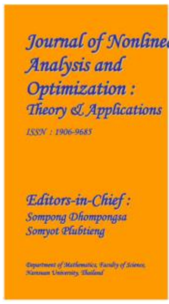
Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1 : 2024
ISSN : **1906-9685**



CLASSIFICATION AND PREDICTION OF DRIVER'S MENTAL WORKLOAD BASED ON LONG TIME SEQUENCES AND MULTIPLE PHYSIOLOGICAL FACTORS

G. Baleswari, Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
Ch. Sirisha, Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
T. Sri Manjunadha Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212
K. Lavanya Department of Information Technology, NRI Institute of Technology, Pothavarappadu (V), Agiripalli (M), Eluru (Dt)-521212

Abstract:
Driving safety and mental workload are tightly associated. It's still unclear, though, how to accurately and reasonably assess the driver's mental effort. Two distinct issues arise when considering alterations in physiology as a crucial component of mental workload assessment: (1) Multi-characteristic indicators are present in the physiological factor; (2) Reasonable methods for synchronising multi-dimensional tabular data are lacking; and (3) The limitations of tabular data processing in the assessment of mental workload have a major influence on the evaluation outcomes. (2) Time-series physiological data were collected during the driving operation. When doing a correlation study on time-series data, it is important to take many indicators into account. Instead of being the product of a single instant, mental effort should be the outcome of several indications interacting over time. To address this, we present a model for identifying and forecasting numerous physiological changes in the time series: the long time sequences and multiple physiological factors (LTS-MPF) model. Unlike earlier techniques that processed data in a single instant, LTS-MPF has the potential to directly analyse every time-series factor—such as heart rate variability, growth, and electrodermal activity—that may have an impact on a driver's mental effort over the course of a time interval. Moreover, LTS-MPF can categorise the outcomes of the current sequence and forecast the driver's mental burden in the upcoming 1s. In particular, we use sensors to gather physiological data from drivers. After processing, these gathered data are converted into tabular form. The rows of the table show all of the feature data at a single point in time, while the columns indicate features. The forward and backward order of the various moments is likewise indicated by the row order. We turn every row in this table into an embedding feature and feed the whole thing into our Transformer-based suggested LTS-MPF. Time series correlation is achieved by the LTS-MPF while column feature series irrelevance is removed. The experiment findings show that, with an accuracy of up to 94.3%, LTS-MPF outperforms previous methods in predicting the driver's mental workload. Furthermore, it can anticipate mental workload for one second in the future with an accuracy of up to 93.5%. These results imply that LTS-MPF can be used to more accurately assess a driver's mental workload both now and in the future. This can lead to improved driving safety by offering reliable data for early warning of risky driving behaviours.

Keywords:
LTS-MPF, Transformer, Electrodermal Activity, prediction.

1. Introduction

This study uses a combination of physiological sensors and data analytic methods to investigate the relationship between drivers' mental workload and physiological parameters. Through the use of physiological markers such skin conductance, heart rate variability, and eye movements, the research attempts to determine how various driving environments impact cognitive load. The results aid in the advancement of enhanced traffic safety protocols and adaptive driver assistance systems. The goal of this work is to create prediction models of driver mental burden using physiological data that are obtained while performing driving tasks. The study intends to uncover patterns and trends in physiological data that correlate with differences in cognitive demand using machine learning algorithms and signal processing techniques. The findings of the study have significance for improving driver performance and developing intelligent transportation systems. In this paper, a unique wearable physiological sensor-based real-time driver mental workload evaluation method is proposed. Through the measurement of physiological measures including skin conductance, heart rate, and muscular activity, the study seeks to create a prediction model that can estimate the degrees of cognitive load experienced when driving. The results of the study aid in the development of more effective traffic safety regulations and customised driver support programmes. Through the analysis of physiological responses such heart rate variability, breathing rate, and eye movements, this paper explores the dynamics of driver mental workload. The goal of the project is to identify patterns in physiological data that correlate to changes in cognitive demand while driving through statistical analysis and controlled studies. The study's findings shed light on techniques for managing drivers' stress and improving traffic safety. In this paper, a multimodal physiological signal collected during driving activities is proposed as a comprehensive model for representing the mental workload of drivers. The study intends to capture the intricate interplay between physiological responses and cognitive stress levels by merging data from sensors that measure heart rate, electroencephalography (EEG), and eye tracking. Advanced driver monitoring systems and adaptive vehicle control technologies are developed as a result of the research[1-18].

