Multimodal Emotion Evaluation – Data Science Approach to Analyzing Multimodal Psychological Data in Emotions



|  |  |
| --- | --- |
| **Name:** | Kiranpreet Kaur Chahal |
| **Student ID:** | 20111118 |
| **Course:** | MSc. Advanced Computer Science |
| **Supervisor Name:** | Aliyu Sambo |
| **Date:** | 10 June, 2021 |

Table of Contents

[Table of Figures 2](#_Toc74240692)

[1. Abstract 2](#_Toc74240693)

[2. Introduction 2](#_Toc74240694)

[a. Background 2](#_Toc74240695)

[b. Aim/Objective 2](#_Toc74240696)

[c. Research Question 2](#_Toc74240697)

[d. Ethical Considerations 2](#_Toc74240698)

[3. Literature Review 2](#_Toc74240699)

[a. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English 2](#_Toc74240700)

[i. Methodology 2](#_Toc74240701)

[ii. Results 2](#_Toc74240702)

[b. Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection 2](#_Toc74240703)

[i. Methodology 2](#_Toc74240704)

[ii. Results 2](#_Toc74240705)

[c. Facial Recognition using Convolutional Neural Networks and Implementation on Smart Glasses 2](#_Toc74240706)

[d. Deep Sparse Representation Classifier for facial recognition and detection system 2](#_Toc74240707)

[e. An Experimental Study of Speech Emotion Recognition Based on Deep Convolutional Neural Networks 2](#_Toc74240708)

[f. Learning Salient Features for Speech Emotion Recognition Using Convolutional Neural Networks 2](#_Toc74240709)

[4. Project Timeline 2](#_Toc74240710)

[5. Methodology 2](#_Toc74240711)

[a. Dataset 2](#_Toc74240712)

[b. Machine Learning 2](#_Toc74240713)

[c. Results 2](#_Toc74240714)

[6. Conclusion 2](#_Toc74240715)

[7. References 2](#_Toc74240716)

[8. Appendix 2](#_Toc74240717)

# Table of Figures

[Figure 1 – Methodology (Livingstone and Russo, 2018) 2](#_Toc74205815)

[Figure 2 – Results (Livingstone and Russo, 2018) 2](#_Toc74205816)

[Figure 3 - Features in the Collected Dataset (Schmidt et al., 2018) 2](#_Toc74205817)

[Figure 4 – Results from the Three-class Classification (Schmidt et al., 2018) 2](#_Toc74205818)

[Figure 5 – Results from the Binary Classification (Schmidt et al., 2018) 2](#_Toc74205819)

[Figure 6 - Confusion Matrix (Schmidt et al., 2018) 2](#_Toc74205820)

[Figure 7 - Results (Cheng et al., 2019) 2](#_Toc74205821)

[Figure 8 – Results (Zheng, Yu and Zou, 2015) 2](#_Toc74205822)

[Figure 9 – Results (Mao, Dong, Huang and Zhan, 2014) 2](#_Toc74205823)

[Figure 10 - Project Timeline 2](#_Toc74205824)

[Figure 11 - Convolutional Neural Network Workflow 2](#_Toc74205825)

[Figure 12 - Neural Network Architecture 2](#_Toc74205826)

[Figure 13 - Training vs Validation 2](#_Toc74205827)

[Figure 14 - Confusion Matrix 2](#_Toc74205828)

# Abstract

The investigation of emotion has progressed quickly throughout the most recent decade, driven by minimal effort brilliant advances and expansive premium from analysts in neuroscience, brain research, psychiatry, audiology, and software engineering. Vital to these investigations is the accessibility of approved and solid articulations of feeling (Livingstone and Russo, 2018). To address these issues, a developing number of emotion-based datasets are now accessible. Many datasets consist of unchanging outward appearances, accounts. Very few have varying media chronicles from the Northside of America. Developing acknowledgement exists for contemplating problems of the nervous system plus their restoration. (Livingstone and Russo, 2018).

This research aims to develop a Machine Learning algorithm/ensemble using modern/state-of-the-art techniques that can classify various human emotions using supervised learning. The dataset used will be from the OASIS study. Since the paradigm is classification and due to multiple emotions, the paradigm is multi-class classification and the dataset this research is using consists of image data, the algorithm that will be used for this research will be the Convolutional Neural Networks.

# Introduction

A pattern in emotion research has been the utilization of full of feeling upgrades that portray feeling in a solitary methodology, basically through outward appearances. Nonetheless, in the normal world, passionate correspondence is fleeting. Research has featured significance the multi-sensory combination possesses while preparing full of feeling improvements (Livingstone and Russo, 2018). The shortfall of approved multimodal sets has roused analysts to make their multimodal boosts (Livingstone and Russo, 2018).

Analysts have likewise made multimodal upgrades by consolidating two free unimodal sets (Livingstone and Russo, 2018), or getting self-made boosts together with a current unimodal set (Livingstone and Russo, 2018). This specially appointed technique might entangle examination done on discoveries along considers since datasets differ concerning highlights, specialized content and expression. This research aims to create a method using Machine Learning to enable the classification and identification of emotions to help further the study of emotions.

## Background

Along these lines, different discoveries might be somewhat inferable from varieties in boost sets. The typical discussion contains an assortment of looks, and faces are infrequently, if at any point, static. However, most sets contain just static facial pictures (Livingstone and Russo, 2018). There is presently significant proof that facial development works full of feeling preparation (Livingstone and Russo, 2018). Imaging considers have uncovered those dynamic articulations bring out differential and improved examples of neural initiation comparative with static articulations (Livingstone and Russo, 2018).

Electromyography contemplates having shown that unique upgrades inspire bigger mimicry reactions from the muscles in the face of eyewitnesses compared when evoked with unchanging articulations (Livingstone and Russo, 2018). Subsequently, dynamic outward appearances may give a more biologically substantial portrayal of feeling than static outward appearances.

## Aim/Objective

This research aims to produce a Machine Learning ensemble/algorithm that can identify and classify various human emotions using supervised learning. To achieve this, the following steps are followed:

* A comprehensive and detailed literature review will be performed for the validation of the integrity of this research.
* Using the OASIS study, the dataset will be collected with the consent of the people participating in the project to train the algorithm for emotion classification.
* The machine learning algorithm, i.e., the Convolutional Neural Network will be designed using extensive test runs on the dataset collected.
* The project evaluation will be done using cross-validation and performance metrics like the confusion matrix.

## Research Question

1. Will machine learning be able to contribute to the study of emotion?

## Ethical Considerations

The information in this venture is being gathered through an investigation called the OASIS study. This investigation records the miniature articulations of individuals in light of different ecological upgrades individuals don't know about. This information gathered is then utilized in this venture to create programming that can gather miniature articulations information from individuals. The General Data Protection Regulation (GDPR) has moral strategies for this in Chapter 3: Rights of the information subject. This section is summed up underneath:

* Transparency and modalities.
  + Transparent information, correspondence regarding the movement of the advantages of the topic of the dataset.
* Information and permission to singular information.
  + Providing data where singular data is assembled from the data subject.
  + Providing data where singular data has not been obtained from the data subject.
  + Accessing rights for the owner/creator.
* Amendments, annihilation.
  + Right to remedy.
  + Right to annihilation ('choose for getting ignored').
  + Notifying responsibility concerning remedy, removal of the individual material.
* Right to dissent and mechanized solitary dynamic.
  + Right to dissent.
* Limits.

The product should hold fast to these standards from the GDPR, request authorization and assent before information assortment from the client.

# Literature Review

A strong literature review provides the researcher with experience from fellow researchers working in the same domain. It also provides strong validation for the integrity of the facts and figures in the research. The papers selected to perform Literature Review are summarized in detail below.

## The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English

The RAVDESS is an approved multimodal data set of enthusiastic discourse and melody. The information base is sexual orientation adjusted comprising of 24 expert entertainers, expressing lexically-coordinated proclamations in an unbiased North American intonation (Livingstone and Russo, 2018). Discourse incorporates quiet, cheerful, tragic, irate, unfortunate, shock, and appall articulations, and tune contains quiet, cheerful, tragic, irate, and unfortunate feelings. Every articulation is delivered at two degrees of enthusiastic power, with an extra nonpartisan articulation (Livingstone and Russo, 2018).

Every state possible is accessible, what're more, voice-just configurations (Livingstone and Russo, 2018). The arrangement of 7356 accounts was each appraised multiple times on passionate legitimacy, force, and validity. Appraisals were given by 247 people who were normal for undeveloped examination members from Northern USA (Livingstone and Russo, 2018). An arrangement of 72 members gave test-retest information. Undeniable degrees of enthusiastic legitimacy and test-retest intrarater unwavering quality were accounted for. Adjusted precision techniques were introduced for helping scientists with upgrade determination (Livingstone and Russo, 2018).

### Methodology

For the approval task, 247 members each appraised a subset of the 7356 documents (Livingstone and Russo, 2018). For the dependability test, seventy-two members gave intra-member test-retest information. Approval was accomplished by requesting that members name the communicated feeling. In a few existing information bases of facial feeling, another rating strategy for approval has been executed utilizing a restricted number of exceptionally prepared members to recognize explicit face muscle withdrawals, activity packets for demonstrating an objective feeling (Livingstone and Russo, 2018). These frameworks were created for nonverbal demeanors of feeling, which include generally still faces. Conversely, vocal creation includes huge orofacial development, where developments attached to lexical substance communicate with developments identified with enthusiastic articulation. Consequently, customary muscle coding frameworks are inadmissible for approving the RAVDESS (Livingstone and Russo, 2018).

