Capstone Project

Real-Time Driver Drowsiness Detection Using Machine Learning and Deep Learning Techniques

By

Lee Jia Sheng

Date: 12 April 2025

Table of Contents

1. Problem statement

Driver drowsiness is a silent yet deadly threat to road safety, contributing to a significant number of traffic accidents and fatalities worldwide. In Malaysia, 54% of surveyed drivers admitted to being involved in accidents after dozing off at the wheel, and 61% reported near misses due to drowsiness [1].

One of the most dangerous aspects of drowsy driving is **microsleep** — brief, involuntary episodes of inattention lasting a few seconds. During microsleep, a driver may appear awake but fails to respond to road conditions, signals, or surrounding vehicles. These brief lapses can occur without warning and are especially deadly when operating a vehicle at high speed, where even a two-second loss of focus can lead to catastrophic consequences.

To address this issue, automotive manufacturers and technology companies have developed several drowsiness detection systems. For instance, Bosch's Driver Drowsiness Detection monitors steering behaviour to recognize early signs of fatigue, prompting the driver to take a break [2]. Similarly, companies like Netradyne have introduced AI-powered monitoring systems that assess driver alertness in real time, using visual and behavioural cues [3].

However, current systems often rely on vehicle-based cues (e.g., steering input, lane departure) or simplistic facial recognition, which may not detect subtle signs of fatigue or microsleep episodes accurately. These methods can also be susceptible to high false positive rates and struggle with real-time responsiveness under diverse lighting or environmental conditions.

This project aims to develop a **real-time driver drowsiness detection system** that leverages **machine learning and deep learning techniques** to overcome these limitations. By analysing visual cues from facial features and eye movements over time using Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks, the system seeks to detect signs of drowsiness and microsleep episodes with higher accuracy and faster response. Ultimately, the goal is to enhance driver safety, reduce fatigue-related accidents, and contribute to the development of intelligent in-vehicle safety systems.

2. Industry / Domain

As an organisation committed to enhancing road safety through innovation, our work aligns closely with the domain of **Advanced Driver Assistance Systems (ADAS)** — a vital area within the automotive industry focused on reducing human error and preventing accidents.

ADAS technologies such as lane-keep assist, adaptive cruise control, and anti-collision warning are now common in modern vehicles. A key emerging component is **driver drowsiness detection**, which addresses fatigue-related accidents — a major cause of road fatalities worldwide.

Using computer vision and deep learning, ADAS-integrated drowsiness detection systems monitor facial cues, eye movements, and behavioural patterns to identify signs of fatigue or microsleep. Upon detection, real-time alerts prompt the driver to take corrective action.

As vehicles progress toward higher levels of autonomy, ADAS solutions like drowsiness detection serve as a crucial safety layer — ensuring that drivers remain alert and responsive, ultimately supporting safer and smarter transportation systems.

3. Stakeholders

The primary stakeholder for this initiative is a local bus company that operates long-distance routes, transporting passengers from Melaka to Penang, Malaysia. The operator faces the ongoing challenge of ensuring driver alertness over extended hours, often during overnight journeys. The goal is to equip their fleet with an onboard drowsiness detection device that monitors driver alertness in real time and issues timely reminders to take breaks when signs of fatigue are detected. This not only enhances the safety of passengers and other road users, but also helps the company meet regulatory and operational safety standards. Additionally, the solution is scalable and applicable to logistics companies involved in long-haul freight transport, where driver fatigue similarly poses serious safety and efficiency risks.

4. Business question

The goal is to develop and implement a reliable, real-time driver drowsiness detection system that alerts drivers to take timely breaks, thereby reducing fatigue-related incidents in long-distance commercial bus operations. This supports broader objectives such as improving passenger safety, lowering accident-related costs, and enhancing the reputation of transport companies.

Assuming each accident involving a commercial bus results in an average cost of RM 100,000—including damages, legal liabilities, medical claims, vehicle downtime, and insurance premiums—and that drowsiness contributes to approximately 15% of such incidents, preventing just five drowsiness-related accidents per year could yield annual savings of up to RM 500,000 for a single operator. Beyond financial benefits, successful implementation can foster public trust, ensure regulatory compliance, and potentially reduce insurance premiums.

A **minimum accuracy of 90%** is targeted to ensure the system is reliable for real-world deployment. The system must carefully manage the trade-off between false positives and false negatives:

- **False Positives** (incorrectly identifying a driver as drowsy) may result in unnecessary alerts, causing driver annoyance or alert fatigue, which can diminish the effectiveness of future warnings.
- **False Negatives** (failing to detect actual drowsiness) present a serious safety risk, as an undetected fatigued driver may continue operating the vehicle, increasing the likelihood of accidents.

Therefore, the system is designed to **minimise false negatives**, even if it results in a slightly higher rate of false positives, in order to prioritise safety.

5. Data question

The data question that needs to be answered is:

How can visual data (e.g. facial features, eye movements, head position) be used to accurately detect signs of driver drowsiness in real time and distinguish between alert and drowsy states with high accuracy and reliability?

To address this question effectively, the data should possess the following characteristics: -

i. **Labelled video frames or image sequences** annotated with drowsiness levels (e.g., alert, drowsy), ideally captured under varying lighting conditions, camera angles, and across diverse driver demographics.

ii. **Temporal sequence data** that captures transitions in driver states over time, enabling the training of models such as LSTM that learn from sequential behaviour patterns.

iii. **Facial landmarks** extracted from images to highlight key physiological indicators commonly associated with fatigue (e.g., eye closure, yawning, head nodding).

iv. **Metadata,** including frame timestamps, drowsiness onset times, and subject IDs to support subject-wise evaluation and avoid data leakage.

v. A **balanced dataset** that contains an adequate representation of both alert and drowsy states across multiple individuals to ensure the model's generalisability to real-world conditions.

6. Data

The dataset used for this project is the **Driver Drowsiness Dataset (DDD)**, which is publicly available on Kaggle [4]. It consists of extracted and cropped facial images of drivers, derived from the videos in the University of Texas at Arlington Real-Life Drowsiness Dataset (UTA-RLDD), originally used in the study by Ghoddoosian, Galib and Athitsos (2019) [5]. The video frames were extracted using VLC software, and the Viola-Jones algorithm was applied to isolate the region of interest (the driver's face) from each image.

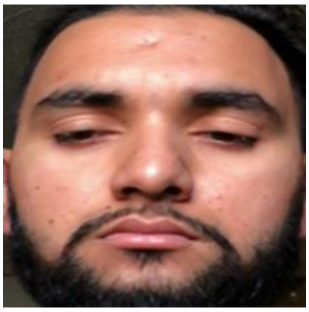
The resulting dataset contains labelled images of drivers in both alert and drowsy states, making it highly suitable for training and evaluating deep learning models for drowsiness detection. The DDD dataset was also used in the research by Nasri et al. (2021) [6], where it served as input for CNN-based models aimed at preventing road accidents through early detection of driver fatigue. Its adoption in peer-reviewed work highlights its practical relevance and reliability for real-world driver monitoring applications.

7. Data science process

7.1 Data analysis

7.1.1 Data pre-processing

The raw dataset consists of a total of 41,793 images, categorised into Drowsy and Non-Drowsy classes. Figure 1 provides example images from both categories to illustrate the visual differences between the two states.



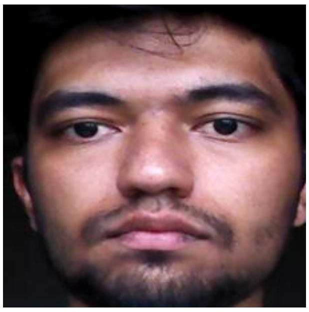


Figure 1 Examples of drowsy (left) and non-drowsy (right) images.

The raw dataset is slightly imbalanced, with Drowsy images making up 53.5% of the total and Non-Drowsy images comprising the remaining 46.5%, as illustrated in Figure 2.

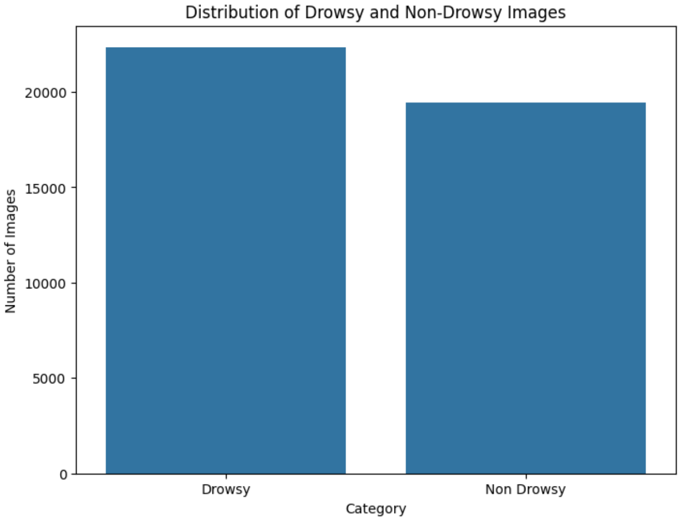


Figure 2 Distribution of drowsy and non-drowsy images in the raw dataset.

From Figure 3, it can be observed that the number of images varies across unique subjects, indicating an uneven sample distribution. Additionally, the class distribution is not consistent among subjects—some are heavily skewed toward either the drowsy or non-drowsy class, while only a few exhibit a balanced representation. Notably, Subjects F and T have zero non-drowsy images, introducing a significant bias that may impact the model's ability to generalise across different individuals.

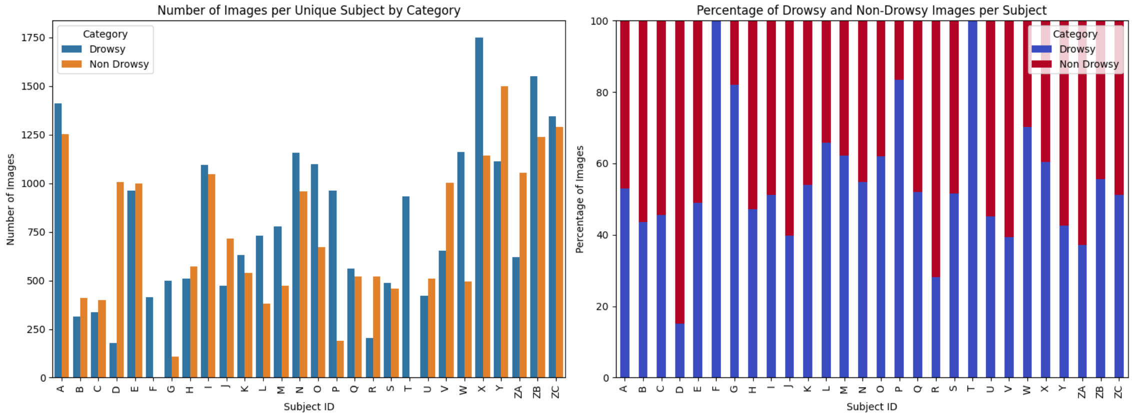


Figure 3 Number of images per unique subject by category (left) and percentage of drowsy and non-drowsy images per subject (right) in the raw dataset.

