AnomaData (Automated Anomaly Detection for Predictive Maintenance)

Project Overview

Anoma Data is a machine learning project designed to predict machine breakdowns by identifying anomalies in time series data. This project aims to provide a predictive maintenance solution, helping industries reduce risks and perform maintenance before equipment failure occurs.

Project Structure

The project is organized into the following main components:

1 Data Collection

2 Exploratory Data Analysis (EDA)

3 Data Cleaning

4 Feature Engineering

5 Model Selection and Training

6 Model Validation

7 Hyperparameter Tuning

8 Model Deployment Plan

Libraries used To run the project, this are the following packages installed:

1 pandas

2 numpy

3 matplotlib

4 seaborn

5 scikit-learn

pip install pandas numpy matplotlib seaborn scikit-learn

Instructions

1. Data Collection

Ensure data.csv is in the same directory as main.py. The script will load the data from this file.

2. Exploratory Data Analysis (EDA)

The script performs EDA to understand the data, identify patterns, and visualize relationships using descriptive statistics and visualizations.

3. Data Cleaning

The script handles missing values, standardizes data, and treats outliers to ensure data quality.

4. Feature Engineering

The script includes feature engineering steps to create new features and transform existing ones for better model performance. It ensures correct datatypes, especially for date columns.

5. Train/Test Split

The script applies a sampling distribution to split the data into training and testing sets, ensuring a representative split.

6. Model Selection and Training

The script selects and trains a suitable model. Models considered include Logistic Regression, Random Forest, and Gradient Boosting. The best parameters are estimated during training.

7. Model Validation

The script evaluates the model's performance on the test set using metrics such as accuracy, precision, recall, and F1-score to ensure it generalizes well to unseen data.

8. Hyperparameter Tuning

The script includes methods for hyperparameter tuning using Grid Search or Random Search to improve model performance.

9. Model Deployment Plan

The script outlines a plan to deploy the model using a web framework like Flask or FastAPI, making the trained model available for production use.

Running the Project

Ensure all required packages are installed.

Place data.csv in the same directory as main.py.

Run the script:

bash

Copy code

python main.py

The script will load the data, perform all steps from EDA to model validation, and output the evaluation metrics.

Future Work

The report (report.pdf) discusses potential future work, including:

Improving feature engineering techniques.

Exploring additional models and ensemble methods.

Implementing a robust deployment pipeline.

Integrating real-time data for continuous monitoring and predictions.

Conclusion

AnomaData provides a comprehensive pipeline for predicting machine breakdowns through anomaly detection. By following this README, you should be able to set up and run the project, gaining insights into your data and predicting potential failures before they occur. For any further details, refer to the detailed report provided.