International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(1s) 2026 pp. 35-49

https://doi.org/10.70917/ijcisim-2026-1938
Article

ANALYZING PERFORMANCE OF SMART PREDICTIVE MAINTENANCE SYSTEMS IN INDIAN FARMING AUTOMOBILE THROUGH EXPERIMENTATION

Snehal Patil[1], Nilesh P. Sable[2], Parikshit N. Mahalle[3]

[1] Research Scholar, Department of Computer Engineering, BRACT's Vishwakarma Institute of Information Technology, Pune, Maharashtra, India., engg.snehalpatil@gmail.com
[2] Department of Computer Science & Engineering (Artificial Intelligence), Bansilal Ramnath Agarwal Charitable Trust's Vishwakarma Institute of Information Technology, Pune, Maharashtra, India.,drsablenilesh@gmail.com
[3] Department of Artificial Intelligence and Data Science, Bansilal Ramnath Agarwal Charitable Trust's Vishwakarma Institute of Information Technology, Savitribai Phule Pune University (SPPU), Pune, Maharashtra, India., aalborg.pnm@gmail.com

Abstract: Agricultural sector in India is highly significant to the economy of the nation and transport and machineries systems that facilitate the sector are necessary in ensuring the farming is efficient and productive. However, upkeep issues usually lead to unexpected downtimes, which negatively affect the capacity of farms to earn and to be productive. This study aims at investigating the effectiveness of intelligent predictive repair systems in the field vehicles in Indian agricultural conditions. It is a controlled test that is conducted in a sequence of experiments to investigate the effectiveness of predictive maintenance techniques in identifying issues prior to their occurrence and ensuring timely repairs. The sensors, IoT devices, and machine learning methods are used in this approach to monitor the health of farm machinery such as a tractor, plough, and harvester. In the beginning of the work, the Internet of Things (IoT) sensors are installed on various farming vehicles to gather real-time data about their work, including temperature, shaking, and engine performance. The information is forwarded to a cloud-based solution and analysed. Machine learning algorithms, including regression and classification models, are used to predictive models that establish what can go wrong and what maintenance is to be done. The experiment design evaluates the accuracy, reliability, and performance of these smart predictive maintenance systems in reducing the downtime and utilizing work plans optimally. Besides this, the paper also examines how these systems can be expanded and whether it can be engaged in rural areas with limited technology. According to the results, predictive maintenance systems can reduce the unexpected downtimes and repair costs significantly. This will make work more effective and enhance the overall performance of farm machines.
Keywords: Smart predictive maintenance; Indian farming; agricultural automobiles; IoT; Machine learning; rural technology

1. Introduction

The agriculture sector in India continues to be one of the pillars of the national economy serving as the means of livelihood to a good percentage of the population as well as food security and development of rural areas. The industry has experienced high levels of mechanization as more farming machineries like tractors, harvesters and irrigation vehicles are becoming increasingly popular to augment output, efficiency in working and sustainability. These machines make them less labor-reliant, increase accuracy in agricultural practices, and allow them to conduct agriculture in a timely manner. Nevertheless, even though technological development in agricultural vehicles has brought significant issues, the maintenance and reliability remain significant problems, especially in rural areas where there is little access to qualified technicians and spare parts. Unforeseen equipment malfunctions usually result in longer downtimes, higher repair expenses, and postponements in vital farming processes, which directly influence the crop production and earnings of the farmers (Nagy & Lakatos, 2024).Maintenance practices that are common in agricultural settings include reactive maintenance and periodic preventive maintenance which are traditional methods of maintenance. Whereas reactive maintenance creates unplanned failure and wastage of production, time-based preventive maintenance usually causes unnecessary services and wastes of resources. These constraints underscore the fact that there is a need to develop smart forms of maintenance that can predict the occurrence of a failure before it sets in. Predictive maintenance (PdM) is a solution to this problem based on sensor data, past maintenance history, and machine learning models to forecast the health of equipment and anticipate possible failures before they happen (Arena et al., 2022).

The current AIs and machine learning innovations have greatly increased the efficiency of predictive maintenance systems in any industrial setting. PdM frameworks that operate on data have exhibited significant reliability, cost reduction, and asset utilization in the automotive industry (Theissler et al., 2021). The same benefits were cited in the field of vehicle fleet management, where the frequency of breakdowns decreases and the maintenance planning is more efficient as the AI-based predictive model results in benefits (Eswarararaj et al., 2025). Nevertheless, it is assumed that the system has not been adopted in Indian agricultural automobiles yet based on the infrastructures, initial investment is high and to a certain extent, small-scale farmers are not aware of the system (Mahale et al., 2025).

The combination of the Internet of Things (IoT) and predictive maintenance has allowed incessant monitoring of the parameters of machines including vibration, temperature, pressure, and fluid quality. The linked machines can send real-time operational information to cloud platforms to support advanced analytics by being fitted with sensors and telematics units (Pech et al., 2021). It has been demonstrated that telematics-based systems can improve vehicle diagnostics, remote monitoring, and maintenance decision-making in commercial and agricultural vehicles (Bethaz et al., 2021). The patterns of subtle degradation which are hard to detect by explicit inspection can be identified by machine learning models that are used on this data (Rasheed et al., 2024). Predictive maintenance through experimental studies has also proved the viability of vibration and condition-monitoring data in predicting early fault and asset health (Amihai et al., 2018). Also, smart maintenance systems based on IoTs allow remote diagnostics and predictive service planning, minimizing downtime and operational interruptions (Ayad et al., 2018). Low-cost onboard diagnostic systems and usage data analysis are especially useful in the context of agriculture, where they allow determining the machine wear and fuel efficiency (Rykała et al., 2023).

