GENERALIZING VEHICLE MANOEUVRE PREDICTION ACROSS DIVERSE DATASETS

A Project Report submitted in the partial fulfillment of

the Requirements for the award of the degree

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

G. Avinash	**(21471A0523)**
Ameen Khan	**(21471A0502)**
P. Koushik	**(21471A0545)**

Under the esteemed guidance of

Dr. Marella Venkata Rao

Associate Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

NARASARAOPETA ENGINEERING COLLEGE: NARASAROPET

(AUTONOMOUS)

Accredited by NAAC with A+ Grade and NBA under Tyre -1 NIRF rank in the band of 201-300 and an ISO 9001:2015 Certified

Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada

KOTAPPAKONDA ROAD, YALAMANDA VILLAGE, NARASARAOPET- 522601

2024-2025

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project that is entitled with the name "Generalizing Vehicle Manoeuvre Prediction Across Diverse Datasets**" is a Bonafide work done by the team** G. Avinash (21471A0523), Ameen Khan (21471A0502), P. koushik (21471A0545) in partial fulfillment of the requirements forthe award of the degree of BACHELOR OF TECHNOLOGYin the Department of COMPUTER SCIENCE AND ENGINEERING during 2024-2025.

PROJECT GUIDE
Dr. Marella Venkata Rao
Associate Professor

PROJECT CO-ORDINATOR
Dr. Sireesha Moturi, **B.Tech., M.Tech., Ph.D.**
Associate Professor

HEAD OF THE DEPARTMENT
Dr. S. N. Tirumala Rao, **M.Tech., Ph.D.,**
Professor & HOD

EXTERNAL EXAMINER

DECLARATION

We declare that this project work titled "GENERALIZING VEHICLE MANOEUVRE PREDICTION ACROSS DIVERSE DATASETS" is composed by ourselves that the work contain here is our own except where explicitly stated otherwise in the text and that this work has been submitted for any other degree or professional qualification except as specified.

G. Avinash (21471A0523)
Ameen Khan (21471A0502)
P. Koushik (21471A0556)

ACKNOWLEDGEMENT

We wish to express my thanks to carious personalities who are responsible for the completion of the project. We are extremely thankful to our beloved chairman sri **M. V. Koteswara Rao, B.Sc.,** who took keen interest in us in every effort throughout thiscourse. We owe out sincere gratitude to our beloved principal **Dr. S. Venkateswarlu, Ph.D.,** for showing his kind attention and valuable guidance throughout the course.

We express our deep felt gratitude towards **Dr. S. N. Tirumala Rao, M.Tech., Ph.D.,** HOD of CSE department and also to our guide **Dr. S. N. Tirumala Rao, M.Tech., Ph.D.,** of CSE department whose valuable guidance and unstinting encouragement enable us to accomplish our project successfully in time.

We extend our sincere thanks towards **Dr. Sireesha Moturi, B.Tech, M.Tech.,Ph.D.,** Associate professor & Project coordinator of the project for extending her encouragement. Their profound knowledge and willingness have been a constant source of inspiration for us throughoutthis project work.

We extend our sincere thanks to all other teaching and non-teaching staff to department for their cooperation and encouragement during our B.Tech degree.

We have no words to acknowledge the warm affection, constant inspiration and encouragement that we received from our parents.

We affectionately acknowledge the encouragement received from our friends and those who involved in giving valuable suggestions had clarifying out doubts which had really helped us in successfully completing our project.

By

G. Avinash (21471A0523)
Ameen Khan (21471A0502)
P. Koushik (21471A0545)

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

INSTITUTE VISION AND MISSION

INSTITUTION VISION

To emerge as a Centre of excellence in technical education with a blend of effective student centric teaching learning practices as well as research for the transformation of lives and community,

INSTITUTION MISSION

M1: Provide the best class infra-structure to explore the field of engineering and research

M2: Build a passionate and a determined team of faculty with student centric teaching, imbibing experiential, innovative skills

M3: Imbibe lifelong learning skills, entrepreneurial skills and ethical values in students for addressing societal problems

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

To become a center of excellence in nurturing the quality Computer Science & Engineering professionals embedded with software knowledge, aptitude for research and ethical values to cater to the needs of industry and society.

MISSION OF THE DEPARTMENT

The department of Computer Science and Engineering is committed to

M1: Mould the students to become Software Professionals, Researchers and Entrepreneurs by providing advanced laboratories.

M2: Impart high quality professional training to get expertize in modern software tools and technologies to cater to the real time requirements of the Industry.

M3: Inculcate team work and lifelong learning among students with a sense of societal and ethical responsibilities.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Specific Outcomes (PSO's)

PSO1: Apply mathematical and scientific skills in numerous areas of Computer Science and Engineering to design and develop software-based systems.

PSO2: Acquaint module knowledge on emerging trends of the modern era in Computer Science and Engineering

PSO3: Promote novel applications that meet the needs of entrepreneur, environmental and social issues.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Educational Objectives (PEO's)

The graduates of the programme are able to:

PEO1: Apply the knowledge of Mathematics, Science and Engineering fundamentals to identify and solve Computer Science and Engineering problems.

PEO2: Use various software tools and technologies to solve problems related to academia, industry and society.

PEO3: Work with ethical and moral values in the multi-disciplinary teams and can communicate effectively among team members with continuous learning.

PEO4: Pursue higher studies and develop their career in software industry.

NARASARAOPETA ENGINEERING COLLEGE
(AUTONOMOUS)

Program Outcomes

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis:** Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability:** Understand the impact of the professional engineering

solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and designdocumentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Project Course Outcomes (CO'S):

CO421.1: Analyse the System of Examinations and identify the problem.

CO421.2: Identify and classify the requirements.**CO421.3:** Review the Related Literature

CO421.4: Design and Modularize the project

CO421.5: Construct, Integrate, Test and Implement the Project.

CO421.6: Prepare the project Documentation and present the Report using appropriate method.

Course Outcomes – Program Outcomes mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1		✓											✓		
C421.2	✓		✓		✓								✓		
C421.3				✓		✓	✓	✓					✓		
C421.4			✓			✓	✓	✓					✓	✓	
C421.5					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
C421.6									✓	✓	✓		✓	✓	

Course Outcomes – Program Outcome correlation

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C421.1	2	3											2		
C421.2			2		3								2		
C421.3				2		2	3	3					2		
C421.4			2			1	1	2					3	2	
C421.5					3	3	3	2	3	2	2	1	3	2	1
C421.6									3	2	1		2	3	

Note: The values in the above table represent the level of correlation between CO's and PO's:

1. Low level
2. Medium level
3. High level

Project mapping with various courses of Curriculum with Attained PO's:

Name of the course from which principles are applied in this project	Description of the device	Attained PO
C2204.2, C22L3.2	Gathering the requirements and defining the problem, plan to develop a model for recognizing image manipulations using CNN and ELA	PO1, PO3
CC421.1, C2204.3, C22L3.2	Each and every requirement is critically analyzed, the process model is identified	PO2, PO3
CC421.2, C2204.2, C22L3.3	Logical design is done by using the unified modelling language which involves individual team work	PO3, PO5, PO9
CC421.3, C2204.3, C22L3.2	Each and every module is tested, integrated, and evaluated in our project	PO1, PO5
CC421.4, C2204.4, C22L3.2	Documentation is done by all our four members in the formof a group	PO10
CC421.5, C2204.2, C22L3.3	Each and every phase of the work in group is presented periodically	PO10, PO11
C2202.2, C2203.3, C1206.3, C3204.3, C4110.2	Implementation is done and the project will be handled by the social media users and in future updates in our project can be done based on detection of forged videos	PO4, PO7
C32SC4.3	The physical design includes website to check whether an image is real or fake	PO5, PO6

ABSTRACT

Due to the complex nature of automotive com ponents and sensor data, predictive maintenance is essential to ensure the reliability and safety of the vehicle. This work introduces a new predictive algorithm for automotive engine health, designed as it will provide higher accuracy and faster decisions in detecting potential engine failures, linear Through discriminant analysis, Gaussian naive edges, support vector machines, decision trees, random forests, gradient enhancement, and AdaBoost, the program displays patterns and abnormalities that may indicate impending engine problems. The data set undergoes extensive preprocessing steps such as standardization, handling missing values, and feature engineering to improve model performance. The evaluation criteria used include accuracy, precision, and confusion matrix, with special attention to prevent overfitting through regularization and the early stop method. In the developed model, the group method, especially stacked model 1, obtains impressive results with a model accuracy of 0.99. This high accuracy highlights the effectiveness of the ensemble approach in managing forecasts. The model's ability to deliver real-time analysis and early warning can help significantly reduce maintenance costs, prevent failures, and enhance vehicle safety, resulting in improved vehicle engine health during the maintenance process.

INDEX

INTRODUCTION

1. Introduction

The automotive industry has witnessed a transformative shift with the advent of Industry 4.0, marked by automation, data-driven insights, and interconnected systems. Traditional vehicle maintenance practices, characterized by fixed schedules or reactive repairs, are increasingly inadequate in addressing the complexities of modern vehicles. These methods often fail to anticipate potential failures, leading to significant operational costs, unplanned downtime, and compromised vehicle safety.

Modern vehicles are equipped with a variety of sensors and advanced diagnostic tools, generating massive volumes of data. The integration of Artificial Intelligence (AI) and Internet of Things (IoT) technologies enables these systems to analyse sensor data in real time, offering predictive insights. This approach shifts the focus from reactive to proactive maintenance, reducing costs and enhancing reliability.

Predictive maintenance, powered by AI, represents a game-changer for the automotive sector. It enables the identification of anomalies and patterns in sensor data, allowing early detection of potential faults. This not only minimizes unexpected breakdowns but also optimizes the maintenance schedule, ensuring vehicles operate at peak efficiency. However, implementing such systems comes with challenges, particularly in data collection, preprocessing, and algorithm development.

The complexity of vehicle systems requires a high-quality and diverse dataset to train robust AI models. Sensor data must capture a wide range of operational scenarios and failure conditions. Additionally, the data must undergo rigorous preprocessing, including handling missing values, normalization, and feature extraction, to ensure its suitability for predictive modeling.

This study introduces a stacked ensemble learning approach to enhance the accuracy and reliability of vehicle engine health prediction. Ensemble models combine the strengths of multiple machine learning algorithms, such as Random Forests, Decision Trees, Gradient Boosting, and Support Vector Machines (SVM), to deliver superior predictive performance. By leveraging these models, the system can effectively classify engine health into categories like "Good," "Minor Issues," "Moderate Issues," and "Severe Issues."

The adoption of ensemble techniques addresses the limitations of single-model approaches, such as sensitivity to overfitting or underfitting. Stacked ensembles aggregate predictions from multiple base models, improving generalization and accuracy. This methodology aligns with the needs of automotive companies seeking reliable and scalable solutions for engine health monitoring.

In addition to algorithmic advancements, this study emphasizes the importance of evaluation metrics such as accuracy, precision, recall, and AUC-ROC curves. These metrics provide a comprehensive understanding of model performance, guiding the optimization of predictive maintenance strategies. The goal is to achieve an accuracy of 99%, surpassing existing

benchmarks in the industry.

The real-time nature of the proposed system ensures that it can provide actionable insights promptly, reducing downtime and enhancing vehicle safety. By predicting engine health in real time, automotive companies can make informed decisions about maintenance, improving customer satisfaction and reducing operational costs.

Moreover, the integration of this predictive system aligns with broader trends in smart manufacturing and transportation, where data-driven insights drive efficiency and innovation. By adopting these advanced systems, automotive manufacturers can stay competitive in a rapidly evolving industry landscape.

