1. Data Preprocessing: Clean and prepare the data set by removing any irrelevant or missing values. Scale the data as needed.
2. Feature Engineering: Create relevant features from the current readings that can help the model learn the pattern and detect anomalies. Features such as FFT (Fast Fourier Transform), RMS (Root Mean Square), crest factor, etc. can be considered.
3. Model Selection: Select a suitable machine learning algorithm for this problem such as Random Forest, Support Vector Machines (SVM), or Artificial Neural Networks (ANN).
4. Model Training: Train the model on the preprocessed and engineered data.
5. Model Evaluation: Evaluate the model's performance using metrics such as accuracy, precision, recall, and F1-score. If the model is not performing well, try different algorithms or adjust the hyperparameters.
6. Model Deployment: Once the model is performing well, it can be deployed to a production environment.

Documentation:

1. What has been done - logic:

The data set of current readings from a 3-phase AC motor has been used to detect anomalies. The data has been preprocessed, relevant features have been engineered, and a machine learning model has been trained and evaluated. The selected algorithm is Random Forest.

1. How it can be set up and tested in an isolated system:

The code can be set up using a Python environment with libraries such as pandas, numpy, scikit-learn, and matplotlib. The code can be tested in an isolated environment using Jupyter Notebook or a similar platform.

1. Creating an anomaly in the data set and showing that the code can detect it:

An anomaly can be created by adding a spike or a drop in the current readings. The code can then be run on this data set to verify that the model is able to detect the anomaly. This can be done using a test-driven development (TDD) approach.

The code can be deployed as a Docker container to ensure that the environment is isolated and the code can be run in any environment. The code can also be hosted on a cloud platform such as AWS or Google Cloud for easier deployment.

Top of Form

Bottom of Form