Legitimacy test conveys proportions containing passionate exactness, power with validity showing improvement across all. To help analysts in the choice of suitable boosts, (Livingstone and Russo, 2018). incorporates a composite "goodness" score, see additionally. This condition characterized by the end goal boosts getting higher proportions of exactness, force, and validity, are appointed higher goodness scores (Livingstone and Russo, 2018).

A late investigation tracked down those vocal articulations of entertainers are just imperceptibly more exact than those of lay-expressers, it is obscure if similar remains constant for outward appearances or dynamic general media articulations. A developing number of feeling sets have effectively utilized proficient or prepared entertainers (Livingstone and Russo, 2018).

Eight feelings were chosen for discourse: nonpartisan, quiet, glad, miserable, irate, unfortunate, shock, and disturb. Quiet and impartial were chosen as a benchmark. Extra states establish arrangement i.e., 6 essential, principal feelings perceived as socially widespread. This idea concerning essential feelings possesses strong ancestry with reasoning (Livingstone and Russo, 2018).

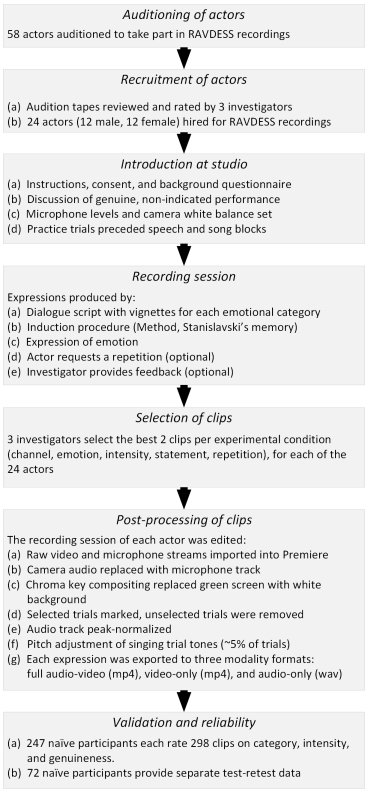


Figure 1 – Methodology (Livingstone and Russo, 2018)

### Results

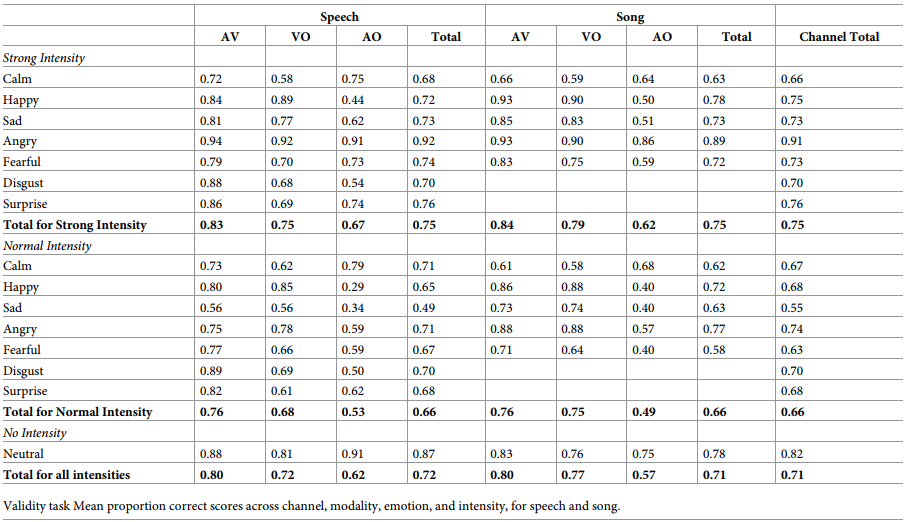


Figure 2 – Results (Livingstone and Russo, 2018)

## Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection

WESAD is freely accessible data conveying pressure and influence identification. The dataset highlights physiological and movement information, taken with a gadget placed on the wrist and the chest from fifteen volunteers (Schmidt et al., 2018).

Also, the dataset overcomes any issues between past lab concentrates on pressure and feelings, by containing three distinctive emotional states (unbiased, stress, delight). Moreover, self-reports of the subjects, which were acquired utilizing a few setup surveys, make up the data. Besides, this data is used as a threshold, utilizing notable highlights furthermore and standard AI strategies (Schmidt et al., 2018).

### Methodology

Because of the characterized study convention, (Schmidt et al., 2018) explicitly focused on graduate understudies at (Schmidt et al., 2018) exploration office. Avoidance rules, expressed in the study greeting, were pregnancy, substantial smoking, mental problems, ongoing and cardiovascular sicknesses (Schmidt et al., 2018). Altogether, 17 subjects took part in the investigation. Because of sensor breakdown, the information of two members must be disposed of. The excess 15 subjects had a mean period of 27.5 ± 2.4 years (Schmidt et al., 2018).



Figure 3 - Features in the Collected Dataset (Schmidt et al., 2018)

The separated highlights, itemized above, fill in as contribution for the arrangement step. Five AI calculations were applied and looked at inside the benchmark: DT, k-NN, RF, AB and LDA (Schmidt et al., 2018). As the whole information preparing chain was executed in Python, (Schmidt et al., 2018) utilized the scikit-learn execution of the previously mentioned classifiers. For the AB outfit student, a choice tree was utilized as the base assessor. For every one of the DT order calculations, data acquire was utilized to quantify the nature of parting choice Nodes, and the base number of tests needed to part a Node was set to 20. The number of base assessors was set to 100 for both of the troupe students (RF and AB).

(Schmidt et al., 2018) utilized precision and F1-score as assessment measurements. Exactness addresses the quantity of effectively arranged examples from total tests. The F1-measure, characterized as the consonant mean of exactness, showing the unwavering quality of the outcomes in a specific class, and review, addressing a proportion of culmination (Schmidt et al., 2018). To get the last F1-score, exactness and review were figured for each class independently and afterwards found the middle value of. Applying the F1-score is suggested for uneven arrangement assignments, which is the situation when utilizing WESAD (since the different conditions were completed at various sizes whilst investigation convention) (Schmidt et al., 2018). All models were assessed utilizing the (LOSO) cross-approval (CV) strategy. Thus, the outcomes show how a model would sum up, what's more, perform on the information of a formerly concealed subject.

### Results

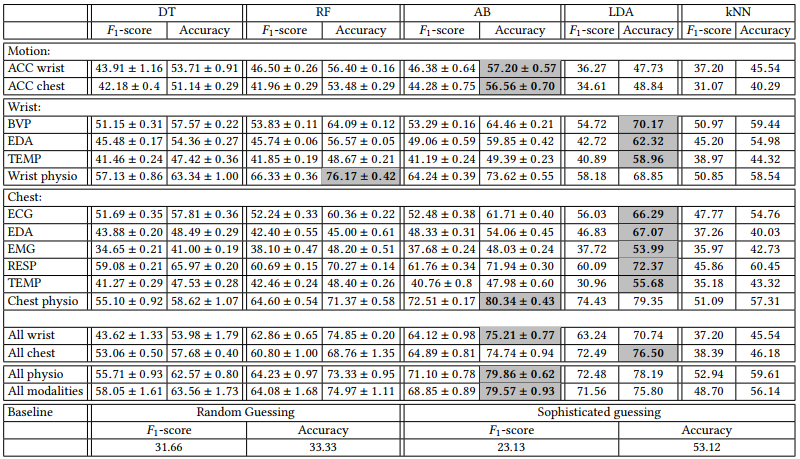


Figure 4 – Results from the Three-class Classification (Schmidt et al., 2018)

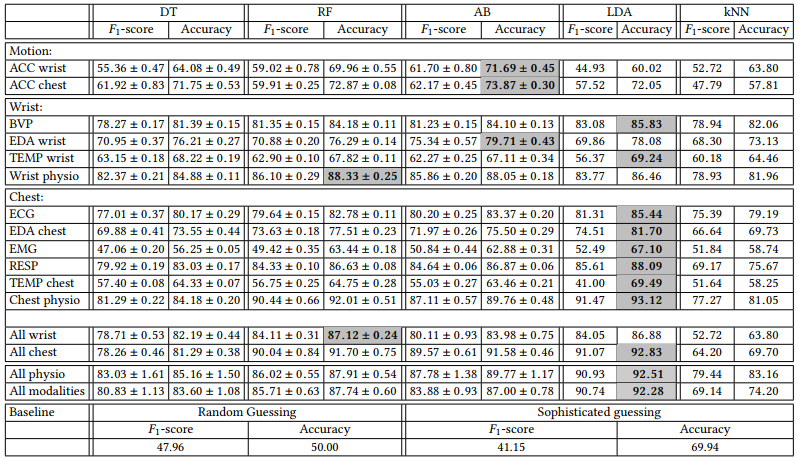


Figure 5 – Results from the Binary Classification (Schmidt et al., 2018)

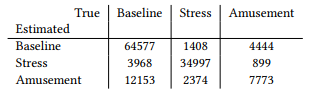


Figure 6 - Confusion Matrix (Schmidt et al., 2018)

## Facial Recognition using Convolutional Neural Networks and Implementation on Smart Glasses

Face acknowledgement is a strategy utilized for check or distinguishing proof of an individual's personality by investigating and relating designs dependent on the individual's facial highlights. The development of facial acknowledgement the way it is imbued is clarified (Khan et al., 2019). Advanced Image Processing is utilized in this task since there are numerous calculations accessible for picture control and issue like commotion can be kept away from. Man-made brainpower ordinarily incorporates Machine discovery that enables the framework to learn things consequently and make certain upgrades in them with its encounters (Khan et al., 2019). This method essentially centers around growing such projects which can get to information all alone and afterwards utilize that information for their learning (Khan et al., 2019).