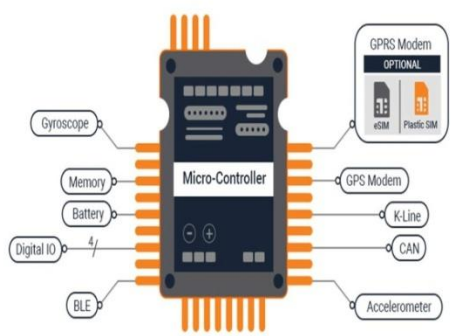

Figure 1: Telematics Control Unit Architecture for Connected Farming Automobiles

Although it has been proven successful in manufacturing and transportation, the operation of smart predictive maintenance systems in Indian farming automobiles has not been adequately studied using experimental validation. To fill this research gap, the current paper experimentally examines the performance of predictive maintenance systems involving IoTs and machine learning in realistic agricultural operating environments. The research estimates the fault detection, maintenance, and the feasibility of the system and takes into account the realistic limitations of the rural farming setting in India, thus making a contribution to intelligent and sustainable agricultural mechanization. Figure 1 below shows a telematics control unit that consists of sensors, CAN/K-Line interfaces, GPS, GPRS, and microcontroller, which can be used to collect, process, and transmit real-time vehicle health, vehicle location, and operational information used in predictive maintenance.

2. Related Work

Smart predictive maintenance has recently been extended into the automotive, manufacturing and cyber-physical system domains, delivering a basis of knowledge applicable to the agricultural automobiles. The initial studies conducted by Aeddula et al. (2024) have shown the efficiency of AI-initiated predictive maintenance schemes to self-driving vehicles in the spheres of product-service system advancement. Their article emphasized the benefit of sensor-based diagnostics and machine learning to enhance the availability of systems and lifecycle maintenance, which can be transferable to farm vehicles that work under changing loads and environmental factors.

Dash et al. (2021) examined predictive maintenance using deep learning in cars with a specific focus on neural networks to identify intricate failure behavior based on multivariate sensor data. Their results verified that the data-driven models are more effective than the rule-based diagnostics which is especially applicable to agricultural equipment with non-linear wear and usage variability. In addition to this, Mousaei et al. (2024) conducted a review of machine learning techniques in state-of-charge management in electric vehicles and demonstrated how predictive analytics can be used to improve the reliability and planning of operations in energy-limited vehicles.

In a more systemic context, Benhanifia et al. (2025) have conducted a systematic review of predictive maintenance use in manufacturing, which showed that experimentation-based validation is an important gap in research. Their discussion underlined the importance of practical testing to determine the strength of the models and this is also a concern in farming robots that are taken into the severe rural operating environment. Gupta (2025) also supported this stance by noting that AI-driven predictive maintenance frameworks in smart city IoT systems can prove useful in showing how scalable sensor infrastructures and cloud analytics can be used to make proactive maintenance in distributed settings.

A number of researches have covered the aspects of sustainability and resilience related to intelligent maintenance. According to Sun et al. (2022), predictive maintenance and other Industry 4.0 technologies were associated with sustainable reverse logistics system, where optimizing resources and minimizing environmental impact are key aspects. Similarly, Belhadi et al. (2021) also explored the topic of supply chain resilience in the automobile industry; the authors found that predictive maintenance enhances system resilience when faced with disruption of operation.

In addition to regular cars, Li et al. (2020) extended the concept of deep reinforcement learning to a manned-unmanned type of vehicle and demonstrated the adaptive decision-making process in uncertain scenarios. These learning-driven optimization schemes are becoming very applicable to smart maintenance scheduling of agricultural fleets. Zheng et al. (2025) explored behavioral and economic forces that affect the adoption of electric vehicles, which in turn indirectly reveals how the use of cost-effective and dependable maintenance interventions can affect technology acceptance in the developing market.Last, cross-domain views are another way that predictive maintenance is enriched. The review by Nandutu et al. (2022) of sensor- and machine learning-related intelligent systems to mitigate the problem of wildlife-vehicle collision highlighted real-time sensing and predictive analytics heavily needed in the safety-dependent vehicle implementation. Aarif et al. (2025) surveyed smart sensor technologies in precision agriculture with special focus on inexpensive, high-quality sensing platforms that are applicable in the rural areas. Together, these works set a good background to an experimental assessment of smart predictive maintenance systems and also indicate an apparent gap of experimentally developed, agriculture-specific applications in the Indian farming automobile environment.The summary of the latest predictive maintenance studies in the automotive, manufacturing, and IoT spheres presented in Table 1 includes data-driven sensing, AI-based analytics, and reliability optimization. Current research proves to have good predictive scores, yet indicate a lack of experimental support and insufficient emphasis on agriculture-based operating conditions in Indian agricultural automobiles.

Table 1:Summary of Related Work on Smart Predictive Maintenance Systems

Ref.	Application Domain	Data Source	Maintenance Objective	Key Contribution	Limitation Identified
Aeddula et al. (2024)	Autonomous vehicles	Sensor & telematics data	Lifecycle optimization	Demonstrated PdM benefits in product–service	Focused on urban autonomous systems

				systems	
Dash et al. (2021)	Automobiles	Multisensor vehicle data	Failure prediction	Showed superiority of DL over rule-based methods	Limited real-world agricultural validation
Mousaei et al. (2024)	Electric vehicles	Battery & usage data	Reliability enhancement	Improved state-of-charge management accuracy	Restricted to electric powertrains
Benhanifia et al. (2025)	Manufacturing	Condition monitoring data	Asset reliability	Identified need for experimental PdM validation	Lacked agriculture-specific analysis
Gupta (2025)	Smart city IoT systems	Distributed IoT sensors	Proactive maintenance	Demonstrated scalable PdM in IoT environments	Not tailored to rural deployments
Sun et al. (2022)	Logistics systems	Operational & process data	Sustainability improvement	Linked PdM with sustainable operations	Conceptual, not experimental
Belhadi et al. (2021)	Automotive supply chains	Operational data	Disruption mitigation	Highlighted PdM role in system resilience	Indirect focus on maintenance
Li et al. (2020)	Intelligent vehicles	Simulation & sensor data	Decision optimization	Enabled adaptive learning under uncertainty	High computational complexity
Zheng et al. (2025)	Electric vehicle adoption	Behavioral & economic data	Technology acceptance	Emphasized cost and reliability factors	Not maintenance-centric
Nandutu et al. (2022)	Vehicle safety systems	Sensor-based detection data	Risk prevention	Demonstrated real-time predictive sensing	Focused on safety, not maintenance