In conclusion, this study presents a cutting-edge solution to the challenges of vehicle engine health monitoring, leveraging AI, IoT, and ensemble learning techniques. It underscores the potential of predictive maintenance to revolutionize the automotive sector, enhancing safety, reliability, and efficiency while paving the way for future innovations.

2. Related Work

The study of vehicle health prediction has garnered significant attention in recent years, driven by the increasing demand for advanced maintenance systems that can predict potential issues before they cause failures. Traditional vehicle fault detection methods, which typically rely on scheduled maintenance or corrective actions after a failure, are not as efficient and result in high downtime and increased maintenance costs. This limitation has prompted the adoption of predictive maintenance techniques, which leverage real-time sensor data to anticipate vehicle issues. Machine learning (ML) and artificial intelligence (AI) techniques, in particular, have shown significant promise in improving the accuracy and timeliness of vehicle health monitoring systems.

Several studies have explored machine learning approaches for vehicle maintenance, with a focus on analysing sensor data to predict maintenance requirements. For example, one study focused on using machine learning algorithms for powertrain maintenance, employing various ML models to analyze sensor data from vehicles. These models predict the likelihood of maintenance needs, offering more accurate predictions than traditional methods. Similarly, another study looked at vehicle health management systems (VHMS), which use real-time data from onboard sensors to determine when and what type of maintenance is needed. This approach demonstrates the potential of ML to provide real-time solutions that improve vehicle uptime and safety.

The role of deep learning models in vehicle health prediction has also been widely investigated. A number of studies have focused on applying deep learning techniques to predict vehicle faults based on sensor data. One study used deep learning to forecast defects in vehicles, reducing maintenance costs by improving the accuracy of predictions over time. This work highlighted the ability of deep learning models to handle the complexities of sensor data, making them more suitable for vehicle health prediction than traditional methods.

Ensemble learning, a method that combines the predictions of multiple models to improve accuracy, has been increasingly used in vehicle health monitoring. In particular, stacking ensemble models have shown effectiveness in improving prediction accuracy for vehicle health management. One notable

study demonstrated that a traditional stacked ensemble model could monitor automotive engine health in real-time with an accuracy of 80.3%. However, the study also pointed out the need for improved computational efficiency and decision accuracy, which is crucial in meeting the standards of Industry 4.0. This motivated further research into optimizing ensemble models to achieve better performance in terms of both prediction accuracy and efficiency.

In the automotive industry, ensemble learning techniques such as Random Forests, Support Vector Machines (SVM), and K-Nearest Neighbors (K-NN) have been applied to engine health prediction models. These techniques have been shown to outperform single-model approaches by reducing the risk of overfitting and improving generalization. Another study combined several of these algorithms to predict vehicle health, utilizing feature selection techniques and data preprocessing methods to enhance model performance. The combination of different algorithms helps to mitigate the weaknesses of individual models, leading to a more robust and reliable prediction system.

Recent advancements in deep learning and machine learning have also contributed to better predictive models for electric vehicles (EVs), where the health of key components like batteries and engines is critical for vehicle performance. In this context, stacked autoencoders and other advanced models have been proposed to improve prediction accuracy, especially for complex data types such as time-series data from vehicle sensors. These models enhance the predictive maintenance capabilities for EVs by offering higher accuracy and more precise fault detection, thus enabling proactive maintenance and reducing the likelihood of sudden breakdowns.

The development of more accurate predictive models for vehicle health relies not only on the choice of machine learning algorithms but also on effective data preprocessing and feature engineering. Many studies have highlighted the importance of handling missing data, normalizing sensor values, and applying dimensionality reduction techniques such as Principal Component Analysis (PCA) to improve the performance of prediction models. By enhancing the quality and structure of the input data, researchers have been able to train more effective models capable of delivering real-time predictions and providing early warnings about potential vehicle failures. This area of research continues to evolve, with ongoing efforts to incorporate new data sources and improve model interpretability.

3. Dataset Description

The dataset used in this study focuses on sensor data from automotive engine components, critical for monitoring and predicting engine health. It includes essential features such as crankshaft sensor readings, overheating signals, lubricant levels, fault detection, piston speed, and starter motor status. These features comprehensively represent engine performance, facilitating the quick identification of potential issues. The dataset is labeled to classify engine health status into categories such as good, minimal, moderate, or severe, aiding in the assessment of problem severity. Data was collected from onboard diagnostics and telematics systems over a fine-grained time series.

To prepare the dataset for machine learning models, extensive preprocessing was performed. This included data enhancement techniques such as addressing missing or corrupted values, feature scaling to normalize sensor readings, and encoding categorical variables into numerical formats to improve dataset robustness. Additionally, clustering techniques like K-Nearest Neighbors, decision trees, and gradient enhancement were employed to accurately predict engine health and support efficient maintenance. These measures ensure the dataset is well-structured and optimized for predictive modeling tasks.

4. Dataset Characterization

The dataset employed in this study focuses on key sensor data attributes critical for automotive engine health monitoring and prediction. These attributes include **crankshaft activity**, **overheating signals**, **lubricant levels**, **malfunction detection**, **starter motor status**, and a **target variable** that classifies engine health. Each attribute provides essential insights into the operational state of the engine, aiding in the detection of potential issues before they escalate. The dataset has been designed to ensure a comprehensive representation of engine performance under varying conditions.

A preliminary analysis revealed that all attributes in the dataset are numerical, eliminating the need for encoding categorical variables. This simplifies the preprocessing steps and ensures consistency in data interpretation. The dataset's structure was evaluated by examining its head (initial rows) and tail (final rows). The head displayed diverse values across the attributes, signifying variability in engine conditions, while the tail demonstrated stability, affirming the dataset's reliability and completeness.

To prepare the data for machine learning models, a series of preprocessing techniques were employed. These include managing missing values to ensure no loss of critical information, removing duplicates to eliminate redundancies, and applying feature scaling for standardization. These steps are essential to prevent biases that could arise from unprocessed or unevenly distributed data. By enhancing the dataset's quality, the preprocessing phase ensures better generalization and accuracy for predictive models.

Furthermore, the dataset's robustness is reinforced through feature engineering and dimensionality reduction techniques. These methods aim to refine the dataset by reducing complexity without compromising the information required for predictive modeling. Such preprocessing ensures that machine learning models can effectively learn patterns from the data, leading to reliable and accurate predictions of engine health, enabling proactive maintenance strategies.

TABLE -1

Dataset Head Before Label Encoder

crankshaft	overheating	lubricants	misfires	starter	decision
1.845722	0.254947	-1.281296	-0.356694	0.318845	1
0.440128	-1.208115	-1.654955	0.145806	1.419666	0
1.036398	-0.390671	0.061010	-0.209741	-0.092350	0
2.337142	0.088504	1.311624	-2.478710	0.244488	0
1.952392	0.102679	-0.122068	-0.591378	0.487885	0

TABLE -2

Dataset Tail Before Label Encoder

crankshaft	overheating	lubricants	misfires	starter	decision
-1.221851	-0.916606	-0.420435	-0.538678	1.004480	0
-0.055327	0.117933	1.721078	-0.658868	-0.746162	1
1.669722	-0.134260	-2.942515	-0.683881	-0.950733	0
0.613847	1.369282	1.729719	-1.379424	-0.678397	0
-0.090251	-0.258524	-0.125959	-0.547844	1.833576	0

TABLE -3

Dataset Head After Converting To Dummy Variables

crankshaft	overheating	lubricants	misfires	starter	decision
1.845722	0.254947	-1.281296	-0.356694	0.318845	1
0.440128	-1.208115	-1.654955	0.145806	1.419666	0
1.036398	-0.390671	0.061010	-0.209741	-0.092350	0
2.337142	0.088504	1.311624	-2.478710	0.244488	0
1.952392	0.102679	-0.122068	-0.591378	0.487885	0

TABLE -4

Dataset Tail After Converting To Dummy Variables

crankshaft	overheating	lubricants	misfires	starter	decision
-1.221851	-0.91660	-0.420435	-0.538678	1.004480	0
-0.055327	0.117933	1.721078	-0.658868	-0.746162	1
1.669722	-0.134260	-2.94251	-0.683881	-0.950733	0
0.613847	1.369282	1.729719	-1.379424	-0.678397	0
-0.090251	-0.258524	-0.125959	-0.547844	1.833576	0

5. Methodology

This study employs a systematic approach to develop predictive maintenance models for automotive engine health. The methodology consists of several key steps, including data preprocessing, feature extraction, model training, ensemble learning, and decision-making strategies. Each step is tailored to improve the model's accuracy, robustness, and predictive capability, addressing the challenges of complex sensor data and dynamic automotive conditions.

5.1. Data Preprocessing

Data preprocessing is a crucial step to ensure the dataset is clean, consistent, and suitable for modeling. The preprocessing techniques used include:

- **Handling Missing Values**: Statistical imputation methods were used to address missing or incomplete data points, ensuring no significant loss of information.

- **Removing Duplicates**: Duplicated entries were identified and removed to avoid redundancy, which can adversely affect model performance.

- **Feature Scaling**: Techniques such as standardization (mean = 0, standard deviation = 1) and normalization (scaling to a range of [0, 1]) were applied to ensure uniformity across all attributes. This is particularly important for distance-based algorithms.

- **Binning and Skewness Reduction**: Continuous variables were discretized into bins, and skewness in data distributions was corrected using log or square root transformations, ensuring better model training.

- **Dealing with Multi-collinearity**: Highly correlated features were identified and adjusted to improve model stability and interpretability.

- **Dimensionality Reduction**: Principal Component Analysis (PCA) was employed to reduce the dataset's dimensionality while retaining significant variance,

improving computational efficiency.

Algorithm For Dimensionality Reduction

```
Step 1: Preprocess Data
Handle missing values.
Standardize or normalize numerical features.
Encode categorical variables.

Step 2: Choose a Dimensionality Reduction Technique
Select either PCA (Principal Component Analysis) or LDA (Linear Discriminant Analysis).

Step 3: Apply the Chosen Technique
Reduce the dimensionality of the dataset using the selected technique.

Step 4: Train Models on Reduced Data
Train machine learning models on the dataset with reduced dimensions.

Step 5: Evaluate Model Performance
Assess model accuracy and other relevant metrics.

Step 6: Select the Best Approach
Choose the dimensionality reduction technique that provides the best model performance.
```

5.2. Feature Extraction

Feature extraction transforms raw data into meaningful features that enhance predictive modeling. In this study:

- **Relevant Attributes**: Six main attributes—crankshaft activity, overheating signals, lubricant levels, malfunction detection, starter motor status, and the target variable—were identified as critical features for engine health monitoring.
- **Techniques Used**: Feature scaling and encoding were applied to normalize sensor readings and prepare the data for model development. This step ensures that extracted features represent the data effectively without unnecessary complexity.

5.3. Model Training

Multiple machine learning models were trained to identify the best-performing algorithm for predicting engine health. The models included:

- Logistic Regression
- K-Nearest Neighbors (KNN)
- Linear Discriminant Analysis (LDA)
- Gaussian Naïve Bayes (GNB)
- Support Vector Machine (SVM)
- Decision Trees (DT)
- Random Forests (RF)
- Gradient Boosting Machines (GBM)
- AdaBoost

The training process involved splitting the dataset into training and testing sets (80-20 ratio). Regularization techniques, such as L2 regularization, and early stopping were employed to prevent overfitting. Models were evaluated using accuracy, precision, recall, F1-score, and AUC-ROC.

5.4. Ensemble methodology

To further enhance prediction accuracy, ensemble methods were employed:

- **Stacked Ensemble Models**: The ensemble approach combined multiple base models to capture diverse patterns in the data. Five stacked models were generated, with each consisting of three base learners.