Based on these features’ Facial recognition algorithms identify a person (Khan et al., 2019). The precision of CNN yield can be expanded by utilizing more variational pictures and performing more emphases while preparing the model. After finishing of cycles the precision of the model is additionally illustrated. The precision accomplished by (Khan et al., 2019) model after 1690 emphases are 98.5% The prepared model is then, at that point utilized for the facial acknowledgement (Khan et al., 2019). The model is imported and is used to arrange the info. Grouping of individuals faces pictures give verification if the information is coordinated (Khan et al., 2019).

## Deep Sparse Representation Classifier for facial recognition and detection system

This paper by (Cheng et al., 2019) conveys a binary layered CNN to get familiar with the undeniable level highlights uses facial input utilizing inadequate portrayal. Highlight extraction assumes a crucial part in certifiable example acknowledgement and grouping errands. The subtleties portrayal of the given information face picture fundamentally improves the presentation of the facial acknowledgement framework (Cheng et al., 2019). SRC classifier is a well-known facial data classifier that meagerly addresses the face picture using a small part of preparing information, thought of as uncaring toward the decision of highlight space. The proposed technique shows the exhibition improvement of SRC through a decisively chosen include exactor. The exploratory outcomes show that the proposed technique beats different strategies on given datasets (Cheng et al., 2019).

Convolutional Neural Network (CNN) takes in reformist depictions from the readiness pictures. Also, the segment maps removed by the CNN-based model is exhibited to be lacking and specific that sufficiently improve the discriminative power of face affirmation structure (Cheng et al., 2019). The proposed CNN model (Cheng et al., 2019) is made out of two convolution layers with max-pooling, additionally, a related layer which makes significantly insignificant and perceptive features for ID work. At the point when the CNN model is ready, its yield incorporate guides are used to play out the conspicuous evidence task using SRC. The proposed CNN designing is executed with the open-source significant learning framework called Caffe (Cheng et al., 2019), which is broadly embraced actually in research identified with significant learning (Cheng et al., 2019).

The nuances designing of the proposed CNN is depicted in (Cheng et al., 2019) houses binary convnet sets with max-pooling, followed by a related layer, and SoftMax yield layer showing character classes in the readiness stage (Cheng et al., 2019). In the test stage, the SoftMax layer is displaced with the SRC and the yield of the related layer is dealt with to the SRC. Despite the basic accomplishment in huge extension picture request, one normal test to CNN is that it can without a doubt encounter the evil impacts of overfitting without a ton of planning data. As (Cheng et al., 2019) prepared a model with outrageous limits and lacking planning data, the models get overfitting issue, which doesn't summarize well to other subtle data (Cheng et al., 2019). Thusly, the overfitted model can faultlessly predict planning data, in any case, bombs when expecting test data (Cheng et al., 2019). An averaging model philosophy is applied to set up a couple of one-of-a-kind models on subsets of a dataset then ordinary the yields of these freely pre-arranged associations. Averaging model is valuable to improve the execution of AI systems; regardless, it is exorbitant to set up a wide scope of huge associations. Also, the gigantic associations generally need a ton of planning data and there may not be adequate data open to get ready different associations on different subsets of the data (Cheng et al., 2019).

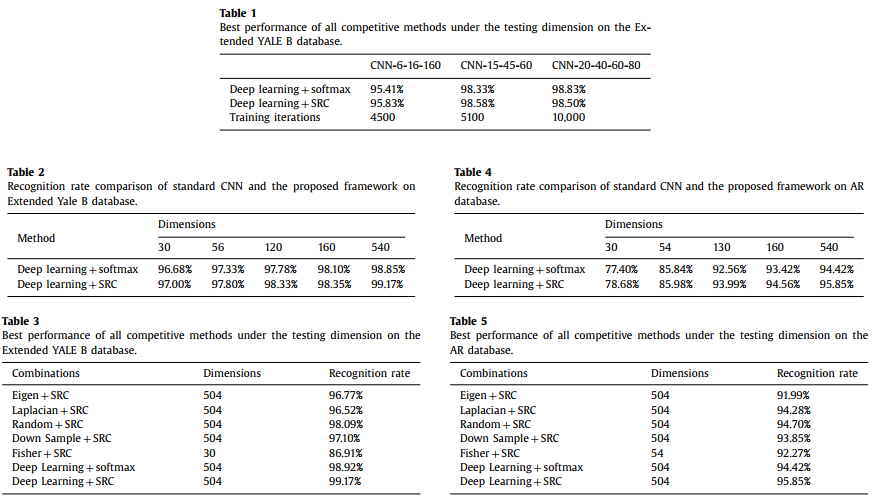


Figure 7 - Results (Cheng et al., 2019)

The paper by (Cheng et al., 2019) proposes a facial acknowledgement model which is created with a two-layer profound CNN for highlight extraction and SRC for characterization (Cheng et al., 2019). SRC gives better grouping result regardless of whether a basic element extraction technique is utilized (Cheng et al., 2019). The proposed technique shows that picking exact component space can improve the presentation of SRC. Likewise, the proposed framework is profoundly impervious to varieties of light and appearance of the facial pictures. Even though CNN has shown prevalent execution in the picture arrangement region, the immense measure of teachable boundaries makes it hard to prepare when a little dataset is utilized (Cheng et al., 2019). Besides, SRC attempt to develop a preparation word reference to inadequately address the test picture; that is, the exhibition of SRC is likewise impacted by the size of the dataset. For future work, the exhibition of the proposed framework would be assessed for an enormous scope dataset (Cheng et al., 2019).

## An Experimental Study of Speech Emotion Recognition Based on Deep Convolutional Neural Networks

Discourse feeling acknowledgement (SER) is a difficult task since it is indistinct what sort of highlights can reflect the attributes of human feeling from discourse (Zheng, Yu and Zou, 2015). Notwithstanding, conventional element extractions perform conflictingly for distinctive feeling acknowledgement assignments. Unique spectrogram gives data reflecting contrast feeling. Regardless of the incredible advances made in man-made brainpower, (Zheng, Yu and Zou, 2015) is still a long way from having the option to normally cooperate with machines, mostly because machines don't comprehend feeling states (Zheng, Yu and Zou, 2015). Feelings assume a significant part in the human-PC connection. As of late, discourse feeling acknowledgement, which expects to break down the feeling states through discourse signals, has been drawing in expanding consideration (Zheng, Yu and Zou, 2015). An experimental study of speech emotion recognition based on deep convolutional neural networks. 2015 International Conference on Affective Computing and Intelligent Interaction (ASCII) (Zheng, Yu and Zou, 2015).

Deep learning is an arising field in AI in ongoing years. An exceptionally encouraging quality of profound neural networks is that they can learn undeniable level invariant highlights from crude information (Zheng, Yu and Zou, 2015), which is possibly useful for feeling acknowledgement. Late investigations show that CNNs can learn higher request information includes and have demonstrated to be enormously fruitful models for arranging picture information (Zheng, Yu and Zou, 2015). CNN's have been utilized explicitly for the errand of discourse feeling acknowledgement. (Zheng, Yu and Zou, 2015) accomplished great execution of discourse feeling acknowledgement by attempting to learn notable element maps utilizing an auto-encoder followed by a CNN. Be that as it may, no fleeting model was used because a straight classifier was prepared across highlights from record-breaking outlines in a given discourse record to foresee the related feeling state (Zheng, Yu and Zou, 2015). Propelled by the extraordinary property of CNNs, (Zheng, Yu and Zou, 2015) applies profound CNNs to gain proficiency with the highlights for discourse feeling acknowledgement from sound spectrogram information, which can be seen as a picture portrayal of sound information along with recurrence and time tomahawks (Zheng, Yu and Zou, 2015). The spectrogram is further log-changed and prepared to utilize PCA brightening (with 60 segments) to lessen the dimensionality and some impedance to the feeling grouping task. Subsequently, primer examinations check the viability of PCA brightening and show the proposed DCNNs based discourse feeling acknowledgement technique outflanks the feeling characterization utilizing the hand-made acoustic highlights (Zheng, Yu and Zou, 2015).

To evaluate the proposed approach in (Zheng, Yu and Zou, 2015), the Intelligent Emotional Dyadic Motion Capture (IEMOCAP) informational index is used. This informational index contains changing media data from 10 performers, and (Zheng, Yu and Zou, 2015) simply utilize the soundtrack in examination (Zheng, Yu and Zou, 2015). Every articulation in the informational collection is set apart by three human annotators using supreme and dimensional imprints. In the assessment (Zheng, Yu and Zou, 2015) utilize supreme names and simply consider articulations with names from five sentiments: energy, disillusionment, ecstasy, unprejudiced and stun. Since three annotators may give different imprints for an articulation, (Zheng, Yu and Zou, 2015) take those articulations which are given something comparative imprint by at any rate two annotators to avoid vulnerability in (Zheng, Yu and Zou, 2015) investigate.

