Milestone 6 – Create a CNN and train and save, and with it create Confusion matrix using dataset#3.

(Parameter setting and reliability test of a sensor system for infant carrier car seat sensing in a car using a dashboard sensor)

Masters of Engineering
Information Technology

Aneeta Antony(1431461)
aneeta.antony@stud.fra-uas.de

Sonam Priya (1427129)
sonam.priya@stud.fra-uas.de

Nitu Shrestha (1428296)
nitu.shrestha@stud.fra-uas.de

I. IMPLEMENTATION -MILESTONE 6

A. *Research on and implementation of improvements .*

1) *Otimization of CNN model and Our MLP model*

- Optimization of the models is done repeatedly by taking into consideration a lot of important things.
- Majorly the hyper parameter tuning helped in creating a stable and reliable model.

a) Model Optimisation for our MLPClassifier and RFClassifier:

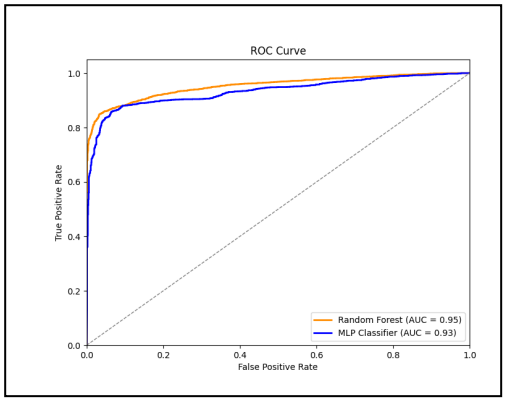
- MLP classifiers and RF classifiers can be optimized by adjusting various hyper parameters such as number of hidden layers, learning rate and, tree depth and minimum sample per leaf.
- In our project we have optimized the values of number of trees in the forest *n_estimators*, maximum depth of tree *max_depth*, minimum samples required to split an internal node *min_samples_split* and minimum number of samples required to be at leaf node *min_samples_leaf.*

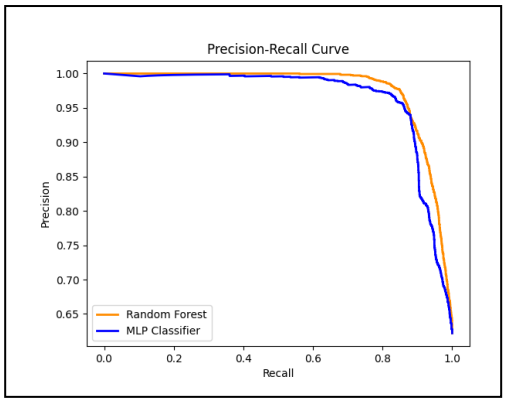
b) Model optimization for CNN model:

- CNN model could be optimized using various hyper parameters such as number of epochs, batch size, learning rate and kernel size.
- We have adjusted our kernel size to achieve better training results.
- We have adjusted our dropout regularization value to achieve better results.

2) *Characteristics after optimization of **MLP Classifier** and **Random Forest**.*

- Receiver Operating Characteristic (ROC) Curve: ROC curve is a useful tool for visualizing and comparing the performance of classification models. AUC provides a single scalar value to compare the overall performance. Also, Precision-Recall Curve like below.