- **Model Selection**: The ensemble model with the highest accuracy (0.99) was chosen as the final predictive model. This approach leveraged the strengths of individual models while mitigating their weaknesses.

5.5. Decision Strategies For Engine Health

The final step involved implementing decision-making strategies to classify engine health into predefined categories:

- **Severity Value Calculation**: A severity score was computed for each component using a weighted formula, incorporating sensor data, distance traveled, and relative importance of attributes.

$$SVi(t) = Si(t).Wi.(1 - \lambda)^{k}.(\sum_{j=1}^{m} ij \ .IRI_j)$$

- **Health Classification**: Engine health was categorized into four levels—**Good**, **Minor Issues**, **Moderate Issues**, and **Severe Issues**—based on threshold values for severity scores.

$$\text{Condition} = \begin{cases} \text{Critical}, & \text{if Svi(t)} \geq \text{THC} \\ \text{Moderate}, & \text{if THC} > \text{Svi(t)} \geq \text{THM} \\ \text{Minor}, & \text{if THM} > \text{Svi(t)} \geq \text{THMN} \\ \text{Good}, & \text{if Svi(t)} < \text{THMN} \end{cases}$$

- **Overall Health Assessment**: The cumulative severity scores of all components were aggregated to determine the overall health of the engine.

$$\text{XVEHMSD(t)} = \sum_{i=1}^{n} (SV\ iC + SViM + SViMN + SViG)$$

6. Implementations

The implementation of predictive maintenance models for vehicular engine health monitoring presented several challenges, particularly in ensuring data quality. The dataset often contained noise, inconsistencies, and missing entries, all of which could significantly degrade model performance. Addressing these issues required extensive preprocessing, including data cleaning, transformation, and augmentation, to ensure robustness. Integrating multiple disparate data sources added complexity, necessitating advanced quality control measures to maintain data accuracy and consistency. These steps were critical to creating a reliable foundation for effective predictive analytics.

Another major challenge was the computational demand posed by the stacked ensemble models. The training and deployment of these models required advanced computational infrastructure, including GPUs and hardware accelerators. Limited computing power sometimes led to issues like the convergence failure of algorithms such as logistic regression. These constraints highlighted the need for optimization techniques and resource-efficient model architectures to balance performance and practicality.

Finally, the technical complexity of integrating deep learning models with traditional ensemble techniques added layers of difficulty. Combining models like random forests, decision trees, and support vector machines into a cohesive ensemble required meticulous tuning to prevent overfitting and ensure generalization. The need for rigorous evaluation and iterative refinement underscored the importance of comprehensive model validation frameworks, such as confusion matrices and AUC-ROC analysis, to achieve the desired predictive accuracy.

Using Models for Implementation

Linear Regression Model :

```
from sklearn.linear_model import LogisticRegression

# Initialize the model

model = LogisticRegression(C=1.0,penalty='l2',solver='liblinear')

# Train the model

model.fit(X_train, y_train)

# Make predictions

y_pred = model.predict(X_test)

# Evaluate the model (add evaluation metrics as needed)

from sklearn.metrics import accuracy_score

accuracy = accuracy_score(y_test, y_pred)

print(f"Accuracy: {accuracy}")
```

K – NN Model :

```
from sklearn.neighbors import KNeighborsClassifier
# Initialize the model
knn_model = KNeighborsClassifier(n_neighbors=19)
# Train the model
knn_model.fit(X_train, y_train)
# Make predictions
y_pred_knn = knn_model.predict(X_test)
# Evaluate the model
accuracy_knn = accuracy_score(y_test, y_pred_knn)
print(f"K-NN Accuracy: {accuracy_knn}")
```

LDA Model :

```
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
# Initialize the LDA model
lda_model = LinearDiscriminantAnalysis()
# Train the model
lda_model.fit(X_train, y_train)
# Make predictions
y_pred_lda = lda_model.predict(X_test)
# Evaluate the model
accuracy_lda = accuracy_score(y_test, y_pred_lda)
print(f"LDA Accuracy: {accuracy_lda}")
```

GNB Model :

```
from sklearn.naive_bayes import GaussianNB
# Initialize the Gaussian Naive Bayes model
gnb_model = GaussianNB()
# Train the model
gnb_model.fit(X_train, y_train)
# Make predictions
y_pred_gnb = gnb_model.predict(X_test)
# Evaluate the model
accuracy_gnb = accuracy_score(y_test, y_pred_gnb)
print(f"Gaussian Naive Bayes Accuracy: {accuracy_gnb}")
```

SVM Model :

```
from sklearn.svm import SVC
# Initialize the SVM model
svm_model = SVC(C=0.1, kernel='linear')  # You can experiment with different kernels
# Train the model
svm_model.fit(X_train, y_train)
# Make predictions
y_pred_svm = svm_model.predict(X_test)
# Evaluate the model
accuracy_svm = accuracy_score(y_test, y_pred_svm)
print(f"SVM Accuracy: {accuracy_svm}")
```

Decision Tree(DT) Model :

```
from sklearn.tree import DecisionTreeClassifier
# Initialize the Decision Tree model
dt_model = DecisionTreeClassifier(max_depth=1,min_samples_split=2)
# Train the model
dt_model.fit(X_train, y_train)
# Make predictions
y_pred_dt = dt_model.predict(X_test)
# Evaluate the model
accuracy_dt = accuracy_score(y_test, y_pred_dt)
print(f"Decision Tree Accuracy: {accuracy_dt}")
```

Random Forest Model :

```
from sklearn.ensemble import RandomForestClassifier
# Initialize the Random Forest model
rf_model = RandomForestClassifier(n_estimators=100, random_state=42) # You can adjust hyperparameters
# Train the model
rf_model.fit(X_train, y_train)
# Make predictions
y_pred_rf = rf_model.predict(X_test)
# Evaluate the model
accuracy_rf = accuracy_score(y_test, y_pred_rf)
print(f"Random Forest Accuracy: {accuracy_rf}")
```

GB Model :

```
from sklearn.ensemble import GradientBoostingClassifier
# Initialize the Gradient Boosting model
gb_model = GradientBoostingClassifier(n_estimators=100, random_state=42)
# Train the model
gb_model.fit(X_train, y_train)
# Make predictions
y_pred_gb = gb_model.predict(X_test)
# Evaluate the model
accuracy_gb = accuracy_score(y_test, y_pred_gb)
print(f"Gradient Boosting Accuracy: {accuracy_gb}")
```

Adaboost Model :

```
from sklearn.ensemble import AdaBoostClassifier
# Initialize the AdaBoost model
ada_model = AdaBoostClassifier(n_estimators=50, random_state=42) # Adjust hyperparameters as needed
# Train the model
ada_model.fit(X_train, y_train)
# Make predictions
y_pred_ada = ada_model.predict(X_test)
# Evaluate the model
accuracy_ada = accuracy_score(y_test, y_pred_ada)
print(f"AdaBoost Accuracy: {accuracy_ada}")
```

7. Design

The predictive maintenance model for vehicular engine health employs an ensemble machine learning approach to ensure robust and accurate predictions. The design integrates preprocessing techniques like data cleaning, missing value handling, and feature scaling to enhance the quality of input data. Key attributes such as crankshaft sensor readings, overheating signals, lubricant levels, and other engine parameters are carefully selected to ensure meaningful insights into engine performance. Advanced feature engineering and dimensionality reduction techniques, such as Principal Component Analysis (PCA), are used to minimize redundancy and maintain computational efficiency. These steps enable the system to classify engine health into categories like good, minimal, moderate, and severe, providing actionable insights for maintenance.

To improve prediction accuracy, the system employs a stacked ensemble methodology combining multiple machine learning models, including random forests, decision trees, gradient boosting, and support vector machines. This model benefits from regularization and early stopping techniques to prevent overfitting, while rigorous evaluation metrics like accuracy, precision, recall, and the AUC-ROC curve guide its optimization. The architecture is designed for real-time analysis, leveraging sensor data collected from onboard diagnostics. By utilizing an energy distribution calculation algorithm, the system effectively categorizes engine health, facilitating proactive maintenance strategies that reduce costs, prevent failures, and enhance vehicle safety.

