Capstone Project Submission

Instructions:

i) Please fill in all the required information.

ii) Avoid grammatical errors.

Team Member's Name, Email and Contribution:
Shubham Srivastava: **E-Mail-** shubham.mach30@gmail.com • Loading Dataset • Data Inspection • Using Deep Face model to predict face emotion • Using Transfer Learning model to predict face emotion • Using CNN model to predict face emotion • Creating Layers using Kaggle kernel GPU model to predict face emotion • Real Time Video Face detection using Local Webcam(Single/Multiple Face) • Deployed using Streamlit Sharing • Challenges • Conclusion
Please paste the GitHub Repo link.
Github Link: https://github.com/Shubhamverse/Face-Emotion-Recognition--Deep-Learning_ML-Engineering **Drive Link:-** https://drive.google.com/drive/folders/1aRQg3c4t4U2qBVfphc1IPY53uFVOCHYp?usp=sharing
Please write a short summary of your Capstone project and its components. Describe the problem statement, your approaches and your conclusions. (200-400 words)

In this Deep Learning Project, we found FER-2013 dataset on Kaggle.

First, we load the dataset. In dataset we have testing and training images separately which are further divided in seven emotions: neutral, happy, sadness, anger, disgust, fear, surprise. After which we directly trained our dataset on Deep Face model. In which we analyze that the actual emotion in picture is angry but using DeepFace it is predicted to be happy. Then, we trained Transfer Learning model which also predicted wrong when used on image from webcam. Then, we trained model by creating layers i.e. CNN we applied model on images from webcams and found them to be accurate enough except for disgust which lacked sample data. Then, to increase accuracy we applied hyperparameter tuning. The training accuracy increased to 82% from the original 69% and the live feed prediction of emotion was accurate enough which is good. Then ,we used cv2 for detecting single /multiple faces for prediction on local webcam.

After saving CNN model as model.h5 file , we made app.py file to deploy webapp on streamlit . But on webapp we cannot use cv2 so, we changed it to streamlit-webrtc to get livefeed from webcam to get images and process them to predict emotions accurately. Here we also used haarcascade_frontalface to determine whether the image has a front face or not. We also made requirements.txt to mention all dependencies and their versions to be used. Then, we uploaded it to GitHub repository. And then deployed the webapp via Streamlit Sharing.