(Zheng, Yu and Zou, 2015) trains the model in a speaker-self-governing way. (Zheng, Yu and Zou, 2015) indiscriminately picks 80% articulations of each feeling portrayal to construct the planning dataset, and use the other 20% articulations for the test. The length of articulation is conflicting and its mean is 4s. = The talk data analyzed at 16 kHz is changed over into diagrams with a 25-ms window sliding at 10-ms each time.

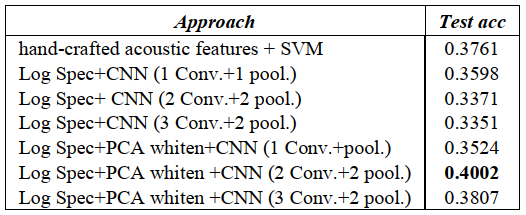


Figure 8 – Results (Zheng, Yu and Zou, 2015)

## Learning Salient Features for Speech Emotion Recognition Using Convolutional Neural Networks

As a fundamental method of human enthusiastic conduct understanding, discourse feeling acknowledgement (SER) has drawn in a lot of consideration in human-focused sign handling (Mao, Dong, Huang and Zhan, 2014). Precision in SER intensely relies upon discovering great influence related, discriminative highlights. In this paper, (Mao, Dong, Huang and Zhan, 2014) proposes to learn the influence of remarkable highlights for SER utilizing convolutional neural organizations (CNN). The preparation of CNN includes two phases. In the primary stage, unlabeled examples are utilized to learn neighborhood invariant highlights (LIF) utilizing a variety of inadequate auto-encoder (SAE) with remaking punishment (Mao, Dong, Huang and Zhan, 2014). In the subsequent advance, LIF is utilized as the contribution to an include extractor, remarkable discriminative component examination (SDFA), to learn influence striking, discriminative highlights utilizing a novel target work that supports include saliency, symmetry, furthermore, separation for SER. The exploratory outcomes on benchmark datasets show that the methodology prompts steady and hearty acknowledgement execution in complex scenes (e.g., with speaker and language variety, and climate bending) and outflanks a few grounded SER highlights (Mao, Dong, Huang and Zhan, 2014).

The influence notable element learning strategy was assessed on four public passionate discourse data sets with various dialects. The first is Surrey Audio-Visual Expressed Feeling (SAVEE) Database [12], which contains passionate discourse expressions covering seven feelings (i.e., outrage, disdain, dread, bliss, trouble, shock, and nonpartisan) purposely showed by four English speakers. The testing rate is 44.1 kHz.

The second is Berlin Emotional Database (Emo-DB) [3], which likewise incorporates seven feelings (i.e., outrage, disdain, dread, satisfaction, misery, weariness and impartial) showed by ten German entertainers. The example rate is 48 kHz. The third is the Danish Emotional Discourse data set (DES) [7], which incorporates enthusiastic discourse expressions covering five feelings (i.e., outrage, satisfaction, shock, bitterness and nonpartisan) showed by four Danish entertainers. The test rate is 48 kHz. The latter is the Mandarin Emotional Speech data set (MES) [9], which contains five feelings (i.e., outrage, satisfaction, shock, bitterness and loathing) showed by seven Mandarin entertainers. The example rate is 11.025 kHz.

Aside from the examinations directed with language change (Area IV-C4), (Mao, Dong, Huang and Zhan, 2014) handles every data set independently. That is, highlights are learned and assessed utilizing the examples from the same enthusiastic discourse data set. In particular, (Mao, Dong, Huang and Zhan, 2014) originally split the information into the preparation set and the testing set. In the unaided highlight learning stage, (Mao, Dong, Huang and Zhan, 2014) trains portions utilizing 33% of arbitrarily chose information in the preparation dataset of every data set. The names are taken out and not utilized in this stage. In the subsequent stage (SDFA), (Mao, Dong, Huang and Zhan, 2014) train with all the discourse expressions in the preparation dataset.

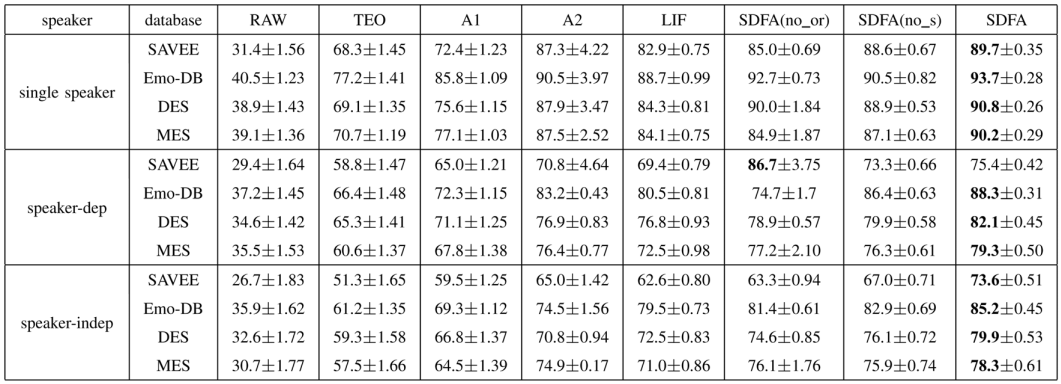


Figure 9 – Results (Mao, Dong, Huang and Zhan, 2014)

# Project Timeline

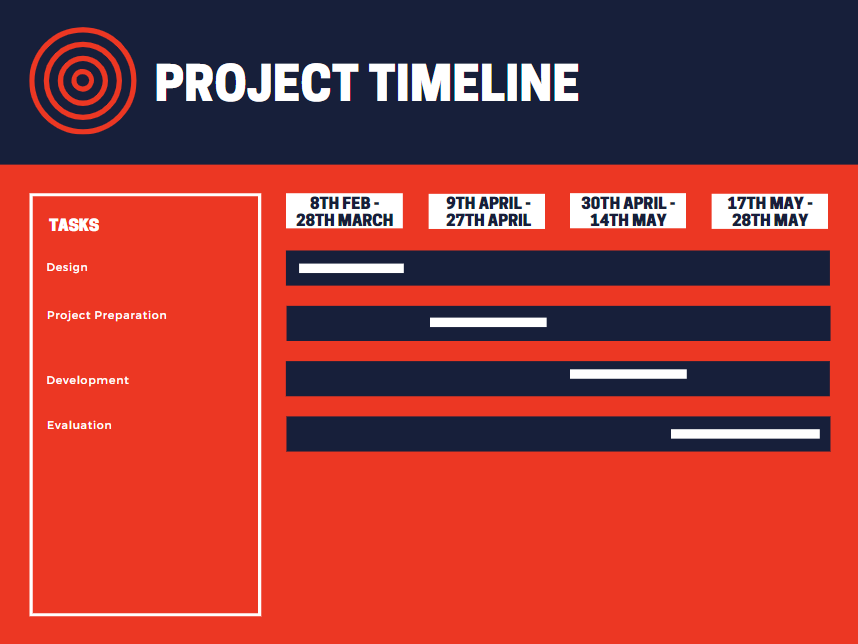


Figure 10 - Project Timeline

# Methodology

This research aims to provide a way to automate the collection of emotion data from human-environment interaction for furthering multimodal emotion research.

## Dataset

The dataset will be provided from the OASIS study. The OASIS study will conduct a study on various micro-expressive reactions based on the pre-planned stimulus to which the subjects will not be privy. This dataset will be the source of training for the Machine Learning Algorithm. The dataset is just a set of 20 images. There are no labels in the images and there is no clue of how the image must be classified. Therefore, the set of images was subjected to a group of 15 people and their expressions were recorded. The expression/emotion that had the highest frequency for each image was the classification of the image.

The images that the dataset contains are colored images. Viewed mathematically, the images are 2D matrices represented by various vectors from the colour spectrum. Therefore, a colored image is the combination of 3 colored 2D matrices; one for each channel of RGB. To pass the image into a neural network of any type for training, the requisite is that the data must be converted into numerical form since the algorithm is a computer that does not understand or possess the power to comprehend text, images and audio. Therefore, the image has to be converted into an array. Each element of the resulting array will then represent a pixel. Therefore, if the image is of the size 48x48, then the number of elements in the array will be 2304. Since each array represents a single image, each array is treated as a data instance, which means that each element of the array becomes a feature. This implies that there are 2304 features in the dataset where the image size is 48x48. But this is considering 1 colour channel per image.

The sheer intensity of features and data generated by 3 2D matrices for a single image requires a lot of resources to process. Therefore, to make this process more efficient, the author converts the dataset that has images of the size 500\*400 into size 48\*48 and grayscale. Converting the image into grayscale removes 2 of the three 2D matrices and makes it easier for the algorithm to digest and predict efficiently. Since the dataset had no labels, the images, now Numpy arrays, are associated with labels determined above. The labels are stored separately as a list in the code for reference. The data preprocessing section of the experiment is complete. The dataset created along with the labels has been saved as a .csv file in the local storage also.

