Electric Motor Temperature Prediction

Presented By: 1) Om Sinkar - From MITAOE
2) Parimal Dode - From MITAOE
3) Prajwal Jadhav - From MITAOE
4) Atharva Pate - From MITAOE

- **Problem Statement:**

In applications like robotics, vehicles, and industrial machines, **Permanent Magnet Synchronous Machines (PMSMs)** are widely used due to their high efficiency, low torque ripple, and excellent performance. However, high temperature in the **rotor** part of the motor can cause performance issues or permanent damage.

The challenge is to **accurately estimate the rotor's temperature** using sensor data. Manual checks are not possible in real-time, so we aim to build a **machine learning model** that can predict the rotor temperature from electrical and mechanical parameters.

- **Proposed System/Solution:**

We are developing an **AI-based predictive model** that takes input features like voltage, current, speed, and other motor parameters, and predicts the **stator temperature components and PM**.

- Feature engineering from sensor data
- Training with multiple ML algorithms
- Selecting the most accurate model

- Saving the model in .pkl format
- Integrating the model into a **Flask web application** with a user-friendly interface

System Developement Approach:

AI/ML - Python, Pandas,Scikit-learn
Model Training - Linear Regression,Decision Tree,Random Forest,SVM
Web Development - Flask(Backend), HTML with inline CSS(Frontend)
Model Saving - Joblib or Pickle
IDE and Tools - Google Collab

Algorithm and Deployment:

Algorithms Used:

We experimented with the following algorithms:

- **Linear Regression** – Basic and fast, but sometimes underperforms on non-linear data.
- **Decision Tree** – Good at handling non-linearity and categorical splits.
- **Random Forest** – Ensemble of decision trees, better accuracy and generalization.
- **Support Vector Machine (SVM)** – Good at finding boundaries, but slower.

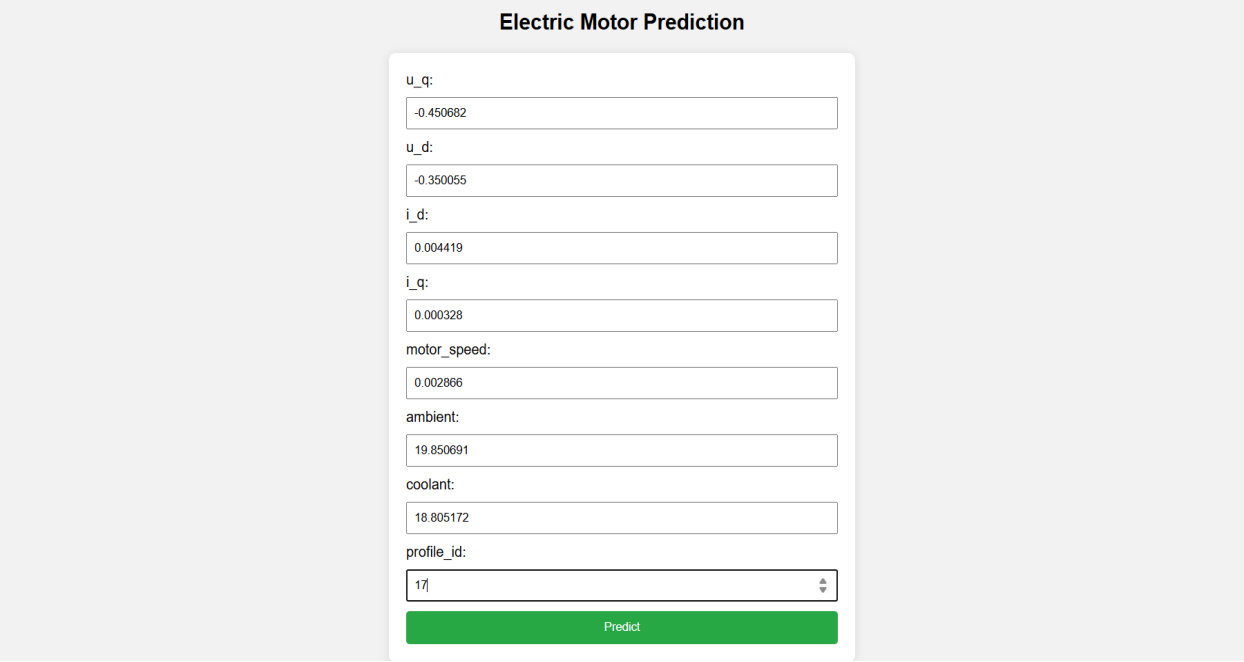
Model Selection:

After training and testing, **Decision Tree** gave the best accuracy in our case. This model was saved in .pkl format using joblib.

Flask Deployment:

- Flask receives input values from the user via HTML form.
- These values are passed to the model to get predictions.
- The predicted stator temperature components and PM is displayed on the interface.

- ## Result:



Predicted Values:

PM: 19.09

Stator Yoke: 18.32

Stator Tooth: 18.29

Stator Winding: 24.48

- ## Conclusion:

This project successfully demonstrates how machine learning can be used in **real-time industrial applications** like motor temperature monitoring. By predicting the rtemperature in advance, it becomes easier to **avoid overheating, reduce downtime**, and **increase machine lifespan**.

- ## Future Scope:

1. **Integrate with live IoT sensors** for real-time monitoring.
2. Add **visual alerts** for overheating (e.g., red for high temperature).
3. Use **deep learning models** (LSTM, GRU) for better prediction with time-series data.
4. Deploy on cloud platforms like **Render, Heroku, or AWS.**
5. Include **motor health diagnostics** in the same platform.