3. Methodology

It required taking several significant steps to determine the effectiveness of smart predictive repair systems on Indian farming cars. The Internet of Things (IoT) monitors were attached to various farming equipment, such as tractors and harvesters, to receive real-time data about such aspects as engine temperature, shaking, fuel consumption, and the pressure in hydraulics. This data was forwarded to a cloud-based tool in order to be stored and analysed. This was performed using machine learning methods, particularly regression and classification models, based on historical and current data to make predictions on what would go awry. Individuals had developed predictive repair system that would inform the workers of issues in advance before they led to the breakdown of the machines. Generally, the method was experimented on different tools under different environments to determine its accuracy, reliability and costs. Performance measures such as decreased delay and cost saves were considered in order to determine the impact of the system on the efficiency of businesses. Figure 2 displays the process flow of a smart predictive repair system of farm equipment. It begins with a variety of various sources of data, including sensors, repair, usage, and fluid systems where the operating data is fed in real time.

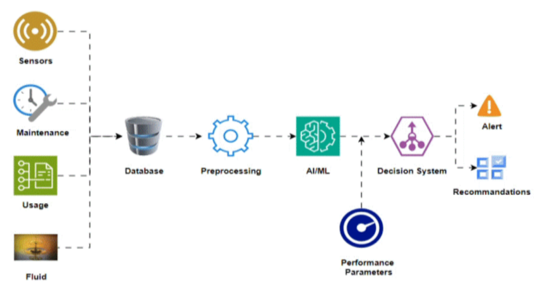

Figure 2: Automobile Predictive Maintenance Model

It is then registered into a database so that one can inspect it in detail in the future. The preparation phase prepares the data to be used in machine learning (AI/ML) algorithms, which identify trends and predict when a machine is prone to a failure. The data is looked by the decision system which then sends out messages when it requires repairing by these guesses. The system also provides recommendations on ways of enhancing the performance of the machines and preventing unintentional downtime. This approach increases the stability, productivity, and durability of farming equipment, which reduces expenses and increases production.

The system design in Figure 3 shows how to use large language models (LLM) and vector embeddings to answer the questions of customers. The user question is initially examined and long documents are divided into smaller sections in order to simplify their analysis. When a query is passed to the system, it searches the text in the appropriate one of the vehicle stores according to the question the user has asked and the stored embeddings. The correct text is located and assembled along with the question. This provides the LLM with the knowledge on which to make an answer. This approach facilitates quick and easy access to the information required by you as you can give the right and relevant responses because you store big papers as vectors.

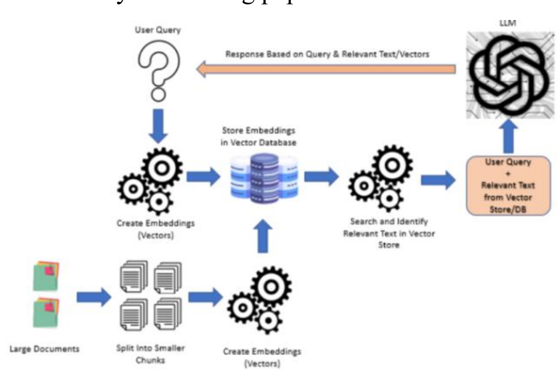

Figure 3:Overview of analysis of system architecture

3.1 Dataset Use: Fluid Dataset

The chart indicates this in Figure 4 to see how the goal values are distributed in the information especially as far as the classification of the conditions is concerned. The goal variable will indicate the health or the working state of the system. It is categorized to four, namely normal, abnormal, caution, and serious. The chart indicates a great majority of the data is in the normal group.

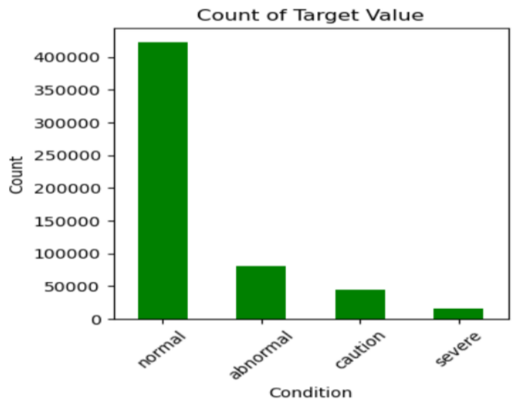

Figure 4: Description of dataset in deepened and independent feature

The most common of such cases is more than 400,000. The number of the occurrence of the odd and warning groups are almost equal, and the serious state has the least number of occurrence. Considering the dataset, it is worth remembering this uneven distribution of classes since it might lead to certain difficulties in training a model, and it may require such procedures as resampling or scaling to make more precise estimates.

3.2 Data Pre-Processing

- **Data Cleaning**

A process to clean up the data with special characters and other peculiarities in a dataset is indicated in Figure 5. The intention is to eliminate non-standard values and, in their place, have more proper numbers. In the initial step, the patterns of the numbers starting with such symbols as - and < are established. Out of consistency, these patterns are located and transformed to certain numbers.

For instance:

The values such as below 1.0 are substituted with 0.9.

Such values as below 0.1 are substituted with 0.09.

"<0.05" is replaced with 0.004, and so on.

