UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Predictive Maintenance Project – Sorbonne Data Analytics

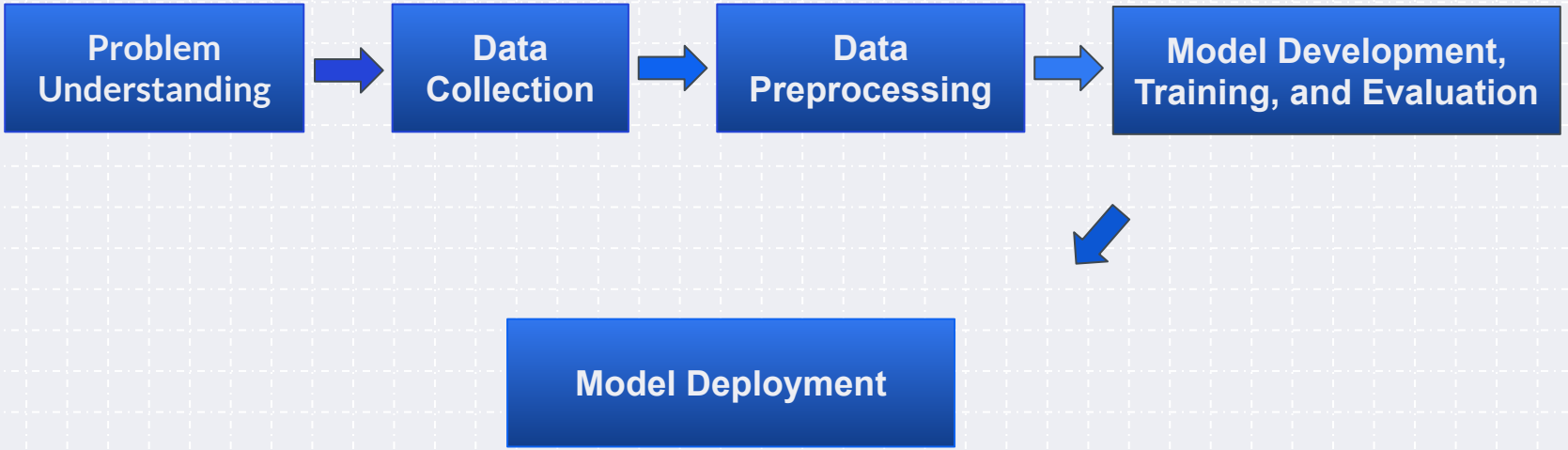
Ibrahim Guoual Belhamidi

Problematic

In a factory with over 100 CNC machines, wear and tear are inevitable, leading to occasional failures that result in repair costs and production downtime. It's not economically feasible to have a technician check each machine's health after every operation.

The goal is to automate this process by creating an end-to-end solution that alerts the repair team to potential issues before a machine breaks down.

Timeline:


Problem Understanding
Data Collection
Data Preprocessing
Model Development, Training, and Evaluation
Model Deployment

About Dataset:

Given the The dataset contains 10,000 rows with 10 features:

- **UID:** Unique identifier (1-10,000)
- **ProductID:** Includes a quality variant (L, M, H) and a serial number.
- **Air temperature [K]:** Random walk process, normalized to 2 K around 300 K.
- **Process temperature [K]:** Air temperature plus 10 K, normalized to 1 K.
- **Rotational speed [rpm]:** Based on 2860 W power with noise.
- **Torque [Nm]:** Normally distributed around 40 Nm (σ = 10 Nm).
- **Tool wear [min]:** Quality variants (H/M/L) add 5/3/2 minutes of tool wear.
- **Machine failure:** Indicates if a machine failure occurred for any failure mode.
- **Target:** Failure or Not.
- **Failure Type:** Type of Failure.

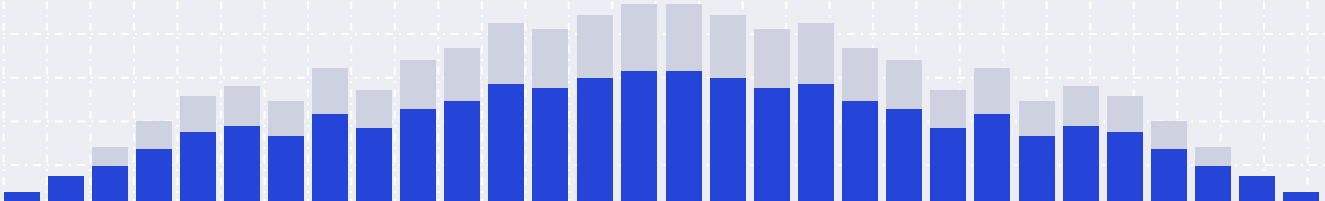
9661 (96.61%) of total machine working in the dataset

