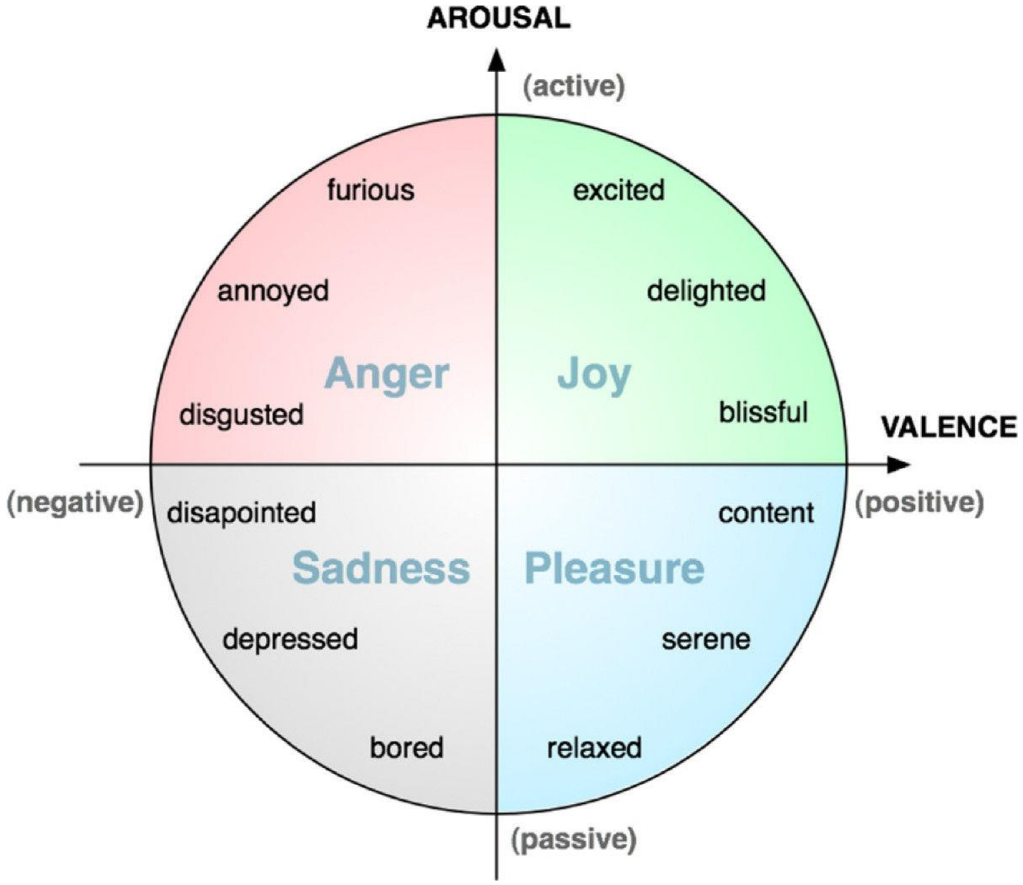
[1] [1] https://managementmania.com/en/six-basic-emotions

Current State of the Project:

The project Emotion Detection and Characterization (https://bitbucket.org/skrish13/gsoc17-krish) is currently in development stage. The main modules added to this project last year were the following:

- Silence vs Non-silence
- Music vs Speech Segment
- Speaker Identification
- Arousal vs Valence

Currently the project detects the emotion and categorizes it based on VAD model.



Proposed Enhancements to the Project:

There are two modules I would like to integrate in this project. One is sentiment Analyzer for textual data and the other is speech emotion recognition module.

- **Sentiment Analysis:** Sentiment-analysis aims to derive the emotion or feeling of a body of text. Sometimes, sentiment analysis is referred to as opinion mining, although the emphasis in this case is on extraction. The two methods of sentiment analysis are Lexicon based model and Machine Learning model.

- **Speech Emotion Recognition:** The most expressive way the humans display emotions are through facial expressions and speech characteristics. Speech Emotion Recognition is a task which relies on the effectiveness of the speech features used for classification. For Speech, Acoustic features like MFCC, LPCC, Intensity, Linguistics Features like phonemes, words, laughter and pauses, and spectrograms are used for recognizing emotions. I will be focusing on acoustic characteristics and spectrograms for speech emotion recognition.

