|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PROJECT OVERVIEW STATEMENT** | **Project Name: “Predictive Maintenance for Industrial Machinery using Machine Learning”** | | **Student Name:**  Pavan Kumar Jidugu | |
| **Problem/Opportunity:** | | | | |
| In the Industries, Unexpected equipment failures in an industrial setting can result in expensive downtime, lost production, and costly repairs. As they don't handle possible breakdowns in real time, traditional maintenance techniques like planned maintenance can be ineffective. By employing machine learning to anticipate equipment breakdowns before they happen, predictive maintenance offers a data-driven solution that minimizes unscheduled downtime and maximizes operational efficiency. | | | | |
| **Goal:** | | | | |
| The main goal of this research is to develop a machine learning model that, using sensor data, including temperature, vibration, pressure, and other machine operational parameters, may be used to anticipate machinery breakdowns in real time. To lower maintenance costs and downtime, the system will suggest maintenance tasks before faults happen. | | | | |
| **Objectives:** | | | | |
| * Objective 1: **Data Collection and Preprocessing**   + **Outcome**: Collect and preprocess data from industrial IoT sensors (e.g., temperature, vibration, pressure) to ensure data quality for predictive modeling.   + **Time Frame**: Within 1 month of project initiation.   + **Measure**: The dataset will be cleaned, with outliers handled, and missing values imputed. Sensor data will be standardized and transformed for better model performance.   + **Action**: Use historical machine data from existing sensors or publicly available datasets (e.g., NASA’s turbofan engine dataset). Perform feature engineering to extract useful features such as rolling averages or peak amplitudes. * Objective 2: **Data Analysis**   + **Outcome**: Identify trends and patterns in sensor data, detect relationships between features, and determine failure patterns.   + **Time Frame**: Within 2 weeks of project initiation.   + **Measure**: Visualize key patterns between variables and failure events using plots, heatmaps, and correlation matrices.   + **Action**: Conduct EDA to analyze data distributions and key factors influencing failures, using libraries like Matplotlib, Seaborn, and Plotly. * Objective 3: **Machine Learning Model Development**   + **Outcome**: Develop and train machine learning models (e.g., Random Forest, XGBoost, or LSTM) to predict when failures are likely to occur.   + **Time Frame**: Within 2 months of project initiation.   + **Measure**: Achieve a predictive accuracy of at least 85% on a validation dataset.   + **Action**: Train multiple models, including classification algorithms for failure prediction and regression models for predicting remaining useful life (RUL). Tune hyperparameters using grid search and cross-validation techniques. * Objective 4: **Model Evaluation and Optimization**   + **Outcome**: Evaluate model performance using appropriate metrics such as accuracy, precision, recall, F1 score, and ROC-AUC curve for classification models, or mean squared error (MSE) for regression models.   + **Time Frame**: Within 2 weeks after model development.   + **Measure**: Select the best-performing model based on validation performance, with a target accuracy of 85% or higher.   + **Action**: Conduct model evaluations and optimize the selected model for higher predictive accuracy. Implement techniques like feature selection and hyperparameter tuning for improved performance. * Objective 5: **Deployment and Dashboard for Real-Time Monitoring**   + **Outcome**: Deploy the predictive model and create a user-friendly dashboard for real-time monitoring of machine health.   + **Time Frame**: Within 1 month of project completion.   + **Measure**: Develop a web-based dashboard that displays the current state of machine health, real-time predictions of failures, and recommended maintenance schedules.   + **Action**: Use Flask or FastAPI to deploy the model as an API and integrate it with IoT sensor data. Implement a dashboard using Plotly Dash or Power BI for visualization. | | | | |
| **Success Criteria:** | | | | |
| * In order to accurately forecast machinery breakdowns, the predictive model must attain a minimum of 85% accuracy. * By giving timely maintenance recommendations, the system should successfully lower maintenance costs and downtime. * Updating the machine health in real time should be displayed on the dashboard, giving maintenance personnel an easy-to-use interface. * All project milestones should be reached, and the project should be finished in the allotted time. | | | | |
| **Assumptions, Risks, Obstacles:** | | | | |
| * **Assumptions**:   + For predictive analysis, sensor data is taken to be accurate and dependable.   + The model is predicated on having access to historical data that precisely captures the operational circumstances and machinery failure history. * **Risks**:   + Incomplete or significantly noisy or outlier-filled sensor data can cause the model's performance to deteriorate.   + Delays could result from unforeseen technological difficulties when integrating real-time sensor data with the model. * **Difficulties**:   + Inadequate high-quality data for model testing and training could affect the model's accuracy.   + Integrating the technology into the existing industrial infrastructure may require additional IT resources and knowledge. | | | | |
| **Prepared By** | **Date** | **Approved By** | | **Date** |
| Pavan Kumar Jidugu | 09/24/2024 |  | |  |