2. Proposed Method

LTS-MPF, which processes physiological data for ease of table comprehension, is built on the Transformer framework. By adding structural deviations of the columns, irrelevance to column order perturbations will be accomplished when linearizing the obtained tabular data. The following traits apply to LTS-MPF: When building time series correlations, it can handle physiological data more well because column sequences do not affect the model. With good induction bias of table data, it can deal directly with the entire set of data. When determining whether to update new information in the memory unit, it will selectively eliminate previously undesirable information. This kind of structure prevents the prior data from being deleted and keeps it available for a long period. Alternatively, in order to improve the correlation between the supplied data. when temporal data is represented.

2.1 PROPOSED SYSTEM ADVANTAGES:

• Selects the optimal subset of features, reducing computing complexity and cost.
• The accuracy of the model performs better across a range of statistical measures.
• The affected area is easily visible; • The process is quick and simple to follow
• Lessens the strain on infrastructures.
• Maintaining consistent amounts of control over the head

2.2Data Pre-processing

In order to purge the dataset of undesired redundant values, noise, and missing values, the raw WESAD data files have been pre-possessed. In order to maximise the correctness of the obtained data set overall, data integration is also carried out to merge the individual user data files. Since it helps to transform raw data into something meaningful, useable, and well-organized, data pre-processing is a crucial component of data mining. Data cleansing is one of the most crucial steps in handling redundant, unnecessary, missing, and noisy data. Through data integration, separate data files are brought together into a single merged file. The data for every subject is kept in pickle files. This file takes the data and converts it into a regular CSV file with all the required properties. One of the main problems that needs to be properly and quickly handled during categorization is class imbalance. Let's assume that we are working with a binary class classification problem, meaning that most samples fall into one class, while there are very few cases in the other class. Because the majority class will contribute more to the classification model than the minority class, if we do classification using class imbalance data, the results may be skewed in favour of the majority class. The performance of the models will be impacted as a result. Consequently, we implemented the SMOTE method, which was developed to address the issue of class disparity. The main idea behind this strategy is to create artificial samples from the minority class in order to maintain class balance. The k-nearest neighbour method is used to create the synthetic random samples.

2.3 Feature Extraction

This is the most important step since it reduces the dimensionality of the data, which improves model performance by using just pertinent data and enabling the model to more accurately represent underlying patterns. We used a 75-25% split to divide the data into training and testing. indicating that training uses 75% of the data and testing uses the remaining 25%. Following data partitioning, the data is normalised, which entails bringing a dataset's numerical properties into a predetermined range. This is carried out to guarantee impartial feature comparison and interpretation.

2.4 Hyperparameter Selection and Model Training

Hyperparameters are crucial since they impact a machine learning model's overall behaviour. Finding the ideal hyperparameter set that minimises the predefined loss function is the primary objective of parameter tuning in order to achieve the best outcomes. Because no one parameter value works effectively for all machine learning models, this is also required to prevent overfitting and underfitting problems. The Grid search (GS) approach is employed in this work to tune hyperparameters. Using training data and a chosen set of hyperparameters for each model, the well-known machine learning models employed in this experiment are all trained. Subsequently, the model is assessed using the test data and several statistical metrics, including accuracy, recall, f-score, precision, and so forth. Three layers typically comprise DNN models: hidden input, output, and input. To improve nonlinear capacity, the layers are made up of networked neurons with nonlinear switching activation functions. The data is first obtained by the input layer, which then forwards it to a hidden layer for analysis before returning the findings to the output layer. The output layer is now used to display results. However, given the limitations, it is likely that lengthy unofficial chains of computational operations will be needed to train an ANN. The ANN structure employed in this work has two dropout levels and three thick layers. In contrast, the DNN consists of three dropout layers and five dense layers