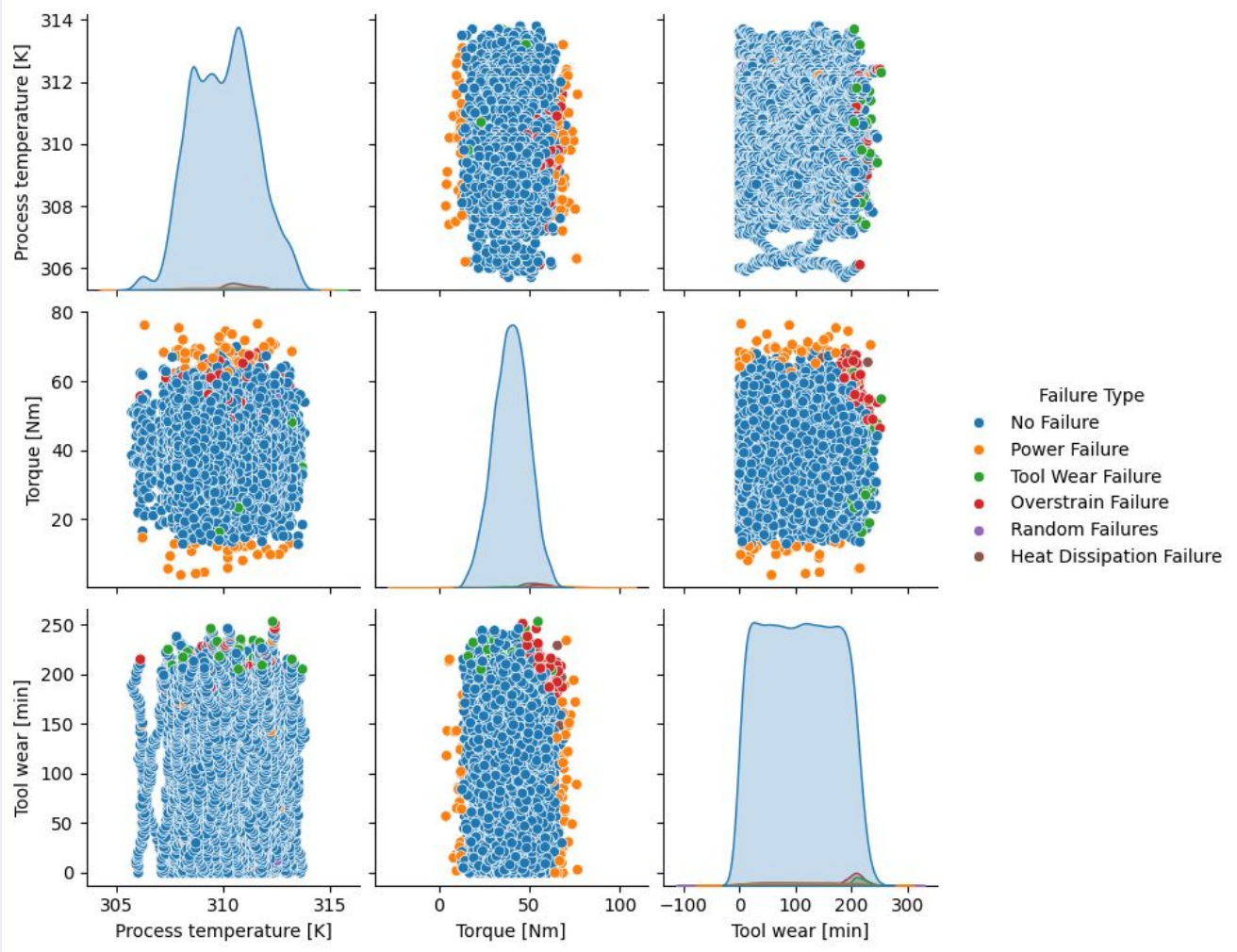
339 (03.39%)

of the total machine experienced failure in the dataset

9661 (96.61%)

Of total machine working in the dataset




Process temperature [K]
Torque [Nm]
Tool wear [min]
Failure Type
No Failure
Power Failure
Tool Wear Failure
Overstrain Failure
Random Failures
Heat Dissipation Failure

Models used and their result:

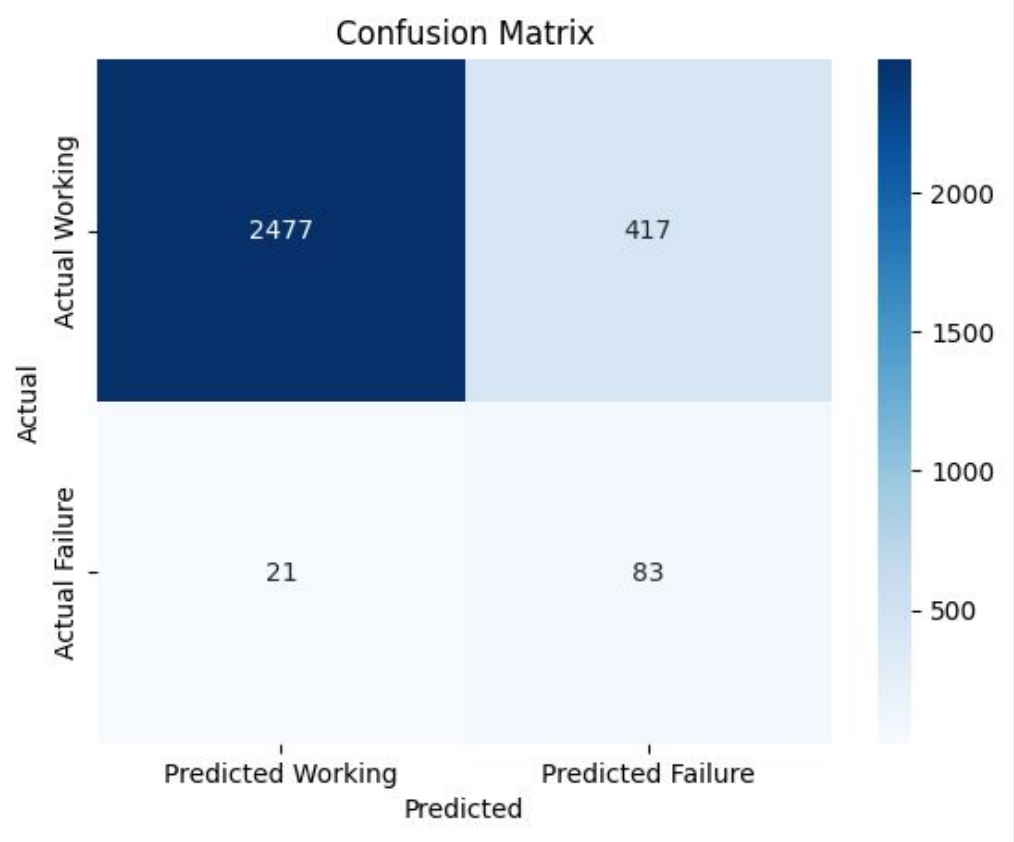
In this case, we used multiple ML models.

- GradientBoostingClassifier.
- RandomForestClassifier.
- RandomForestClassifier with Class Weight Adjustment (Auto).
- RandomForestClassifier with Combining Oversampling and Undersampling.
- CatBoostClassifier with Class Weight Adjustment.
- CatBoostClassifier Combining Oversampling and Undersampling.
- Neural network model.
- Neural network model with Resampling.
- Neural network model with custumasition.

Models used and their result:

GradientBoostingClassifier.