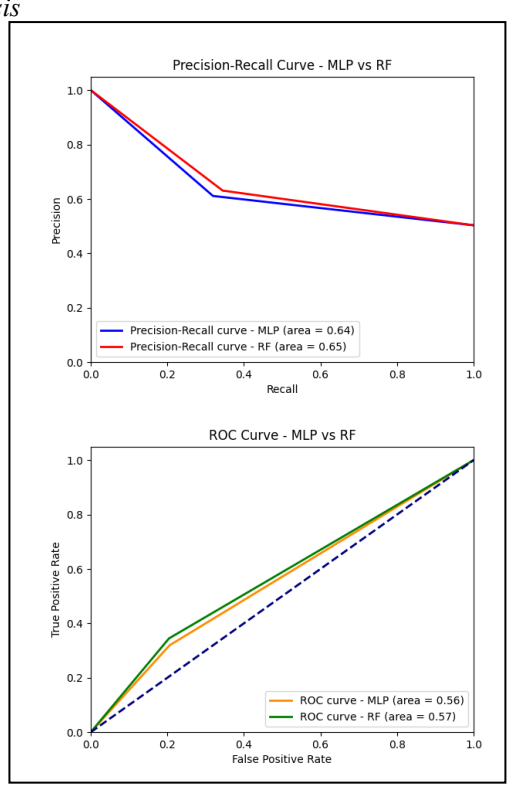




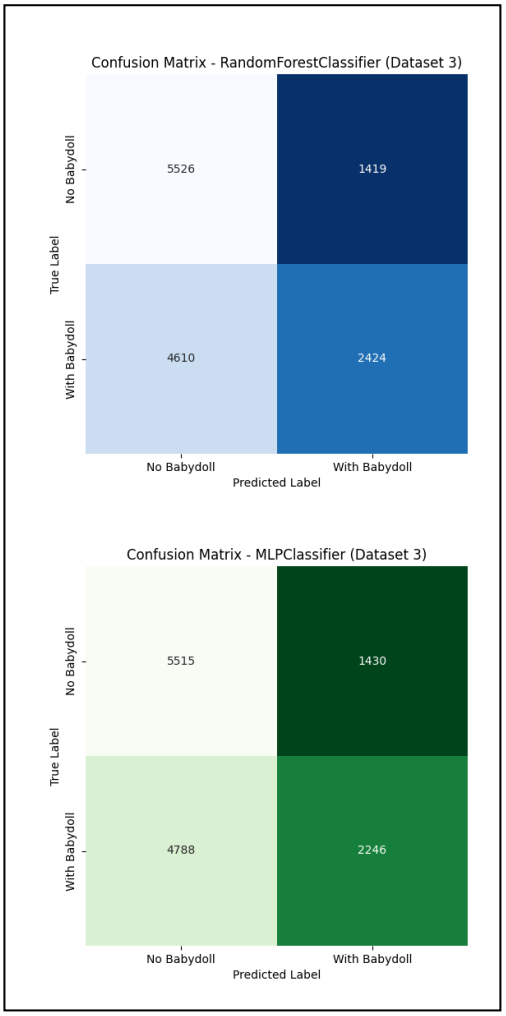
	Metric	Random Forest	MLP Classifier
1.	Accuracy	0.887924	0.888187
2.	Precision	0.930667	0.935730
3.	Recall	0.885787	0.880711
4.	F1-score	0.907672	0.907387

Table 1: Performance Metrics Table

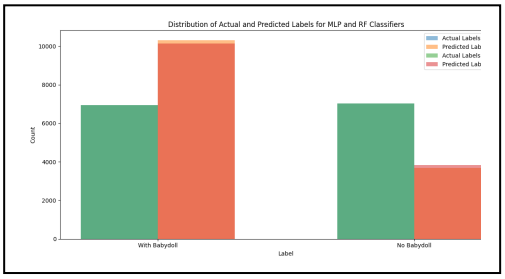
3) ***Dataset 3 on MLP and RFClassifier:*** *Performance analysis*



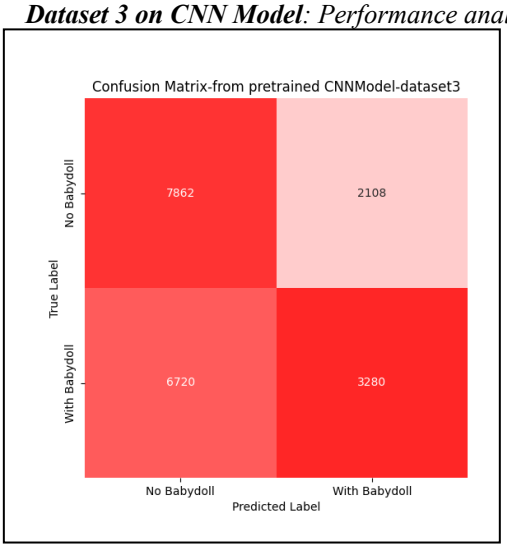
- Confusion matrix like below



- Predicted Label Distribution for both models.