To ensure data quality and improve model generalisation, several pre-processing steps were applied to the raw dataset: -

i. **Removed Subjects F and T**

- These subjects had only drowsy images and no non-drowsy samples, introducing significant class imbalance.

ii. **Smart down-sampling**

- The number of frames per subject is limited to a maximum of 240.
- Frames were selected at even intervals to maintain the sequence integrity and temporal context.
- These steps helped reduce redundancy while preserving key behavioural transitions needed for sequence-based models like LSTM.

From Figure 4, the dataset was reduced to 6,143 drowsy images (50.3%) and 6,059 non-drowsy images (49.7%), achieving a nearly balanced class distribution after pre-processing. As shown in Figure 5, the distribution has become more consistent across subjects, with most now having a relatively equal number of drowsy and non-drowsy samples—addressing the imbalance issues present in the raw dataset.

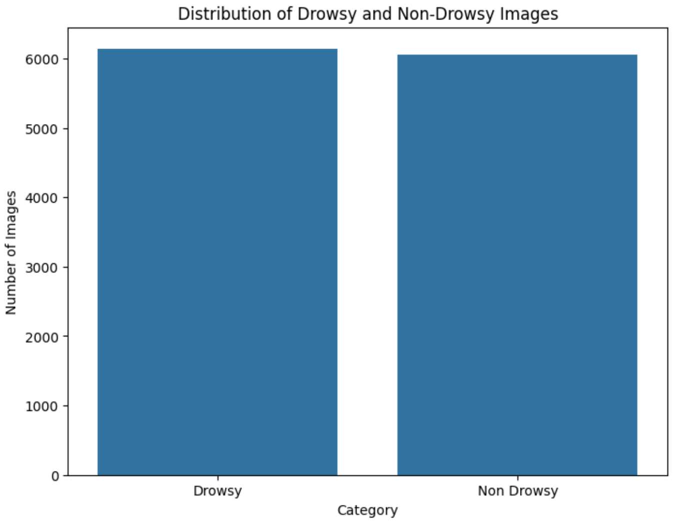


Figure 4 Distribution of drowsy and non-drowsy images after pre-processing.

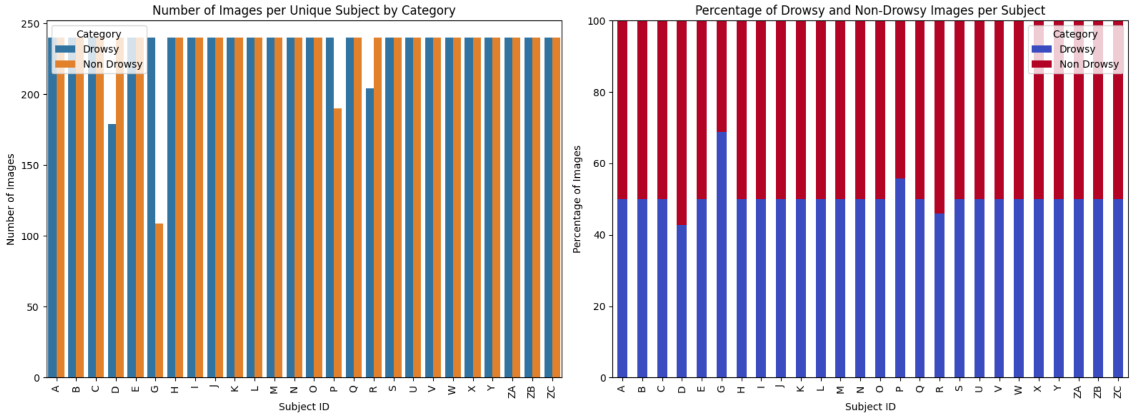


Figure 5 Number of images per unique subject by category (left) and percentage of drowsy and non-drowsy images per subject (right) after pre-processing.

7.1.2 Facial features extraction

Facial features critical to detecting drowsiness were extracted using *MediaPipe Face Mesh*, which is a machine learning solution developed by Google that provides 468 3D facial landmarks with high accuracy and real-time performance. *MediaPipe* was chosen over traditional facial landmark detection libraries like *dlib* due to several advantages: -

i. **Higher landmark density**: While *dlib* provides 68 facial landmarks, *MediaPipe* offers 468 landmarks, enabling more detailed feature extraction, especially for subtle facial movements.

ii. **Real-time performance**: *MediaPipe* is optimised for speed and can run efficiently on both CPU and GPU, making it suitable for real-time applications like drowsiness detection.

iii. **Robustness to occlusion and head movement**: *MediaPipe* maintains landmark accuracy even under challenging conditions, such as partial occlusions or non-frontal faces, which are common in driver monitoring scenarios.

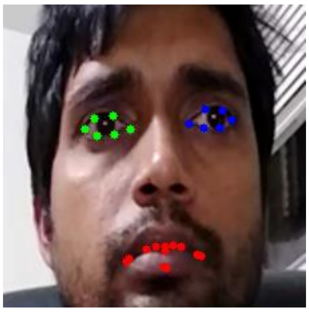


Figure 6 An example of facial landmarks at the left eye, right eye and mouth region detected by *Mediapipe*.

Once the facial landmarks have been detected, the facial features tabulated in Table 1 were extracted to be used as indicators of drowsiness. These features were selected because they are computationally efficient and have been validated in prior research as strong indicators of driver alertness levels.

Table 1 Facial features extracted from facial landmarks.

Feature	Description
Eye Aspect Ratio (EAR)	• Measures the vertical distance between eyelid landmarks divided by the horizontal distance between the eye corners. • Decreases when the eyes begin to close. • $EAR = \frac{\|\|p_2 - p_6\|\| + \|\|p_3 - p_5\|\|}{2 \cdot \|\|p_1 - p_4\|\|}$, where the p_i’s represent the landmark points at the eye region.
Mouth Aspect Ratio (MAR)	• Represents the mouth opening based on the vertical distance between upper and lower lip landmarks and the horizontal distance across the mouth. • Increases when the mouth is open, which may indicate yawning. • $MAR = \frac{\|p_3 - p_9\| + \|p_2 - p_{10}\| + \|p_4 - p_8\|}{3 \cdot \|p_0 - p_6\|}$, where the p_i’s represent the landmark points at the mouth region.
Mouth-Eye Ratio (MER)	• A derived feature calculated as the ratio between MAR and EAR, helping to enhance sensitivity to simultaneous yawning and eye closure. • $MER = \frac{MAR}{EAR}$
Pupil-to-Eye Center Distance (PECD)	• Measures the distance between the pupil and the geometric center of the eye landmarks. • Deviations can suggest gaze direction or eye movement, providing cues for distraction or drowsiness.

The extracted facial features were computed for each frame and stored in a structured DataFrame, along with the corresponding Subject ID and category label, in preparation for further exploratory data analysis (EDA) and machine learning model development. The resulting dataset consists of 12,194 rows and 6 columns, where each row represents the features extracted from a single frame.

7.1.3 Exploratory data analysis (EDA) of extracted facial features

A box plot was generated to compare the distribution of the extracted facial features across the two drowsiness categories (0: alert, 1: drowsy), with some key observations summarised in

Table 2.

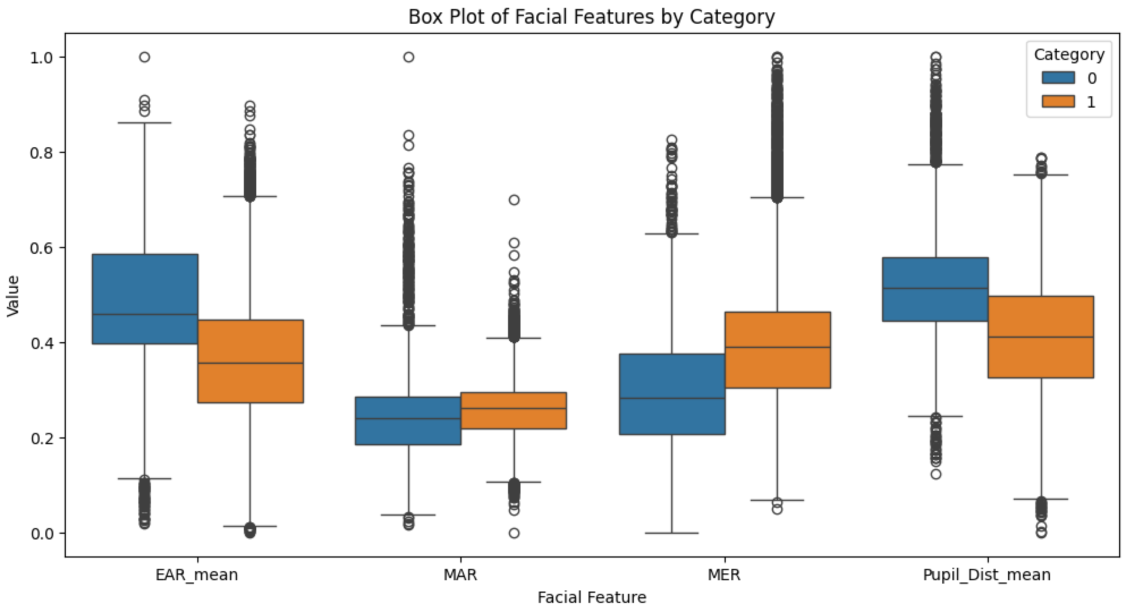


Figure 7 Box plot of facial features by category.

Table 2 Summary of key observations of box plot.

Feature	Observation	Insight
EAR	Higher median and wider IQR in Category 0; lower median in Category 1.	Strong indicator of eye closure in drowsy states.
MAR	Slightly higher median in Category 1; more variability.	Suggests mouth opening (i.e. yawning) is more frequent in drowsy individuals.
MER	Noticeably higher median in Category 1.	Indicates increased mouth opening relative to eye openness during drowsiness.
PECD	Higher median in Category 0; lower and more variable in Category 1.	Suggests focused gazed in alert individuals vs relaxed or unfocused gaze in drowsy ones.
General pattern	Clearer separation between alert and drowsy states in EAR and PECD; more outliers in MAR and MER.	EAR and PECD are more stable indicators.

To better understand the relationships between extracted facial features and drowsiness states, both a pair plot and a correlation heatmap were generated as shown in Figure 8 and Figure 9. Table 3 provides a summary of insights that can be deduced from the pair plot and correlation heatmap.