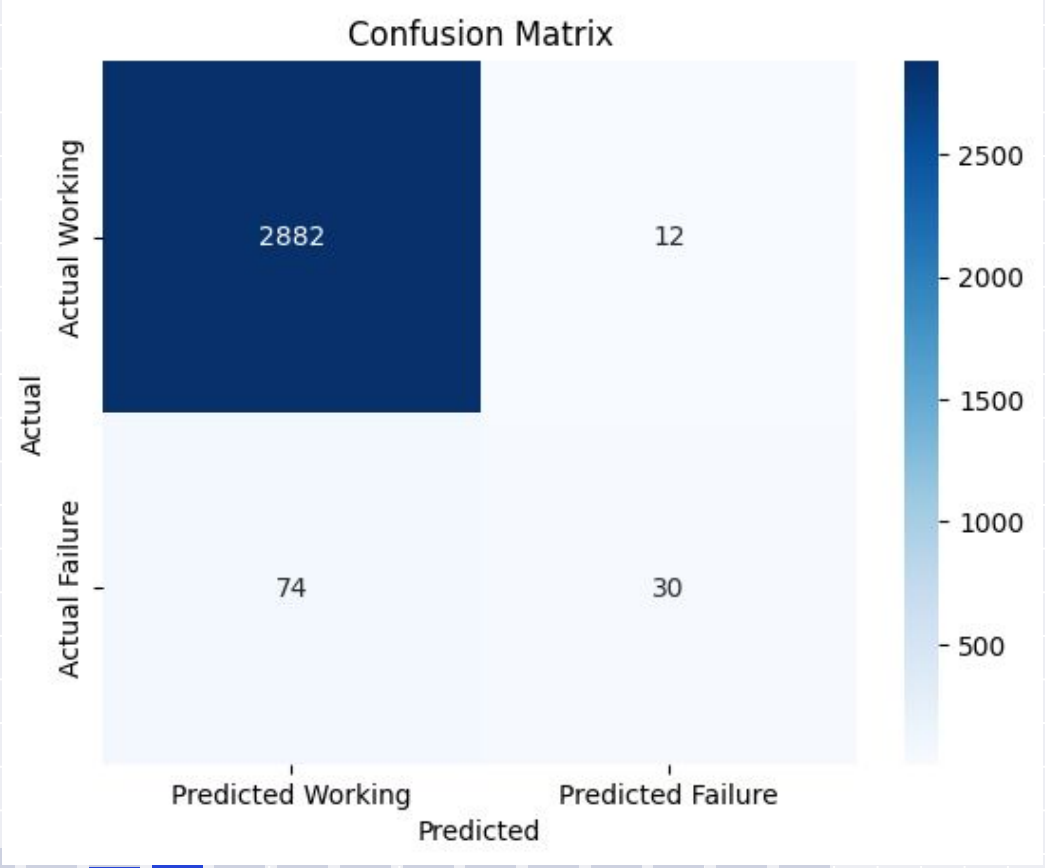
In this case, This Model Scored 85%.



Models used and their result:

RandomForestClassifier.

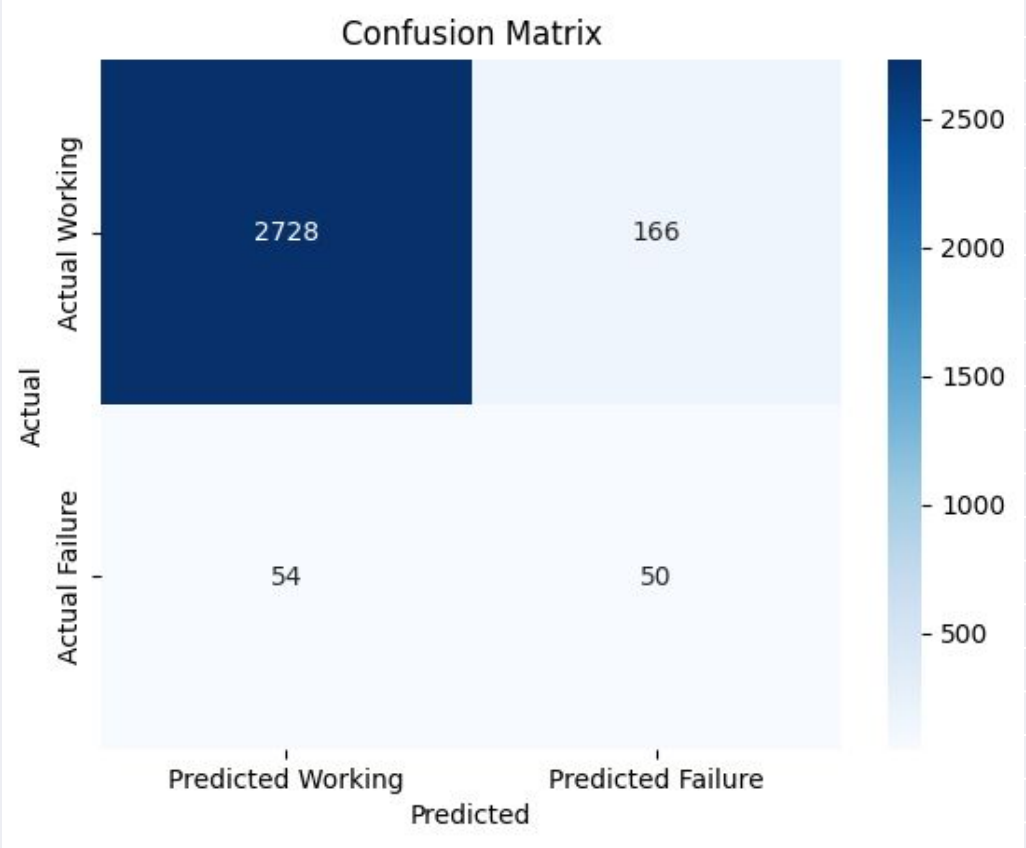
In this case, This Model Scored 89.09%.



Models used and their result:

RandomForestClassifier with Class Weight Adjustment (Auto).