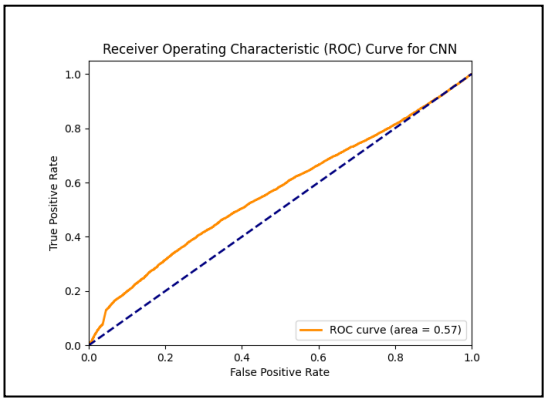


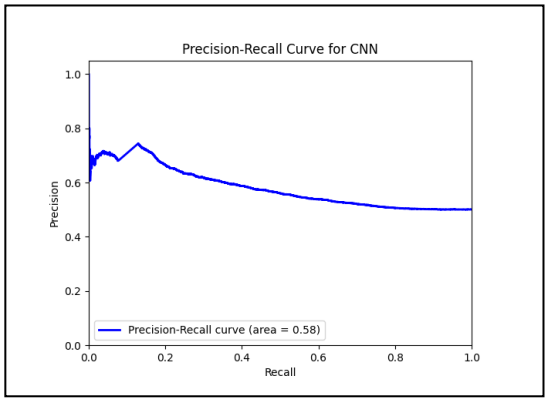
4) ***Dataset 3 on CNN Model****: Performance analysis*



Accuracy: 0.56
Precision: 0.61
Recall: 0.33
F1-score: 0.43
False Negative Rate: 0.672
False Positive Rate: 0.21143430290872617

Plots representing CNN model after optimizing and predicting labels in the 2 figures below.





B. Test the limits of the classification. Find out how the classifiers (CNN, MLP) can be obfuscated:

1. The CNN Model as well as the MLP Created has been trained and tested with various data sets:

- We observed the addition of extra measurement label helps in achieving better accuracy in terms of predicting a model with other data sets. But this sometimes leads to ***over-fitting models.***
- Overfitting models are seen to obfuscate any models' prediction to be true to self.
- The FFT magnitude is to be seen with upmost importance in having prediction to be more accurate and precise.

2. Hyper Parameter modifications:

- Some of the hyper parameters plays a crucial role in totally damaging the prediction capability.
- Adding more layers with activation parameters improves the model's capability while training to deeply access each label and its values while creating a model.

3. Obfuscated and Damaged data collected:

- This also can cause poor model classification ability and prediction precision.

4. Model Complexity:

- The models we have used in our project, including CNN and MLP has very high probability of capturing very complex patterns. Because of this, there are high chances of overfitting, which eventually leads to uncertain results.

5. Noise and Variability:

- Dataset captured for our experiments are not completely noise free. Noise appears in the dataset because of multiple reasons. Because of this, model might not be able to identify the meaningful patterns, which could lead to uncertainty in the classification of result.

6. Data Imbalance:

- Even distribution of the classes in the dataset is a key requirement for the success of our model. However, due to the imbalance in class distribution of our data, model might output uncertain classifications.

7. Boundary Cases:

- Classification results are not very accurate when the predictions revolve around the edge cases scenarios. Minor changes near the edge case could produce false positive results, leading to underconfident classifications.