Sentiment Analysis Implementation:

Dataset:

The dataset I will be using is from ISEAR[2], SemEVAL[3], and Saif Mohammad's hash tag emotion corpus[4]. All three of these data sets are textual and contains annotations based on universal emotions like anger, sadness, fear, happiness etc.

Methodology of Implementation:

The two ways of implementing sentiment analysis are Lexicon-based (Bag of words model) and Machine learning based. I will be applying machine learning based method for building our sentiment analyzer.

Reason for choosing machine learning model is that because the learning approach is much more accurate. I will be using Long Short-Term Memory Neural Networks for building our sentiment analyzer.

Commonly, algorithms like Naïve-bayes and Support Vector Machines are used for classifying textual data. But there are several issues with these algorithms. Naïve-bayes algorithm is very volatile. I have implemented all the Naïve-Bayes Algorithms provided in the scikit-learn library (https://github.com/WorldHellow/Sentiment_Analysis/blob/master/Classifiers_Comparison.py) and they all provide very different results every time. Support Vector Machines are great for small datasets, but they get very complex if the dataset is large. These

[2] ISEAR Databank (http://emotion-research.net/toolbox/toolboxdatabase.2006-10-13.2581092615

[3] SemEVAL (http://nlp.cs.swarthmore.edu/semeval/tasks/task14/data.shtml)

[4] http://saifmohammad.com/WebPages/lexicons.html

algorithms also require extensive feature engineering and do not achieve high recall due to diverse ways of representing emotions.[5]

Deep learning approach for detecting emotions in text has been very successful. Unlike lexicon-based approach, Deep learning using LSTMs can also detect emotions like irony and sarcasm[6]. They can understand subtleties because they create abstract representations of a text of what they learn. Such generalizations are called vectors.

I have broken down the tasks of building Sentiment Analyzer in five steps:

1. Data preprocessing: Vectorizing our generalized model
2. Building Deep RNN (LSTM)
3. Training
4. Testing
5. Hyperparameter tuning

Speech Emotion Detection and Recognition Implementation:

Dataset:

The datasets I will be using are from IEDMOCAP[7], Toronto Emotional Speech Set[8], and RAVDESS[9]. All three of these databases are audio and labelled according to the Paul Ekman model of emotions.

Methodology of Implementation:

I will build our Speech Emotion Recognizer using either the DCNN or RNN. Previously, Emotion Recognition has been done using HMMs, but I am preferring Deep Learning due to its robust self-learning mechanism and proven performance.

Recurrent Neural Networks do sequence-wise learning while the HMM uses a probabilistic model which predicts what a word is given the phenomes it was made up of.

CNNs are trained for Speech Emotion Recognition using spectrograms. Spectrograms are a visual representation of a speech signal. These speech signals are fed in to Convolution Neural Networks.

[5] https://arxiv.org/pdf/1707.06996.pdf

[6] http://publications.lib.chalmers.se/records/fulltext/251695/251695.pdf

[7] IEDMOCAP: http://sail.usc.edu/iemocap/iemocap_info.htm

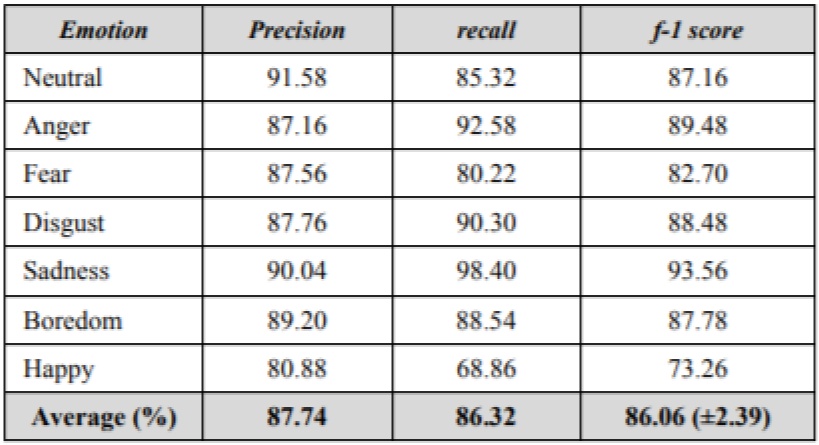
[8] Toronto Emotional Speech Set: https://tspace.library.utoronto.ca/handle/1807/24487