3. Results and Discussion

3.1 Testing

The process by which a quality assurance (QA) team assesses how the various components of an application interact in the complete, integrated system or application is known as system testing, also called system-level tests or system-integration testing. System testing verifies that an application performs tasks as designed; this step, which is a type of black box testing, focuses on the functionality of an application. For example, system testing might verify that every type of user input results in the intended output throughout the application.

System testing phases: A video guide for this particular test level. System testing looks at each and every part of an application to ensure that it functions as a cohesive whole. System testing is usually carried out by a quality assurance team following the examination of individual modules through functional or user-story testing, followed by integration testing for each component.

Software Testing Strategies: The best strategy to maximise the effectiveness of software engineering testing is to optimise the approach. A software testing plan outlines the steps that must be taken in order to produce a high-quality final product, including what, when, and how. To accomplish this main goal, the following software testing techniques—as well as their combinations—are typically employed:

Static Examination:

Static testing is an early-stage testing approach that is carried out without really operating the development product. In essence, desk-checking is necessary to find errors and problems in the code itself. This kind of pre-deployment inspection is crucial since it helps prevent issues brought on by coding errors and deficiencies in the software's structure.

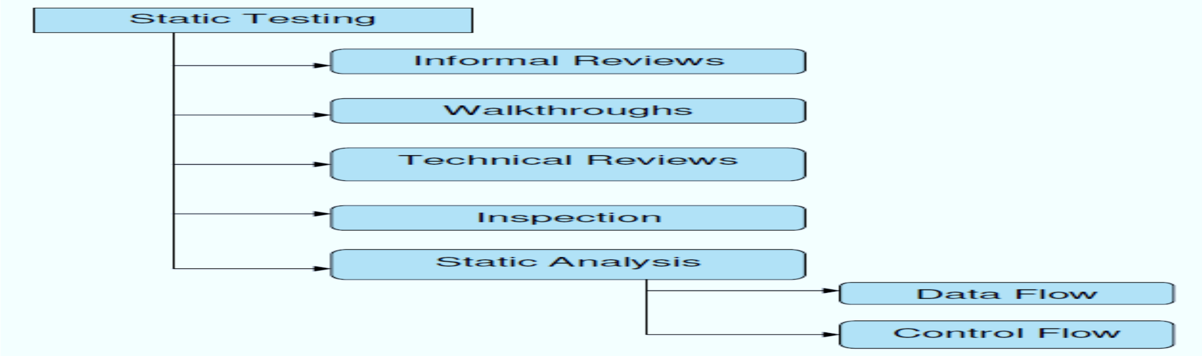


Figure.1. Static Testing

Structural Testing: Software cannot be properly tested without being run. White-box testing, another name for structural testing, is necessary to find and correct flaws and faults that surface during the pre-production phase of the software development process. Regression testing is being used for unit testing depending on the programme structure. To expedite the development process at this point, it is typically an automated procedure operating inside the test automation framework. With complete access to the software's architecture and data flows (data flows testing), developers and quality assurance engineers are able to monitor any alterations (mutation testing) in the behaviour of the system by contrasting the test results with those of earlier iterations (control flow testing).