C. Dataset#4 , Dataset#5.

Figure 1 Baby doll covered with scarf

Figure 2 Baby doll with sunshade pulled up

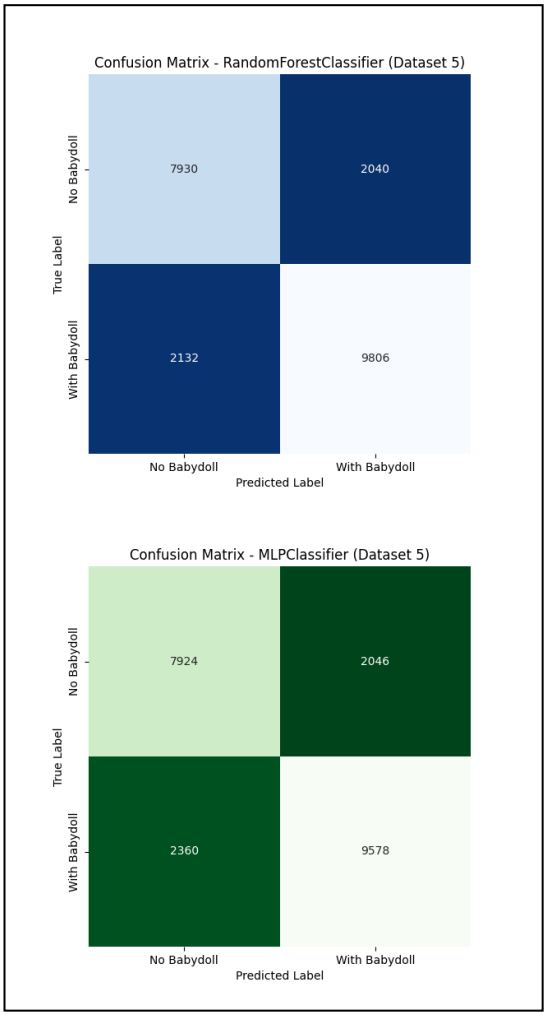
Towards this milestone, we conducted experiments at parking lot with baby doll in baby seater covered with scarf and also with Sunshade pulled up on the baby carriage. These are named as Dataset4 and 5 respectively.

The dataset#4 and dataset#5 Data is both uploaded in the OneDrive Link - Dataset4 , Dataset5

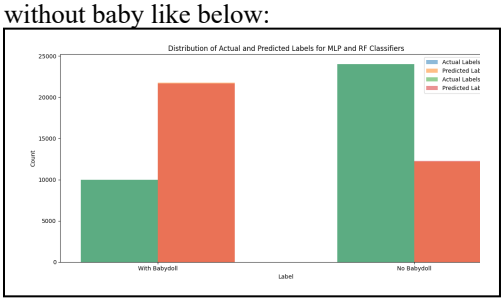
*D. **Dataset 4: baby covered in scarf:***

We applied the new data set to the Created MLP model , to verify the label prediction precision and efficiency.

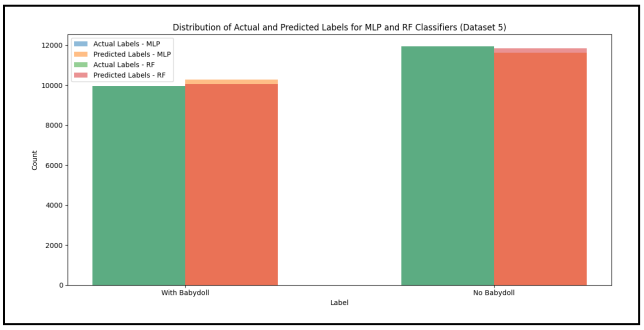
MLPClassifier Evaluation:	RandomForestClassifier Evaluation:
False Negative Rate: 0.5769166666666666 False Positive Rate: 0.20521564694082248 Cross-validation scores: [0.70650574 0.70650574 0.70650574 0.70650574 0.70650574] Mean CV score: 0.7065057403591404 Accuracy: 0.53 Precision: 0.83 Recall: 0.42 F1-score: 0.56	False Negative Rate: 0.5725833333333333 False Positive Rate: 0.2046138415245737 Cross-validation scores: [0.69634972 0.67692081 0.65911098 0.69561378 0.69443627] Mean CV score: 0.6844863114512806 Accuracy: 0.54 Precision: 0.83 Recall: 0.43 F1-score: 0.57



The predicted Label Distribution over Baby with scarf and without baby like below:



*E. **Dataset 5: baby in carriage with sunshade pulled up :***



This gave a much better prediction precision and accuracy for both the MLPs.

MLPClassifier Evaluation:	RandomForestClassifier Evaluation:
False Negative Rate: 0.19768805495057798 False Positive Rate: 0.20521564694082248 Cross-validation scores: [0.70949338 0.77110908 0.72341397 0.78429582 0.75850262] Mean CV score: 0.7493629757392901 Accuracy: 0.80 Precision: 0.82 Recall: 0.80 F1-score: 0.81	False Negative Rate: 0.17858937845535267 False Positive Rate: 0.2046138415245737 Cross-validation scores: [0.8767686 0.775445 0.72706527 0.78361105 0.87674047] Mean CV score: 0.8079260772030086 Accuracy: 0.81 Precision: 0.83 Recall: 0.82 F1-score: 0.82