The next step is the algorithm initialization with the following specs:

1. Input Conv2D layer
2. Conv2D layer: Nodes
3. Maxpool2D layer
4. Dropout (0.25)
5. Input Conv2D layer
6. Conv2D layer
7. Maxpool2D layer
8. Dropout (0.25)
9. Flatten
10. Dense Artificial Neural Network Layer: Nodes = 128, activation function: ReLU
11. Dropout (0.5)
12. Output Dense Layer: 3 Nodes, activation function = SoftMax

Upon observing the last layer of the Neural Network, it can be seen that while the other layers have 32, 64 and 128 nodes respectively, the last layer has only 3. The reason behind that is the existence of only 3 output classes as labels in the dataset. The neurons in the output layer must be = the total number of expected outputs from the dataset. The optimizer has been set to RMSprop. The data has also been split into training and test datasets. The reason behind this is for the cross-validation of the results and the algorithm after training.

Once the data preprocessing and model initialization is complete, it is time for the training of the algorithm. The algorithm was allowed to run at about 20 epochs during which the accuracy of the algorithm kept fluctuating but it came to a max value of value accuracy = 50% at epoch 17. The final epoch brought the accuracy of the Neural Network to 84.62%.

## Machine Learning

The paradigm based on the research question and the dataset selected will be the multi-class classification paradigm. The algorithm used for this would be Convolutional Neural Networks. The reason behind this selection is because the algorithm will be dealing with real-time image data. The number of layers and nodes in each layer will be decided after further testing and research. Using LSTMs or R-CNNs is also an option to give the algorithm the power to digest video input as well.

The machine learning algorithms will be provided by Python-based Machine Learning libraries like tensor-flow and Keras. These libraries have an excellent implementation of Convolutional Neural Networks (CNNs) which will be beneficial to this project. The CNN from these libraries will be trained using the dataset collected from the OASIS study. The possibility of pairing CNNs with Long-Short Term Memory (LSTM) networks does exist although this will be decided after further testing.

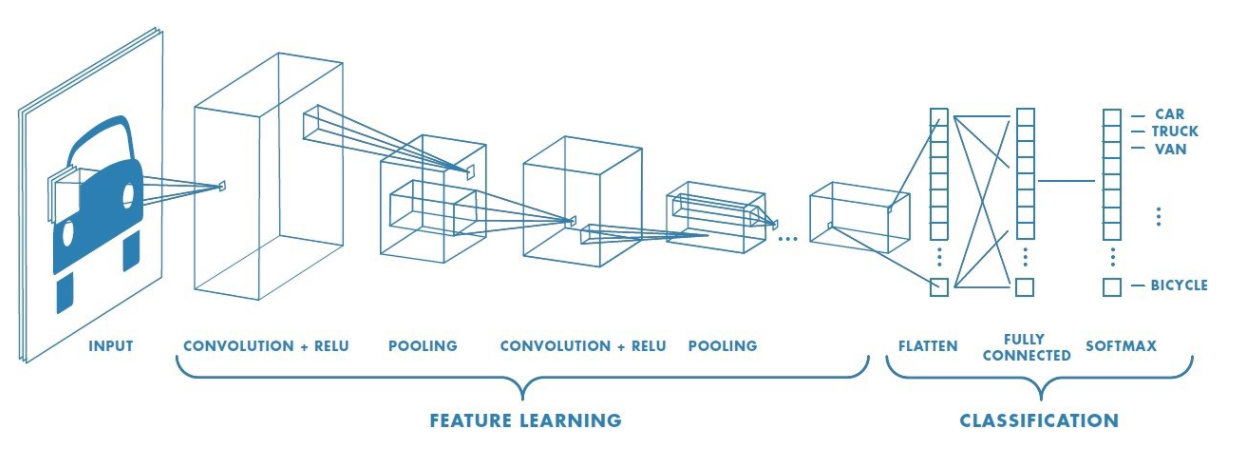


Figure 11 - Convolutional Neural Network Workflow

Convolution Neural Networks (CNNs) are neural networks that utilize the convolution activity (rather than a completely associated layer) as one of its layers (Ketkar, 2017). CNNs are an unfathomably effective innovation that has been applied to issues wherein the information on which expectations are to be made has a realized lattice-like geography like a period arrangement (which is a 1-D matrix) or a picture (which is a 2-D lattice) (Ketkar, 2017).

Profound convolutional neural organizations (CNNs) have illustrated critical enhancements over customary methodologies in many examples’ acknowledgement errands (Liu et al., 2017), for example, picture characterization (Liu et al., 2017) and video grouping (Liu et al., 2017). All the more as of late, profound CNNs have been utilized as capacity approximators in profound support figuring out how to remove powerful portrayals and help decide, which has prompted human-level execution in astute undertakings, for example, Atari games (Liu et al., 2017) and the round of Go. CNNs are an uncommon sort of neural organization for preparing information that has known, lattice-like geography. An ordinary CNN is organized as an arrangement of stages. In the initial not many stages, there are two sorts of layers: convolutional layers and pooling layers. In a convolutional layer, every neuron is associated with nearby fixes of the past layer through a bunch of loads (Liu et al., 2017).

The aftereffect of this neighborhood weighted entirety is then the contribution to an actuation work. An actuation work is a non-straight change that can keep CNNs from learning minor direct mixes of the inputs. While the convolutional layer means to recognize nearby blends of highlights from the last layer, the pooling layer means to combine semantically comparative highlights into one (Liu et al., 2017). A pooling activity processes a synopsis measurement (e.g., limit) of a nearby fix of the information. Receiving pooling in a CNN has two advantages. In the first place, pooling permits the yield to change almost no when the information shift in position and appearance. Second, pooling can altogether diminish computational expense when the network contains numerous layers (Liu et al., 2017).

A few phases of convolution, initiation work and pooling are stacked, trailed by one or a few completely associated layers (Liu et al., 2017). Then, at that point, at the yield of the model, a misfortune work is received as a method for estimating execution, in particular, the contrast between the yield of a CNN and a genuine picture name (i.e., the misfortune). The objective of preparing a CNN is to limit the misfortune work. This is typically accomplished with stochastic slope drop an advancement strategy that initially computes the slope of the misfortune work concerning the heaviness of each edge in the organization and afterwards refreshes the load as per the figured inclination (Liu et al., 2017).

The understanding of the concept and working of the neural network is of the utmost importance before diving into the understating of the convolutional neural network. Artificial neural networks models can be perceived as a bunch of fundamental preparing units, which are firmly interconnected and work on the offered contributions to handle the data and create wanted yields (Khan, Rahmani, Shah and Bennamoun, 2018). Neural networks can be assembled into two conventional classes because of how the data is proliferated in the network.

* **Feed-forward networks:** The information stream in a feed-forward network happens only one way. In case the organization is considered as an outline with neurons as its Nodes, the relationship between the Nodes is with the ultimate objective that there are no circles or cycles in the diagram. These organization models can insinuate Directed Acyclic Graphs (DAG). Models join MLP and CNNs, which will be analyzed in nuances in the approaching regions by (Khan, Rahmani, Shah and Bennamoun, 2018).
* **Feedback networks:** As the name proposes, criticism networks have affiliations that design composed cycles (or circles). This plan grants them to deal with and produce groupings of abstract sizes. Input networks show maintenance limit and can store information and progression associations in their inside memory. Cases of such models join Recurrent Neural Network (RNN) and Long-Short Term Memory (LSTM) (Khan, Rahmani, Shah and Bennamoun, 2018).

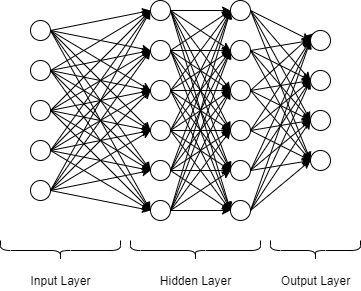


Figure 12 - Neural Network Architecture

In most straightforward terms, the network can be treated as a discovery, which works on a bunch of sources of info and produces some outputs (Khan, Rahmani, Shah and Bennamoun, 2018).

Layered Architecture: Neural organizations include a request for getting ready levels. Each level is known as a "network layer" and involves different taking care of "hubs" (moreover called "neurons" or "units"). Usually, the data is dealt with through an information layer and the last layer is the yield layer which makes assumptions. The moderate layers play out the planning and are insinuated as the mysterious layers. Due to this layered plan, this neural organization is called an MLP or a Multilayer Perceptron (Khan, Rahmani, Shah and Bennamoun, 2018).

Hubs: The individual planning units in each layer are known as the Nodes in neural organization designing. The Nodes essentially complete an "actuation work" which given data, picks if the Node will fire (Khan, Rahmani, Shah and Bennamoun, 2018).

Thick Connections: The Nodes in a neural organization are interconnected and can talk with each other. Each affiliation has a weight that shows the strength of the relationship between two Nodes. For The essential example of feed-forward neural organizations, the information is moved successively one path from the commitment to the yield layers. Thusly, every Node in a layer is connected with all Nodes in the speedy past layer (Khan, Rahmani, Shah and Bennamoun, 2018). loads of a neural organization portray the relationship between neurons. These loads ought to be laid out appropriately with the objective that an ideal yield can be gotten from the neural organization. The loads encode the "model" made from the arrangement data that is used to allow the organization to play out an appointed endeavour (e.g., object area, affirmation, just as portrayal). In realistic settings, the number of loads is gigantic which requires a customized procedure to invigorate their characteristics appropriately for a given task (Khan, Rahmani, Shah and Bennamoun, 2018).