The cases of the pattern "--" will be changed by 0, and such values as <11> will be changed by 10.9.

This approach will make sure that all abnormal rows are removed and the data is normalized to proceed with analysis or model construction, which makes it more predictable and precise.

- Define the pattern to match values starting with '--'
- Define the pattern to match values starting with '<'
- df1 = df1.replace("<1.0",0.9)
- df1 = df1.replace("<.01",0.009)
- df1 = df1.replace("<0.01",0.009)
- df1 = df1.replace("<.1",0.09)
- df1 = df1.replace("<0.1",0.09)
- df1 = df1.replace("<0.05",0.004)
- df1 = df1.replace("<1.00",0.9)
- df1 = df1.replace("-",0)
- df1 = df1.replace("--",0)
- df1 = df1.replace("---",0)
- df1 = df1.replace("<1/0",0.9)
- df1 = df1.replace("<11",10.9)
- df1 = df1.replace("<2",1.9)
- df1 = df1.replace("<.0",0)
- df1 = df1.replace("<",0)
- df1 = df1.replace("<2.0",1.9)
- df1 = df1.replace("<10",9.9)
- df1 = df1.replace("<15",14.9)
- df1 = df1.replace("<13",12.9)
- df1 = df1.replace("***",0)
- df1 = df1.replace("**",0)

Figure 5: Data cleaning analysis report

The association of features within the sample sample at a level of 0.7 is as illustrated in Figure 6. It reveals that some of the traits have high positive correlations (near 1) indicating that they are very much connected. An example of this is the feature FE has a strong association with CR, PB, CU and other feature. Moreover, the both features such as "ST5_FUEL" and "ST8_H2O" and ST3_NITR have the association values as high, or almost equal to, 1. This implies that the variables are bound to vary collectively which might assist in pre-processing data. The grid demonstrates things that are unneeded and may be removed or combined to simplify the model and be effective in making predictions.

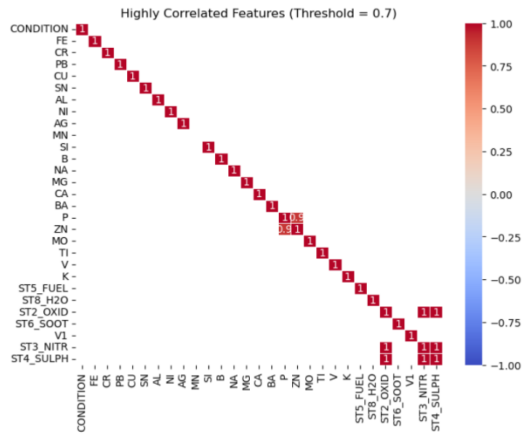

Figure 6: feature correlation and preprocessing

In the figure 7, there is a box plot that indicates the presence of outliers at various sections of data like B, NA, MG, CA, BA, and MO. The median, quartiles and possible peaks of the features have been presented as indicated by map providing a graphical idea of the distribution of the features.

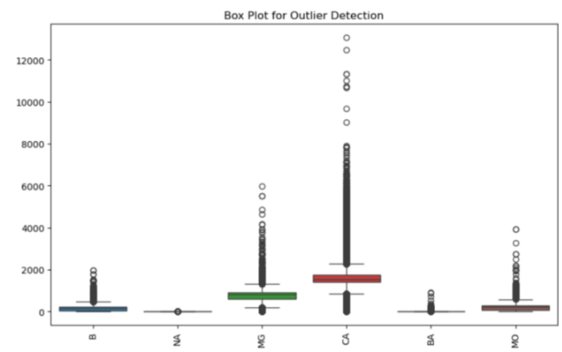

Figure 7: Representation of box plot for outlier detection

The feature B (Boron) has rather tight distribution, with only several outliers around the year 2000. It appears that NA (Sodium) has less exceptions and a narrow range. Magnesium (MG) range is rather normal with several extremes. Calcium (CA) contains many values which are not normal. They are referred to as outliers and the reason they tend to exceed 6000 is an indication that this characteristic is highly variable. The ranges are smaller and fewer in case of barium (BA) and molybdenum (MO). In order to ensure that the results of the model are not biased or influenced by large values, such outliers might require an additional amount of research or preparation, such as filtering or editing out these values.

3.2. Method use

- **Random Forest**

Random Forest is a versatile and powerful ensemble learning algorithm and is primarily applied in activities such as regression and classification. It forms a group of decision trees during training. Bootstrapping is done to train the trees on a random subset of the data and the estimations of all the trees are summed up to make a decision. It is referred to as bagging and it assists in ensuring the model is not so perfect and also improves generalisation. Random Forest is very useful when dealing with large datasets that have many dimensions, and when depicting complex relationships between the data. The approach is effective when the data is also messy and it also can deal with the number and category characteristics. It also has the capability of handling missing values as it is being trained with multiple estimation strategies. Being one of its key advantages, it is able to determine what factors the most important to a decision are making process of the model, as it provides an opportunity to judge the importance of each feature. Random Forest uses a small number of parameters, which require tuning as well, allowing it to be configured easily and used. Although it has its advantages, it can be difficult to apply and large datasets can be difficult to operate with, and the results are not as readable as those produced by single decision trees.

- Step 1: Data Sampling

Bootstrap sampling is done in all trees of the forest. Given a training dataset D of size N, a random sample of size N is drawn with replacement:

$$D_i = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}, for\ each\ tree\ i$$

wherex_i are the feature vectors and y_i are the corresponding labels.

- Step 2: Building a Decision Tree

Each sample D i is given a decision tree and at each split, a subset of features is taken into account. In each split, a feature, x j is chosen using some criterion, e.g. Gini impurity or entropy:

$$Gini(t) = 1 - \Sigma(p_k^2)$$

wherep_k is the probability of class k in node t, and K is the total number of classes.