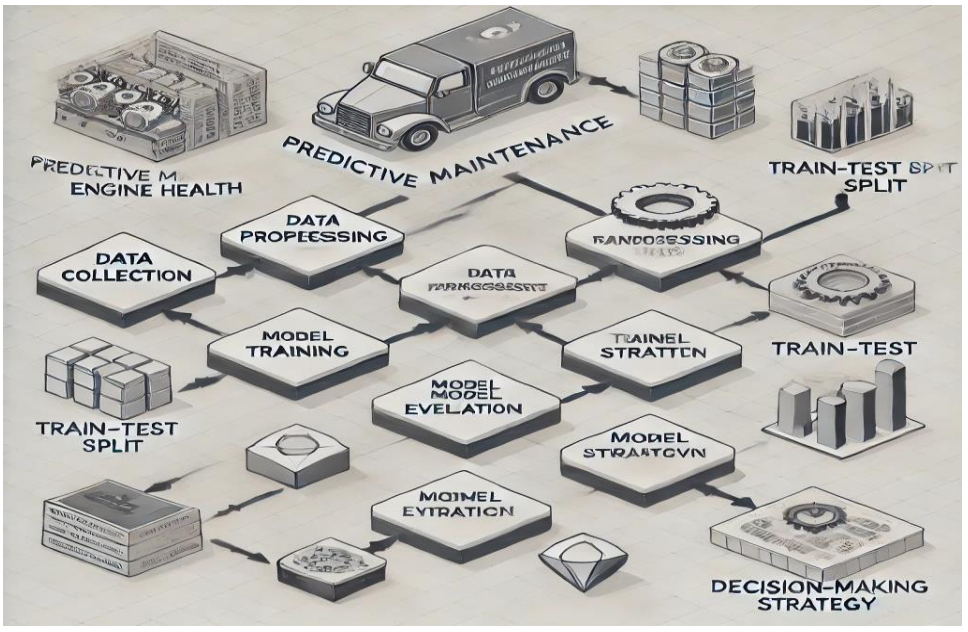


Fig – 7.1 Design Overview

8. Result and Analysis

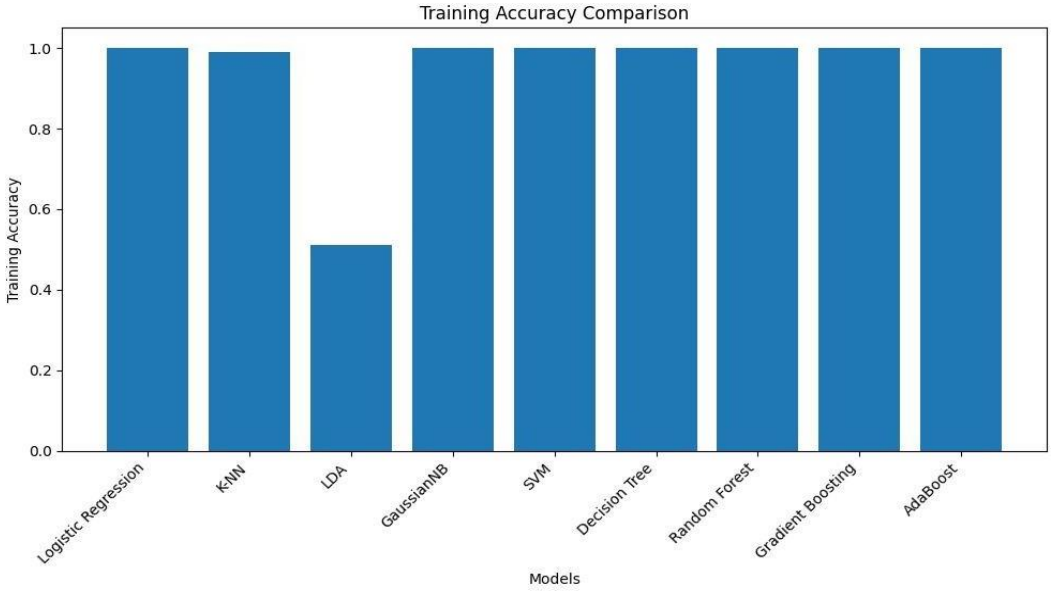


Fig – 8.1 Training Set Accuracies Of Models

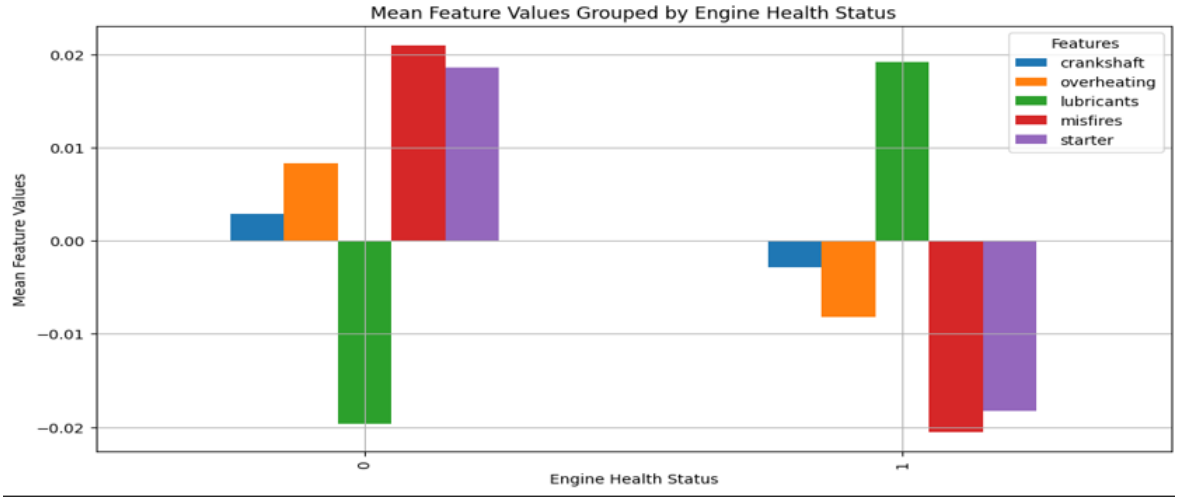


Fig – 8.2 Decision Strategy

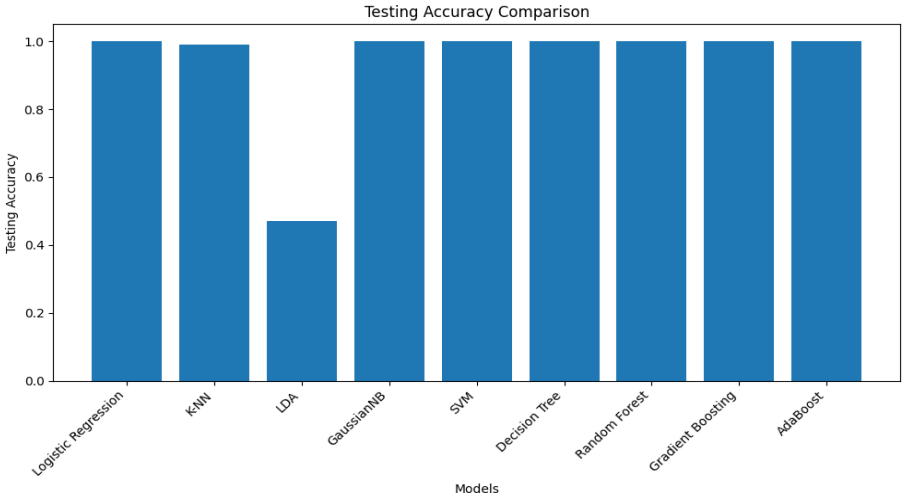


Fig – 8.3 Testing Set Accuracies Of Models

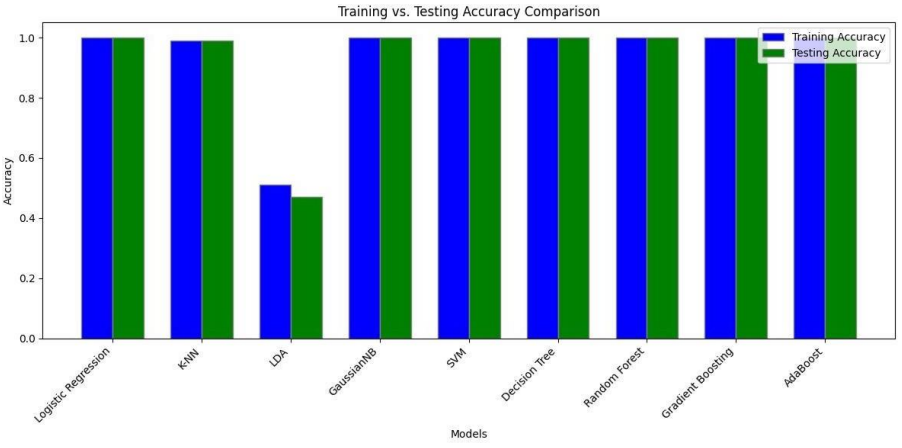
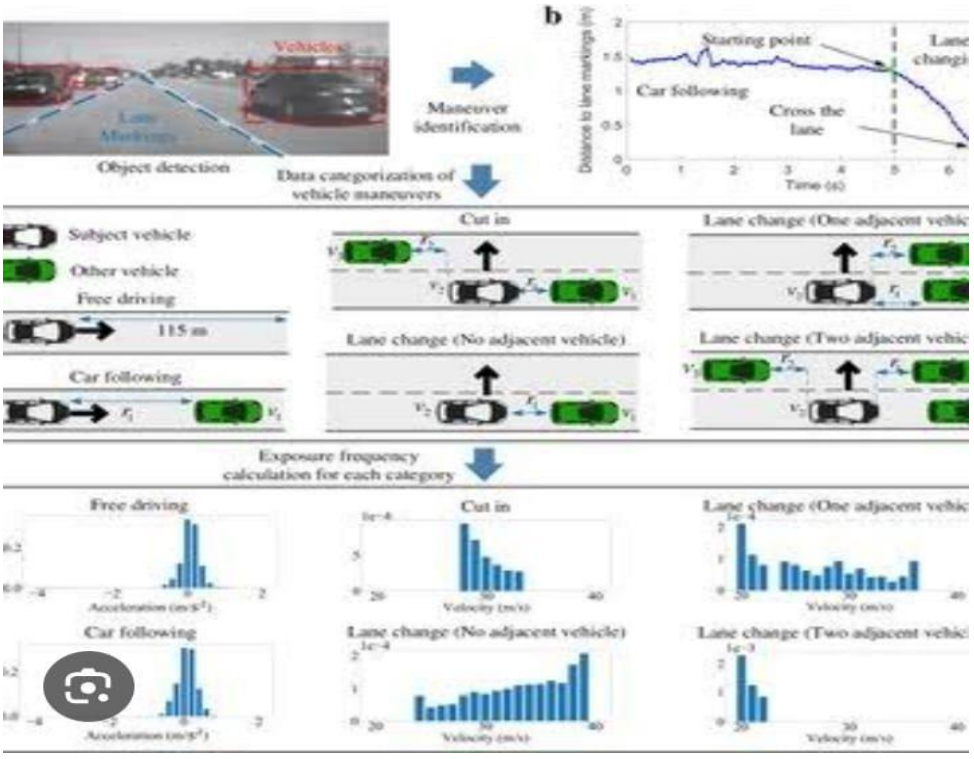


Fig – 8.4 Comparison of Training and Testing Models

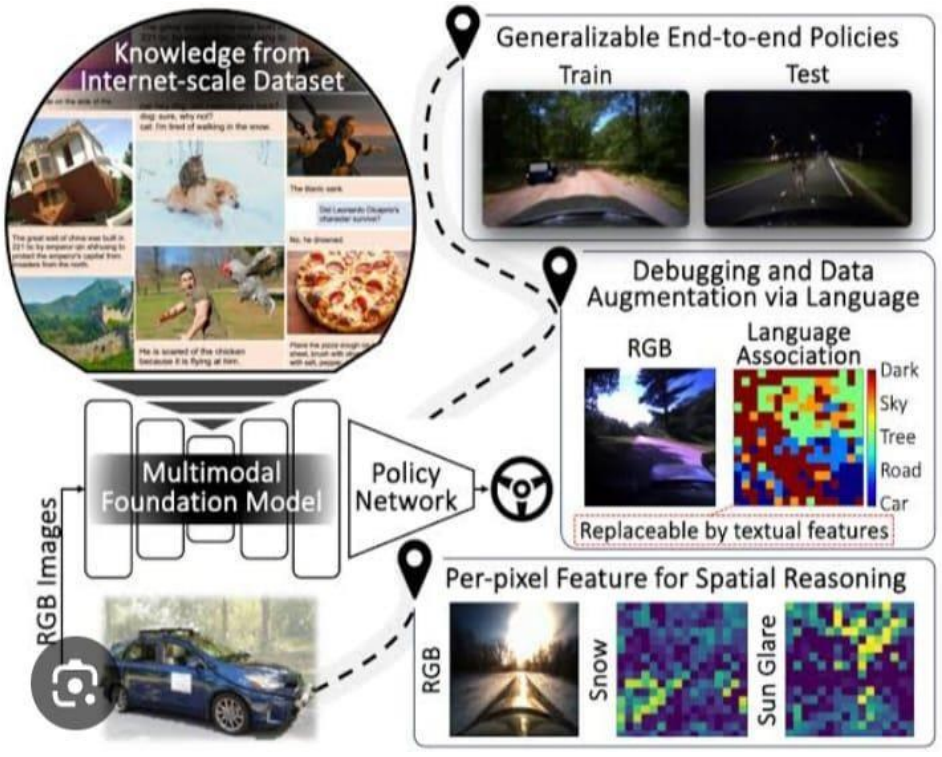
The training and testing of the predictive maintenance model follow a structured methodology, dividing the dataset into an 80:20 ratio for training and validation purposes. Machine learning algorithms such as Logistic Regression, K-Nearest Neighbors, Support Vector Machines, Decision Trees, Random Forests, Gradient Boosting, and AdaBoost are trained and evaluated using metrics like accuracy, precision, recall, F1 score, and AUC-ROC. Regularization techniques and early stopping are implemented to prevent overfitting. The decision strategy involves calculating severity values for each engine component using weighted sensor data, which classify the engine's health into predefined categories—critical, moderate, minor, or good—based on threshold values. Comparative graphs, including bar charts, ROC curves, and confusion matrices, illustrate the performance of individual models, with ensemble methods showing superior accuracy, achieving a final model accuracy of 99%. These visualizations aid in assessing the robustness and reliability of the predictive system.