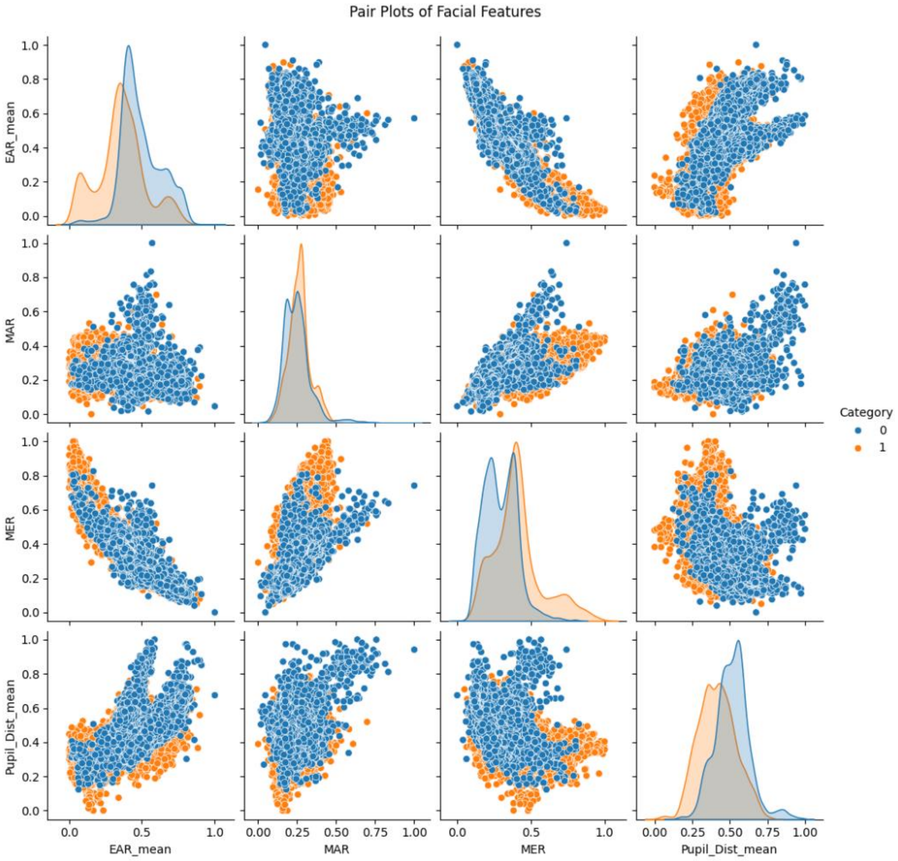


Figure 8 Pair plots of facial features

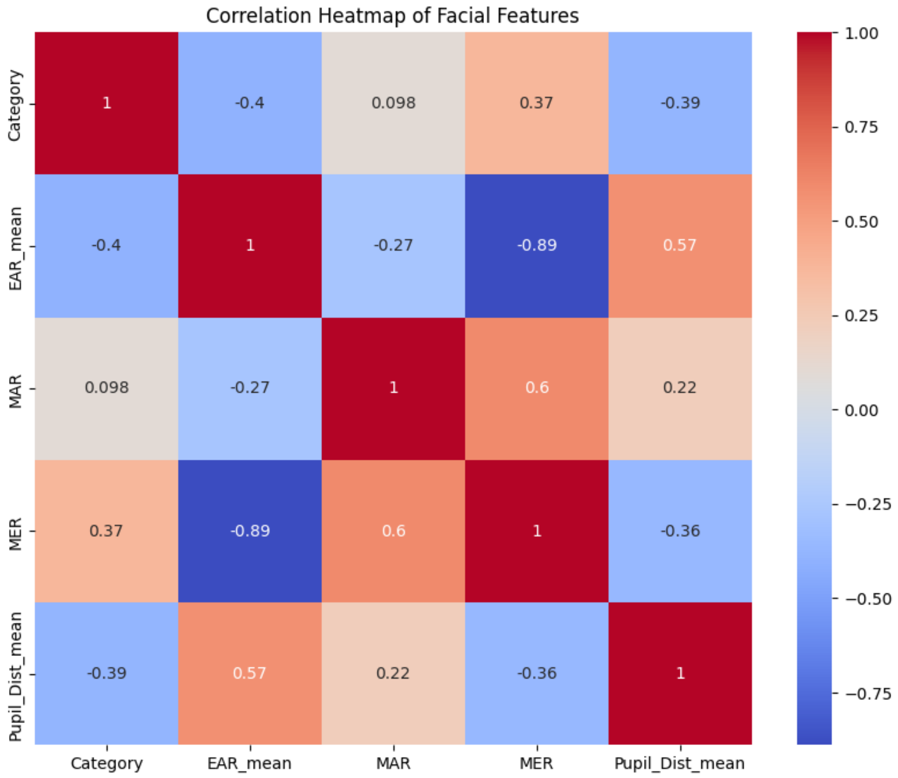


Figure 9 Correlation heatmap of facial features.

Table 3 Summary of insights from pair plot and correlation heatmap.

Aspect	Feature(s)	Observation	Insight
Key features	EAR & PECD	Clear separation between alert and drowsy categories.	Strong indicators for drowsiness detection.
	MAR & MER	Overlapping distributions across categories.	Useful but more variable; may capture yawning and facial relaxation.
Strongest correlations	EAR & MER	-0.89 → strong negative correlation.	As eyes close, mouth engagement tends to increase (i.e. yawning).
	MAR & MER	+0.60 → moderate positive correlation.	Mouth-related features increase together during drowsiness.
	EAR & PECD	+0.57 → moderate positive correlation.	More open eyes often coincide with more focused gaze in alert individuals.
Category relationship	EAR	-0.40 correlation with drowsiness	Eye closure is a strong sign of fatigue.
	PECD	-0.39 correlation with drowsiness	Relaxed or unfocused gaze is common in drowsy individuals.
	MER	+0.37 correlation with drowsiness	Increased mouth-eye ratio during drowsiness.

7.2 Classification and ensemble models

7.2.1 Feature selection

For training the machine learning models, all four extracted features— **EAR**, **MAR**, **MER**, and **PECD** — were used. These features were selected because each contributes uniquely to detecting signs of drowsiness. EAR and PECD showed strong separation between alert and drowsy states, making them reliable indicators of eye-related fatigues. MAR and MER, while more variable, capture mouth movements such as yawning, which are also important behavioural cues of drowsiness. Including all four features allows the model to learn from a more comprehensive set of visual cues, combining both eye and mouth dynamics for improved classification performance.

7.2.2 Train-test split

The dataset was split into training and testing sets using an 80:20 ratio. This resulted in 9,755 samples for training and 2,439 samples for testing. A fixed random state of 42 was used to ensure reproducibility of the split. This approach provides a balanced setup for training the model while reserving sufficient data for unbiased evaluation.

7.2.3 Model selection

To identify the most suitable model for driver drowsiness classification using extracted facial features, four commonly used supervised learning algorithms were selected: **Logistic Regression**, **Support Vector Machine (SVM)**, **Random Forest**, and **XGBoost**. Each of these models offers distinct advantages in terms of interpretability, performance, and ability to handle non-linear relationships. Table 4 below summarises their respective pros and cons in the context of this project, providing justification for their inclusion in the evaluation process.

Table 4 Pros and cons of the selected machine learning models and reasons for their selection.

Model	Pros	Cons	Reason for Selection
Logistic Regression	• Simple and interpretable • Fast to train • Good baseline model	• Assumes linear relationship • May underperform on complex patterns	Provides a strong and interpretable baseline for binary classification.
Support Vector Machine (SVM)	• Effective in high-dimensional spaces • Works well with clear class separation	• Computationally expensive on large datasets • Requires careful tuning of parameters	Suitable for capturing non-linear relationships between features and class labels.
Random Forest	• Handles non-linearity well • Robust to overfitting • Provides feature importance	• Less interpretable • May require more memory and computation	Good balance of performance, generalisation and interpretability.
XGBoost	• High predictive accuracy • Efficient with imbalanced data • Supports regularisation	• More complex and harder to interpret • Sensitive to parameter tuning	Powerful gradient boosting model suitable for fine-grained performance optimisation.

To further enhance the performance of the Random Forest and XGBoost models, hyperparameter tuning was performed using grid search. This process helps identify the optimal combination of parameters that balance bias and variance, hence improving generalisation to unseen data. The key hyperparameters selected for tuning and their respective value ranges are summarised in Table 5.

Table 5 Hyperparameters for tuning using GridSearchCV.

Model	Hyperparameters
Random Forest	• No. of estimators: [50, 75, 100] • Max. depth: [5, 7, 10] • Min. samples split: [5, 10, 20] • Min. samples leaf: [5, 10, 20]
XGBoost	• No. of estimators: [50, 75, 100] • Max. depth: [5, 7, 10] • Learning rate: [0.01, 0.1, 0.2] • Subsample: [0.6, 0.8, 1.0]

7.2.4 Evaluation metrics selection

To evaluate model performance, **accuracy** was chosen as the primary metric since the dataset was relatively balanced following preprocessing. In addition, to ensure the robustness and generalisability of the models, 5-fold cross-validation was performed. This helped reduce the impact of variability in training/testing splits and provided a more reliable estimate of each model's performance.

Beyond accuracy, several other evaluation tools were used to gain deeper insights:

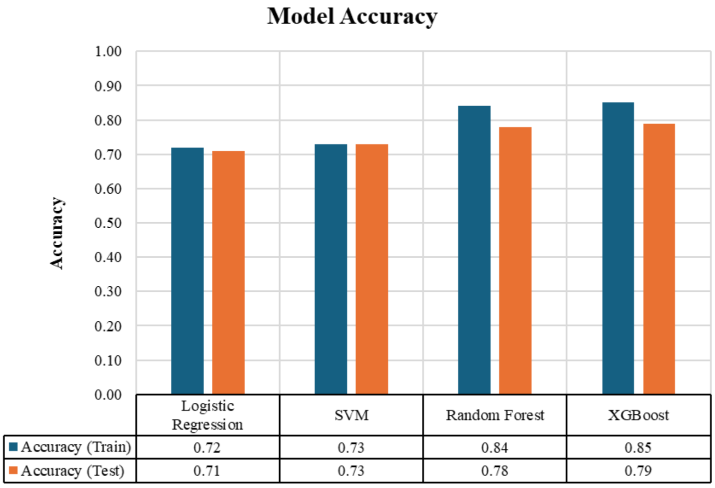
- **Confusion Matrix** to visualise the distribution of true vs. predicted labels.
- **ROC-AUC Curve** to assess the model's ability to distinguish between classes.

- **Precision-Recall Curve** to evaluate performance on positive class detection.

- **Classification Report** to summarise precision, recall, f1-score, and support for each class.