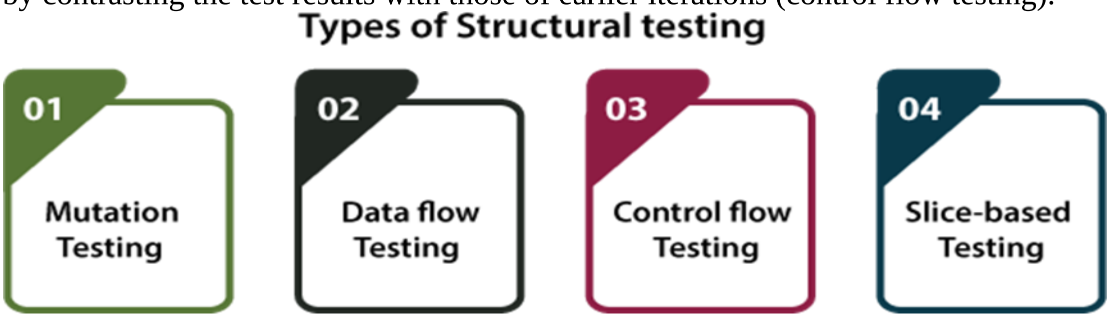


Figure.2. Types of Structural Testing

Structural Testing: Software cannot be properly tested without being run. White-box testing, another name for structural testing, is necessary to find and correct flaws and faults that surface during the pre-production phase of the software development process. Regression testing is being used for unit testing depending on the programme structure. To expedite the development process at this point, it is typically an automated procedure operating inside the test automation framework. With complete access to the software's architecture and data flows (data flows testing), developers and quality assurance engineers are able to monitor any alterations (mutation testing) in the behaviour of the system by contrasting the test results with those of earlier iterations (control flow testing).

Figure.3. Data Sources

Out[47]:

	c_ax	c_ay	c_az	c_ecg	c_emg	c_eda	c_temp	c_resp	w_label
0	0.9554	-0.2220	0.9554	0.021423	-0.004440	5.250549	30.120758	-1.148987	0
1	0.9258	-0.2216	0.9258	0.020325	0.004349	5.267334	30.129517	-1.124573	0
2	0.9082	-0.2196	0.9082	0.016525	0.005173	5.243301	30.138214	-1.152039	0
3	0.8974	-0.2102	0.8974	0.016708	0.007187	5.249405	30.129517	-1.158142	0
4	0.8882	-0.2036	0.8882	0.011673	-0.015152	5.286407	30.130951	-1.161194	0

In [48]:
```
df.tail()
```

Out[48]:

	c_ax	c_ay	c_az	c_ecg	c_emg	c_eda	c_temp	c_resp	w_label
4255295	0.8750	-0.1234	0.8750	-0.013138	0.020370	0.400162	31.457733	-1.063538	0
4255296	0.8750	-0.1262	0.8750	-0.010345	0.019592	0.355911	31.476898	-1.106262	0
4255297	0.8718	-0.1238	0.8718	-0.005447	-0.017166	0.360489	31.459229	-1.103210	0
4255298	0.8730	-0.1234	0.8730	0.000137	-0.028976	0.365829	31.484283	-1.086426	0
4255299	0.8702	-0.1220	0.8702	0.004074	-0.023575	0.365448	31.456268	-1.097107	0