[9] RAVDESS: https://smartlaboratory.org/ravdess

RNNs and CNNs have both been implemented and have given promising results. In the paper “**Speech Emotion Recognition using Convolutional and Recurrent Neural Networks**”[10], RNNs resulted with an average accuracy of 79% and CNNs resulted with an average accuracy of 87.74%. The combination of both, CNN and RNN resulted with an average accuracy of 88.01%. When CNN is combined with RNN, the result is a model that learned to recognize and synthesize sequential dynamics in speech signal.

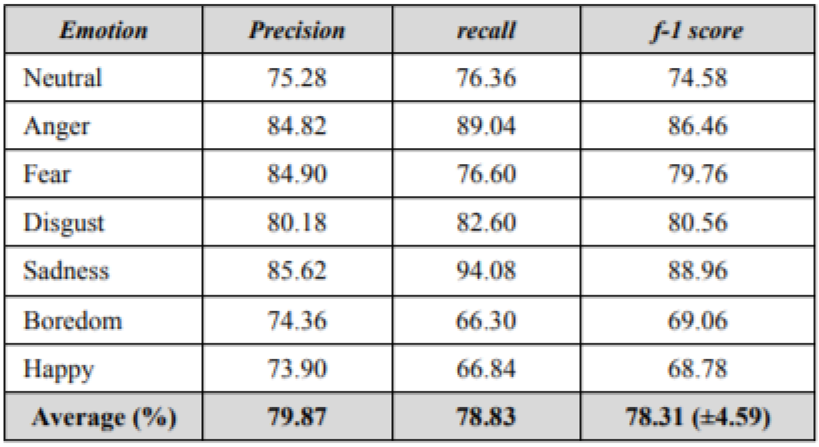
Following are the results of CNN, RNN and CNN with RNN approaches in the paper mentioned above:

TABLE II. RESULT OF EXPERIMENT 1 (CNNS)



Emotion	*Precision*	*recall*	*f-1 score*
Neutral	91.58	85.32	87.16
Anger	87.16	92.58	89.48
Fear	87.56	80.22	82.70
Disgust	87.76	90.30	88.48
Sadness	90.04	98.40	93.56
Boredom	89.20	88.54	87.78
Happy	80.88	68.86	73.26
Average (%)	**87.74**	**86.32**	**86.06 (±2.39)**

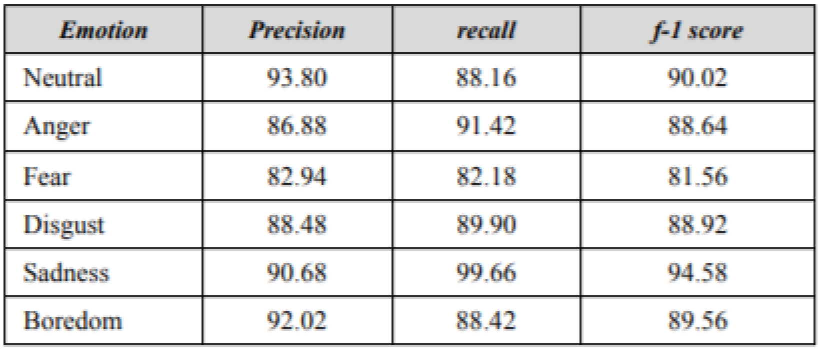
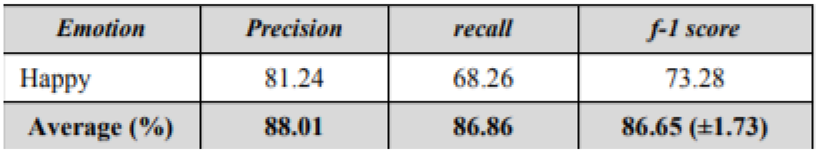
TABLE III. RESULT OF EXPERIMENT 2 (LSTM)