7.2.5 Model performance of baseline model

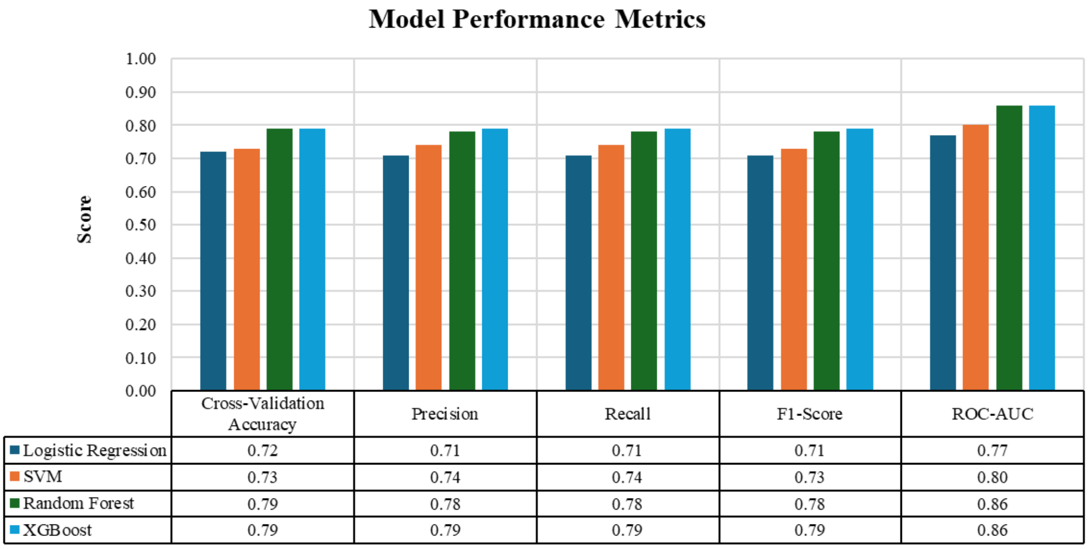
Figure 10 compares training and testing accuracy for all four machine learning models used in the baseline model. Logistic Regression and SVM achieved consistent but modest performance, with accuracies around 0.72 to 0.73, suggesting limited model complexity but stable generalisation. In contrast, Random Forest and XGBoost demonstrated significantly higher training and testing accuracies, reaching up to 0.85 and 0.79 respectively. Although there is a slight overfitting between the training and test sets, the models still generalise well.



	Logistic Regression	SVM	Random Forest	XGBoost
🟦 Accuracy (Train)	0.72	0.73	0.84	0.85
🟧 Accuracy (Test)	0.71	0.73	0.78	0.79

Figure 10 Model accuracy on training and testing sets of the baseline model.

Figure 11 provides a more detailed comparison of the four baseline models across multiple evaluation metrics, including cross-validation accuracy, precision, recall, F1-score, and ROC-AUC. Consistent with the accuracy comparison in Figure 10, Random Forest and XGBoost outperformed Logistic Regression and SVM across all metrics. Both ensemble models achieved higher precision and recall, resulting in stronger F1-scores (0.78–0.79) and ROC-AUC values of 0.86, indicating better overall classification ability. Meanwhile, Logistic Regression and SVM showed more limited performance, with all scores around the 0.71–0.74 range.



	Cross-Validation Accuracy	Precision	Recall	F1-Score	ROC-AUC
■ Logistic Regression	0.72	0.71	0.71	0.71	0.77
■ SVM	0.73	0.74	0.74	0.73	0.80
■ Random Forest	0.79	0.78	0.78	0.78	0.86
■ XGBoost	0.79	0.79	0.79	0.79	0.86

Figure 11 Model performance metrics of the baseline model.

7.2.6 Feature engineering

To improve model performance beyond the baseline, additional features were engineered based on existing research in drowsiness detection and facial behaviour analysis. The goal was to capture more specific and dynamic facial cues related to fatigue that complement the original features. The engineered features in Table 6 were added to the dataset and used in subsequent training to evaluate whether they could improve model accuracy and class separation compared to the baseline setup.

Table 6 Facial features added from feature engineering.

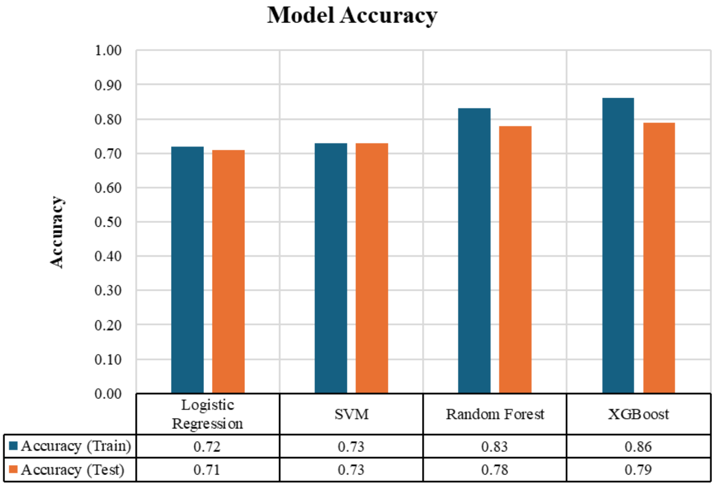
Feature	Description
Mouth Openness	• Reflects the overall degree of mouth opening. • Higher values may correspond to yawning or relaxed jaw position, commonly seen in drowsy states. • $Mouth\ Openness = MAR + MER$
Mouth-Eye Interaction	• Captures the relationship between mouth movement and eye openness, which often changes concurrently during fatigue. • $Mouth - Eye\ Interaction = MAR \times EAR$
Mouth Difference	• Highlights sudden or irregular mouth movements, which may be indicative or yawning or brief lapses in alertness. • $Mouth\ Difference = \|MAR - MER\|$

Figure 12 and Figure 13 illustrate the model performance after incorporating the engineered features. The results show consistent performance improvements for the more complex Random Forest and XGBoost models. Although the overall accuracy and metric scores remained similar to the baseline, the models demonstrated more stable and generalisable behaviour.

- XGBoost maintained the highest performance, with a training accuracy of 0.86 and test accuracy of 0.79, along with strong cross-validation and ROC-AUC scores (0.86).

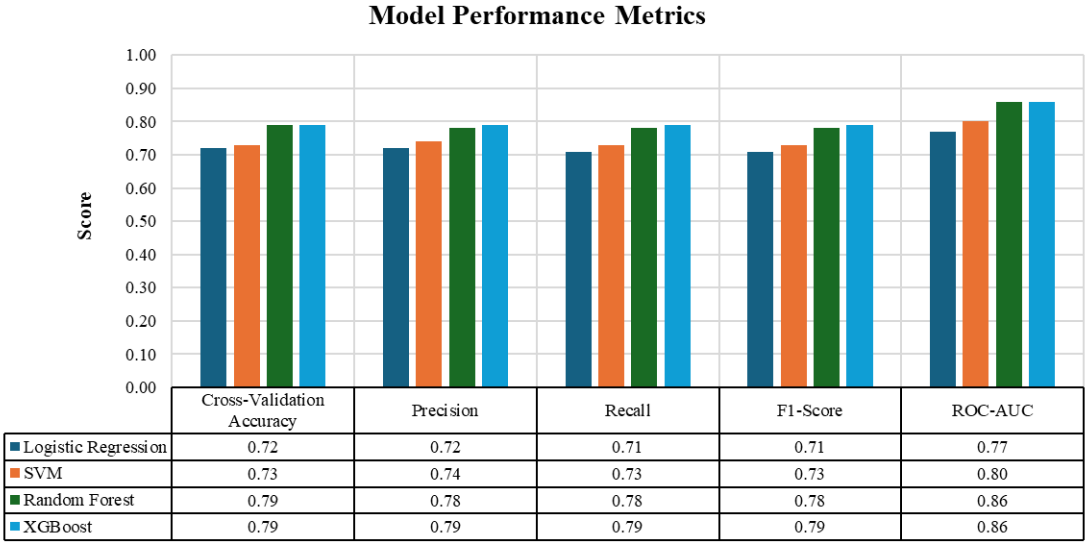
- Random Forest also performed reliably, with slightly reduced training accuracy (0.83) but unchanged test accuracy (0.78), suggesting reduced overfitting after feature engineering.
- Logistic Regression and SVM showed no significant change, with performance metrics remaining around 0.72–0.74.

Overall, the addition of engineered features provided marginal but meaningful gains in performance, particularly for tree-based models. This indicates that the new features added useful information, especially for models capable of capturing complex feature interactions.



	Logistic Regression	SVM	Random Forest	XGBoost
■ Accuracy (Train)	0.72	0.73	0.83	0.86
■ Accuracy (Test)	0.71	0.73	0.78	0.79

Figure 12 Model accuracy on training and testing sets after feature engineering.



	Cross-Validation Accuracy	Precision	Recall	F1-Score	ROC-AUC
■ Logistic Regression	0.72	0.72	0.71	0.71	0.77
■ SVM	0.73	0.74	0.73	0.73	0.80
■ Random Forest	0.79	0.78	0.78	0.78	0.86
■ XGBoost	0.79	0.79	0.79	0.79	0.86

Figure 13 Model performance metrics after feature engineering.

7.2.7 Principal component analysis

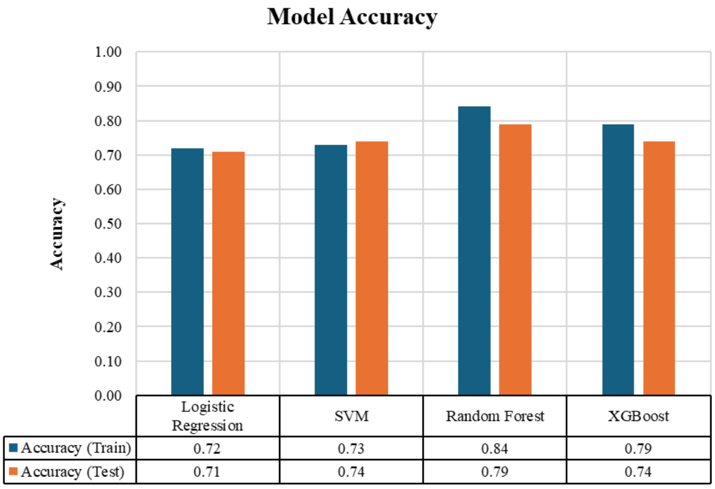
To further improve model performance and reduce the risk of overfitting, Principal Component Analysis (PCA) was applied to the engineered feature set. A sensitivity study was carried out to determine the optimal number of principal components for each model. The goal was to retain the majority of variance while minimising unnecessary complexity. As shown in Table 7, most models used 4 components to retain 100% of the variance, while XGBoost achieved near-identical performance using just 2 components, retaining 99.5% of the variance. This indicates that XGBoost can generalise well even with fewer features, reducing the risk of overfitting.

Table 7 Number of principal components used and cumulative explained variance retained by each model.