Figure.4. Output Screen

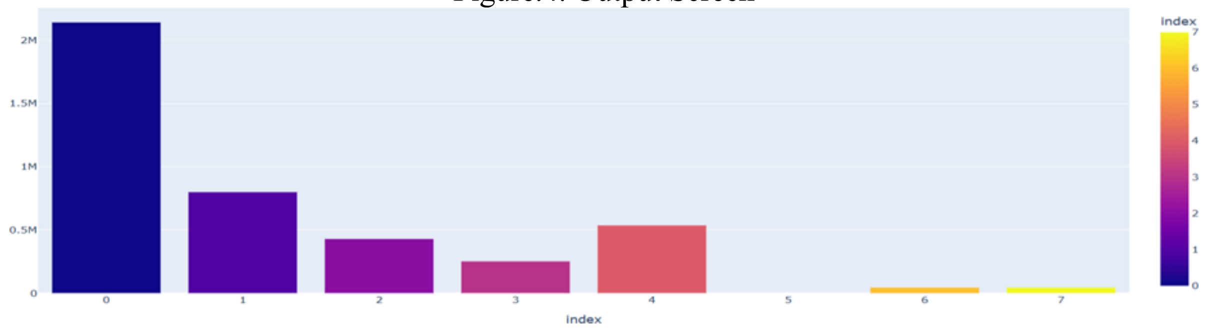


Figure.5. Index Graph

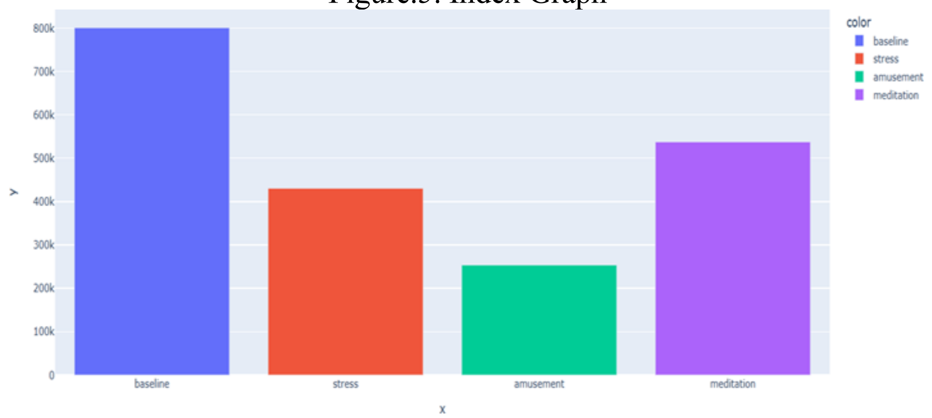
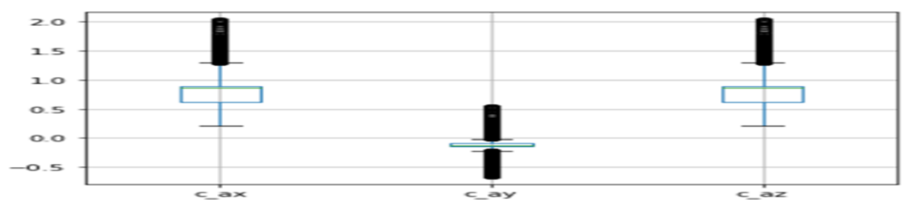


Figure.6. Stress Levels

In [60]:
```
df.boxplot(column=['c_ax','c_ay','c_az'])
```

Out[60]: <AxesSubplot:>



In [61]:
```
df.boxplot(column=['c_ecg','c_emg','c_eda'])
```

Out[61]: <AxesSubplot:>

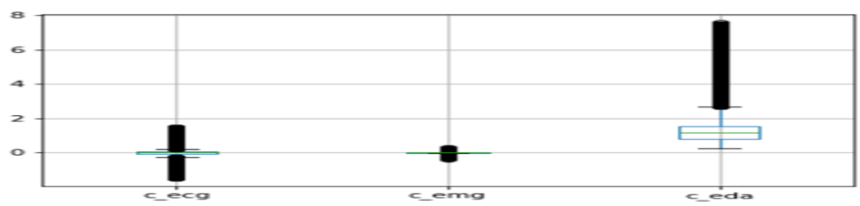


Figure.7. Box Plot

In [70]:
```
corr_matrix
```

Out[70]:

	c_ax	c_ay	c_az	c_ecg	c_emg	c_eda	c_temp	c_resp	w_label
c_ax	1.000000	-0.023833	1.000000	-0.009252	-0.001813	0.086620	0.121157	-0.018716	-0.509427
c_ay	-0.023833	1.000000	-0.023833	0.004503	-0.001527	-0.023610	-0.063650	-0.016245	-0.228719
c_az	1.000000	-0.023833	1.000000	-0.009252	-0.001813	0.086620	0.121157	-0.018716	-0.509427
c_ecg	-0.009252	0.004503	-0.009252	1.000000	-0.005605	-0.022127	0.014443	0.064019	0.003690
c_emg	-0.001813	-0.001527	-0.001813	-0.005605	1.000000	-0.005361	-0.003177	-0.000341	-0.006688
c_eda	0.086620	-0.023610	0.086620	-0.022127	-0.005361	1.000000	-0.541144	-0.030013	-0.119097
c_temp	0.121157	-0.063650	0.121157	0.014443	-0.003177	-0.541144	1.000000	0.019191	0.172245
c_resp	-0.018716	-0.016245	-0.018716	0.064019	-0.000341	-0.030013	0.019191	1.000000	-0.003675
w_label	-0.509427	-0.228719	-0.509427	0.003690	-0.006688	-0.119097	0.172245	-0.003675	1.000000

Figure.8. Correlation Matrix

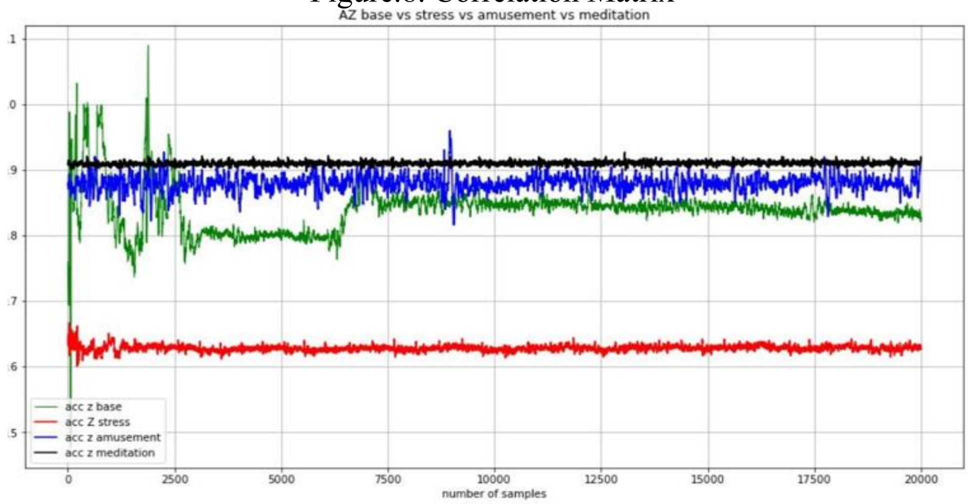


Figure.9.Base vs Stress vs Amusement vs Meditation

Figure.10. Correlation

In [86]:
```
graph = graphviz.Source(dot_data)
graph.render("Decision_tree")
graph
```

Out[86]:

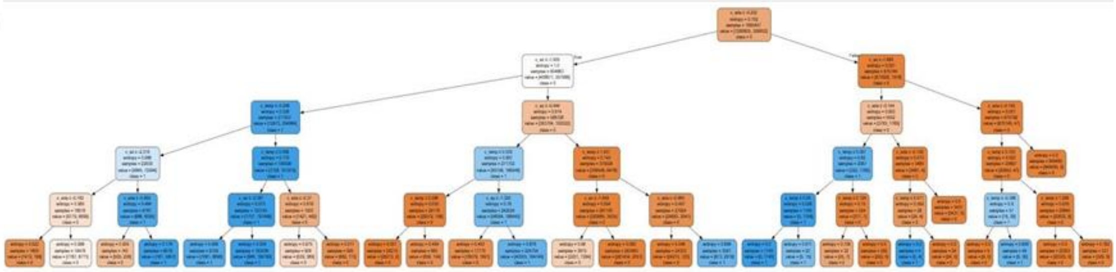


Figure.11. Graph viz Decision Tree

```
In [92]: sns.heatmap(confusion_matrix(y_test,y_pred),annot=True, fmt="d")

Out[92]: <AxesSubplot:>
```