Emotion	*Precision*	*recall*	*f-1 score*
Neutral	75.28	76.36	74.58
Anger	84.82	89.04	86.46
Fear	84.90	76.60	79.76
Disgust	80.18	82.60	80.56
Sadness	85.62	94.08	88.96
Boredom	74.36	66.30	69.06
Happy	73.90	66.84	68.78
Average (%)	**79.87**	**78.83**	**78.31 (±4.59)**

[10] http://www.apsipa.org/proceedings_2016/HTML/paper2016/137.pdf

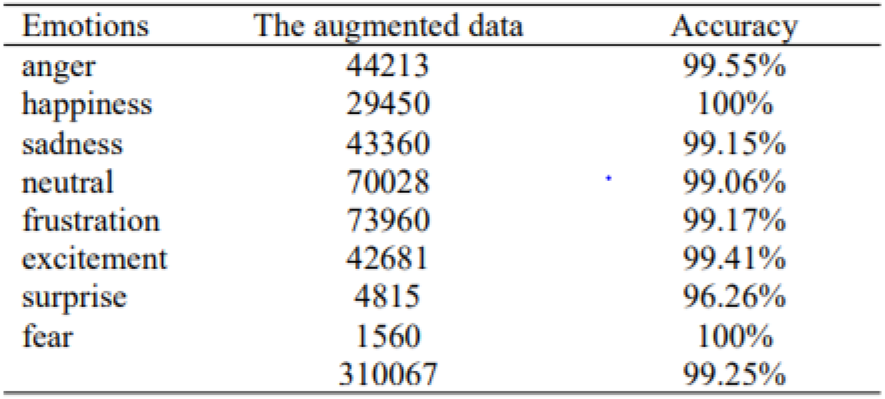
TABLE IV. RESULT OF EXPERIMENT 3 (TIME DISTRIBUTED CNNS)

Emotion	*Precision*	*recall*	*f-1 score*
Neutral	93.80	88.16	90.02
Anger	86.88	91.42	88.64
Fear	82.94	82.18	81.56
Disgust	88.48	89.90	88.92
Sadness	90.68	99.66	94.58
Boredom	92.02	88.42	89.56
Happy	81.24	68.26	73.28
Average (%)	**88.01**	**86.86**	**86.65 (±1.73)**

In a paper[11] "**A breakthrough in Speech emotion recognition using Deep Retinal Convolution Neural Networks**", Deep CNN resulted with an average accuracy of over 99% on IEMOCAP and EMO-DB databases. Idea is to first convert the speech signals in to Spectrograms using Data Augmented Algorithm Based on Retinal Imaging Principle (DAARP) and then feed these spectrograms to AlexNet[12].

TABLE 5. EXPERIMENT ON AUGMENTED DATA

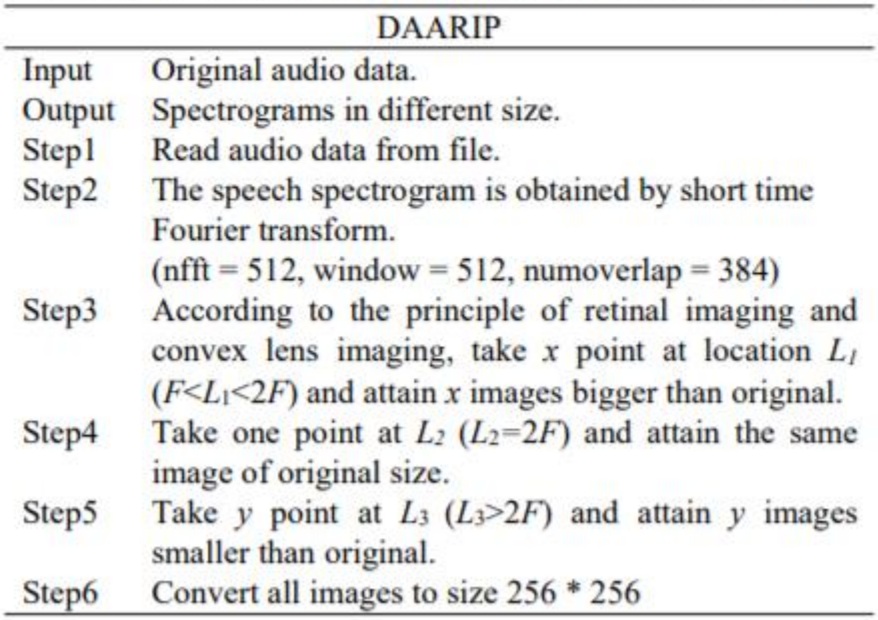


Emotions	The augmented data	Accuracy
anger	44213	99.55%
happiness	29450	100%
sadness	43360	99.15%
neutral	70028	99.06%
frustration	73960	99.17%
excitement	42681	99.41%
surprise	4815	96.26%
fear	1560	100%
	310067	99.25%

