Electric Motor Temperature Prediction Using Machine Learning

Abstract

This project focuses on predicting the temperature of an electric motor using machine learning techniques based on various operational parameters such as current, speed, torque, ambient temperature, vibration, and motor load. Overheating is one of the major causes of motor failure in industrial systems. By predicting motor temperature in advance, preventive maintenance can be performed, reducing downtime and improving efficiency. The model is trained using historical data and deployed using a Streamlit-based web application for real-time prediction.

1. Introduction

Electric motors are widely used in industries, manufacturing plants, and automation systems. Continuous monitoring of motor temperature is essential to ensure safety, efficiency, and long lifespan of the equipment. Traditional monitoring systems only provide current temperature values but do not predict future overheating conditions. This project introduces a machine learning-based approach to predict motor temperature using multiple sensor parameters.

2. Problem Statement

The main problem addressed in this project is the overheating of electric motors due to excessive load, high current, vibration, and environmental factors. Manual monitoring is inefficient and cannot predict future temperature trends. Therefore, an intelligent prediction system is required to forecast motor temperature and prevent system failure.

3. Objectives of the Project

- To collect and analyze motor operational data. - To preprocess and clean the dataset. - To build a machine learning model for temperature prediction. - To deploy the model using a Streamlit web application. - To provide real-time temperature prediction for users.

4. Dataset Description

The dataset used in this project contains various parameters affecting motor temperature, including current, speed, torque, ambient temperature, vibration, and motor load. The dataset was stored in CSV format and used for training and testing the machine learning model.

5. Data Preprocessing

Data preprocessing is a crucial step in machine learning. The dataset was cleaned by removing missing values and outliers. Feature scaling was applied using StandardScaler to normalize the input features and improve model performance. The dataset was then split into training and testing sets.

6. Model Building

A regression-based machine learning model was developed to predict motor temperature. The model was trained using historical motor data and evaluated using performance metrics such as Mean Squared Error (MSE) and R-squared score. The trained model was saved as a .pkl file for deployment.

7. Application Development

The user interface was developed using Streamlit, a Python-based web framework. The application allows users to input parameters such as current, speed, torque, ambient temperature, vibration, and motor load. Based on the input values, the trained machine learning model predicts the motor temperature in real time.

8. Results and Discussion

The developed model successfully predicts motor temperature with good accuracy. The Streamlit application provides a simple and interactive interface for users to test different input values. This system can be used in industries for predictive maintenance and performance monitoring.

9. Advantages of the Project

- Real-time temperature prediction - Prevents motor overheating - Improves system reliability - Easy-to-use web interface - Supports predictive maintenance

10. Future Scope

In the future, this project can be enhanced by integrating IoT sensors for live data collection, using advanced deep learning models for higher accuracy, and deploying the application on cloud platforms for large-scale industrial usage.

11. Conclusion

This project demonstrates the successful implementation of a machine learning-based system for electric motor temperature prediction. By using operational parameters and a trained regression model, the system provides accurate temperature predictions through a Streamlit web application. The solution helps in early fault detection, efficient maintenance, and improved industrial safety.

12. Tools and Technologies Used

- Python - Machine Learning (Scikit-learn) - Pandas and NumPy - Streamlit - VS Code - CSV Dataset