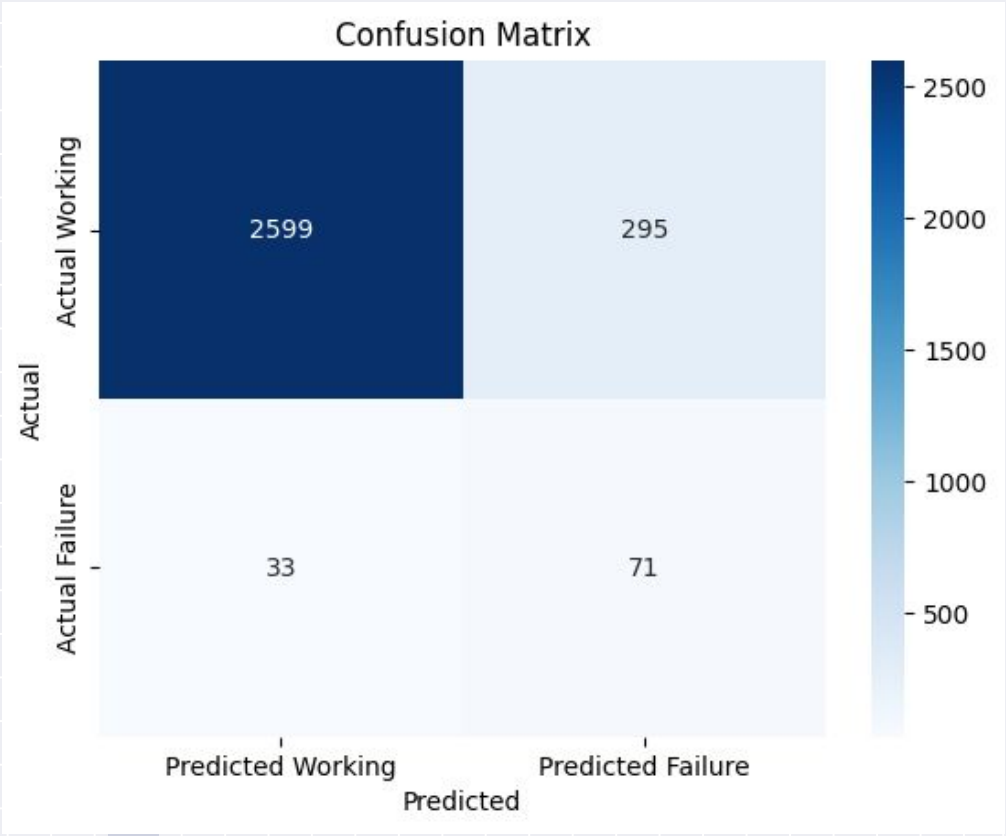
In this case, This Model Scored 90.24%.



Models used and their result:

RandomForestClassifier with Combining Oversampling and Undersampling

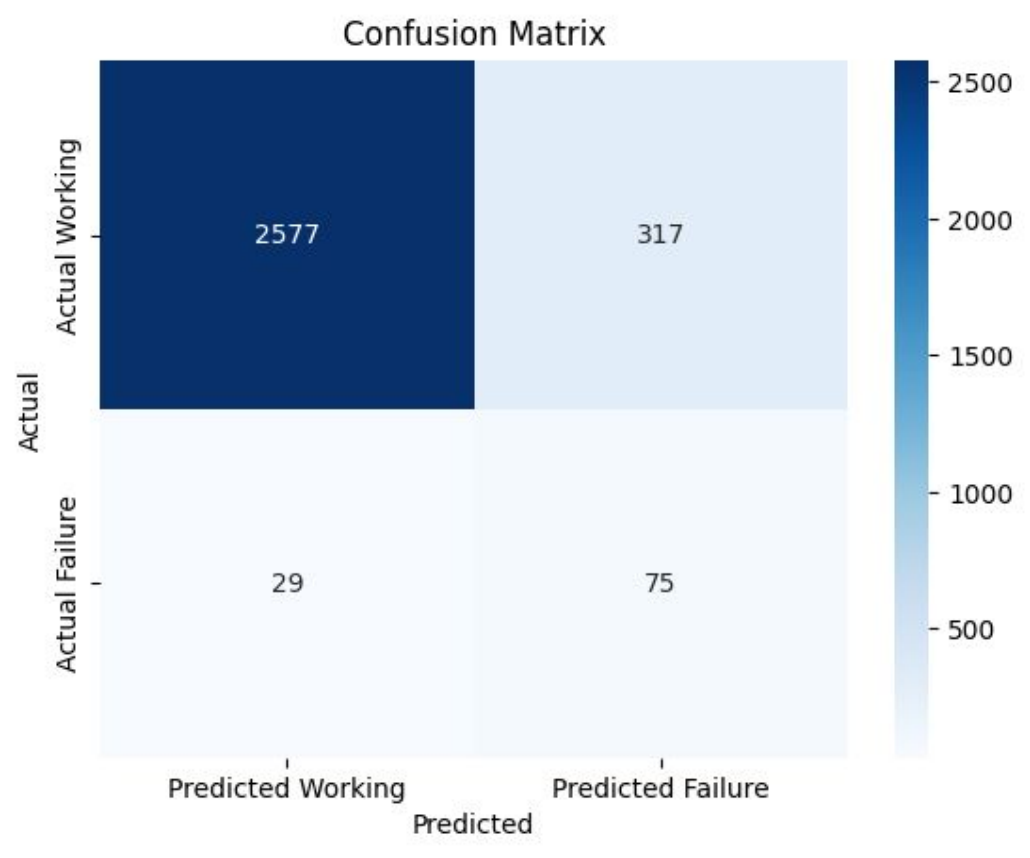
In this case, This Model Scored 90.31%.



Models used and their result:

CatBoostClassifier with Class Weight Adjustment

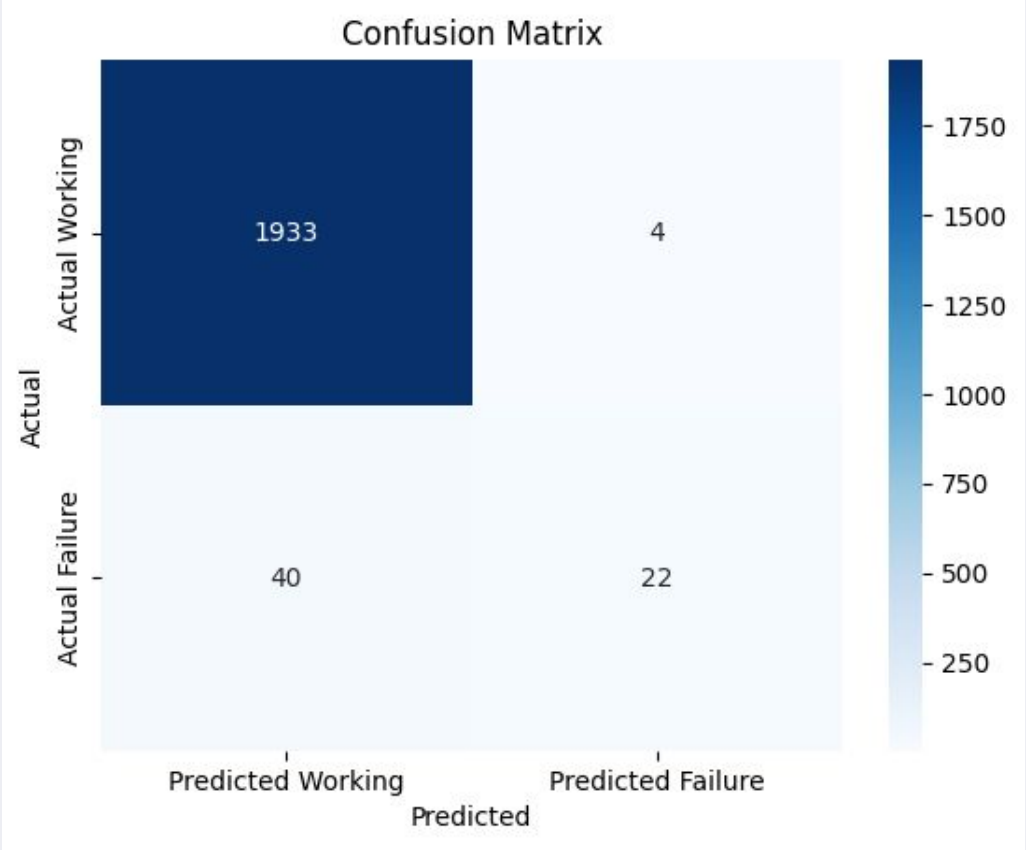
In this case, This Model Scored 86.78%.