[11] https://arxiv.org/ftp/arxiv/papers/1707/1707.09917.pdf
[12] https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

Algorithm for DAARP is as follows:

TABLE 1. PSEUDO-CODE OF DAARIP ALGORITHM



DAARIP	
Input	Original audio data.
Output	Spectrograms in different size.
Step1	Read audio data from file.
Step2	The speech spectrogram is obtained by short time Fourier transform. (nfft = 512, window = 512, numoverlap = 384)
Step3	According to the principle of retinal imaging and convex lens imaging, take x point at location L_1 ($F<L_1<2F$) and attain x images bigger than original.
Step4	Take one point at L_2 ($L_2=2F$) and attain the same image of original size.
Step5	Take y point at L_3 ($L_3>2F$) and attain y images smaller than original.
Step6	Convert all images to size 256 * 256

The problem of developing a speech emotion detection and recognition system can be broken down into the following steps for each of the approaches:

RNN:

1. Choose an emotional speech Corpus.
2. Extract Acoustic Features
3. Build a classifier (RNN)
4. Develop a set of evaluation metrics
5. Testing and Validating

CNN:

1. Choose an emotional speech Corpus.
2. Convert the audio data to spectrograms.
3. Build a classifier (DCNN)
4. Develop a set of evaluation metrics
5. Testing and Validating

Timeline:

Expected Deliverables:

By Week 4: Sentiment Analyzer Testing and Validation Results**. (**Difficulty level= Intermediate)

By Week 8: Classifier Built for Speech Emotion Recognizer**. (**Difficulty level= Hard)

By Week 10: SER Testing and Validation Results**. (**Difficulty level= Intermediate)

By Week 12: Linux Terminal based api for Speech Emotion Recognition and Sentiment Analysis using Emotion Classification integrated in the present Emotion Detection and Characterization tool**. (**Difficulty level= Intermediate)

Detailed Breakdown with weekly implementation details:

I will be breaking down all the details of week and making sure the deadlines are followed so everything goes smoothly.

Community Bonding period:

During community bonding period, I will learn more about the organization from my mentors. Meanwhile, I will understand all the code in the Multimodal Emotion Detection on Videos using CNN-RNN repository. I will also finalize what kind of datasets we will be using for our Sentiment Analyzer and Speech Emotion Recognition tool. I will also discuss which approach to use for our Speech Emotion Recognition tool with my mentors.

Week 1:

In the first week I will be preprocessing the data we will be using. The steps for preprocessing the data involves loading the data and then converting it into matrices by padding using numpy. Then we will convert labels to binary vectors.

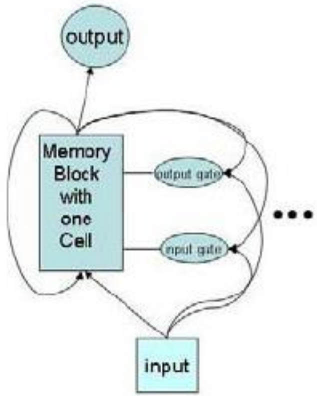
We will also need some natural language processing techniques like stemming, parts of speech tagging and tokenizing. I have implemented these techniques in the link below. https://github.com/WorldHellow/naturalLanguageProcessing

Week 2:

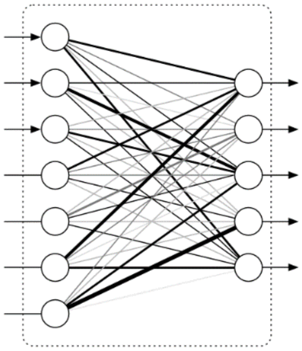
We will build our neural network using tensorflow. First layer will be our input layer and then after it will be our embedding layer. Third layer will be our LSTM layer which will remember data from the beginning of the sequences which will improve our prediction.