Model	No. of Principal Components	Cumulative Explained Variance Retained
Logistic Regression	4	100%
SVM	4	100%
Random Forest	4	100%
XGBoost	2	99.5%

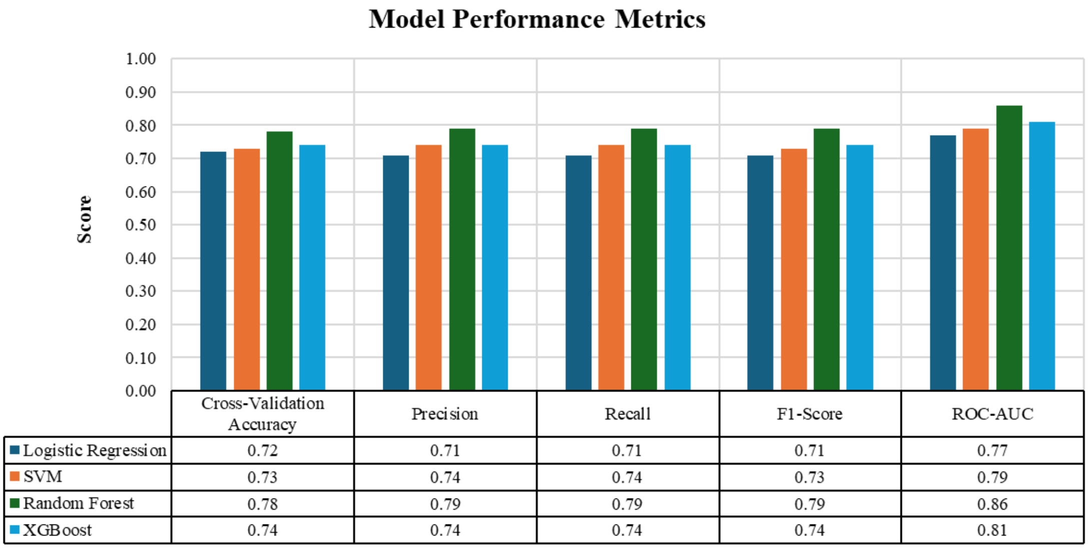
Figure 14 and Figure 15 illustrate the model performance after applying feature engineering and PCA. While PCA helped reduce feature dimensionality and slightly improved generalisation in some cases, neither technique resulted in significant gains in classification accuracy.

Among the models, Random Forest continued to perform well with a test accuracy and F1-score of 0.79, showing no degradation from dimensionality reduction. XGBoost, which used only 2 principal components, experienced a minor drop in test accuracy (from 0.79 to 0.74), suggesting potential loss of fine-grained feature information. SVM showed a small improvement in test accuracy (0.74), indicating that PCA may have benefited linear models, while Logistic Regression maintained consistent performance across all stages.



	Logistic Regression	SVM	Random Forest	XGBoost
■ Accuracy (Train)	0.72	0.73	0.84	0.79
■ Accuracy (Test)	0.71	0.74	0.79	0.74

Figure 14 Model accuracy on training and testing sets after PCA.



	Cross-Validation Accuracy	Precision	Recall	F1-Score	ROC-AUC
■ Logistic Regression	0.72	0.71	0.71	0.71	0.77
■ SVM	0.73	0.74	0.74	0.73	0.79
■ Random Forest	0.78	0.79	0.79	0.79	0.86
■ XGBoost	0.74	0.74	0.74	0.74	0.81

Figure 15 Model performance metrics after PCA.

Overall, while feature engineering and PCA simplified the feature space and introduced some variation in secondary metrics (e.g. precision, recall), they did not significantly improve model accuracy. As accuracy is essential for ensuring the reliability of real-time driver drowsiness detection systems, a more advanced model which is better equipped to model complex spatial and temporal patterns in facial behaviour was explored.

7.3 Convolution Neural Network + Long Short-Term Memory model

To capture both spatial and temporal dynamics of facial behaviour during drowsiness, a **CNN + LSTM hybrid model** was implemented. **Convolutional Neural Networks (CNNs)** are effective at extracting spatial features from individual image frames, such as eye closure and mouth movements. However, drowsiness is a gradual process that unfolds over time, which calls for modelling temporal dependencies. **Long Short-Term Memory (LSTM)** networks are well-suited for this, as they can learn patterns across sequences of frames. By combining CNN and LSTM, the model can effectively learn how facial expressions change over time, improving the accuracy and robustness of drowsiness detection.

7.3.1 Data preparation

Each image was converted to grayscale and resized to 96×96 pixels to reduce dimensionality and ensure uniform input size. The frames were then organised into sequences of 30 consecutive frames, approximately representing one second of visual data. Each sequence was assigned a label based on its drowsiness category (0 for alert, 1 for drowsy).

To support model training and evaluation, two different dataset splitting strategies were applied:

i. **Random Split**: Sequences were randomly divided into 80% training and 20% testing sets. This approach offers quick insight into model performance but may result in subject overlap between sets.

ii. **Subject-Wise Split**: To ensure a more realistic and generalisable evaluation, a subject-independent split was also performed, where all sequences from certain subjects were reserved entirely for testing. This prevents data leakage and better reflects performance on unseen individuals.

7.3.2 Model architecture

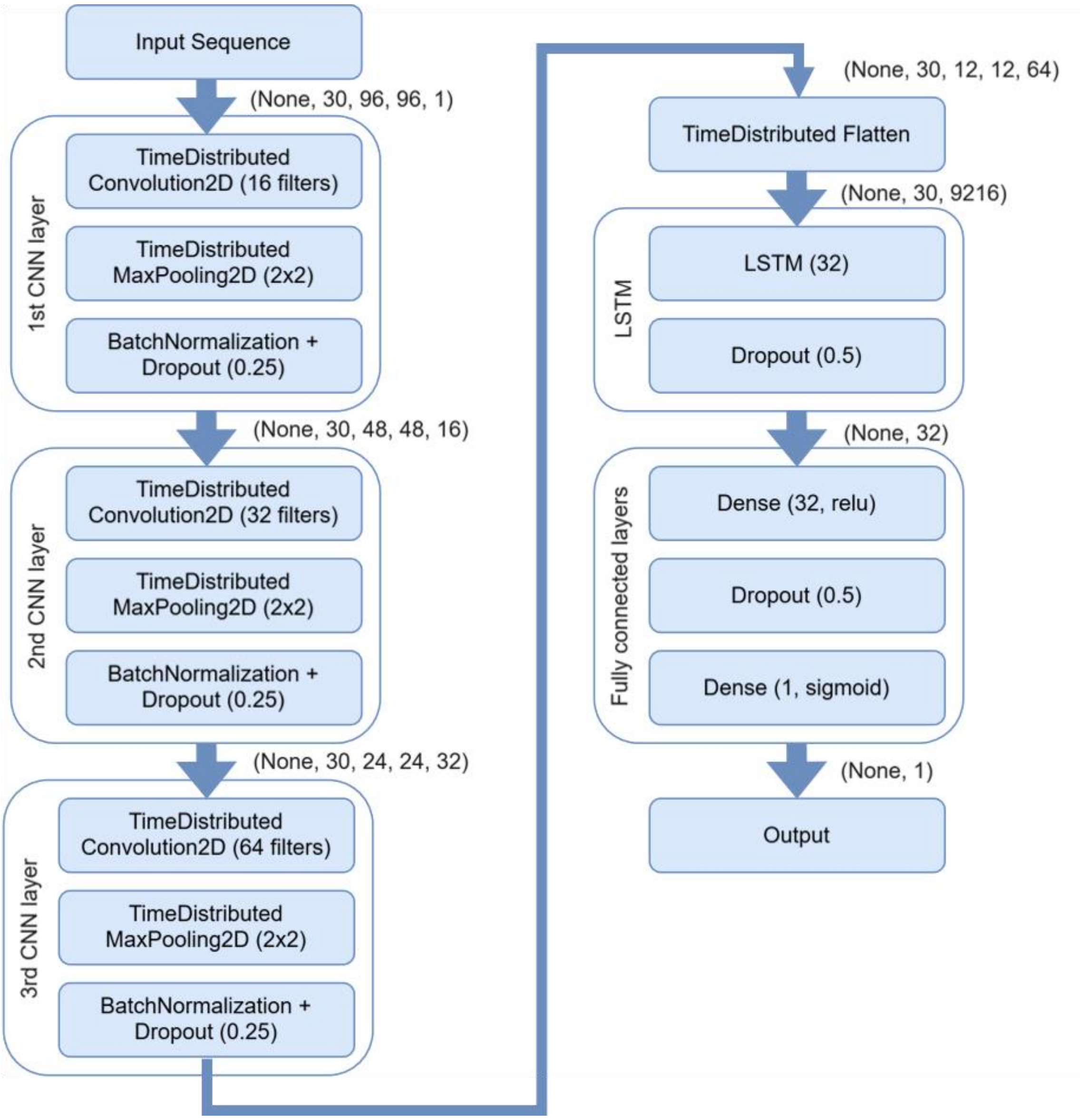

Input Sequence
(None, 30, 96, 96, 1)
1st CNN layer
TimeDistributed Convolution2D (16 filters)
TimeDistributed MaxPooling2D (2x2)
BatchNormalization + Dropout (0.25)
(None, 30, 48, 48, 16)
2nd CNN layer
TimeDistributed Convolution2D (32 filters)
TimeDistributed MaxPooling2D (2x2)
BatchNormalization + Dropout (0.25)
(None, 30, 24, 24, 32)
3rd CNN layer
TimeDistributed Convolution2D (64 filters)
TimeDistributed MaxPooling2D (2x2)
BatchNormalization + Dropout (0.25)
(None, 30, 12, 12, 64)
TimeDistributed Flatten
(None, 30, 9216)
LSTM
LSTM (32)
Dropout (0.5)
(None, 32)
Fully connected layers
Dense (32, relu)
Dropout (0.5)
Dense (1, sigmoid)
(None, 1)
Output

Figure 16 Flow chart of CNN + LSTM model architecture.

Figure 16 illustrates the flow chart of the CNN + LSTM model architecture. It takes an input of 30 grayscale frames, each of size 96×96, and processes them through three *TimeDistributed Conv2D* layers with increasing filter sizes (16, 32, and 64). Each convolutional layer is followed by *MaxPooling2D* to reduce spatial dimensions, *BatchNormalization* to stabilise training and improve convergence, and *Dropout* to reduce overfitting by randomly deactivating neurons during training.

To avoid overfitting and encourage generalisation, *L2 regularization* (weight decay) is applied in all convolutional layers. The use of *BatchNormalization* normalises the outputs of convolutional layers, reducing internal covariate shift and enabling faster and more stable training. *Dropout* is strategically used after each CNN block, the LSTM and dense layers to prevent co-adaptation of neurons.

After spatial features are extracted frame-wise and flattened, they are passed to a single LSTM layer with 32 units to capture the temporal dynamics across the frame sequence. This is followed by a fully connected layer with 32 units and a final sigmoid activation for binary classification (alert vs. drowsy).

The model is compiled with the *Adam optimizer* and *binary cross-entropy loss*, suitable for the binary nature of the problem.

7.3.3 Model performance of random split

The CNN + LSTM model was trained for a maximum of 20 epochs with early stopping based on validation loss to avoid overfitting. Training on the dataset with random split stopped at epoch 14, where the validation loss stabilised and model performance reached optimal levels.