Models used and their result:

CatBoostClassifier Combining Oversampling and Undersampling

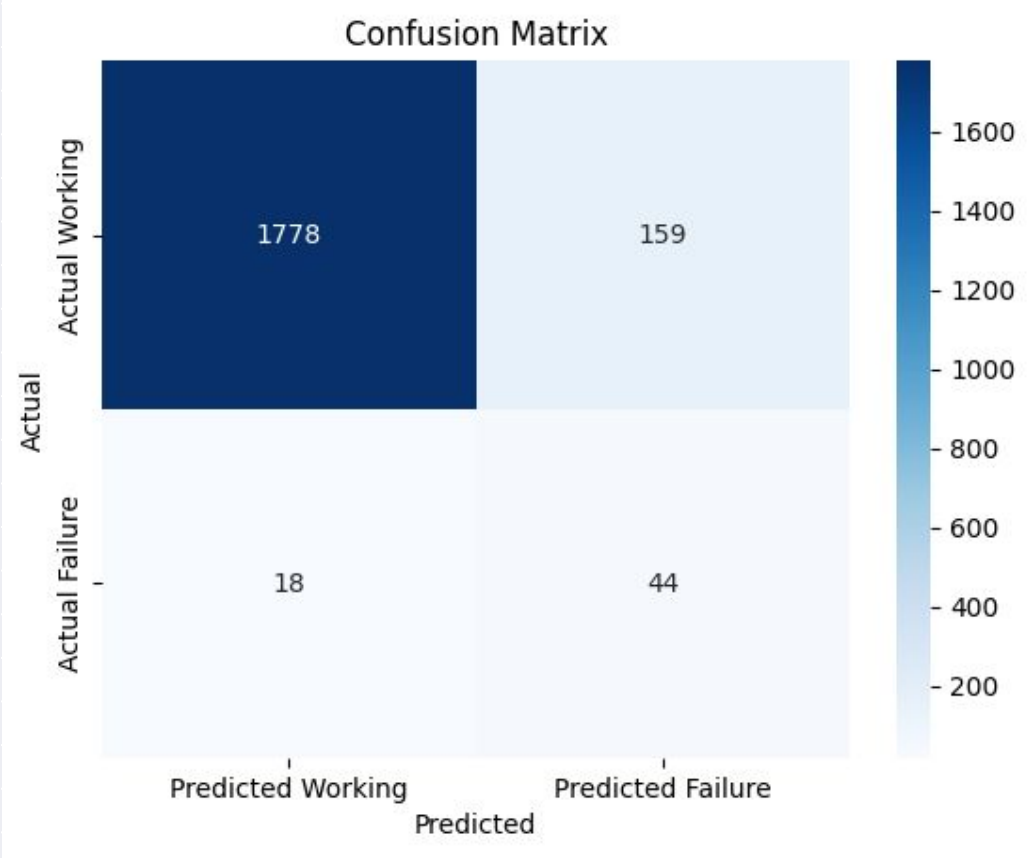
In this case, This Model Scored 90%.



Models used and their result:

Neural network model

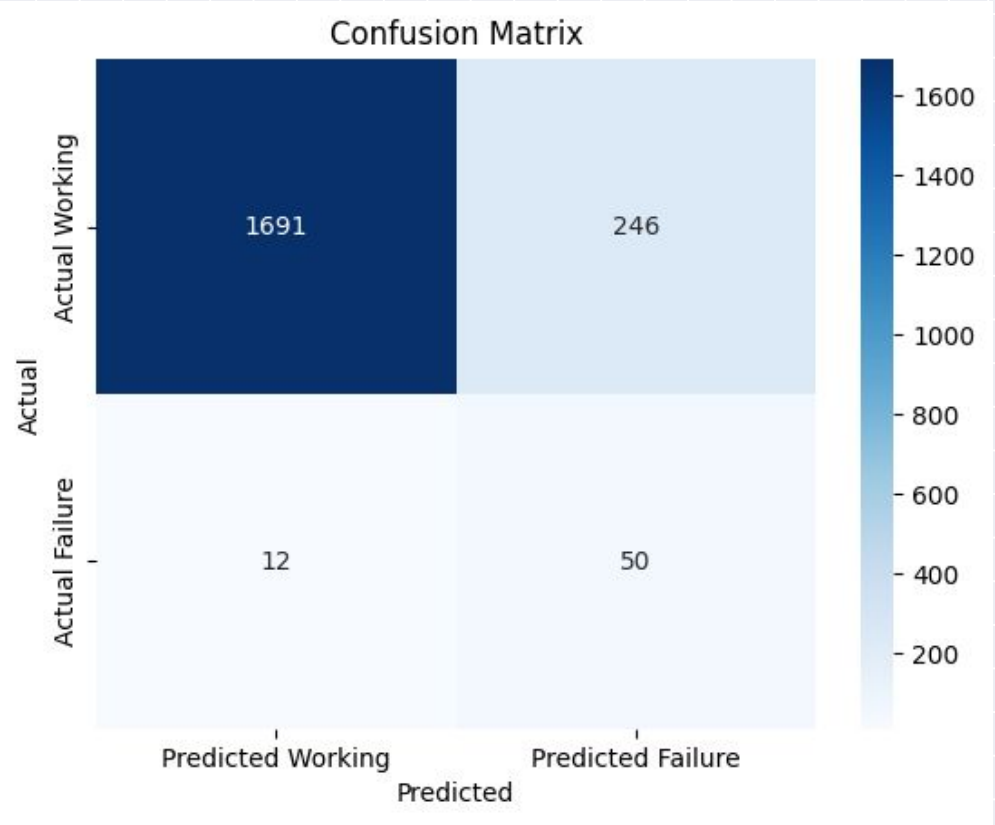
In this case, This Model Scored 89.40%.