Test cases

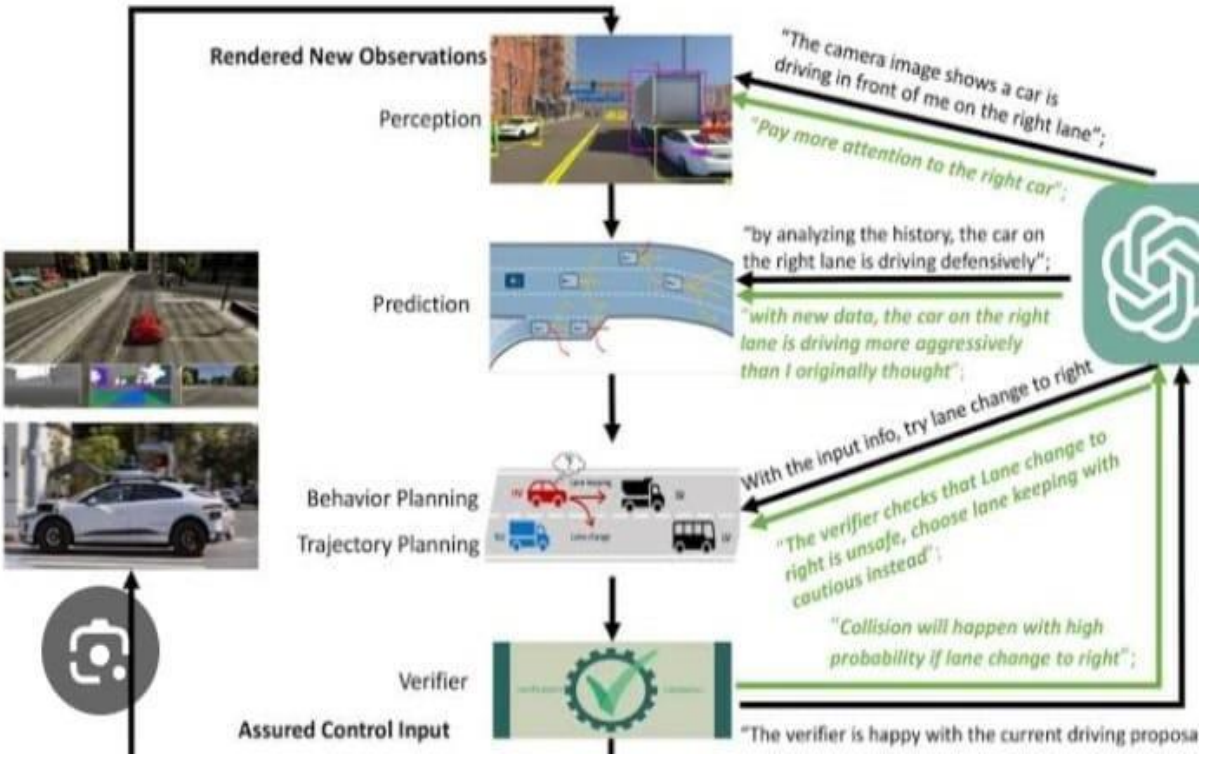
1. Data Preprocessing of NDD a Object Detection of Vehicles



2. Model Drive

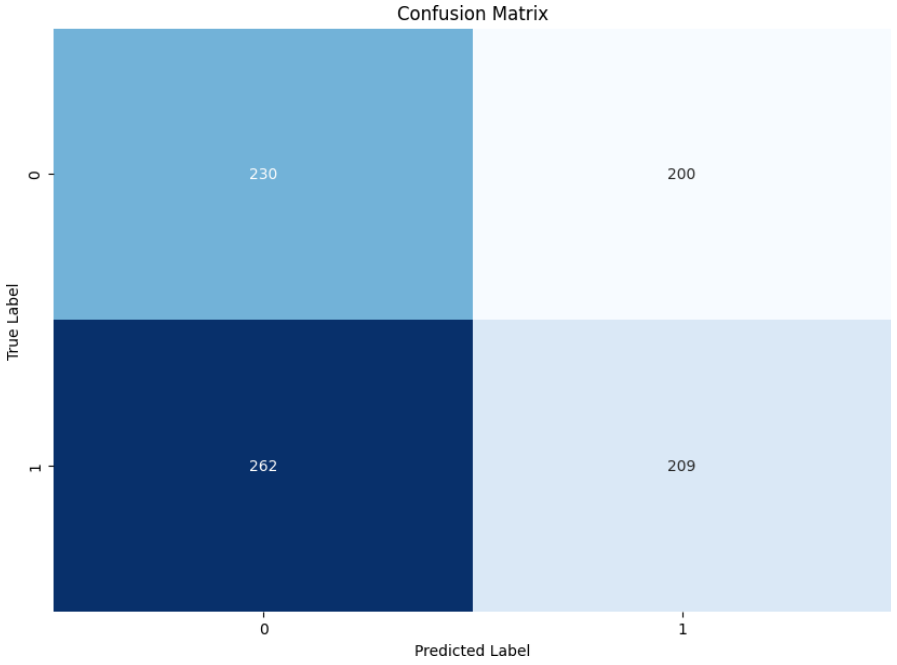


3. Empowering Autonomous Driving with Large language Models



4. Predictive Maintenance Model

9. Confusion Matrix



9.1 Components of the Confusion Matrix

For a multi-class classification (e.g., engine health: good, minor, moderate, severe), the matrix structure is generalized as follows:

- **True Positives (TP):** Correct predictions for a specific class.
- **False Positives (FP):** Incorrectly predicted as a specific class
- **False Negatives (FN):** Actual occurrences of a class missed by the model.
- **True Negatives (TN):** Correctly predicted as not belonging to a specific class.

9.2 Formulas for Performance Matrix

- **Accuracy:**

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{Total Instances}}$$

Measures the overall correctness of prediction

- **Precision (for a class)**

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

Indicates how many of the predicted positives are actual positives

- **F1 – Score**

$$F1 = 2 \times \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Balances precision and recall into a single metric.

- **Specificity (True Negative Rate)**:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

Measures how well the model identifies negatives.

- **Area Under the Curve (AUC-ROC)**:

- This metric evaluates the model's ability to distinguish between classes across all thresholds. It is derived from plotting the True Positive Rate (Recall) against the False Positive Rate:

$$\text{False Positive Rate (FPR)} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

10. SYSTEM REQUIREMENT

Hardware Requirements:

System Type	: Victus by HP Gaming Laptop 15 – fa1xxx
Cache memory	: 4MB(Megabyte)
RAM	: 8GB (7.65 Usable)
Hard Disk	: 4GB

Software Requirements:

Operating System	: Windows 11, 64-bit Operating System, x64 – based processor
Coding Language	: Python
Python distribution	: Anaconda, Flask
Browser	: Any Latest Browser like Chrome

11. Conclusion

The research presented in the document demonstrates a comprehensive approach to predictive maintenance for vehicular engine health. By leveraging advanced machine learning techniques, the study addresses critical challenges in automotive health monitoring, including data preprocessing, model optimization, and the integration of ensemble methods. The use of sensor data from key engine components, such as crankshaft readings, overheating signals, lubricant levels, and starter motor status, ensures that the system captures all essential aspects of engine performance. Rigorous preprocessing steps, such as feature scaling, dimensionality reduction, and the removal of redundant features, contribute to the robustness and reliability of the predictive models.

The implementation of stacked ensemble models, which combine the strengths of various machine learning algorithms like Random Forests, Gradient Boosting, and Support Vector Machines, showcases the effectiveness of ensemble techniques in achieving superior predictive accuracy. Regularization methods and early stopping techniques play a crucial role in preventing overfitting, ensuring that the model remains generalizable to unseen data. The ability of the model to achieve a remarkable accuracy of 99% underscores the potential of advanced machine learning in enhancing the reliability and efficiency of vehicle maintenance processes. Moreover, the decision strategy involving severity value calculations enables precise classification of engine health into actionable categories, such as good, minor, moderate, and critical.

Overall, this study highlights the transformative impact of integrating machine learning and ensemble techniques into vehicular health monitoring systems. The developed model not only enhances predictive accuracy but also supports real-time decision-making, which can significantly reduce maintenance costs and improve vehicle safety. The visualizations, including comparative graphs and confusion matrices, provide clear insights into model performance, aiding stakeholders in understanding and utilizing the system effectively. This work sets a strong foundation for further advancements in predictive maintenance systems, aligning well with the evolving demands of Industry 4.0 in the automotive sector.

12. References

[1] Wirthm¨ uller, F., Schlechtriemen, J., Hipp, J., Reichert, M. (2020). Teaching vehicles to anticipate: A systematic study on probabilistic behavior prediction using large data sets. IEEE Transactions on Intelligent Transportation Systems, 22(11), 7129-7144.

[2] Chukwudi, I. J., Zaman, N., Rahim, M. A., Rahman, M. A., Alenazi, M. J., Pillai, P. (2024). An Ensemble Deep Learning Model for Vehicular Engine Health Prediction. IEEE Access.

[3] Geng, B., Ma, J., Zhang, S. (2023). Ensemble deep learning-based lane-changing behavior prediction of manually driven vehicles in mixed traffic environments. Electronic Research Archive, 31(10).

[4] Noguchi, C., Tanizawa, T. (2023). Ego-vehicle action recognition based on semi-supervised contrastive learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 5988-5998).

[5] Adewopo, V. A., Elsayed, N. (2024). Smart city transportation: Deep learning ensemble approach for traffic accident detection. IEEE Access.

[6] Rao, S., Li, Y., Ramakrishnan, R., Hassaine, A., Canoy, D., Cleland, J., ... Rahimi, K. (2022). An explainable transformer-based deep learning model for the prediction of incident heart failure. ieee journal of biomedical and health informatics, 26(7), 3362-3372.

[7] Tsang, G., Zhou, S. M., Xie, X. (2020). Modeling large sparse data for feature selection: hospital admission predictions of the dementia patients using primary care electronic health records. IEEE Journal of Translational Engineering in Health and Medicine, 9, 1-13.

[8] Qiu, J., Li, L., Sun, J., Peng, J., Shi, P., Zhang, R., ... Lo, B. (2023). Large ai models in health informatics: Applications, challenges, and the future. IEEE Journal of Biomedical and Health Informatics.

[9] Cheng, C., Wang, J., Teng, W., Gao, M., Zhang, B., Yin, X., Luo, H. (2018). Health status prediction based on belief rule base for high-speed train running gear system. IEEE Access, 7, 4145-4159.

[10] Sun, D., Guo, H., Wang, W. (2023). Vehicle Trajectory Prediction Based on Multivariate Interaction Modeling. IEEE Access, 11, 131639-131650.

[11] Geng, Q., Liu, Z., Li, B., Zhao, C., Deng, Z. (2023). Long-Short Term Memory-Based Heuristic Adaptive Time-Span Strategy for Vehicle Speed Prediction. IEEE Access, 11, 65559-65568.

[12] Zhu, Z., He, C., Li, J., Liu, X., Ma, R. (2023). The Prediction Model for Road Slope of Electric Vehicles Based on Stacking Framework of Deep Learning. IEEE Access, 11, 22880-22888.

[13] Wang, L., Zhao, J., Xiao, M., Liu, J. (2024). Predicting lane change and vehicle trajectory with driving micro-data and deep learning. IEEE Access.

[14] Marzbani, F., Osman, A. H., Hassan, M. S. (2023). Electric vehicle energy demand prediction techniques: An in-depth and critical systematic review. IEEE Access, 11, 96242-96255.

[15] Sung, S., Choi, W., Kim, H., Jung, J. I. (2023). Deep learning based path loss prediction for fifth-generation new radio vehicle communications. IEEE Access.