Fourth layer will be a fully connected layer. This layer will use softmax function as activation function. This function squashes the input of values into an output of vector probabilities between 0 and 1. The last layer will be our regression layer. This layer will use an optimization function called as “stochastic gradient descent” or adam’s optimizer. We will use cross-entropy as our loss function.

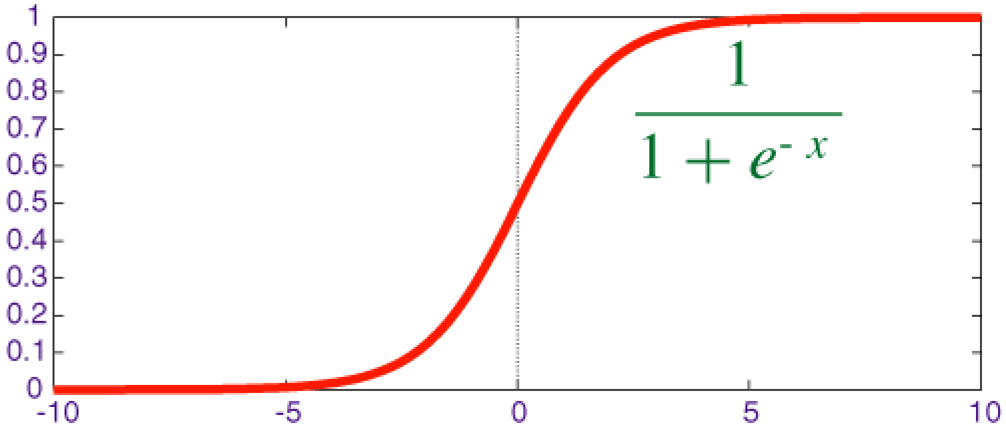
LSTM Layer:



Fully Connected Layer:



Softmax Function:



We will start the hyperparameters from the following values:
Dropout rate: 0.8
Learning rate: 0.0001

Dropout helps us prevent overfitting by randomly turning on and off different pathways in our network. Learning rate is a tradeoff between accuracy and time. If the learning rate is less, our model will be accurate but will take more time to train. If our learning rate is higher, our model will be trained quickly.

Week 3 and 4:

I will spend the third week training and testing our model. In this week, I will also tune hyperparameters to see which one works the best on which value. I will also start integrating our sentiment analyzer with the current pipeline of Emotion Detection of Red Hen's Lab.

Week 5:

First evaluation period. I will not do any further work during this week. During this week, I will make sure that everything I have previously done is working correctly and accurately.

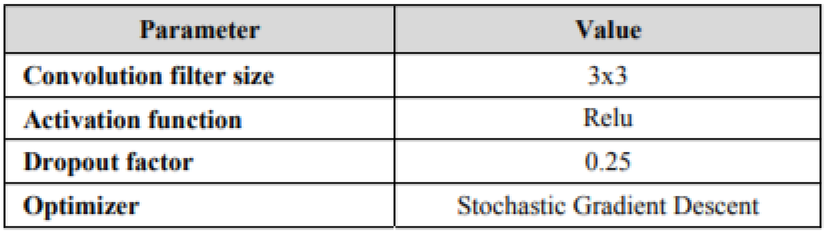
Week 6:

From week 6, I will start working on our Speech Emotion Detection and Recognition (SEDR) tool. During this week, I will work on extracting acoustic features like MFCC and LPCC values for our RNN – LSTM method. For our CNN method, I will work on converting speech signals to spectrograms.