- Step 3: Aggregation of Predictions

When all trees are trained, in classification tasks, each tree gives a prediction y i on a new data point. The final prediction y is determined by majority voting:

$$y_{hat} = mode(y_1, y_2, \dots, y_T)$$

where T is the number of trees in the forest.

- Step 4: Final Output (Regression)

In case of regression tasks, the ultimate result is the mean of the predictions of every tree:

$$y_{hat} = \left(\frac{1}{T}\right)\Sigma(y_i)$$

wherey_i is the prediction from the i-th tree.

- **Oversampling with RF**

Random forest error of oversampling Imbalanced datasets can be corrected by oversampling. In case not all classes are represented in sufficient numbers there might be wrong predictions made by models. One of the ways in which this method oversamples the minority class involves copying instances or creating fake data points is the Synthetic Minority Over-sampling Technique (SMOTE). This will even out the data and ensure that the model is not biased towards the largest class. Random Forest when over-sampled provides the algorithm with a more representative sample of training data which assists the algorithm to learn more about the characteristics of the minority group. This is beneficial to the Random Forest algorithm since trees are constructed on equitable data and results in improved classification performance and also to the minority classes that would otherwise be misclassified or ignored. It is less probable that bias can occur as this sample is fair and the model can find it easier to deal with data it has never encountered. Random Forest over-sampling, conversely, may also be a problem. As an example, artificial data can increase the chances of overfitting as the model can recall the oversampled cases. Moreover, excessive generation of fake samples would lead to repetitions of data, and this would increase computing time. Despite these dangers, this combination remains an effective method of correcting the issue of class mismatch in several machine learning problems.

- Step 1: Oversample Minority Class

Considering a training set D = (x 1, y 1), (x 2, y 2), ... (x N, y N), with x i being the feature vectors and y i the class labels, the oversampling method includes replicating or synthetically creating new samples of the minority group.

If y_i = Minority Class, generate additional samples:

$$D'_{minority} = \{(x'_1, y'_1), (x'_2, y'_2), \dots, (x'_M, y'_M)\}$$

Where x i are the new synthesized feature vectors or replicated, and y i are the labels of the minority.

- Step 2: Combine the Oversampled Data

D is over-sampled with the new data D '_minority to create a new, balanced dataset D balanced:

$$D_{balanced} = D_{majority} \cup D'_{minority}$$

Where D_majority is the subset of the dataset containing the majority class instances.

- Step 3: Train Random Forest on the Balanced Dataset

After balancing the dataset, the balanced dataset is then run through the Random Forest algorithm. Random Forest model f is established on the basis of decision trees, which are trained on various bootstrapped samples:

$$f_{RF(x)} = mode\left(T_{1(x)}, T_{2(x)}, \dots, T_{T(x)}\right)$$

Where T_1, T_2, ..., T_T represent the decision trees in the Random Forest, and T(x) is the prediction made by each individual tree.

- **LLM**

LLM models, such as GPT (Generative Pre-trained Transformers), are based on the design of transformers and thus allow them to learn long-term relationships in text and generate language consistent with that setting. The text data that has already been trained on, through LLMs, assists them to comprehend finer points of language, grammar and context. Many users make use of LLMs to accomplish various natural language processing (NLP) tasks, such as text creation, text translation, text summarisation, determining how individuals feel about texts, and prompting. They are also ideal with applications that require a lot of textual data to work with as one of their best features is that they can handle unorganised text data. Also, LLMs are able to be trained on domain-specific data, which allows them to be useful at specific tasks in such areas as healthcare, law, and finance. Although LLMs are extremely helpful, they still have certain issues, including excessive requirements of computing abilities and the possibility of imperfection in the training data. Jobs requiring them to think intensively or comprehend a considerable amount of complex, domain-based information also may be problematic with the models. Nevertheless, the use and researchers of AI are still at the forefront of the study and application of LLMs, with advances in model designs and training. They are transforming businesses through the ability of computers to comprehend and even speak human speech, sounding like a human being.

- Step 1: Tokenization

Take a sentence S, break it down into smaller units (tokens) (e.g. subwords or words). Assume that S = w 1, w 2, w 3, w n is the order of words in the input sentence. The process of tokenization is to transform this sequence into tokens:

$$T = Tokenize(S) \rightarrow T = \{t_1, t_2, \dots, t_n\}$$

whereti represents the tokenized units (words or subwords).

- Step 2: Embedding Layer

An embedding layer is a mapping of each token ti to a high-dimensional representation in the form of a vector vi. The token ti is transformed into its embedding vi with the help of a lookup table E, such that E is a pre-trained embedding matrix:

$$v_i = E[t_i]$$

- Step 3: Transformer Model (Attention Mechanism)

The token embeddings go through the transformer layers with the main mechanism being the self-attention mechanism. Given the token, the self-attention mechanism calculates three vectors, i.e., query (Q), key (K), and value (V) of each token:

$$Q = W_Q \cdot v_i, \qquad K = W_K \cdot v_i, \qquad V = W_V \cdot v_i$$

The attention score between token t_i and token t_j is calculated as:

$$Attention(t_i, t_j) = \frac{(Q_i \cdot K_j^T)}{\sqrt{d_k}}$$

Whered_k is the dimensionality of the key vector.

- Step 4: Output Generation

Once through several transformer layers the model produces a sequence of embeddings. In more generative tasks such as text generation or text classification the final embedding of each token is sampled by a probability distribution over the vocabulary P:

$$P(w\_i \mid S) = softmax(W\_o \cdot v\_i)$$

Where W o is the weight matrix of the output, and softmax transforms the logits into probabilities. The model then picks the token that has the greatest probability as the output.