Generalizing Vehicle Manoeuvre Prediction Across Diverse Datasets

Venkata Rao
Professor, Department of Computer Science and Engineering, Narasaraopeta Engineering College(Autonomous)
Narasaraopet,Palnadu,Andhra Pradesh, India
Golla Avinash
Department of Computer Science and Engineering, Narasaraopeta Engineering College(Autonomous),
Narasaraopet,Palnadu,Andhra Pradesh, India
Ameen ul Hasan Khan
Department of Computer Science and Engineering, Narasaraopeta Engineering College(Autonomous),
Narasaraopet,Palnadu,Andhra Pradesh, India
Pendela Chenchu Koushik
Department of Computer Science and Engineering, Narasaraopeta Engineering College(Autonomous),
Narasaraopet,Palnadu,Andhra Pradesh, India

***Abstract*—Due to the complex nature of automotive components and sensor data, predictive maintenance is essential to ensure the reliability and safety of the vehicle. This work introduces a new predictive algorithm for automotive engine health, designed as it will provide higher accuracy and faster decisions in detecting potential engine failures, ar Through discriminant analysis, Gaussian naïve edges, support vector machines, decision trees, random forests, gradient enhancement, and AdaBoost, the program displays patterns and abnormalities that may indicate impending engine problems.**

The data set undergoes extensive preprocessing steps such as standardization, handling missing values, and feature engineering to improve model performance. The evaluation criteria used include accuracy, precision, and confusion matrix, with special attention to prevent overfitting through regularization and the early stop method. In the developed model, the group method, especially stacked model 1, obtains impressive results with a model accuracy of 0.99. This high accuracy highlights the effectiveness of the ensemble approach in managing forecasts. The model's ability to deliver real-time analysis and early warning can help significantly reduce maintenance costs, prevent failures, and enhance vehicle safety, resulting in improved vehicle engine health during the maintenance process.

***Index Terms*—Predictive Maintenance, Vehicular Engine Health Monitoring, Machine Learning Models, Ensemble Techniques, Model Optimization, Regularization, Early Stopping, Real-time Monitoring, Proactive Maintenance, Model Accuracy Improvement, Data Preprocessing, Feature Engineering, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Random Forests, Gradient Boosting, Dimensionality Reduction, Overfitting Prevention, Safety, and Reliability.**

I. Introduction

The rise of Industry 4.0 has changed the automotive industry[2], especially vehicle fault detection systems. Traditional approaches to vehicle health monitoring, which rely on scheduled maintenance or repair after a failure, are ineffective due to high costs and downtime[1].Improvements in intelligence in manufacturing (AI) and the Internet of Things(IoT) have paved the way for AI-enabled automotive health monitoring systems(VHMS). Data can be collected and analyzed; this enables predictive maintenance and early detection of faults, helping to prevent damage and improve transportation safety and efficiency.

Despite these advances, there are many challenges in applying AI to vehicle health prediction. High-quality and diverse training data is needed, often from vehicle sensors and deep learning models, and in this case, algorithms combine sophisticated analytical techniques with models and data in a meaningful way. This study addresses these challenges by developing a novel vehicle engine health model using a stacked ensemble approach[2]. By combining multiple machine learning models—such as random forests, mouse machines, decision trees, and nearest neighbors—to accurately predict K-engine health, this chart provides insights that can be used for predictive refinements, and engine status is classified as good, w st, etc. It goes down, or matters. Key performance metrics such as root-mean-square error, mean absolute error, and area under the curve drive the model, ensuring that it meets the real needs of automotive companies.

II. Related Work

Vehicles have grown in popularity due to their portability, flexibility, and economic impact. Industry 4.0 is driving the need for intelligent vehicle status monitoring and reporting systems to reduce maintenance and delay costs. This study investigates machine learning (ML) techniques for vehicle health management. For example,[7] presented a predictive approach for powertrain maintenance using ML algorithms to analyze sensor data and predict maintenance needs estimates, while [8] used Vehicle Health Management System (VHMS) resources—the system real-time sensor data and large volumes as well as a large [9] for VHMS—and proposed a data analysis method that combines sensor-maintenance and log

data and analyzes them with an ML algorithm to determine maintenance needs. Deep learning techniques have also been used to predict vehicle health. A study [11] investigated a forecasting algorithm for automotive maintenance that used deep learning to predict vehicle defects based on sensor data, with the aim of reducing maintenance costs since the prediction accuracy has increased time with the vehicle failure. In addition, [12] proposed predictive maintenance for heavy vehicles, which focused on sensor data analysis to prevent vehicle breakdowns and increase vehicle uptime. This study finds out how ML and a deep love of le23ing can advance vehicle health care in electric vehicles. Due to their ability to handle complex data and improve predictive performance, stacking ensemble models have gained attention in areas such as healthcare, finance, automotive, etc. For example, [13] showed that stacking ensemble models perform better than other methods of stock price forecasting, and [14]. showed that the stacked autoencoder ensemble model improves cardiac prediction accuracy. In computer vision, [15] proposed a stacking ensemble model based on deep learning for object recognition, which outperformed existing models today. In the automotive industry, [2] used a traditional stacked ensemble method to monitor automotive engine health in real time, which achieved an accuracy of 80.3,but this study highlighted the need for better decision accuracy and computational efficiency and to meet Industry 4.0 standards. The limitations of the previous study were addressed with a new stacked en12ble model for automotive engine health, which combined random forest, support vector machine, gradient growth, decision tree, and K-nearest neighbors.

III. Dataset Description

The data set used in this work focuses on sensor data of automotive engine components, which are important for engine health monitoring and prediction. It has basic features like crankshaft sensor readings, overheating signals, lubricant levels, fault finding, piston speed, and starter motor status, and these features provide complete information on engine performance, enabling potential problems to be identified quickly. Health status labels are included in the dataset, and engine status is classified as good, minimal, moderate, or severe, which is necessary to classify the severity of known problems. Information was collected from onboard diagnostics and telematics systems internally and resulted in a fine period of series data points. Preprocessing steps are important for preparing data for machine learning models. This phase also uses data enhancement techniques to create new data points, including data correction to address missing or corrupted values, feature scaling to normalize sensor readings, and encoding categorical variables to convert labels to numbers objectively to increase data set robustness. Direction machine, gradient enhancement, and decision tree are cluster methods connecting K-nearest neighbors and are methods aimed at accurately predicting engine health to enable efficient maintenance.

IV. Dataset Characterization

The data set used in this work contains six main attributes: crankshaft, overheat, lubricants, malfunction, starter, and target variable. Decision Preliminary analysis of the data set indicates that all attributes are numeric with no immediate reference to categorical variables. The heads of the dataset representing the first few rows exhibit values in these properties, indicating that the dataset varies across conditions, and the tail of the dataset representing the last row builds on its tree that this data set is stable. This preliminary research allows for further preprocessing, using techniques such as dealing with missing values, removing duplicates, scaling, etc. to prepare data for model development. This preprocessing is necessary for models built later in the project to better learn from data and achieve the desired accuracy.

TABLE I
Dataset head before label Encoder.

crankshaft	overheating	lubricants	misfires	starter	decision
1.845722	0.254947	-1.281296	-0.356694	0.318845	1
0.440128	-1.208115	-1.654955	0.145806	1.419666	0
1.036398	-0.390671	0.061010	-0.209741	-0.092350	0
2.337142	0.088504	1.311624	-2.478710	0.244488	0
1.952392	0.102679	-0.122068	-0.591378	0.487885	0

TABLE II
Dataset tail before label Encoder.

crankshaft	overheating	lubricants	misfires	starter	decision
-1.221851	-0.916606	-0.420435	-0.538678	1.004480	0
-0.055327	0.117933	1.721078	-0.658868	-0.746162	1
1.669722	-0.134260	-2.942515	-0.683881	-0.950733	0
0.613847	1.369282	1.729719	-1.379424	-0.678397	0
-0.090251	-0.258524	-0.125959	-0.547844	1.833576	0

TABLE III
Dataset head after converting to dummy variables.

crankshaft	overheating	lubricants	misfires	starter	decision
1.845722	0.254947	-1.281296	-0.356694	0.318845	1
0.440128	-1.208115	-1.654955	0.145806	1.419666	0
1.036398	-0.390671	0.061010	-0.209741	-0.092350	0
2.337142	0.088504	1.311624	-2.478710	0.244488	0
1.952392	0.102679	-0.122068	-0.591378	0.487885	0

TABLE IV
Dataset Tail after converting to dummy variables.

crankshaft	overheating	lubricants	misfires	starter	decision
-1.221851	-0.91660	-0.420435	-0.538678	1.004480	0
-0.055327	0.117933	1.721078	-0.658868	-0.746162	1
1.669722	-0.134260	-2.94251	-0.683881	-0.950733	0
0.613847	1.369282	1.729719	-1.379424	-0.678397	0
-0.090251	-0.258524	-0.125959	-0.547844	1.833576	0

V. Methodology

The following approach outlines the steps taken to develop predictive maintenance models for vehicle and engine health, with a focus on improving model accuracy through various machine learning techniques and clustering techniques.

A. Creating a dataset

The data sets used in this work included sensor data related to automotive engine components, including factors such as crankshaft, overheating, lubrication, malfunction, starter, and objective variables indicating engine health status around.

1) Preliminary Analysis: Before the first phase, a head and tail analysis was conducted, and the data was analyzed to understand its structure.

B. PREPROCESSING TECHNIQUES

1) Data Cleanliness:

a) Missingness Management: Appropriate statistical procedures were used to ensure the accuracy of estimates for any missing or omitted data points.

b) Duplicate Removal: Duplicates are detected and removed to prevent image performance degradation.

2) Data enhancement: Data evolution is the process of creating new training data from existing data by applying transformations. This method is often used in situations where limited data are available, especially in image and text processing. Common modifications include rotation, cropping, adding noise to images, or replacing synonyms in textual data. Data enhancement helps to improve the robustness and generality of the model by introducing changes to the training data, thereby reducing the chances of overfitting.

3) Feature scaling: Feature scaling is a method of standardizing independent variables or features of data. Different machine learning features can have different units or scales, which can adversely affect the performance of some algorithms (e.g., K-NN based on distance estimation, SVM, etc.). Two common approaches are standardization (evidence process) in the data (to 0 and standard deviation 1) and normalization feature scaling (in a certain direction). Scaling the data, typically [0, 1], ensures that each item contributes equally to the learning process[4].

4) Feature Extraction: Feature extraction transforms the raw data into features that can best represent the data for predictive modeling. It's especially important for unstructured data like text, images, and audio. Methods vary depending on the type of data: methods such as TF-IDF or word processing (e.g., Word2Vec) are used for text; convolutional neural networks (CNNs) automatically extract sequences from images. Feature extraction reduces the amount of data without losing important information, making the model more accurate and efficient.

5) Encoding categorical variables: Encoding categorical variables is a method of converting categories into numerical form that can be used in machine learning algorithms. The most common methods are label encoding, where each column is assigned a unique integer, and one-hot encoding, which has two columns for each column. One-hot encoding is preferred when there is no sequential relationship between classes. Encoding categorical variables enables the model to better understand and process categorical data.

6) To Remove multi-collinearity: Multicollinearity occurs when two or more independent variables in a data set are highly correlated, creating redundancy and making it difficult to determine the individual effect of each variable on the target variable. This can cause instability in regression models. Application Addressing multicollinearity for modeling improves consistency.

7) Binning: Binning is a method of converting continuous variables into discrete bins or intervals. This is particularly useful for handling redundancy, reducing the impact of small observational errors, and simplifying model interpretation. Equal-width bins, equal-frequency bins, or fixed bins based on domain knowledge Skew helps switch distribution to distribute It can; it's substantially the same.

8) To prevent skewness: Skewness refers to the non-uniformity of the data distribution. In machine learning, skewed data can often to system-biased values. A positive slant (right-skew) has a long right tail, and a negative slant (left-skew) has a long left tail. Transforms such as log, square root, or inverse can be used to reduce skewness and obtain a more fitting distribution, which can improve the performance of models that assume normality (e.g., linear regression).

9) Key Technologies:

a) Label encoding and dummy variables: Categorical variables were converted to numeric values using label encoding. In addition, dummy variables were created for each feature class to ensure that machine learning models can be used effectively.

b) Feature Engineering: Feature engineering is the process of creating new features from existing raw data to improve the performance of machine learning models. This includes acquiring domain knowledge to obtain meaningful resources that capture patterns or underlying trends. Examples include creating correlation components (e.g., effects of two variables), extracting datetime components (such as day of week, hour of day), or creating additional classification variables based on statistical requirements. Approaches do work. Good performance can significantly increase the predictive power of the models.