The model loss curves in Figure 17 indicate effective learning and convergence. The training loss decreased from 0.63 to approximately 0.06, while the validation loss dropped to 0.03. Correspondingly, the model achieved high classification accuracy, with training accuracy reaching 98.6% by epoch 12 while validation accuracy achieving 100% at the final early stopping point. The alignment of training and validation curves also demonstrates that the model generalised well, with no signs of overfitting.

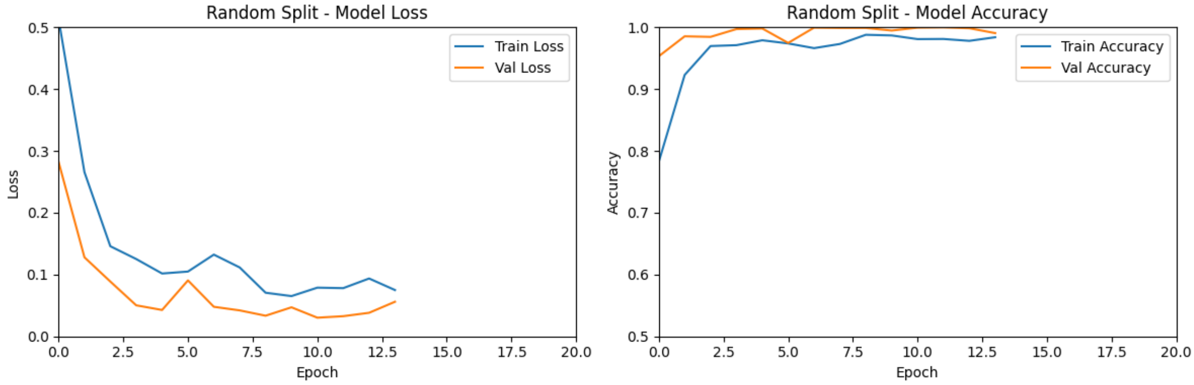


Figure 17 Model loss and accuracy of CNN+LSTM model on the random split sequences.

The confusion matrix in Figure 18 confirms the model's ability to achieve near-perfect classification, with only one false positive. The ROC curve and precision-recall curve further support this, with both the AUC and average precision score reaching 1.0.

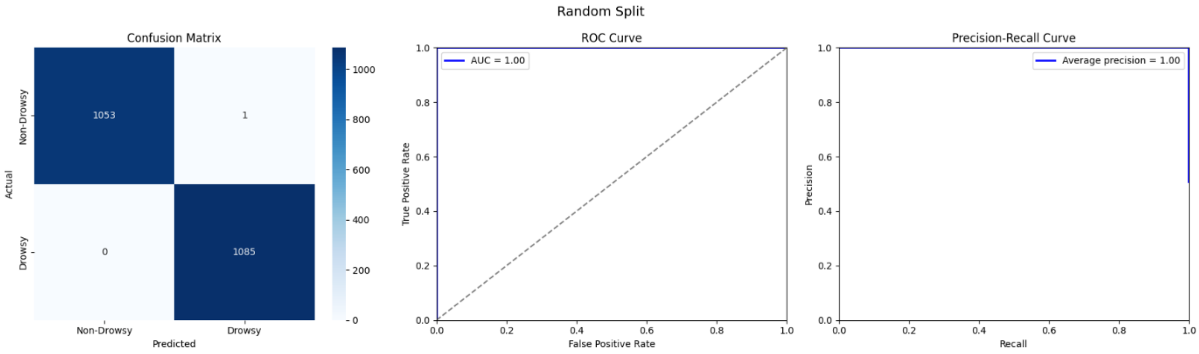


Figure 18 Confusion matrix, ROC-AUC curve and precision-recall curve of CNN+LSTM model on the random split sequences.

7.3.4 Model performance of subject-wise split

While the results in Section 7.3.3 are impressive, such outcomes prompt the question: ***Is this too good to be true?*** A potential issue with random splitting is data leakage across subjects. In the training dataset, multiple sequences may come from the same driver. A random split might allow the model to see similar facial features during training and testing, inadvertently making the task easier. As a result, the model might perform well not because it has learned to detect drowsiness robustly, but because it has memorised individual identities or characteristics.

To validate this concern, a more rigorous evaluation was performed using a subject-wise split, ensuring that individuals in the test set were not seen during training.

Under subject-wise splitting, training stopped at epoch 4, where the validation loss started to increase slightly, suggesting the optimal stopping point (see Figure 19). Training accuracy ranges between 98.7% to 99.3% over 4 epochs, which validation accuracy peaked at 99.1% at epoch 1 before some slight fluctuations in the later epochs. Despite the more challenging setup, the model maintained exceptionally high performance, confirming its ability to generalise across unseen drivers.

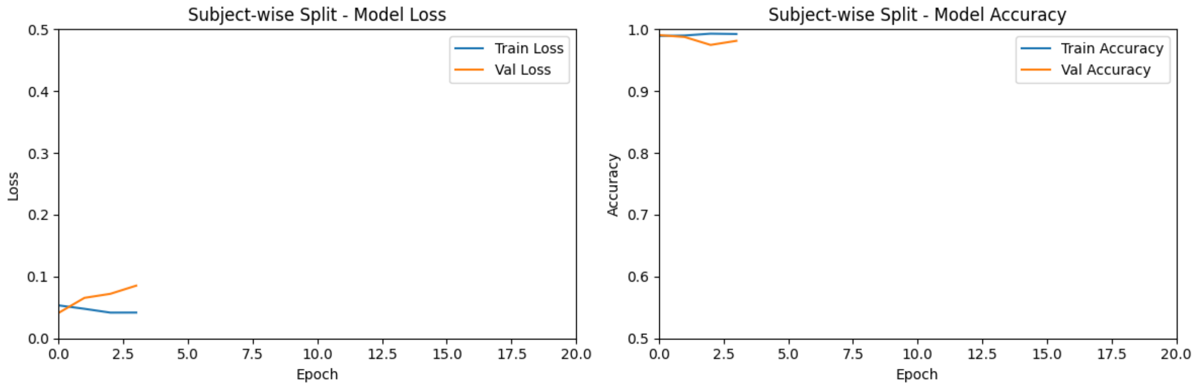


Figure 19 Model loss and accuracy of CNN+LSTM model on the subject-wise split sequences.

From the confusion matrix in Figure 20, although 21 non-drowsy samples were incorrectly classified as drowsy, the model correctly identified all drowsy instances, ensuring zero false negatives — a crucial property in safety-critical applications like drowsiness detection. Area under ROC curve and average precision remained as 1.0, indicating the model's strong discriminative capability.

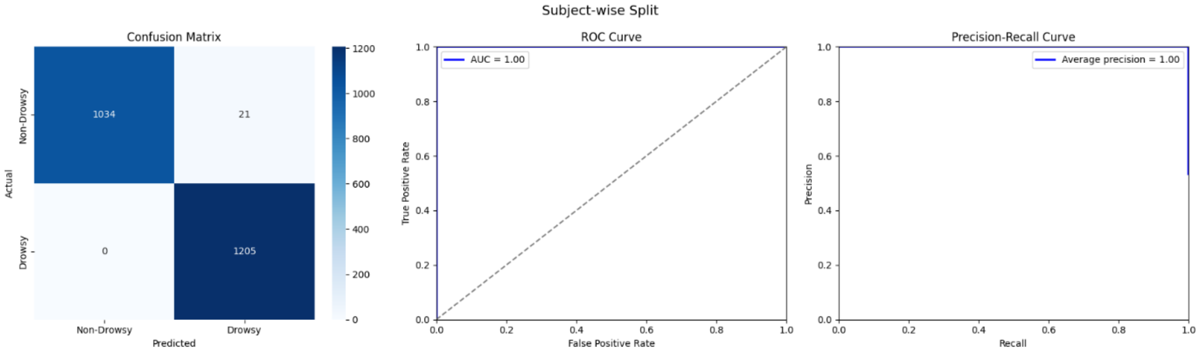


Figure 20 Confusion matrix, ROC-AUC curve and precision-recall curve of CNN+LSTM model on the subject-wise split sequences.

7.4 Outcomes

Both traditional classification models and a deep learning-based CNN + LSTM model were evaluated using the DDD dataset for driver drowsiness detection.

Key findings are summarised as follows: -

1. Baseline classification models, built on handcrafted features such as EAR, MAR, and other facial metrics, achieved reasonable performance and offered interpretability and computational efficiency. However, these models were limited in their ability to capture temporal patterns and were sensitive to variations in lighting, pose, and individual differences.

2. In contrast, the CNN + LSTM model demonstrated significantly superior performance by learning spatial features from raw video frames and temporal dependencies across sequences. Under a random data split, the model achieved:

- Accuracy: **99.95%**
- AUC & Average Precision: **1.00**
- **Near-perfect classification**, with only one misclassification.

3. To validate the model's generalisation capability, a subject-wise split was adopted to prevent data leakage across individuals. The model retained high performance with:

   - Accuracy: **99.07%**
   - AUC & Average Precision: **1.00**
   - **Zero false negatives**, ensuring high reliability in safety-critical applications.

4. These results confirm the robustness and generalisability of the CNN + LSTM approach for real-world driver monitoring scenarios.

7.5 Implementation

As an initial step towards real-world deployment, the best-performing model from the subject-wise split was deployed on Streamlit Cloud for testing. The web-based interface allows users to upload a short 5-second video recorded at 30 frames per second. From the uploaded video, the system extracts multiple 30-frame sequences and computes the average drowsiness probability across those segments to generate a final prediction. In preliminary tests using self-recorded video clips, the model successfully detected drowsiness states, demonstrating its potential for real-time application in real-world conditions.

To improve the model for production, the following enhancements are recommended: -

i. **Ensure real-time inference performance:** Optimise model architecture and inference pipeline to achieve low latency, allowing predictions to be made within milliseconds to support real-time driver monitoring.

ii. **Integrate face detection and alignment:** To ensure consistent input quality regardless of lighting, head pose, or camera placement.

iii. **Optimise for edge deployment:** Apply model compression techniques such as quantization and pruning or convert the model to TensorFlow Lite for faster inference on low-power devices.

iv. **Expand and diversify the dataset:** Include a broader range of driver demographics, lighting conditions, facial accessories (e.g., glasses, masks), and environmental scenarios to improve robustness.

v. **Enhance temporal modelling with multi-modal data:** Combine visual cues with additional signals like blink rate, head tilt, or steering behaviour to improve accuracy in ambiguous cases.

vi. **Implement continuous evaluation and feedback mechanisms:** Enable model performance tracking and updates post-deployment to adapt to real-world data over time.