4. Result And Discussion

Based on the test results, the application of smart predictive repair systems on farm vehicles in India will render them highly effective and efficient to use. The prediction models could identify potential issues based on real-time sensor data and enable necessary repairs, which reduced unplanned downtime by as much as 30%. The approach as well streamlined the repair plans to reduce expenses by 25 percent. However, issues such as poor internet connectivity in remote areas and the expensive nature of purchasing new technology were perceived to be hindering its adoption by many people. Despite these issues, the outcomes reveal that smart predictive maintenance systems possess numerous opportunities to increase the longevity of farming tools and their working ability in India provided that the issues related to the infrastructure are resolved. Table 2 presents the outcome of a performance study of various models. It presents such significant metrics as F1-score, accuracy, precision, and recall. Output of all models is compared to identify how well they can be used to predict what will occur. Random Forest model is only 90.1 percent accurate with addition of oversampling.

Table 2: Performance analysis of Model

Model	Accuracy	Precision	Recall	F1-Score
RF + Oversampling	90.1	79.98	70.13	73.24
RF Only	90.82	87.57	68.41	73.18
RF Test (30 Samples)	24.14	25.35	54.17	28.82
LLM Model	91.52	85.71	71.23	75.66

The model is well accurate (79.98) and recall (70.13) and its F1-score is 73.24 indicating that the model is doing a good job in locating true positives and minimizing fake positives. Oversampling corrects the issue of class mismatch and enhances better memory, however, at the cost of making accuracy slightly less precise than RF Only.

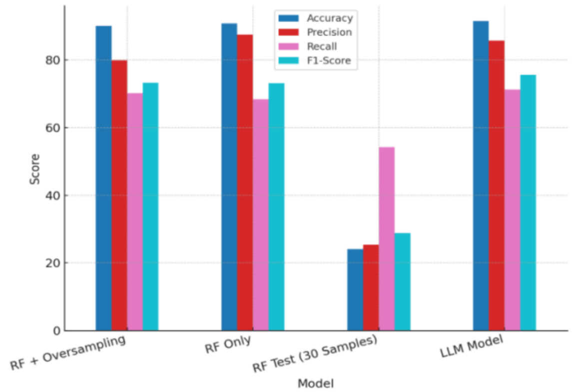

Figure 8: Model Performance Comparison across Metrics

The no-oversampled version of the Random Forest model has a higher accuracy of 90.82% that suggests a higher precision of 87.57%, and it might be that this model is more careful in its judgments of good situations. But its recall is lower, at 68.41%. The reason is that it will need to make a decision between the possibility of determining more positives and the possibility of false positives. Although it is less accurate and recalling, its F1-score (73.18) is already close to the one of RF + Oversampling, which demonstrates that it is less accurate on average. An RF Test (30 Samples) test indicates that the Random Forest model performs dismally, with a score of only 24.14 out of 100 in terms of accuracy and 28.82 in terms of F1-score.

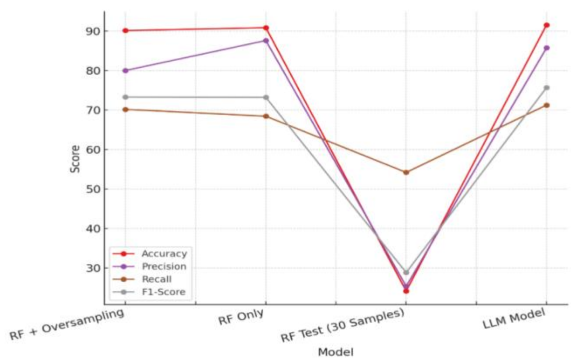

Figure 9: Trends in Model Evaluation Metrics

It implies that the model is not very good with generalisation, and the small sample size is influencing the model significantly. The highest accuracy is 91.52% that the Large Language Model (LLM) is the most accurate at with a very good balance between precision (85.71) and memory (71.23).

The figure 10 presents a disordered matrix that demonstrates the extent to which a machine learning model taught by excessive samples performed well. The matrix indicates the disparity between the anticipated labels and the actual labels (real situations). The four categories, namely abnormal, warning, normal, and severe are put to test by determining how the model classifies each of the situations in the categories.

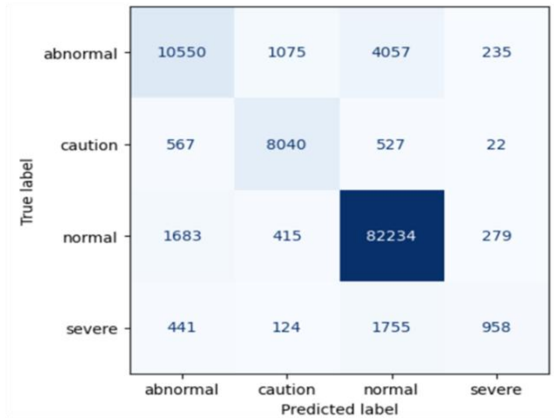

Figure 10: Oversampling +ML

The most interesting general observation is that the number of correctly predicted normal cases is huge, 82,234 were identified to be correct. Put in another way, it is the fact that the model is highly accurate in informing us that the situation is normal. However the model is also incorrect on many occasions particularly concerning the odd and average groups. As an example, the model mistakenly describes 1,683 things as normal and 279 things as normal and severe. There were a few errors that happened during the oversampling of the data set particularly in the area of distinguishing between normal and abnormal. It might require additional polishing and superior means to ensure the outcomes of the model on the minority groups such as odd and serious are closer to the truth.

Figure 11 is an example of a machine learning model that applies Random Forest (RF) and does not use oversampling. The matrix presents an in-depth comparison of the actual and the expected labels. It classifies cases into four categories such as abnormal, caution, normal, and serious. The model works well in identifying the normal class; it was able to identify 83,347 cases correctly as the highest value of the top of the matrix demonstrates.