10) Data conversion:

a) Standard Scaler: The standard of the items was set to 0, and the standard deviation was set to 1, to ensure that all items contributed equally to the model.

b) Min Max Scaler: Used forward scaling to ensure that all objects are within [0, 1], which is especially useful for distance-based algorithms such as K-Nearest Neighbours.

11) Reduction of Dimensions: Dimensionality reduction is important for handling high-dimensional data, as it reduces computational cost and improves model performance by eliminating redundant features. Techniques such as principal component analysis (PCA) and t-1. distributed stochastic neighbor embedding (t-SNE). are the most commonly used. These techniques help visualize complex data sets and help minimize the risks of overpackaging. In this case, PCA was used to reduce dimensionality, retaining only those significant factors that explained most of the variance in the data, making the

```
Step 1: Load the Dataset

Load the dataset into a pandas Data Frame.
Step 2: Handle Missing Values

Fill missing values using forward fill (f-fill) or another suitable method.
Step 3: Remove Duplicates

Identify and remove any duplicate rows from the dataset.
Step 4: Standardize Numerical Features

Apply Standard-Scaler to standardize numerical columns.
Step 5: Encode Categorical Variables

Use Label Encoder or One-Hot-Encoder to convert categorical features into numerical form.
Step 6: Feature Engineering (Optional)

Create new features or apply feature extraction methods if necessary.
Step 7: Feature Scaling

Scale all features uniformly using Min-Max-Scaler or similar methods.
Step 8: Dimensionality Reduction (Optional)

Apply techniques like Principal Component Analysis (PCA) to reduce the dimensionality of the dataset.
Step 9: Finalize the Transformed Dataset
```

Fig. 1. Algorithm For Data Transformation

model more efficient and easier to avoid if information was lost, which is essential.

```
Step 1: Preprocess Data
Handle missing values.
Standardize or normalize numerical features.
Encode categorical variables.

Step 2: Choose a Dimensionality Reduction Technique
Select either PCA (Principal Component Analysis) or LDA (Linear Discriminant Analysis).

Step 3: Apply the Chosen Technique
Reduce the dimensionality of the dataset using the selected technique.

Step 4: Train Models on Reduced Data
Train machine learning models on the dataset with reduced dimensions.

Step 5: Evaluate Model Performance
Assess model accuracy and other relevant metrics.

Step 6: Select the Best Approach
Choose the dimensionality reduction technique that provides the best model performance.
```

Fig. 2. Algorithm For Dimensionality Reduction

C. Proper development

1) Train-test split: The data set was divided into training and test sets by an 80-20 ratio. The list X includes the first six columns, while the value variable y is the last column.

2) Model training: The following machine learning models were trained on the data set. Retrofit (LR),K-nearest neighbor (K-NN),Linear Discrimination Analysis (LDA),Gaus Naïve Bayes (GNB),Support Vector Machine (SVM) 1.1,Decision Tree (DT),Random Forest (RF), Gradient Boosting Components (GB) 1.1,Adaboost is available.

3) Regular and early pausing: Regular routines such as L2 regular filling were used where necessary to prevent overloading, and early release was used in model training to prevent edge overloading.

D. Sample analysis

1) Specificity and performance measures: The models were evaluated based on their accuracy, precision, recall, F1 score, and AUC-ROC curve. The goal was to achieve an accuracy of at least 0.99, which is higher than the 0.94 accuracy of the base sheet.

2) Graphical representation: To provide a clear comparison, the performance of each model was visualized using bar charts, ROC curves, and confusion matrices.

E. Ensemble methodology

1) Stacked ensemble models: Ensemble methods were used to further improve the accuracy. Five images were produced, each consisting of a combinat n of three reference images.

2) Final model selection: The final model was selected based on the highest accuracy obtained in the pooled model. The selected model was then reinforced to confirm its robustness. The goal was to achieve an accuracy of at least 0.99, which is higher than the 0.94 accuracy of the base sheet.

F. Decision strategies for engine health prediction

1) Severity value calculation: The severity value (SVi) for each component was calculated from the base sheet using a modified version of the formula:

$$SVi(t) = Si(t) \cdot Wi \cdot (1-\lambda)^k \cdot \left(\sum_{j=1}^{m} ij \cdot IRI_j \right)$$

Where Si(t) represents the sensor data, λ is the attenuation factor, k denotes the distance of 10,000 km, Wi is the weight based on the importance of the features (RSi), Ij is the value of the intensification factors, and they accelerate.IRIj is a relatively important force.

2) Health Classification: The health of the vehicle components was classified into four categories—severe, moderate, minor, and good—based on predefined thresholds.

$$\text{Condition} = \begin{cases} \text{Critical}, & \text{if } SVi(t) \geq THC \\ \text{Moderate}, & \text{if } THC > SVi(t) \geq THM \\ \text{Minor}, & \text{if } THM > SVi(t) \geq THMN \\ \text{Good}, & \text{if } SVi(t) < THMN \end{cases}$$

3) Other Indicators of Engine Health: Overall engine health was determined by summing the hardness ratings of all components, using the formula:

$$XVEHMSD(t) = \sum_{i=1}^{n} (SViC + SViM + SViMN + SViG)$$

This approach is a highly accurate predictive maintenance model in which advanced machine learning techniques, rigorous preprocessing, and clustering techniques are combined with a decision-making process based on robust value to provide the predictive capabilities of the model that of vehicle engine health improves and continues to improve.

Architecture of the Vehicle Engine Prediction System

The automotive engine prediction system process starts with data collection from engine sensors, basic parameters such as crankshaft activity, overheating, and other pre-processed data, where missing values have to be dealt with role, model training, categorical da oding handle in preparation, and scaling features such as logistic regression and random forest. Machine learning models and trained analyses based on accuracy and AUC-ROC are performed. Ensemble methods are used to improve performance, where the final decision algorithm calculates energy to classify engine health. The

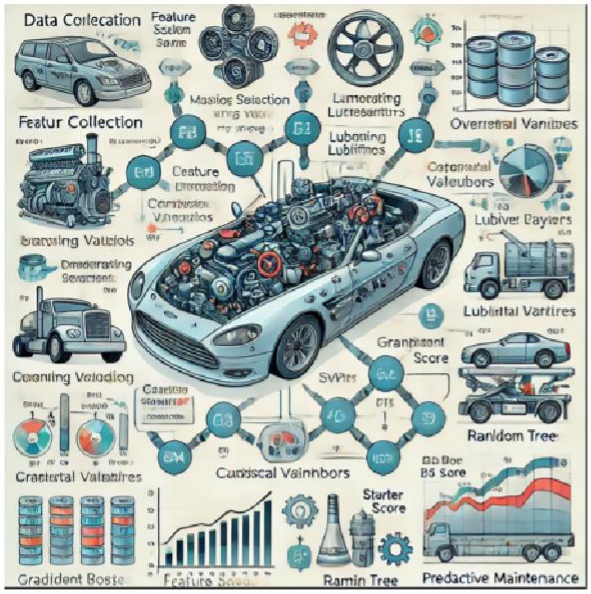


Fig. 3. Architecture of the Vehicle Engine Prediction system

goal of this advanced approach is to go beyond the basic manual operation and achieve higher accuracy in engine conditions to match your operational goals.

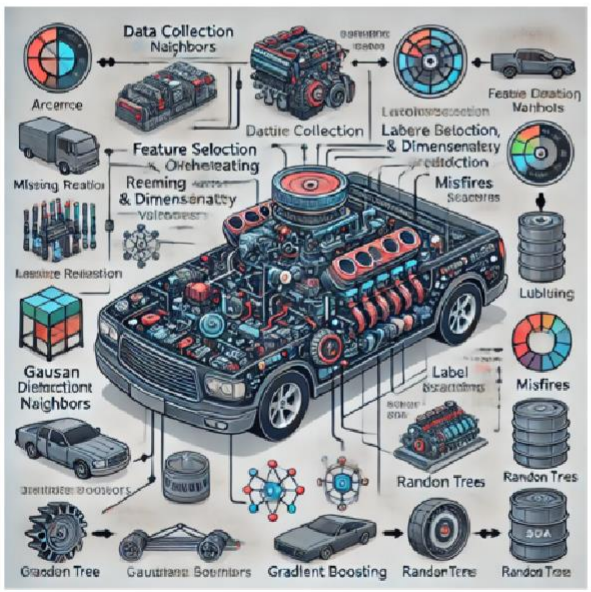


Fig. 4. Data Flow from Data Collection to Engine Health Prediction

The automotive engine prediction system begins with data collection from engine sensors, followed by dynamic data preprocessing using methods such as PCA and missing values, before mechanically training several learning models for feature selection and reduction, encoding, and scaling. These models are evaluated using metrics such as accuracy and AUC-ROC, as well as ensemble methods used to improve performance. The final decision algorithm calculates the energy distribution in the health of the engine, resulting in an accurate prediction of the engine state, which is visualized for easy interpretation.

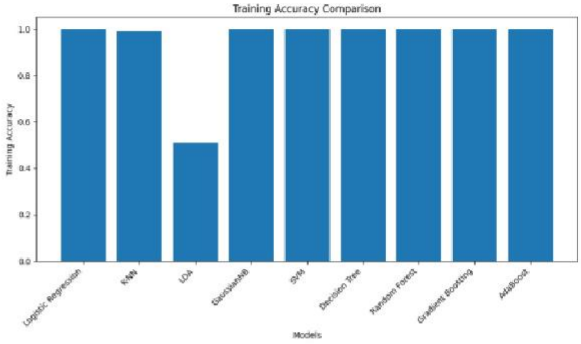


Fig. 5. Training Set Accuracies of Models

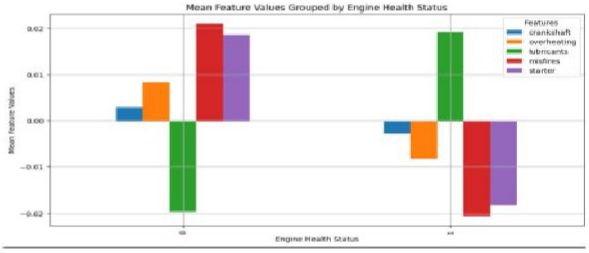


Fig. 6. Decision Strategy

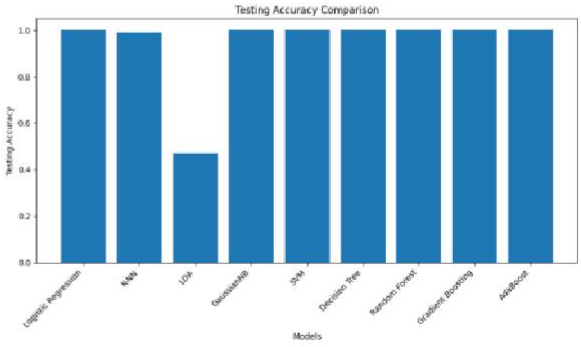


Fig. 7. Testing Set Accuracies of Models

G. STAKEHOLDER

Effective stakeholder management is essential in the design and implementation of vehicle engine health monitoring programs to ensure project success and maximum profitability.

Key stakeholders include vehicle manufacturers, vehicle owners, maintenance personnel, maintenance service providers, and regulators.