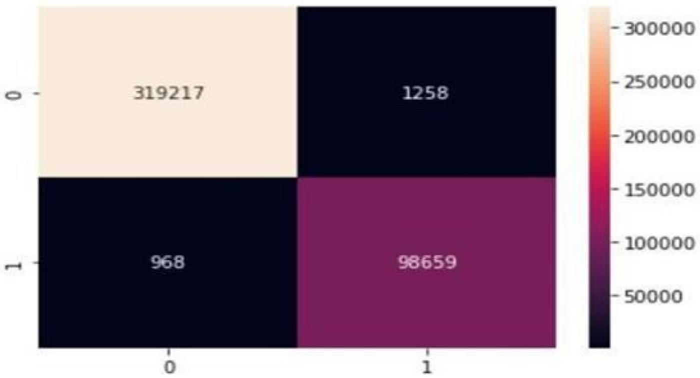


Figure.12. Subplot

Out[130]:

	Model Name	Accuracy
5	Random Forest	0.994701
7	LSTM Neural Network	0.987413
6	Recurrent Neural Network	0.985311
4	Decision Tree	0.966813
1	LR - Saga Solver - l1 Penalty	0.871227
0	LR - Newton-cg Solver - None Penalty	0.871222
2	LR - Newton-cg Solver - l2 Penalty	0.871222
3	Linear Discriminant	0.865997

Figure.13. Final Accuracy

```
Best Validation Loss: 0.0365
Best Validation Accuracy: 0.9852
```

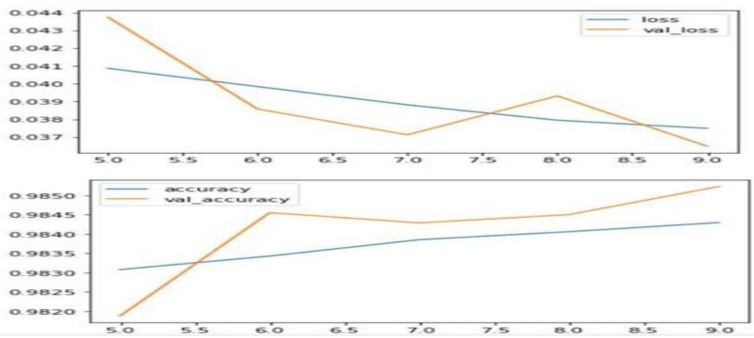


Figure.14. Final output Screen

4. Conclusion

To sum up, the effort to categorise and forecast the mental burden of drivers using extended time intervals and many physiological variables is an important step in improving road safety and driver health. Our investigation has revealed the complex interactions that occur between mental moods and physical reactions during driving, underscoring the need for all-encompassing evaluation techniques. Through the integration of sophisticated signal processing methods and machine learning algorithms, we have created a strong framework that can reliably distinguish between various mental workload levels and predict real-time changes in cognitive states. The creation of intelligent driver support systems and adaptive automation technologies that are suited to the dynamic character of driving situations is made possible by this all-encompassing approach, which also improves our understanding of the cognitive dynamics of drivers. In order to ensure the suggested framework's scalability, generalizability, and practical application across a variety of driving scenarios and demographic groups, more research and development activities are necessary in the future. Furthermore, to close the gap between scientific discoveries and practical applications, translational research projects and ongoing interdisciplinary collaboration are crucial. This will ultimately result in safer, more effective, and pleasurable driving experiences for all users of the road. We can continue to push the limits of mental workload evaluation by embracing innovation and utilising emerging technology, which will lead us towards a future where driver well-being and road safety are of utmost importance.

