Electric Motor Temperature Prediction Using Machine Learning

(Project Report)

1. INTRODUCTION

Electric motors are widely used in industries, electric vehicles, and automation systems.
Excess heat during operation can reduce efficiency and cause motor failure.
This project builds a Machine Learning model to predict motor temperature using sensor data
such as voltage, current, torque, speed, and ambient temperature.
The trained model is integrated with a Flask web application for real-time prediction.

1.2 Purpose

- Predict motor temperature accurately
- Prevent overheating and motor damage
- Enable predictive maintenance
- Provide real-time monitoring through a web interface

2. PROBLEM STATEMENT

Motor overheating leads to unexpected breakdowns and financial losses.
The goal is to build a Machine Learning regression model that predicts motor temperature based on operational parameters.

3. REQUIREMENT ANALYSIS

Functional Requirements:

- User inputs motor parameters
- System predicts temperature
- Output displays predicted temperature

Non-Functional Requirements:

- Fast response time
- High prediction accuracy
- User-friendly interface

4. TECHNOLOGY STACK

Frontend: HTML, CSS, Bootstrap
Backend: Python Flask
Machine Learning: Pandas, NumPy, Scikit-learn
Model Saving: Pickle (.pkl)
Tools: VS Code, Jupyter Notebook

5. PROJECT DESIGN

Dataset is collected and preprocessed.
Features are scaled and selected.
Regression model is trained and evaluated.
Model is saved as .pkl file.
Flask loads model and predicts temperature based on user input.

6. PERFORMANCE METRICS

- Mean Absolute Error (MAE)
- Mean Squared Error (MSE)
- R^2 Score

7. ADVANTAGES

- Early overheating detection
- Reduced maintenance cost
- Improved system reliability
- Supports predictive maintenance

8. DISADVANTAGES

- Depends on dataset quality
- Requires retraining for new conditions
- Sensor errors affect prediction

9. CONCLUSION

The project successfully implements a Machine Learning-based electric motor temperature prediction system integrated with a Flask web application.
It improves reliability and prevents motor failures through predictive monitoring.

10. FUTURE SCOPE

- Implement Deep Learning models
- Integrate IoT real-time sensors
- Deploy on cloud platform
- Add alert notification system