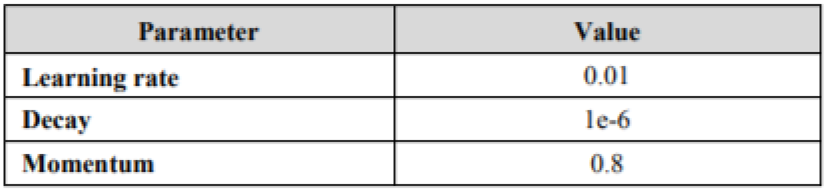
Week 7 & 8:

During this week, I will build our classifier using tensorflow. The hyperparamets will be set according to the values given in the paper **Speech Emotion Recognition using Convolutional and Recurrent Neural Networks.**

TABLE I. THE HYPER PARAMETERS AND SETTINGS OF PROPOSED NETWORK

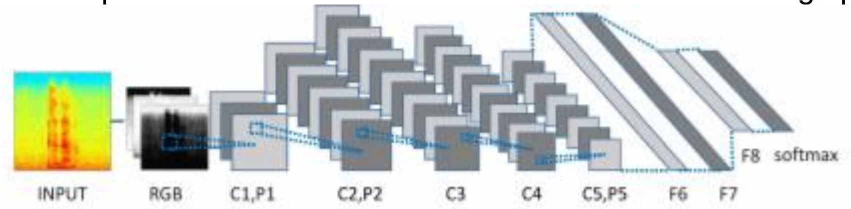


Parameter	Value
Convolution filter size	3x3
Activation function	Relu
Dropout factor	0.25
Optimizer	Stochastic Gradient Descent

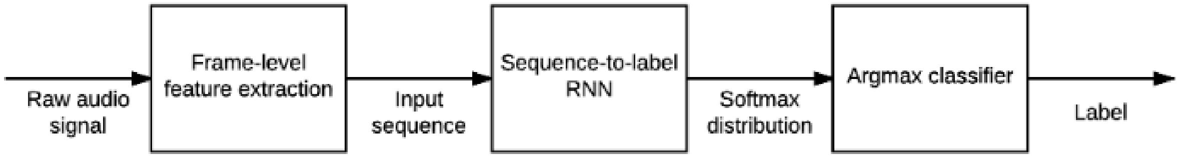


Parameter	Value
Learning rate	0.01
Decay	1e-6
Momentum	0.8

How Deep Convolutional Neural Network will look like using spectrograms:



How Recurrent Neural Network will look like using acoustic features:



Week 9:

Second evaluation Period. During this week, I will work on developing an evaluation metric for our classifier.

Week 10 and 11:

I will start with training and testing our SEDR system. After training and testing, if there is a need, I will optimize our hyperparameters. Lastly, I will integrate our SEDR tool to Red Hen's Lab Emotion Detection pipeline.

Week 12:

During this week, I will test the api to see if everything is working smoothly.

Week 13:
This week will be the final evaluation period and during this period, I will submit all my submissions for final evaluation.

Time commitment and communication:

- For the first 3 weeks, I will give 4 hours on weekdays and 3 hours on weekends.
- I will be having my finals from 4th June – 8th June so I won't be working during this week.
- From week 5th-13th, I will work for 9 hours a day on weekdays and 5 hours on weekends.

I will discuss the communication over the community bonding period with my mentors and we will finalize weekly meetings. I will email my progress to the mentors on daily basis.

Scope of the Project After GSoC:

One of the reason for using Paul Ekman Emotion Model, apart from ease of data availability according to his categorization of emotions, is that these emotions can be further used for detecting emotions using Facial Expressions. The next step in this project will be detecting emotions through facial expressions and gestures. The IEDMOCAP datasets also hold annotated visual expressions which can be used for developing facial emotion detection pipeline.

Why do I want to want to join this project?

I have developed interest in machine learning for quite some time. I find the challenge of using learning algorithms and data to train a model to be invigorating. I want to take my skills to the next level and become a part of this project under the banner of Google Summer of Code. By participating in this project, I will be able to learn to collaborate, apply various deep learning techniques and most importantly, get exposure to the psychological and linguistics branches of computer science.

About me:

I am a Computer Science Junior at School of Electrical Engineering and Computer Sciences (SEECS) NUST, Pakistan. I am interested in artificial intelligence and machine learning research. I work as a Research Assistant at my school and my contract ends by the end of March 2018. I have worked on an industrial research project "Speech based interaction for Vehicle Navigation System”. Before that I spent my summer interning at TUKL-NUST lab. My job was to investigate LIUM Speaker Diarizer on higher level.

Terms and Conditions:

I agree to work in the same manner as I explained above and any failure will hold me accountable. I understand the commitment required for this project; I am committing myself only to this project in summer so that I can complete it in the best way.