8. Data answer

The data question was answered satisfactorily using the DDD dataset, which fulfilled the majority of the specified requirements. The dataset provided images organised into clearly labelled subfolders, allowing for reliable differentiation between alert and drowsy states. Temporal sequences, which is essential for modelling transitions in driver alertness, were successfully constructed to support the use of sequence-based models such as CNN + LSTM. Facial landmarks were indirectly utilised through raw image frames, enabling the convolutional layers to automatically learn critical spatial cues, including eye closure, yawning, and head movement. The inclusion of subject IDs and frame-level metadata facilitated robust subject-wise evaluation, effectively mitigating the risk of data leakage. Although the dataset had some limitations in terms of demographic diversity and environmental variation, it remained sufficiently balanced to train a model with strong generalisation capability. This is evidenced by the model's high accuracy, low misclassification rate, and perfect recall of drowsy cases during subject-wise testing. Overall, the confidence level in the data answer is high, as the model consistently demonstrated reliable performance across unseen individuals—an essential requirement for deployment in real-world, safety-critical scenarios.

9. Business answer

The business question was answered satisfactorily through the development of a CNN + LSTM-based driver drowsiness detection system that achieved over 99% accuracy and zero false negatives under subject-wise evaluation. These results exceed the targeted 90% accuracy threshold and demonstrate strong potential for real-world deployment. By reliably identifying drowsy states in real time and prioritising the reduction of false negatives, the system aligns with the primary business objective of enhancing road safety in long-distance commercial bus operations. Given the model’s high accuracy, generalisation to unseen drivers, and safety-focused design, the confidence level in the business answer is high. This suggests a strong likelihood that the system can contribute to reducing fatigue-related incidents, resulting in significant cost savings and supporting broader goals such as improving passenger safety, maintaining regulatory compliance, and enhancing the operator's brand reputation.

10. Response to stakeholders

This project directly addresses the concerns of the primary stakeholder by delivering a robust, real-time drowsiness detection system. The solution meets the operational need to monitor driver alertness continuously and issue timely alerts when fatigue is detected, thereby enhancing passenger safety and reducing accident risk. With its high accuracy and zero false negatives under subject-wise evaluation, the system offers strong assurance for real-world deployment. It supports the company's compliance with safety regulations and operational standards, while also laying the groundwork for broader adoption across the long-haul freight sector, where similar fatigue-related risks are present. The system's scalability ensures it can be integrated across various fleet types, delivering both safety and financial benefits.

11. End-to-end solution

The proposed end-to-end solution integrates the trained CNN + LSTM model into a robust, real-time driver monitoring system tailored for long-distance commercial bus operations.

1. **Camera System & Video Capture**
   - A dashboard-mounted camera captures 5-second video clips at 1-minute intervals.
   - This prevents continuous recording, reducing data redundancy, storage load, and the risk of model degradation from repetitive inputs.
2. **On-Device (Edge) Inference**
   - The trained CNN + LSTM model runs on an edge device, enabling real-time predictions without requiring internet connectivity.
   - This ensures low latency, high responsiveness, and strong data privacy.
3. **Drowsiness Alerts**
   - When drowsiness is detected, a built-in buzzer provides an immediate audio alert to the driver.
   - Alerts prioritise safety by minimising false negatives, even at the cost of slightly more false positives.
4. **Low-Light Condition Handling**
   - The system uses infrared (IR) cameras or low-light enhancement to ensure reliable face detection at night or in dim environments, without distracting the driver.

5. **Wearable Device Integration**
   - The system can pair with wearable devices (e.g. smartwatches) to collect physiological signals like heart rate variability for enhanced prediction accuracy and provide additional alerts via vibration as a secondary warning channel.
6. **Mobile App for Updates**
   - A companion mobile app supports over-the-air (OTA) model updates, allowing easy deployment of improvements without physical access to the system.
7. **Scalable and Privacy-Conscious Design**
   - Built to be scalable across fleets, compliant with privacy regulations, and adaptable for use in freight transport and logistics sectors.

References

[1] AsiaOne (2019). https://www.asiaone.com/malaysia/malaysian-driver-slammed-not-honking-fellow-driver-who-fell-asleep-wheel (Accessed: 25 March 2025).

[2] Driver drowsiness detection (no date). https://www.bosch-mobility.com/en/solutions/assistance-systems/driver-drowsiness-detection/ (Accessed: 25 March 2025).

[3] Truckinginfo (2024) NetraDyne introduces new Drowsy Driver Detection System. https://www.truckinginfo.com/10231311/netradyne-introduces-new-drowsy-driver-detection-system (Accessed: 25 March 2025).

[4] Driver Drowsiness Dataset (DDD) (2022). https://www.kaggle.com/datasets/ismailnasri20/driver-drowsiness-dataset-ddd (Accessed: 25 March 2025).

[5] Ghoddoosian, R., Galib, M. and Athitsos, V. (2019) 'A realistic dataset and baseline temporal model for early drowsiness detection,' 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 178–187. https://doi.org/10.1109/cvprw.2019.00027.

[6] Nasri, I. et al. (2021) 'Detection and prediction of driver drowsiness for the prevention of road accidents using deep neural networks techniques,' in Lecture notes in electrical engineering, pp. 57–64. https://doi.org/10.1007/978-981-33-6893-4_6 .

Appendix A1

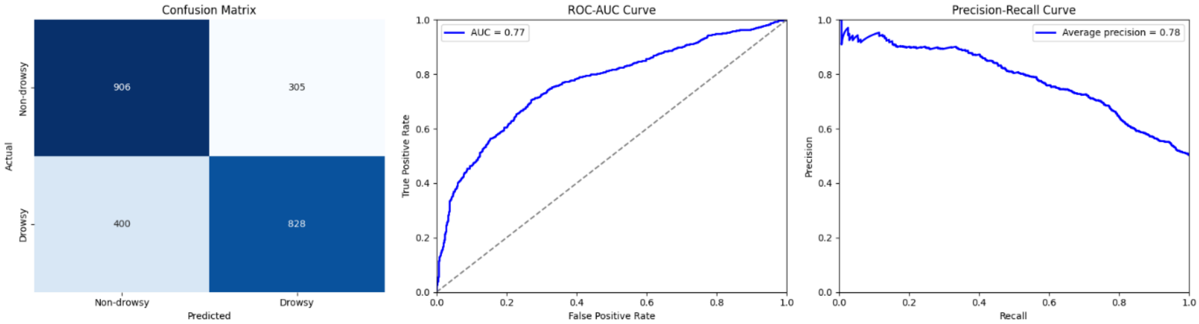


Figure A1-1 Confusion matrix, ROC-AUC curve and precision-recall curve of the Logistic Regression baseline model.

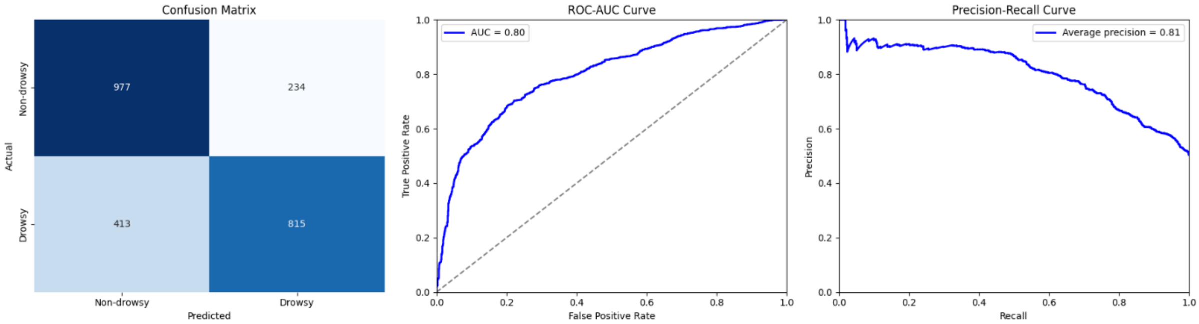


Figure A1-2 Confusion matrix, ROC-AUC curve and precision-recall curve of the SVM baseline model.

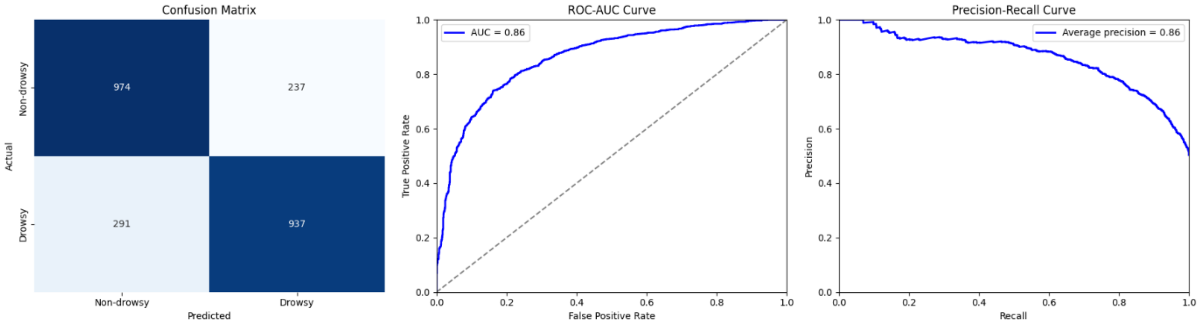


Figure A1-3 Confusion matrix, ROC-AUC curve and precision-recall curve of the Random Forest baseline model.

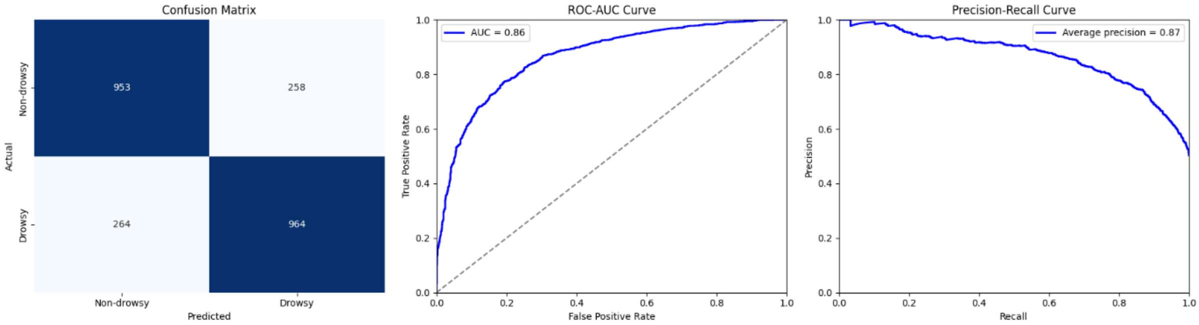


Figure A1-4 Confusion matrix, ROC-AUC curve and precision-recall curve of the XGBoost baseline model.