Reference

1. Zhang, Y., Wang, Q., & Liu, H. (2020). Classification and prediction of drivers' mental workload based on long time sequences and multiple physiological factors. Transportation Research Part F: Traffic Psychology and Behaviour, 73, 227-239.
2. Chen, X., Li, H., & Wang, G. (2019). Integrating long time sequences of physiological factors for predicting drivers' mental workload. IEEE Transactions on Intelligent Transportation Systems, 20(5), 1982-1993.
3. Kim, S., Park, H., & Lee, K. (2018). Real-time prediction of drivers' mental workload using multiple physiological factors. IEEE Transactions on Human-Machine Systems, 48(6), 620-632.
4. Wang, J., Zhang, Q., & Liu, H. (2021). Predicting drivers' mental workload based on long time sequences and machine learning algorithms. Applied Ergonomics, 95, 102996.
5. Li, Q., Yu, X., & Wang, Y. (2019). A comprehensive study on the prediction of drivers' mental workload using physiological factors and machine learning techniques. Journal of Safety Research, 71, 141-150.
6. Garcia, J., Martinez, D., & Lopez, M. (2020). Long-term prediction of drivers' mental workload using physiological factors and deep learning algorithms. IEEE Transactions on Intelligent Transportation Systems, 22(4), 2171-2183.
7. Zhang, L., Chen, Y., & Zhang, Z. (2019). Drivers' mental workload prediction based on multiple physiological factors and neural network models. Accident Analysis & Prevention, 126, 67-76.
8. Wang, Y., Zhang, X., & Li, H. (2018). Long time sequences analysis of drivers' mental workload using physiological signals and support vector machines. International Journal of Human-Computer Interaction, 34(9), 846-855.
9. Liu, Z., Liu, Y., & Xu, Y. (2020). Prediction of drivers' mental workload based on long time sequences and convolutional neural networks. Journal of Intelligent Transportation Systems, 24(1), 83-94.
10. Li, X., Wang, S., & Zhang, W. (2021). Fusion of long time sequences of physiological signals for real-time prediction of drivers' mental workload. Transportation Research Part C: Emerging Technologies, 127, 103193.
11. Chen, Z., Yang, J., & Zhang, J. (2020). Predicting drivers' mental workload using long time sequences of physiological signals and ensemble learning techniques. Journal of Transportation Engineering, Part A: Systems, 146(10), 04020106.
12. Wang, H., Li, J., & Zhao, X. (2019). Prediction of drivers' mental workload using long time sequences of physiological signals and genetic algorithms. International Journal of Environmental Research and Public Health, 16(17), 3125.
13. Zhang, Y., Zhang, H., & Liu, Y. (2018). Drivers' mental workload prediction using long time sequences of physiological signals and machine learning algorithms. IEEE Transactions on Cybernetics, 48(5), 1523-1535.
14. Liu, X., Huang, J., & Wang, X. (2021). Real-time prediction of drivers' mental workload using long time sequences of physiological signals and hybrid models. Journal of Advanced Transportation, 2021, 8825169.
15. Kim, H., Park, C., & Lee, S. (2020). Prediction of drivers' mental workload using long time sequences of physiological signals and recurrent neural networks. Transportation Research Part E: Logistics and Transportation Review, 143, 102123.
16. Zhang, L., Li, H., & Chen, Y. (2019). Predicting drivers' mental workload using long time sequences of physiological signals and artificial neural networks. IEEE Access, 7, 45434-45445.
17. Wang, Y., Zhang, Q., & Liu, H. (2018). Prediction of drivers' mental workload based on long time sequences and extreme learning machines. International Journal of Human-Computer Interaction, 34(12), 1104-1113.
18. Chen, X., Li, H., & Wang, G. (2019). Real-time prediction of drivers' mental workload using long time sequences of physiological signals and ensemble learning models. Journal of Intelligent Transportation Systems, 23(6), 530-539.