Models used and their result:

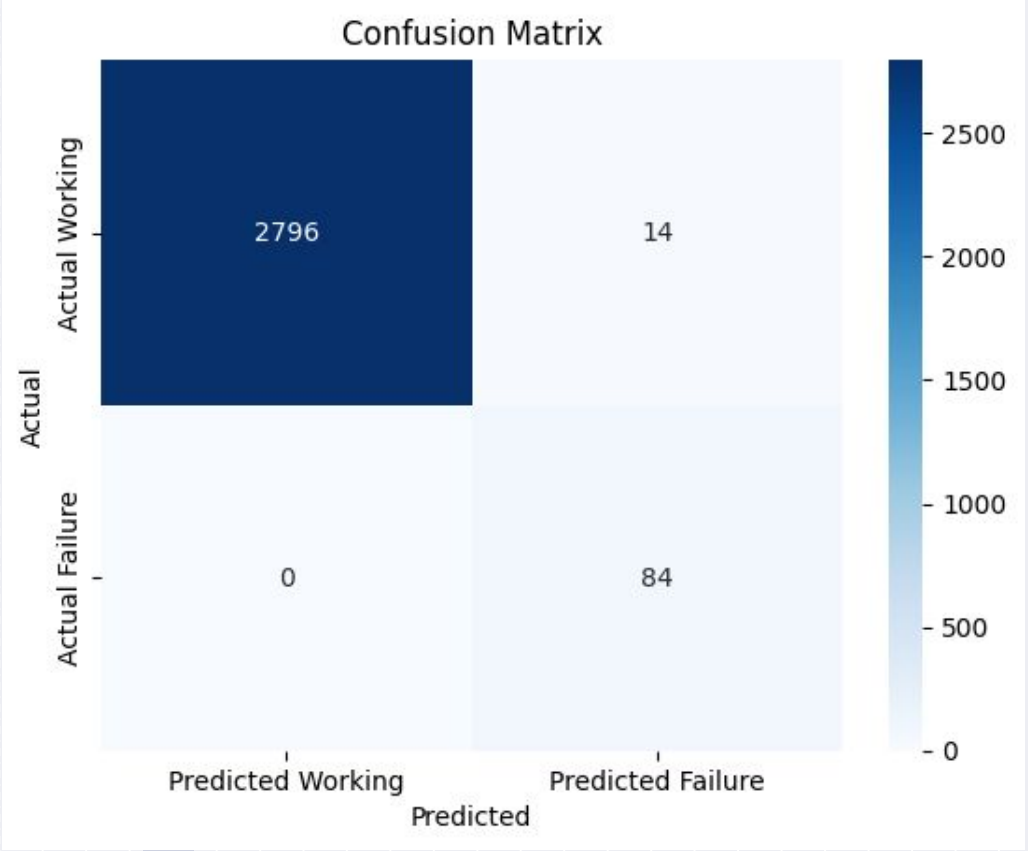
Neural network model using Dropout(0.5) to reduce overfitting and Early Stopping.

In this case, This Model Scored 90.39%.



ML models comparison and showing the best model:

In our case, a perfect model is one that has 0 false negatives (i.e., all actual failures are predicted correctly) and the minimum possible number of false positives (i.e., the fewest incorrect predictions of failure among actual working cases), as presented.



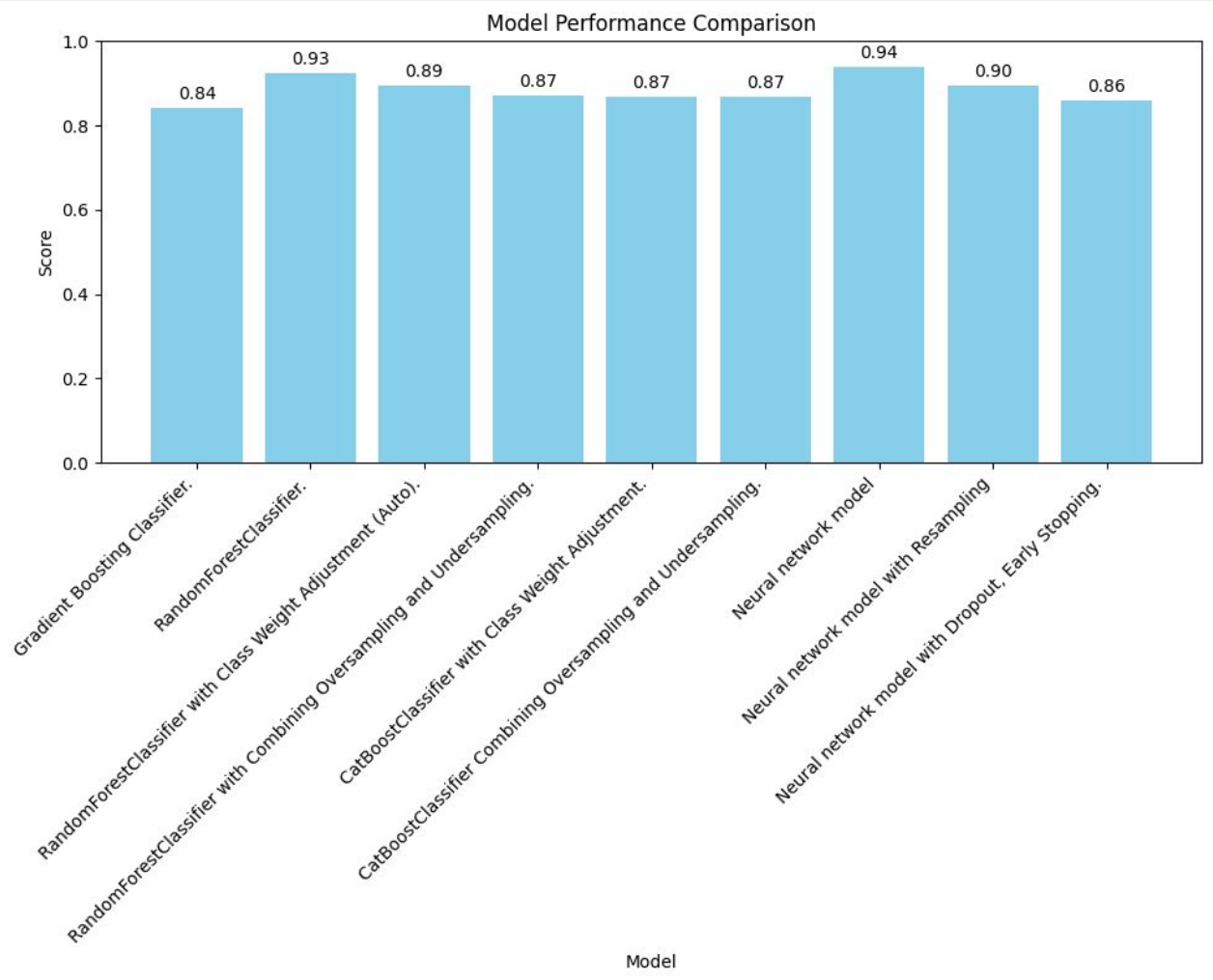
ML models comparison and showing the best model:

cince in our case, a perfect model is a model that has 0 (False Negatives) and the minimum possible of FP (False Positives), we calculate our score based on this formula:

$$\text{Score} = \frac{(TN + TP)}{(TN + TP + \alpha \times FP + \beta \times FN)}$$

Where:

- TN (True Negatives): Actual Working and Predicted Working.
- TP (True Positives): Actual Failure and Predicted Failure.
- FP (False Positives): Actual Working and Predicted Failure.
- FN (False Negatives): Actual Failure and Predicted Working.

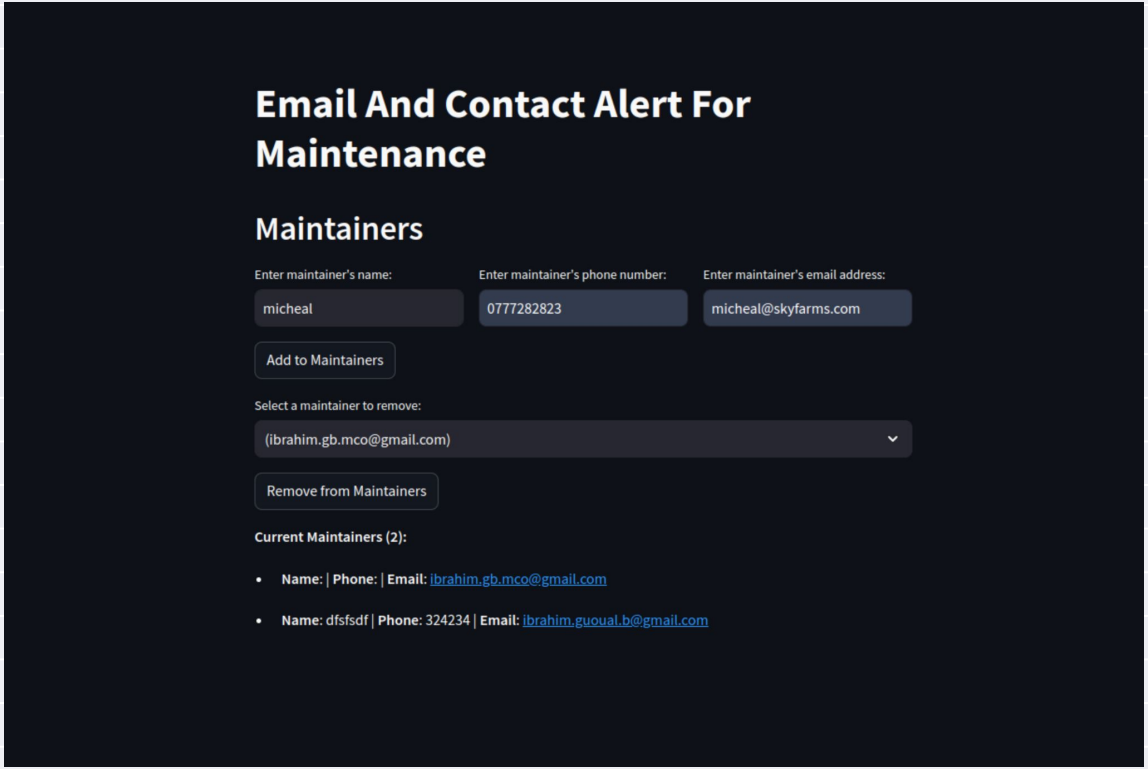

Model Performance Comparison
Score
1.0
0.8
0.6
0.4
0.2
0.0
0.84
0.93
0.89
0.87
0.87
0.87
0.94
0.90
0.86
Gradient Boosting Classifier.
RandomForestClassifier.
RandomForestClassifier with Class Weight Adjustment (Auto).
RandomForestClassifier with Combining Oversampling and Undersampling.
CatBoostClassifier with Class Weight Adjustment.
CatBoostClassifier Combining Oversampling and Undersampling.
Neural network model
Neural network model with Resampling
Neural network model with Dropout, Early Stopping.
Model

suggested solution

We have created a solution that alerts the maintenance team if our machine learning model detects a potential issue with a specific machine. The interface will allow administrators to add contact information for their employees.

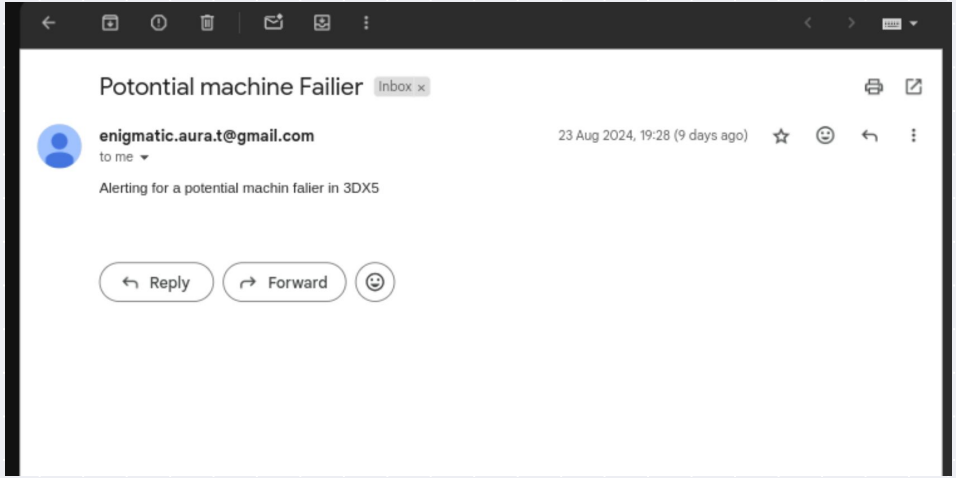
Additionally, we have developed a backend solution that sends notifications to the maintenance team via email.