The pattern of normally tuning the organization limits is arranged "acknowledging" which is developed during the planning stage (instead of the test stage where enlistment/assumption is made on "subtle data," i.e., data that the organization has not "seen" during getting ready) (Khan, Rahmani, Shah and Bennamoun, 2018). This cycle incorporates showing examples of the needed task to the organization with the objective that it can sort out some way to recognize the right course of action of associations between the data sources and the essential yields. For example, in the perspective of directed learning, the data sources can be media (talk, pictures) and the yields are the ideal game plan of "names" (e.g., the character of a person) which are used to tune the neural organization limits (Khan, Rahmani, Shah and Bennamoun, 2018).

CNNs are quite possibly the most well-known classifications of neural networks, particularly for high-dimensional information (e.g., pictures and recordings). CNN's work in a manner that is the same as standard neural networks (Khan, Rahmani, Shah and Bennamoun, 2018). A key contrast, notwithstanding, is that each unit in a CNN layer is a two- (or high-) dimensional channel which is convolved with the info of that layer. This is fundamental for situations where the need to take in designs from high-dimensional input media, e.g., pictures or recordings arises. CNN channels join spatial setting by having a comparative (in any case, more modest) spatial shape as the info media, and use boundary sharing to fundamentally lessen the quantity of learn-capable factor (Khan, Rahmani, Shah and Bennamoun, 2018). CNNs are a helpful class of models for both administered and unaided learning standards. The directed learning component is the one where the contribution to the framework and the ideal yields (true labels) are known and the model learns planning between the two. In an unsupervised learning instrument, the genuine marks for a given arrangement of information sources are not known also, the model expects to assess the fundamental dissemination of the information sources information tests (Khan, Rahmani, Shah and Bennamoun, 2018).

A convolutional layer is the main segment of a CNN. It includes a bunch of channels (likewise called convolutional portions) which are convolved with an offered contribution to creating a yield highlight map. Each channel in a convolutional layer is a matrix of discrete numbers (Khan, Rahmani, Shah and Bennamoun, 2018). Loads of each channel (the numbers in the framework) are gotten the hang of during the preparation of CNN. This learning methodology includes an irregular instalment of the channel loads toward the beginning of the preparation. Subsequently, given info yield matches, the channel loads are tuned in various cycles during the learning method (Khan, Rahmani, Shah and Bennamoun, 2018).

## Results

The performance metrics used for the measurement of the performance of the Neural Network algorithm are the Training vs Validation graphs and the confusion matrix. The first performance metric is shown in Figure 10. The first graph to look at is the Training loss vs Validation loss. The graph has nothing unique going on, it is a normal and expected graph, wherewith time and epochs, the loss of the neural network decreases till the graph becomes asymptotic. The second graph however is a bit peculiar. The graph for training accuracy vs validation accuracy although has the correct flow, have behaved in a very unique way. There is a lot of jaggedness in the training accuracy line and the validation accuracy stays constant for a while, faces a massive spike and immediately drops down to an asymptote. The reason behind this may be the inconsistency of the images in the dataset. The images in the dataset must have a lot of variables that might have created an inherent bias of some sorts which causes such jaggedness in the accuracy graphs. Maybe the neural network requires more epochs and more time for training before anything can be said for certain. The result even after just 20 epochs is accuracy of 84.62%. Even though there is a large amount of jaggedness present in the graphs, the high accuracy can mean that the neural network might be overfitted.

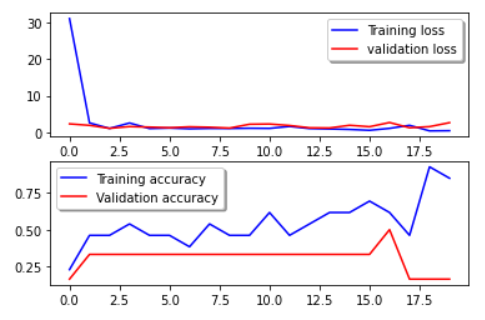


Figure 13 - Training vs Validation

The next performance metric is the confusion matrix. The confusion matrix is based on the precision and recall concepts for each output class in the dataset. The following figure 11 shows the confusion matrix.

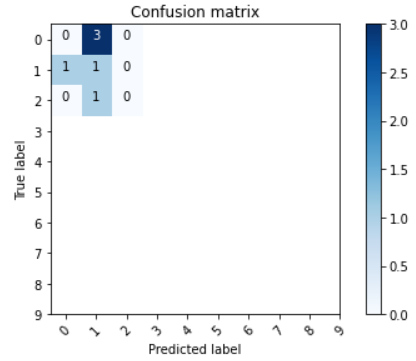


Figure 14 - Confusion Matrix

# Conclusion

This research aimed to design a Machine Learning algorithm that can detect multimodal emotional data from involuntary facial expressions using the OASIS dataset for training. To this extent, the objectives for achieving the final aim were the following:

* Data Preprocessing: The dataset is a set of images known as the OASIS dataset. The dataset contains a very random set of images with no relation of any kind between each other. The images are converted into grayscale and then flattened into a Numpy array that was then stored into a file. This was done with all the images. Since the dataset had no labels, and the work cannot proceed without any labels, the author conducted a survey that recorded the expressions and emotions of each user interacting with the OASIS dataset. This resulted in three labels namely happy, sad and null.
* The survey was conducted on 15 random people and the image was assigned the value from the responses of each image whose frequency was the highest in the list of responses in each list of responses for each image.
* A Convolutional Neural Network was selected for training and testing in this research. The algorithm was configured and trained on the OASIS dataset and 86.42% results.
* The performance of the algorithm was measured and observed using Training vs validation and confusion matrices.

The significance of emotion research can be found in a wide range of utilizations like GUI plan and critical thinking particularly issues identified with everyday environments in social orders in metropolitan zones. The compulsory responses of people to a specific ecological boost can pass on a ton of data. In any case, these aren't difficult to gather as a dataset. This examination expects to computerize the cycle of this information assortment utilizing AI and PC vision.

# References

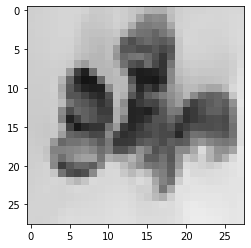
1. Livingstone, S. and Russo, F., 2018. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLOS ONE, 13(5), p.e0196391.
2. Schmidt, P., Reiss, A., Querichen, R., Marberger, C. and Van Laerhoven, K., 2018. Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. Proceedings of the 20th ACM International Conference on Multimodal Interaction.
3. Ketkar, N., 2017. Convolutional Neural Networks. Deep Learning with Python, pp.63-78.
4. Khan, S., Rahmani, H., Shah, S. and Bennamoun, M., 2018. A Guide to Convolutional Neural Networks for Computer Vision. Synthesis Lectures on Computer Vision, 8(1), pp.1-207.
5. Liu, M., Shi, J., Li, Z., Li, C., Zhu, J. and Liu, S., 2017. Towards Better Analysis of Deep Convolutional Neural Networks. IEEE Transactions on Visualization and Computer Graphics, 23(1), pp.91-100.
6. Khan, S., Javed, M., Ahmed, E., Shah, S. and Ali, S., 2019. Facial Recognition using Convolutional Neural Networks and Implementation on Smart Glasses. 2019 International Conference on Information Science and Communication Technology (ICISCT),
7. Livingstone, S. and Russo, F., 2018. The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLOS ONE, 13(5), p.e0196391.
8. Schmidt, P., Reiss, A., Duerichen, R., Marberger, C. and Van Laerhoven, K., 2018. Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. Proceedings of the 20th ACM International Conference on Multimodal Interaction.
9. Cheng, E., Chou, K., Rajora, S., Jin, B., Tanveer, M., Lin, C., Young, K., Lin, W. and Prasad, M., 2019. Deep Sparse Representation Classifier for facial recognition and detection system. Pattern Recognition Letters, 125, pp.71-77.
10. Zheng, W., Yu, J. and Zou, Y., 2015. An experimental study of speech emotion recognition based on deep convolutional neural networks. 2015 International Conference on Affective Computing and Intelligent Interaction (ASCII).
11. Mao, Q., Dong, M., Huang, Z. and Zhan, Y., 2014. Learning Salient Features for Speech Emotion <newline/>Recognition Using Convolutional <newline/>Neural Networks. IEEE Transactions on Multimedia, 16(8), pp.2203-2213.
12. Chen, J., Zhang, P., Mao, Z., Huang, Y., Jiang, D. and Zhang, Y., 2019. Accurate EEG-Based Emotion Recognition on Combined Features Using Deep Convolutional Neural Networks. IEEE Access, 7, pp.44317-44328.
13. Issa, D., Fatih Demirci, M. and Yazici, A., 2020. Speech emotion recognition with deep convolutional neural networks. Biomedical Signal Processing and Control, 59, p.101894.
14. Khorrami, P., Le Paine, T., Brady, K., Dagli, C. and Huang, T., 2016. How deep neural networks can improve emotion recognition on video data. 2016 IEEE International Conference on Image Processing (ICIP).
15. Levi, G. and Hassner, T., 2015. Emotion Recognition in the Wild via Convolutional Neural Networks and Mapped Binary Patterns. Proceedings of the 2015 ACM on International Conference on Multimodal Interaction.
16. Levi, G. and Hassner, T., 2015. Emotion Recognition in the Wild via Convolutional Neural Networks and Mapped Binary Patterns. Proceedings of the 2015 ACM on International Conference on Multimodal Interaction.
17. Li, J., Zhang, Z. and He, H., 2017. Hierarchical Convolutional Neural Networks for EEG-Based Emotion Recognition. Cognitive Computation, 10(2), pp.368-380.
18. Lim, W., Jang, D. and Lee, T., 2016. Speech emotion recognition using convolutional and Recurrent Neural Networks. 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA).
19. Song, T., Zheng, W., Song, P. and Cui, Z., 2020. EEG Emotion Recognition Using Dynamical Graph Convolutional Neural Networks. IEEE Transactions on Affective Computing, 11(3), pp.532-541.
20. Vryzas, N., Vrysis, L., Matsiola, M., Kotsakis, R., Dimoulas, C. and Kalliris, G., 2020. Continuous Speech Emotion Recognition with Convolutional Neural Networks. Journal of the Audio Engineering Society, 68(1/2), pp.14-24.

