# Capstone Project Submission

**Instructions:**
i) Please fill in all the required information.
ii) Avoid grammatical errors.

| |
|---|
| **Team Member's Name, Email and Contribution:** |
| **Mahin Arvind Chanthira Sekaran (mahinc@hotmail.com):**<br>• Data Preprocessing<br>• Data Augmentation<br>• Model Building<br>• Model Deployment<br>• Evaluation<br>• Web App Development<br>• Cloud Deployment<br>• Conclusion |
| **Please paste the GitHub Repo link.** |
| Github Link:- **https://github.com/mahin-arvind/FaceEmotionRecognition**<br><br>[If the Colab file takes too long to render, kindly use the '***Open in Colab***' chrome extension to open.] |
| **Please write a short summary of your Capstone project and its components. Describe the problem statement, your approaches and your conclusions.**<br>**(200-400 words)** |
| In this project, the Face Emotion Recognition 2013 dataset is used to effectively identify student emotions using minimum reference images and perform face emotion recognition on a webcam video feed in real-time. A web application to access the model is created and deployed on the cloud as an end-to-end solution for an e-learning platform.<br>The Keras deep learning neural network library has been used to perform image data augmentation via the ImageDataGenerator class to produce augmented images of varying brightness, horizontal orientations and zooming scales. Apart from augmenting images, all the image pixel values in the dataset are rescaled from the [0, 255] range to the [0,1] range.<br><br>During the model compilation and training phase, the loss and metrics monitored were categorical cross entropy and categorical accuracy. Adam Optimization was used with a learning rate of 0.001. Adam is a replacement optimization algorithm for stochastic gradient descent for training deep learning models. It combines the properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems. The model was trained in data with a batch size of 64 for a period of 50 epochs. For efficient training, early stopping was used to stop training when the delta value of monitored validation loss was less than zero for ten consecutive epochs (patience). A callback for reducing the learning rate by a factor of 0.02 was used to find global minima when the validation loss was observed to plateau.<br><br>The performance of five convolutional network architectures, namely, Generic ConvNet, VGG block ConvNet, Inception block ConvNet, VGG-16 and Resnet-50 were evaluated and compared to effectively classify face emotion. Precision, Recall, F1 score and Balanced Accuracy were the metrics used to evaluate and compare models using the test dataset.<br><br>The Generic ConvNet achieved a weighted F1 score of 64.38 % and a balanced accuracy score of 61.12 %. The Inception block ConvNet achieved a weighted F1 score of 49.94 % and a balanced accuracy score of 46.62 %. The VGG block ConvNet achieved a weighted F1 score of 64.95 per cent and a balanced accuracy score of 62.26 per cent.<br>The VGG-16 architecture with ImageNet weights achieved a weighted F1 score of 54.22 % and |

a balanced accuracy score of 52.47 %. The ResNet50 architecture with ImageNet weights achieved a weighted F1 score of 35.46 % and a balanced accuracy score of 30.24 %.
It was observed that the ConvNet with VGG-blocks produced the most optimal results for classifying human face emotions effectively using the evaluation scores.

For model deployment, OpenCV's Haarscascade Face Frontal file was used to capture the real-time video feed to detect and extract faces from video for emotion classification. Once the model was successfully executable, it was deployed into a web application for a user-friendly interface. Streamlit library was used to build the front-end for the application.
After running saliently on the local host, the model was deployed in the cloud using Heroku's platform and Streamlit Cloud for remote access. Remote access was observed to have a considerably long loading time as the compressed application size crossed the slug size soft limit of 300 megabytes. (Links: Heroku, Streamlit).

In conclusion of the project, five models were built and compared to effectively identify student emotions using the FER-2013 dataset. The best classifying model, ConvNet with VGG blocks, had an F1 score of 65 % and a balanced accuracy score of 62.2 %.
It was observed from the confusion matrix that the model could effectively classify emotions like Happy, Angry, Surprised and Neutral while emotions like Fear and Disgusted were classified with some difficulty. However, this issue only posed mild concern as those emotions were considered irrelevant in the context of digital learning.
The model was used to create a web application to access the model and was deployed as a model on the cloud as an end-to-end solution.
Further improvements that can be sought to be made are the inclusion of images with different levels of illumination and using sophisticated tools to make the web application lighter to suit the parallel processing of multiple video snippets for a real-life classroom environment.