While this solution is not intended to replace employees, it is designed to optimize their time by identifying potential failures for service and repair alongside regular maintenance."

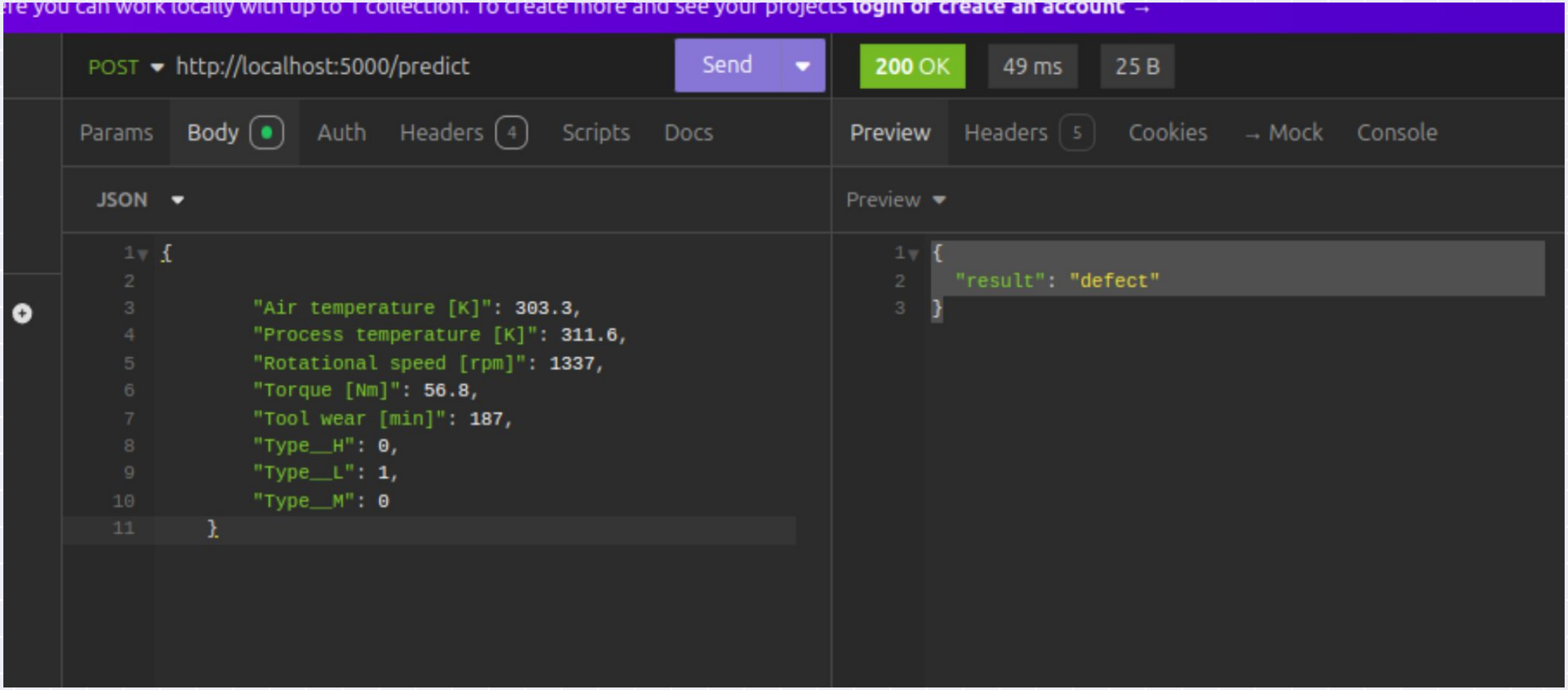


suggested solution

The email will be received by an employee, alerting him of the potential failure and alerting him with the machine ID

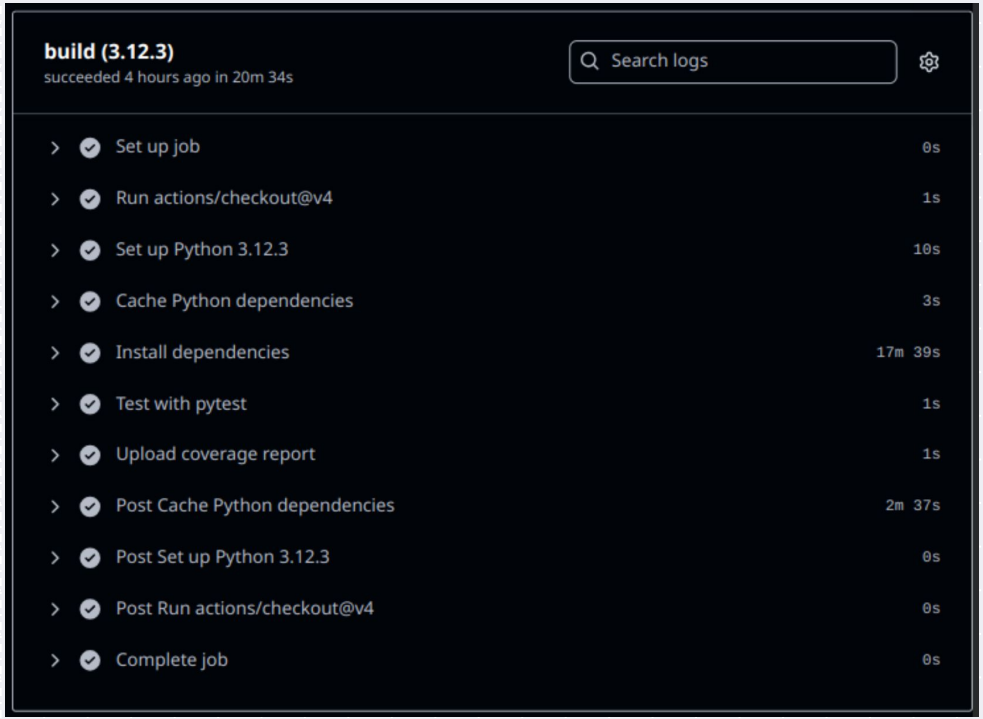


suggested solution



suggested solution

The project includes test script and pipeline for deployments, the project is available on https://github.com/ibrahimgb/BigProject



Merci !