# Appendix

import NumPy as npimport pandas as pdimport matplotlib.pyplot as pltfrom matplotlib import imagefrom PIL import Imageimport itertoolsimport tensorflow as tffrom keras.utils.np\_utils import to\_categoricalfrom keras.models import Sequentialfrom keras.layers import Dense, Dropout, Flatten, Conv2D, MaxPool2Dfrom keras.optimizers import RMSpropfrom keras.preprocessing.image import ImageDataGeneratorfrom keras.callbacks import ReduceLROnPlateaufrom sklearn.model\_selection import train\_test\_splitfrom sklearn.metrics import confusion\_matrix

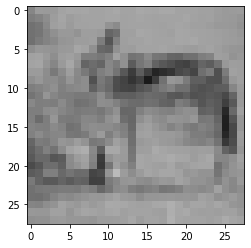
gimage = []**for** i in range(1, 20): im = Image.open('./Images/image ('+ str(i) +').jpg').convert('L').resize((28, 28)) *# Loading Images and Converting to Grayscale* gimage.append(np.asarray(im)) *# Appending in List after converting to np.array*

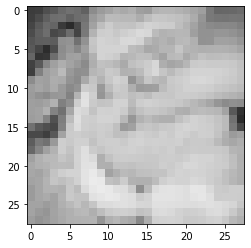
*# Plotting images from arrays in gimage list***for** j in range(len(gimage)): plt.imshow(gimage[j], cmap = 'gray', vmin = 0, vmax = 255) plt.show()

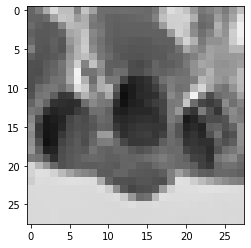


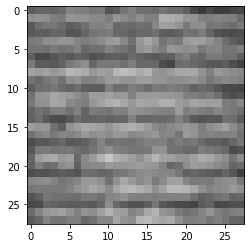


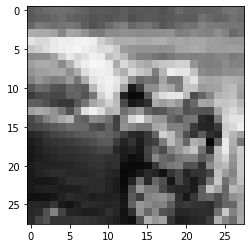


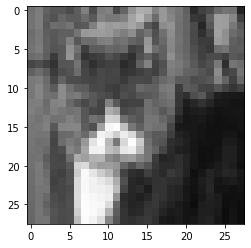


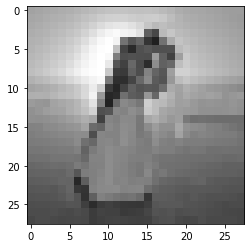


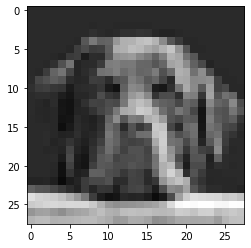


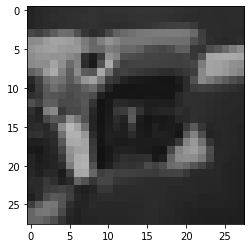






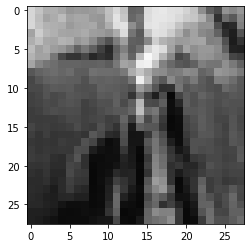


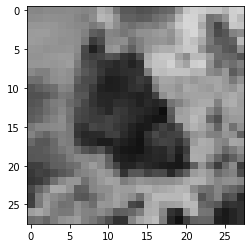


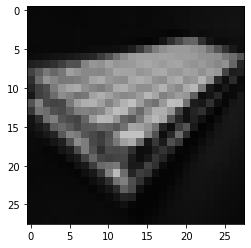


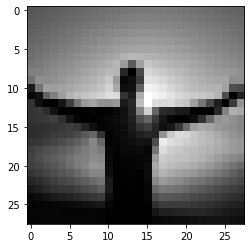


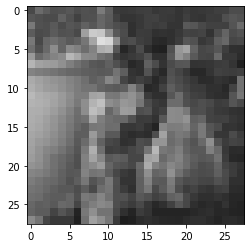








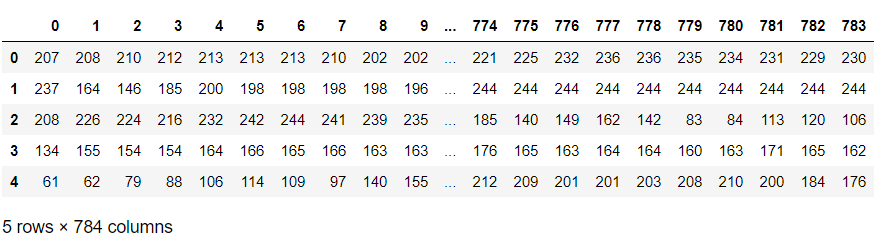




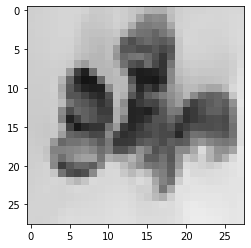
*# flattening the arrays in gimage list to convert to dataframe*fimage = []**for** k in range(len(gimage)): fimage.append(gimage[k].flatten())

*# converting to dataframe*dataset = pd.DataFrame(fimage)

dataset.head(5)

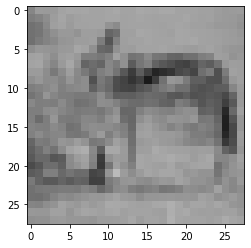


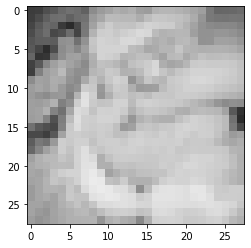
*# Plotting images from the flattened fimage list for verification***for** m in range(5): plt.imshow(fimage[m].reshape(28, 28), cmap = 'gray', vmin = 0, vmax = 255) plt.show()







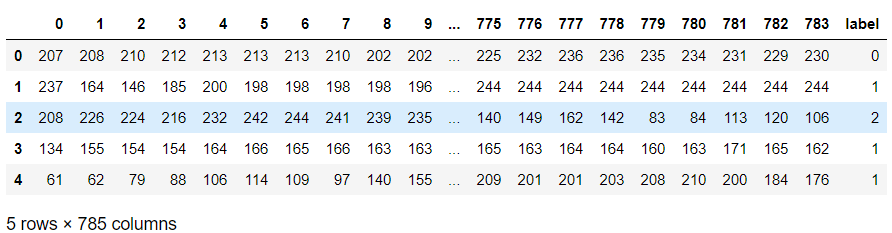




*# Adding labels to the dataset {0: null, 1: happy, 2: surprised}  
# label = [[0.,1.,0.], [0.,0.,1.], [1.,0.,0.], [0.,0.,1.], [0.,0.,1.], [0.,1.,0.], [0.,1.,0.], [1.,0.,0.], [0.,0.,1.], [0.,0.,1.], [0.,0.,1.], [1.,0.,0.], [1.,0.,0.], [0.,0.,1.], [0.,1.,0.], [0.,0.,1.], [0.,1.,0.], [0.,0.,1.], [0.,1.,0.]]*label = [0, 1, 2, 1, 1, 0, 0, 2, 1, 1, 1, 2, 2, 1, 0, 1, 0, 1, 0]dataset['label'] = label

*# Saving dataframe to directory for further use*dataset.to\_csv(r'dataset.csv')

dataset = pd.read\_csv('dataset.csv').drop(columns = 'Unnamed: 0')dataset.head(5)

**

dataset['label']

0 01 12 23 14 15 06 07 28 19 110 111 212 213 114 015 116 017 118 0Name: label, dtype: int64

dataset.shape

(19, 785)

x = dataset.drop(columns = 'label')x.shape

(19, 784)

X = dataset.drop(columns = 'label')y = dataset['label']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.30, random\_state=42)

X\_train = X\_train.values.reshape(-1, 28, 28, 1)X\_test = X\_test.values.reshape(-1, 28, 28, 1)

y\_train = tf.one\_hot(y\_train, 3)y\_test = tf.one\_hot(y\_test, 3)

print(X\_train.shape, y\_train.shape, X\_test.shape, y\_test.shape)