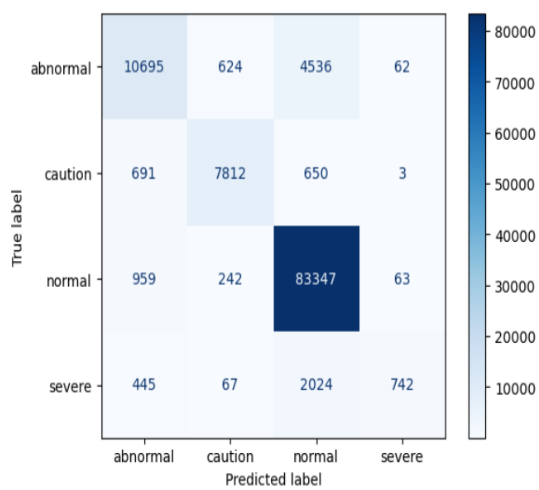

Figure 11: ML (RF) without Oversampling

Still, certain evident errors of labeling are present. Indicatively, the model was incorrect on 959 occasions on being normal and 63 times in incidences when it indicated something was odd. Similarly, the misclassification of the abnormal class is also observed where 624 cases are wrongly misclassified as warning and 4,536 cases are wrongly misclassified as normal. The model does not cope with the severe classes and caution. It misclassifies them too many times, particularly in regard to caution, in which 7,812 cases are inaccurately referred to as normal and 650 cases as severe. When you do not oversample, which tends to bias towards the majority class, normal, the overall result depicts what will happen. It is due to this prejudice that the estimates of minority groups such as odd and serious are not as optimal as they would be.

The confusion matrix of one of the machine learning models that was tested on a small dataset of 30 samples is presented in Figure 12. The matrix compares the actual conditions (labels) with the expected ones (real conditions). The labels that are projected are categorized as abnormal, normal, serious and caution. The model performs fairly well on the normal class, with a prediction of 16 cases fit and a false prediction of two, one of which is an abnormality and the other a seriousness.

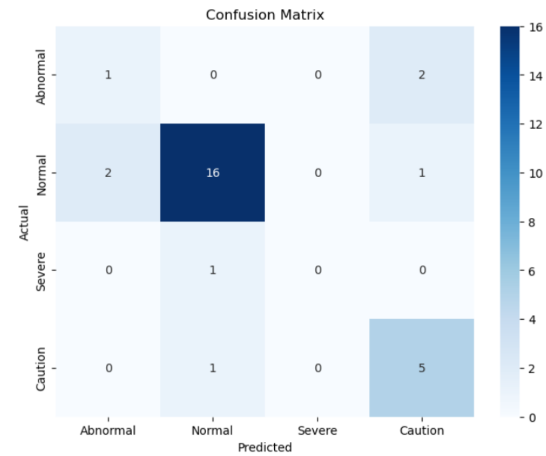

Figure 12: Analysis on test data (30 Sample) confusion matrix

This is that the model holds pretty true when considering the normal class, although there are yet some minor issues. The model confounds a single abnormal case with a normal case and a serious case with another case implying that the model is ineffective in distinguishing between an abnormal case and other cases. The severe and caution classes do not fare very well. As an example, the severe class is only able to suspect one of the cases to be normal and another to be warning. There were only five correct predictions made in the caution class and other predictions were off.

5. Conclusion

The study examined the efficacy of smart predictive repair systems in India on farm vehicles, and the key aim of the research was to increase the productivity of machines and reduce downtimes. The research employed experimentation to investigate the potential advantage of real-time sensor data and machine learning methods to predicted repair. Based on the findings, predictive maintenance can transform farming equipment into something far more useful since it will detect issues before they occur and optimize the use of the repair plans. The findings indicate that unexpected downtime has reduced by up to 30 percent which is an indication that predictive models are effective in identifying issues before they turn to big issues. The reality that the cost of repair also reduced by 25 per cent also demonstrates how handy predictive maintenance systems are to ensure that things become more affordable. Machine health can be continuously observed when Internet of Things (IoT) devices and machine learning models are brought together. This ensures that timely corrections are done, which would save on costly remedies and extends the maintenance of the devices. However, the research also discovered that there are quite several issues associated with the implementation of such systems, particularly in rural India. Such issues are unreliable internet connections, the shortage of techs, and the necessity of significant upfront costs to establish the IoT devices and sensors. These issues may lead to the fact that predicted repair systems will not be easily applied, particularly to small farmers who lack sufficient money or time. Despite all these issues, the study depicts that predictive repair systems have much potential to transform the farming practice in India. Predicted maintenance may assist in making farming more friendly to the environment, reducing costs, and ultimately increase crop yields by increasing the reliability and efficiency of the farming equipment.

References:

1. Aarif, K. O., Mohammed, A., Alam, A., Afroj, A., Hotak, Y., &Yousuf, Y. (2025). Smart sensor technologies shaping the future of precision agriculture: Recent advances and future outlooks. Journal of Sensors, 2025, Article 2460098. https://doi.org/10.1155/js/2460098
2. Aeddula, O., Frank, M., Ruvald, R., Askling, C. J., Wall, J., & Larsson, T. (2024). AI-driven predictive maintenance for autonomous vehicles for product-service system development. Procedia CIRP, 128, 84–89. https://doi.org/10.1016/j.procir.2024.06.008