Appendix A2

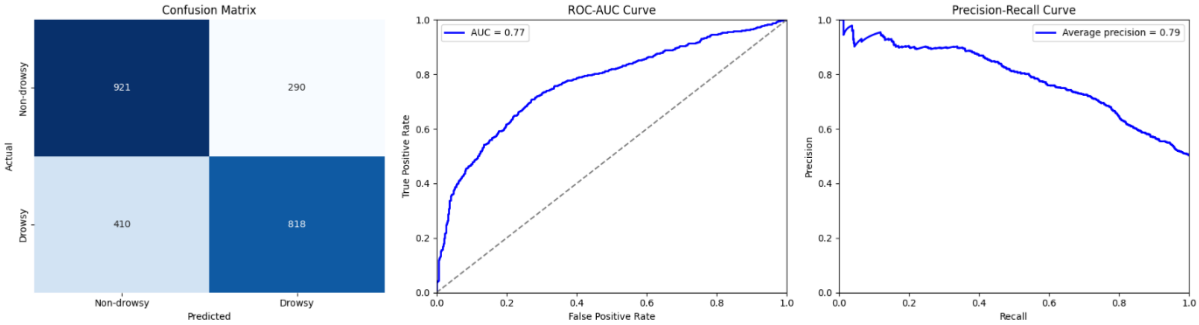


Figure A2-1 Confusion matrix, ROC-AUC curve and precision-recall curve of the Logistic Regression after feature engineering.

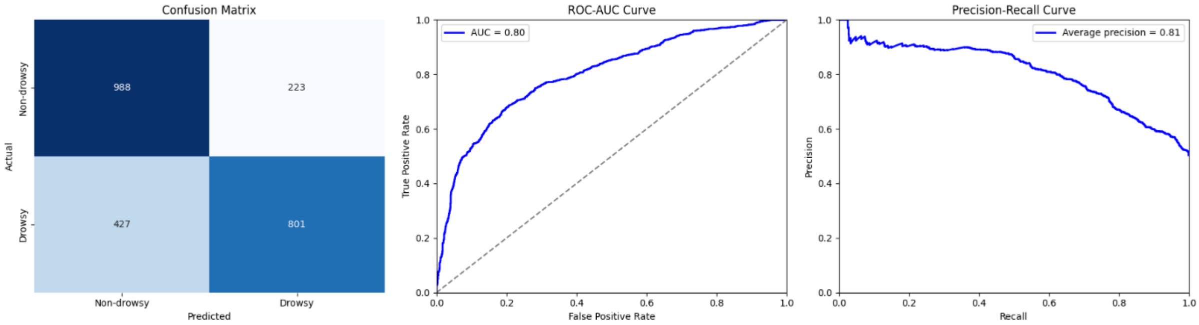


Figure A2-2 Confusion matrix, ROC-AUC curve and precision-recall curve of the SVM baseline model after feature engineering.

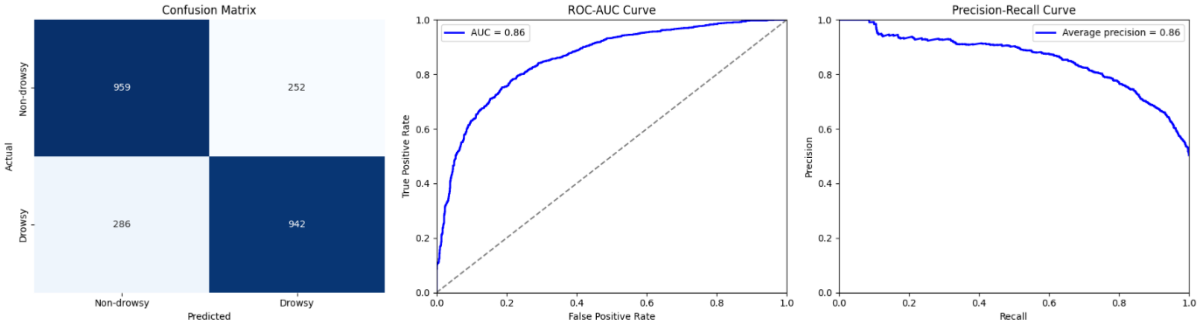


Figure A2-3 Confusion matrix, ROC-AUC curve and precision-recall curve of the Random Forest baseline model after feature engineering.

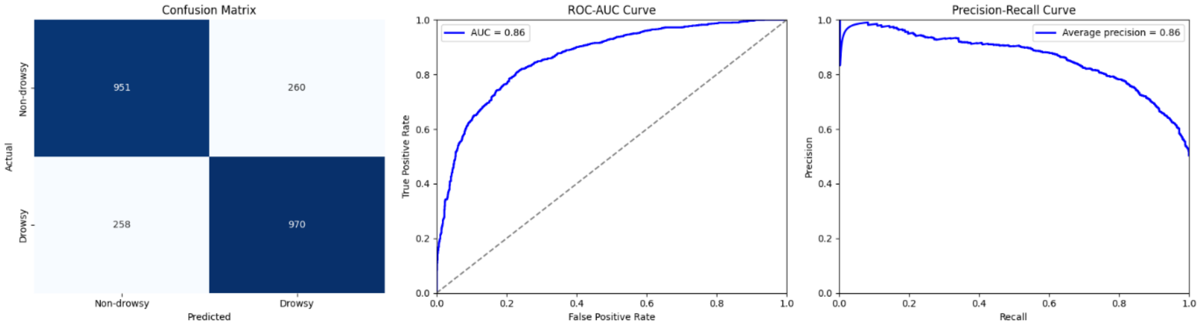


Figure A2-4 Confusion matrix, ROC-AUC curve and precision-recall curve of the XGBoost baseline model after feature engineering.

Appendix A3

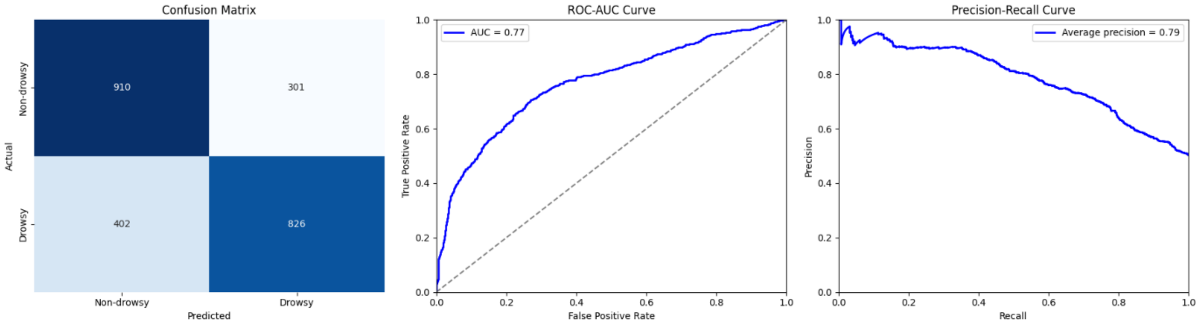


Figure A3-1 Confusion matrix, ROC-AUC curve and precision-recall curve of the Logistic Regression after PCA.

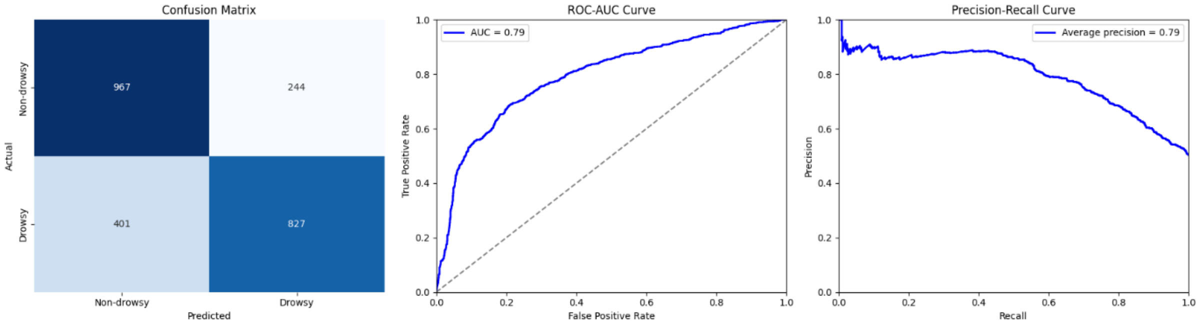


Figure A3-2 Confusion matrix, ROC-AUC curve and precision-recall curve of the SVM baseline model after PCA.

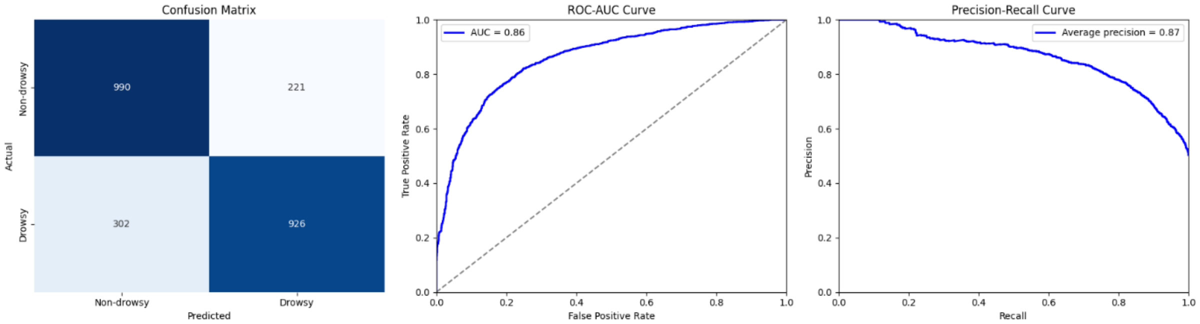


Figure A3-3 Confusion matrix, ROC-AUC curve and precision-recall curve of the Random Forest baseline model after PCA.

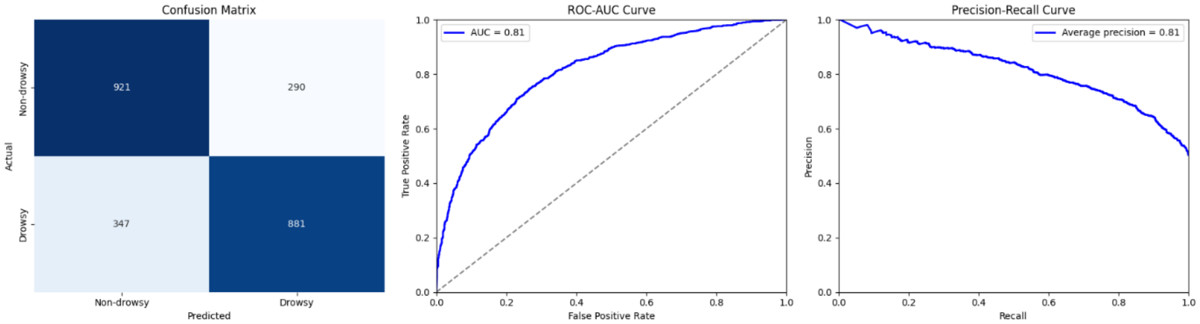


Figure A3-4 Confusion matrix, ROC-AUC curve and precision-recall curve of the XGBoost baseline model after PCA.