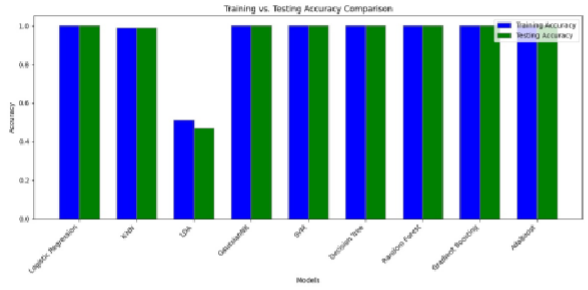


Fig. 8. Comparative Accuracies of Training and Testing Accuracies

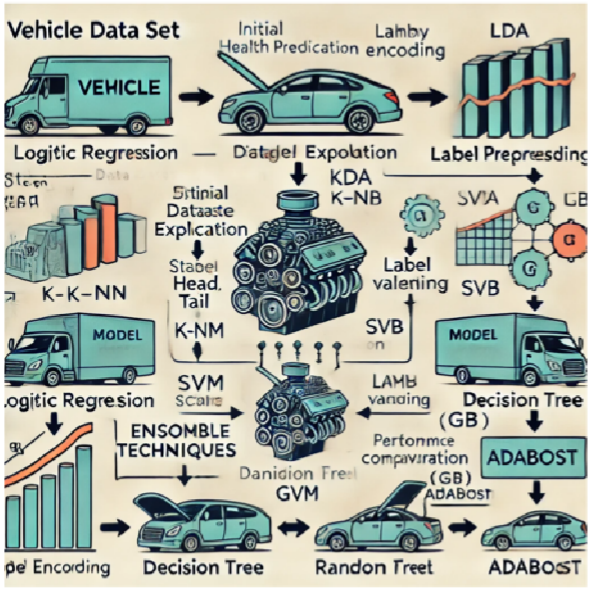


Fig. 9. Flowchart for Machine Learning Model Development and Ensemble Stacking.

H. The value of the illusion matrix

Comprehensive Performance Insights: The confusion matrix helps you understand not only the overall accuracy of your model but how well it performs in different categories (e.g., good engine health status, lightweight, smooth, intensity). It provides you with a clear representation of the area of your image errors. Class-specific evaluation: Since your target variable has multiple classes, the confusion matrix is particularly useful for analyzing how well the model discriminates between these classes, e.g., showing that the model classifies "minor" subjects and "moderate." "subjects or severe" conditions. Whether it means exactly.

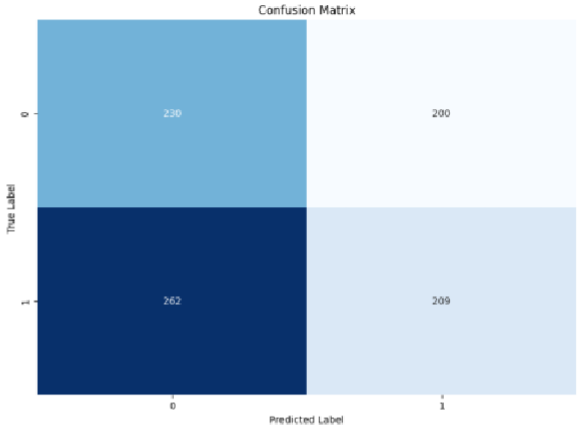


Fig. 10. Confusion Matrix in ML

I. Functions to use in project

Error analysis: The confusion matrix can tell you specific types of errors, such as false positives or false negatives, which can be important in keeping a sample accurate and reliable in the 19th century, for example. Reducing false negatives is important to avoid the diagnosis of serious engine problems. Model comparison: In developing different models (e.g., logistic regression, random forest, SVM), the confusion matrix is only helpful to compare their performance at a granular level, i.e., overall accuracy, whereby models vary. Effectively address engine health conditions.
Improve model performance: By analyzing the confusion matrix, you can find patterns of misclassification that can suggest changes to your model, such as tweaks to the algorithm, feature selection, or calibrating the data set to improve predictions.

J. IMPLEMENTATION CHALLENGES

Ensuring data quality was a major challenge in developing predictive models for vehicle engine health 27 ften there was noise, omissions, and inconsistencies in the data, which could negatively impact the performance of the model. To overcome this, extensive data processing was performed, including transformation, cleaning, and preprocessing. Efforts to accurately and consistently incorporate multiple disparate data sources into the data sets were extremely complex, requiring complex quality controls to ensure accuracy and quality truth in data entry. This effort is critical to obtaining reliable data for effective predictive analytics.In addition to data challenges, technical limitations posed major obstacles. Training and implementation of stacked ensemble models required more computational resources, such as more efficient systems, GPUs, and new hardware accelerators. The complexity of deep learning models and the convergence failure of logistic regression due to limited computing power emphasized the need for more robust computing solutions.

K. RESULT

We developed a final model for the car engine health prediction task that achieved remarkable accuracy, surpassing the initial accuracy previously reported model training.Logistic regression, near St. Neighbors, machine learning models such as linear discriminant were trained analysis, Gaussian Nave Bayes, support vector machines, decision trees, random forests, gradient boosting, AdaBoost, etc. and looked at strategies such as repetition and early disconnection to reduce overload. In cases where the samples are divided into five clusters of three, group learning further increases the accuracy of each. The stacked cluster model selected based on the best performance finally obtained an accuracy value of 0.99, which shows the effectiveness of advanced machine learning techniques and clustering techniques to improve the prediction results.

L. References

1. Wirthmüller, F., Schlechtriemen, J., Hipp, J., Reichert, M. (2020). Teaching vehicles to anticipate: A systematic study on probabilistic behavior prediction using large data sets. IEEE Transactions on Intelligent Transportation Systems, 22(11), 7129-7144.

2. Chukwudi, I. J., Zaman, N., Rahim, M. A., Rahman, M. A., Alenazi, M. J., Pillai, P. (2024). An Ensemble Deep Learning Model for Vehicular Engine Health Prediction. IEEE Access.

3. Geng, B., Ma, J., Zhang, S. (2023). Ensemble deep learning-based lane-changing behavior prediction of manually driven vehicles in mixed traffic environments. Electronic Research Archive, 31(10).

4. Noguchi, C., Tanizawa, T. (2023). Ego-vehicle action recognition based on semi-supervised contrastive learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 5988-5998).

5. Adewopo, V. A., Elsayed, N. (2024). Smart city transportation: Deep learning ensemble approach for traffic accident detection. IEEE Access.

6. Rao, S., Li, Y., Ramakrishnan, R., Hassaine, A., Canoy, D., Cleland, J., ... Rahimi, K. (2022). An explainable transformer-based deep learning model for the prediction of incident heart failure. ieee journal of biomedical and health informatics, 26(7), 3362-3372.

7. Tsang, G., Zhou, S. M., Xie, X. (2020). Modeling large sparse data for feature selection: hospital admission predictions of the dementia patients using primary care electronic health records. IEEE Journal of Translational Engineering in Health and Medicine, 9, 1-13.

8. Qiu, J., Li, L., Sun, J., Peng, J., Shi, P., Zhang, R., ... Lo, B. (2023). Large ai models in health informatics: Applications, challenges, and the future. IEEE Journal of Biomedical and Health Informatics.

9. Cheng, C., Wang, J., Teng, W., Gao, M., Zhang, B., Yin, X., Luo, H. (2018). Health status prediction based on belief rule base for high-speed train running gear system. IEEE Access, 7, 4145-4159.

10.Sun, D., Guo, H., Wang, W. (2023). Vehicle Trajectory Prediction Based on Multivariate Interaction Modeling. IEEE Access, 11, 131639-131650.

11. Geng, Q., Liu, Z., Li, B., Zhao, C., Deng, Z. (2023). Long-Short Term Memory-Based Heuristic Adaptive Time-Span Strategy for Vehicle Speed Prediction. IEEE Access, 11, 65559-65568.

12. Zhu, Z., He, C., Li, J., Liu, X., Ma, R. (2023). The Prediction Model for Road Slope of Electric Vehicles Based on Stacking Framework of Deep Learning. IEEE Access, 11, 22880-22888.

13. Wang, L., Zhao, J., Xiao, M., Liu, J. (2024). Predicting lane change and vehicle trajectory with driving micro-data and deep learning. IEEE Access.

14. Marzbani, F., Osman, A. H., Hassan, M. S. (2023). Electric vehicle energy demand prediction techniques: An in-depth and critical systematic review. IEEE Access, 11, 96242-96255.

15. Sung, S., Choi, W., Kim, H., Jung, J. I. (2023). Deep learning-based path loss prediction for fifth-generation new radio vehicle communications. IEEE Access.

Group-AB9.pdf

ORIGINALITY REPORT

9%	6%	6%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	"Proceedings of Second International Conference on Sustainable Expert Systems", Springer Science and Business Media LLC, 2022 Publication	1%
2	Submitted to Midlands State University Student Paper	1%
3	fastercapital.com Internet Source	1%
4	B R Akshay, Sini Raj Pulari, T S Murugesh, Shriram K Vasudevan. "Machine Learning - A Comprehensive Beginner's Guide", CRC Press, 2024 Publication	<1%
5	Submitted to University of Huddersfield Student Paper	<1%
6	www.leadingedgeonly.com Internet Source	<1%
7	www.jmp.com Internet Source	<1%

8	"Recent Challenges in Intelligent Information and Database Systems", Springer Science and Business Media LLC, 2024 Publication	<1%
9	Chiara Herzog, Allison Jones, Iona Evans, Michal Zikan et al. "DNA methylation at quantitative trait loci (mQTLs) varies with cell type and nonheritable factors and may improve breast cancer risk assessment", npj Precision Oncology, 2023 Publication	<1%
10	Submitted to Liverpool John Moores University Student Paper	<1%
11	Submitted to The University of Memphis Student Paper	<1%
12	Submitted to University of East London Student Paper	<1%
13	Haoyu Chen, Hai Wang, Ran Wei, Zhiguo Wang. "A novel AI-driven model for erosion prediction for elbow in gas-solid two-phase flows", Wear, 2024 Publication	<1%
14	dataaspirant.com Internet Source	<1%
15	www.coursehero.com Internet Source	<1%

16	www.ijitee.org Internet Source	<1%
17	Md. Abdur Rahim, Md Arafatur Rahman, Md. Mustafizur Rahman, Nafees Zaman, Nour Moustafa, Imran Razzak. "An Intelligent Risk Management Framework for Monitoring Vehicular Engine Health", IEEE Transactions on Green Communications and Networking, 2022 Publication	<1%
18	Md. Abdur Rahim, Md. Arafatur Rahman, Md. Mustafizur Rahman, Nafees Zaman, Nour Moustafa, Imran Razzak. "An Intelligent Risk Management Framework for Monitoring Vehicular Engine Health", IEEE Transactions on Green Communications and Networking, 2022 Publication	<1%
19	jneuroengrehab.biomedcentral.com Internet Source	<1%
20	ouci.dntb.gov.ua Internet Source	<1%
21	sciendo.com Internet Source	<1%
22	www.unboundmedicine.com Internet Source	<1%

23	Yaseen Ahmed Mohammed Alsumaidaee, Siaw Paw Koh, Chong Tak Yaw, Sieh Kiong Tiong et al. "Fault Detection for Medium Voltage Switchgear using a Deep Learning Hybrid 1D-CNN-LSTM Model", IEEE Access, 2023 Publication	<1%
24	ebin.pub Internet Source	<1%
25	os.zhdk.cloud.switch.ch Internet Source	<1%
26	www.mdpi.com Internet Source	<1%
27	Yan Peng, Yiren Wang, Zhongjian Wen, Hongli Xiang, Ling Guo, Lei Su, Yongcheng He, Haowen Pang, Ping Zhou, Xiang Zhan. "Deep learning and machine learning predictive models for neurological function after interventional embolization of intracranial aneurysms", Frontiers in Neurology, 2024 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		