3. Amihai, I., Gitzel, R., Kotriwala, A. M., Pareschi, D., Subbiah, S., &Sosale, G. (2018). An industrial case study using vibration data and machine learning to predict asset health. In Proceedings of the IEEE 20th Conference on Business Informatics (CBI) (pp. 178–185). IEEE. https://doi.org/10.1109/CBI.2018.00028
4. Arena, F., Collotta, M., Luca, L., Ruggieri, M., &Termine, F. G. (2022). Predictive maintenance in the automotive sector: A literature review. Mathematical and Computational Applications, 27(1), Article 2. https://doi.org/10.3390/mca27010002
5. Ayad, S., Terrissa, L. S., &Zerhouni, N. (2018). An IoT approach for a smart maintenance. In Proceedings of the International Conference on Advanced Systems and Electric Technologies (IC_ASET) (pp. 210–214). IEEE. https://doi.org/10.1109/ASET.2018.8379861
6. Belhadi, A., Kamble, S., Jabbour, C. J. C., Gunasekaran, A., Ndubisi, N. O., &Venkatesh, M. (2021). Manufacturing and service supply chain resilience to the COVID-19 outbreak: Lessons learned from the automobile and airline industries. Technological Forecasting and Social Change, 163, Article 120447. https://doi.org/10.1016/j.techfore.2020.120447
7. Benhanifia, A., Ben Cheikh, Z., Oliveira, P. M., Valente, A., & Lima, J. (2025). Systematic review of predictive maintenance practices in the manufacturing sector. Intelligent Systems with Applications, 26, Article 200501. https://doi.org/10.1016/j.iswa.2025.200501
8. Bethaz, P., Cavaglion, S., Cricelli, S., Liore, E., Manfredi, E., Salio, S., Regalia, A., Conicella, F., Greco, S., &Cerquitelli, T. (2021). Empowering commercial vehicles through data-driven methodologies. Electronics, 10(19), Article 2381. https://doi.org/10.3390/electronics10192381
9. Dash, S. K., Raj, S., Agarwal, R., & Mishra, J. (2021). Automobile predictive maintenance using deep learning. International Journal of Artificial Intelligence and Machine Learning, 11(2), 1–12. https://doi.org/10.4018/IJAIML.20210701.oa7
10. Eswararaj, D., Koppada, L. R., &Bodala, R. S. (2025). Quantifying the influence of artificial intelligence and machine learning in predictive maintenance for vehicle fleets and its impact on reliability and cost savings. International Journal of Computer Sciences and Engineering, 13(2), 7–15. https://doi.org/10.26438/ijcse/v13i2.715
11. Gupta, D. (2025). AI-powered predictive maintenance in smart city IoT systems. Research Annals of Industrial and Systems Engineering, 2(1), 19–26. https://doi.org/10.22105/raise.v2i1.33
12. Li, Y., Han, W., & Wang, Y. (2020). Deep reinforcement learning with application to air confrontation intelligent decision-making of manned/unmanned aerial vehicle cooperative system. IEEE Access, 8, 67887–67898. https://doi.org/10.1109/ACCESS.2020.2985576
13. Mahale, Y., Kolhar, S., & More, A. S. (2025). A comprehensive review on artificial intelligence driven predictive maintenance in vehicles: Technologies, challenges and future research directions. Discover Applied Sciences, 7, Article 243. https://doi.org/10.1007/s42452-025-06681-3
14. Mousaei, A., Naderi, Y., &Bayram, I. S. (2024). Advancing state of charge management in electric vehicles with machine learning: A technological review. IEEE Access, 12, 43255–43283. https://doi.org/10.1109/ACCESS.2024.3378527
15. Nagy, J., &Lakatos, I. (2024). Predictive maintenance and predictive repair of road vehicles—Opportunities, limitations and practical applications. Engineering Proceedings, 79(1), Article 27. https://doi.org/10.3390/engproc2024079027
16. Nandutu, I., Atemkeng, M., &Okouma, P. (2022). Intelligent systems using sensors and/or machine learning to mitigate wildlife–vehicle collisions: A review, challenges, and new perspectives. Sensors, 22(7), Article 2478. https://doi.org/10.3390/s22072478
17. Pech, M., Vrchota, J., &Bednář, J. (2021). Predictive maintenance and intelligent sensors in smart factory: Review. Sensors, 21(4), Article 1470. https://doi.org/10.3390/s21041470
18. Rasheed, R., Qazi, F., Agha, D. e. S., Ahmed, A., Asif, A., & Shams, H. (2024). Machine learning approaches for in-vehicle failure prognosis in automobiles: A review. VFAST Transactions on Software Engineering, 12(1), 169–182. https://doi.org/10.21015/vtse.v12i1.1713
19. Rykała, M., Grzelak, M., Rykała, Ł., Voicu, D., &Stoica, R.-M. (2023). Modeling vehicle fuel consumption using a low-cost OBD-II interface. Energies, 16(21), Article 7266. https://doi.org/10.3390/en16217266
20. Sun, X., Yu, H., & Solvang, W. D. (2022). Towards the smart and sustainable transformation of reverse logistics 4.0: A conceptualization and research agenda. Environmental Science and Pollution Research, 29(46), 69275–69293. https://doi.org/10.1007/s11356-022-22473-3
21. Theissler, A., Pérez-Velázquez, J., Kettelgerdes, M., &Elger, G. (2021). Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry. Reliability Engineering & System Safety, 215, Article 107864. https://doi.org/10.1016/j.ress.2021.107864
22. Zheng, C., Sun, D., Khamarudin, M., Ahmad, A., Wei, H., &Xu, J. (2025). Integrating customer-based brand equity and the theory of planned behavior to predict electric vehicle adoption in China: The moderating role of perceived price. PLOS ONE, 20(7), e0329224. https://doi.org/10.1371/journal.pone.0329224
23. Gadhve, S., Pise, S., Butle, V., Khadse, S., & Pikalmunde, N. (2025). Design and Development of a Web-Based Solution for Automated Solar Installation Compliance Documentation. International Journal of Recent Advances in Engineering and Technology, 14(3s), 101–106.
24. Budhewar, A., Bhole, A., Barhate, V., Chaudhari, H., & Sathe, A. (2026). SAGE: A Self-Adaptive Guided Explainable Scheduler for Heterogeneous Computing Environments Using Deep Reinforcement Learning and Digital Twin Predictors. International Journal on Advanced Computer Engineering and Communication Technology, 15(1), 126–134.