(13, 28, 28, 1) (13, 3) (6, 28, 28, 1) (6, 3)

model = Sequential()

model.add(Conv2D(filters = 32, kernel\_size = (5,5),padding = 'Same', activation ='relu', input\_shape = (28,28,1)))model.add(Conv2D(filters = 32, kernel\_size = (5,5),padding = 'Same', activation ='relu'))model.add(MaxPool2D(pool\_size=(2,2)))model.add(Dropout(0.25))

model.add(Conv2D(filters = 64, kernel\_size = (3,3),padding = 'Same', activation ='relu'))model.add(Conv2D(filters = 64, kernel\_size = (3,3),padding = 'Same', activation ='relu'))model.add(MaxPool2D(pool\_size=(2,2), strides=(2,2)))model.add(Dropout(0.25))

model.add(Flatten())model.add(Dense(128, activation = "relu"))model.add(Dropout(0.5))model.add(Dense(3, activation = "softmax"))

optimizer = RMSprop(lr=0.001, rho=0.9, epsilon=1e-08, decay=0.0)

model.compile(optimizer = optimizer , loss = "categorical\_crossentropy", metrics=["accuracy"])

learning\_rate\_reduction = ReduceLROnPlateau(monitor='val\_acc', patience=3, verbose=1, factor=0.5, min\_lr=0.00001)

print(X\_train.shape, X\_test.shape, y\_train.shape, y\_test.shape)

(13, 28, 28, 1) (6, 28, 28, 1) (13, 3) (6, 3)

history = model.fit(X\_train, y\_train, batch\_size = 2, epochs = 20, validation\_data = (X\_test, y\_test), verbose = 2)

Epoch 1/207/7 - 1s - loss: 31.0880 - accuracy: 0.2308 - val\_loss: 2.3177 - val\_accuracy: 0.1667Epoch 2/207/7 - 0s - loss: 2.5851 - accuracy: 0.4615 - val\_loss: 1.9106 - val\_accuracy: 0.3333Epoch 3/207/7 - 0s - loss: 1.0754 - accuracy: 0.4615 - val\_loss: 1.0973 - val\_accuracy: 0.3333Epoch 4/207/7 - 0s - loss: 2.5568 - accuracy: 0.5385 - val\_loss: 1.5742 - val\_accuracy: 0.3333Epoch 5/207/7 - 0s - loss: 1.0356 - accuracy: 0.4615 - val\_loss: 1.4359 - val\_accuracy: 0.3333Epoch 6/207/7 - 0s - loss: 1.1778 - accuracy: 0.4615 - val\_loss: 1.2673 - val\_accuracy: 0.3333Epoch 7/207/7 - 0s - loss: 0.9661 - accuracy: 0.3846 - val\_loss: 1.5159 - val\_accuracy: 0.3333Epoch 8/207/7 - 0s - loss: 1.0742 - accuracy: 0.5385 - val\_loss: 1.4009 - val\_accuracy: 0.3333Epoch 9/207/7 - 0s - loss: 1.0593 - accuracy: 0.4615 - val\_loss: 1.1706 - val\_accuracy: 0.3333Epoch 10/207/7 - 0s - loss: 1.1214 - accuracy: 0.4615 - val\_loss: 2.2162 - val\_accuracy: 0.3333Epoch 11/207/7 - 0s - loss: 1.0830 - accuracy: 0.6154 - val\_loss: 2.2801 - val\_accuracy: 0.3333Epoch 12/207/7 - 0s - loss: 1.6218 - accuracy: 0.4615 - val\_loss: 1.8752 - val\_accuracy: 0.3333Epoch 13/207/7 - 0s - loss: 1.0345 - accuracy: 0.5385 - val\_loss: 1.2430 - val\_accuracy: 0.3333Epoch 14/207/7 - 0s - loss: 0.9274 - accuracy: 0.6154 - val\_loss: 1.1948 - val\_accuracy: 0.3333Epoch 15/207/7 - 0s - loss: 0.7964 - accuracy: 0.6154 - val\_loss: 1.9358 - val\_accuracy: 0.3333Epoch 16/207/7 - 0s - loss: 0.5593 - accuracy: 0.6923 - val\_loss: 1.5384 - val\_accuracy: 0.3333Epoch 17/207/7 - 0s - loss: 1.0909 - accuracy: 0.6154 - val\_loss: 2.6862 - val\_accuracy: 0.5000Epoch 18/207/7 - 0s - loss: 1.9196 - accuracy: 0.4615 - val\_loss: 1.2563 - val\_accuracy: 0.1667Epoch 19/207/7 - 0s - loss: 0.4025 - accuracy: 0.9231 - val\_loss: 1.5625 - val\_accuracy: 0.1667Epoch 20/207/7 - 0s - loss: 0.4491 - accuracy: 0.8462 - val\_loss: 2.6482 - val\_accuracy: 0.1667

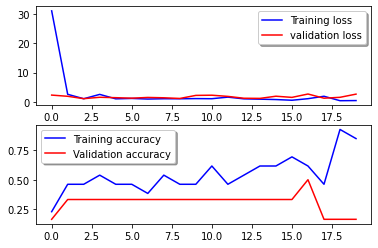
datagen = ImageDataGenerator( featurewise\_center=False, *# set input mean to 0 over the dataset* samplewise\_center=False, *# set each sample mean to 0* featurewise\_std\_normalization=False, *# divide inputs by std of the dataset* samplewise\_std\_normalization=False, *# divide each input by its std* zca\_whitening=False, *# apply ZCA whitening* rotation\_range=10, *# randomly rotate images in the range (degrees, 0 to 180)* zoom\_range = 0.1, *# Randomly zoom image* width\_shift\_range=0.1, *# randomly shift images horizontally (fraction of total width)* height\_shift\_range=0.1, *# randomly shift images vertically (fraction of total height)* horizontal\_flip=False, *# randomly flip images* vertical\_flip=False) *# randomly flip images*

datagen.fit(X\_train)

print(history.history)

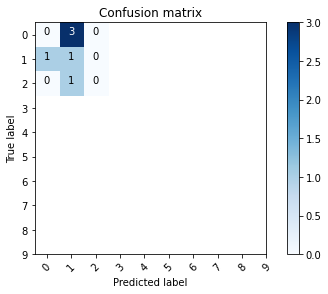
{'loss': [31.087982177734375, 2.585083484649658, 1.0753746032714844, 2.556833028793335, 1.0356119871139526, 1.177750825881958, 0.9661179780960083, 1.0741758346557617, 1.0592682361602783, 1.1213538646697998, 1.0829664468765259, 1.6217753887176514, 1.0344955921173096, 0.9274314641952515, 0.7963706254959106, 0.5592663884162903, 1.0908727645874023, 1.919561505317688, 0.4025065302848816, 0.4491156339645386], 'accuracy': [0.23076923191547394, 0.4615384638309479, 0.4615384638309479, 0.5384615659713745, 0.4615384638309479, 0.4615384638309479, 0.38461539149284363, 0.5384615659713745, 0.4615384638309479, 0.4615384638309479, 0.6153846383094788, 0.4615384638309479, 0.5384615659713745, 0.6153846383094788, 0.6153846383094788, 0.692307710647583, 0.6153846383094788, 0.4615384638309479, 0.9230769276618958, 0.8461538553237915], 'val\_loss': [2.317700147628784, 1.9105762243270874, 1.097281575202942, 1.5742197036743164, 1.4358612298965454, 1.2672605514526367, 1.5158854722976685, 1.4009289741516113, 1.170578956604004, 2.216231346130371, 2.2801029682159424, 1.8752355575561523, 1.2429604530334473, 1.1948405504226685, 1.9358230829238892, 1.5383585691452026, 2.6862475872039795, 1.2562830448150635, 1.5624996423721313, 2.6482338905334473], 'val\_accuracy': [0.1666666716337204, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.3333333432674408, 0.5, 0.1666666716337204, 0.1666666716337204, 0.1666666716337204]}

fig, ax = plt.subplots(2,1)ax[0].plot(history.history['loss'], color='b', label="Training loss")ax[0].plot(history.history['val\_loss'], color='r', label="validation loss",axes =ax[0])legend = ax[0].legend(loc='best', shadow=True)ax[1].plot(history.history['accuracy'], color='b', label="Training accuracy")ax[1].plot(history.history['val\_accuracy'], color='r',label="Validation accuracy")legend = ax[1].legend(loc='best', shadow=True)



**def** plot\_confusion\_matrix(cm, classes, normalize=False, title='Confusion matrix', cmap=plt.cm.Blues): plt.imshow(cm, interpolation='nearest', cmap=cmap) plt.title(title) plt.colorbar() tick\_marks = np.arange(len(classes)) plt.xticks(tick\_marks, classes, rotation=45) plt.yticks(tick\_marks, classes) **if** normalize: cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis] thresh = cm.max() / 2. **for** i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])): plt.text(j, i, cm[i, j], horizontalalignment="center", color="white" **if** cm[i, j] > thresh **else** "black") plt.tight\_layout() plt.ylabel('True label') plt.xlabel('Predicted label')

Y\_pred = model.predict(X\_test)Y\_pred\_classes = np.argmax(Y\_pred, axis = 1) Y\_true = np.argmax(y\_test, axis = 1) confusion\_mtx = confusion\_matrix(Y\_true, Y\_pred\_classes) plot\_confusion\_matrix(confusion\_mtx, classes = range(10))



results = model.predict(X\_test)results = np.argmax(results, axis = 1)results = pd.Series(results, name="Label")