**Machine-Learning-Based Electrical Signal Analysis for Intrusion Detection in Industrial Systems**

In diesem Schritt, habe ich einen Autoencoder, eine Support-Vector-Maschine (SVM), Local Outlier Factor (LOF) und Density-Based Spatial Clustering of Applications with Noise (DBSCAN) Methoden implementiert und geprüft. Die Methoden wurden für verschiedene Scaler und verschiedene Initialisierungsparameter jeder Methode geprüft.

Für SVM, LOF und DBSCAN habe ich die folgenden Scaler angewendet:

- Kein Scaler, die Daten werden nicht wie in den Methoden vorverarbeitet und gesendet.

- Standard Scaler, entfernt den Mittelwert und skaliert die Daten auf Einheitsvarianz.

- MinMaxScaler skaliert den Datensatz neu, sodass alle Feature-Werte im Bereich [0, 1] liegen.

- MaxAbsScaler ähnelt MinMaxScaler, außer dass die Werte im Bereich [0, 1] abgebildet werden.

- Robust Scaler, die Zentrierungs- und Skalierungsstatistik von RobustScaler basiert auf Perzentilen und wird daher nicht von einigen wenigen sehr großen Randausreißern beeinflusst.

- PowerTransformer wendet eine Leistungstransformation auf jedes Feature an, um die Daten Gauß-ähnlicher zu machen, um die Varianz zu stabilisieren und die Schiefe zu minimieren. Derzeit werden die Yeo-Johnson- und Box-Cox-Transformationen unterstützt und der optimale Skalierungsfaktor wird bei beiden Methoden über die Maximum-Likelihood-Schätzung bestimmt.

- QuantileTransformer wendet eine nichtlineare Transformation an, sodass die Wahrscheinlichkeitsdichtefunktion jedes Merkmals auf eine gleichmäßige oder Gaußsche Verteilung abgebildet wird.

- Der Normalizer skaliert den Vektor für jede Probe neu, um unabhängig von der Verteilung der Proben eine Einheitsnorm zu erhalten.

Wir vergleichen jedes Ergebnis für jeden Scaler, um den Scaler auszuwählen, der uns das Ergebnis für die Erkennung von Anomalien liefert.

Um die besten Ergebnisse für jede Methode zu erzielen, habe ich mit den Parametern jeder Methode gespielt.

Für SVM habe ich den Parameter "nu" geändert, der eine Obergrenze für den Anteil der Trainingsfehler und eine Untergrenze für den Anteil der Unterstützungsvektoren ist. Durch Verringern dieses Parameters erhalten wir bessere Ergebnisse für die Erkennung von normalem Verhalten, aber schlechte Ergebnisse für die Erkennung von Anomalien. Durch Erhöhen dieses Parameters können wir bessere skalierungsabhängige Ergebnisse erzielen.

Für LOF spielen wir mit der Anzahl der Nachbarn, um die Dichteabweichung einer Probe zu berechnen. Durch Ändern dieses Parameters werden je nach Scaler unterschiedliche Verhaltensweisen beobachtet, was bei einigen Scaler zu guten Ergebnissen bei einem niedrigen Wert der Anzahl der Nachbarn führen kann, während bei anderen Scaler eine große Anzahl von Nachbarn erforderlich ist, um eine korrekte Anomalieerkennung zu erzielen.

Für DBSCAN spiele ich mit den Parametern, die den maximalen Abstand zwischen zwei Samples darstellen, damit sie in demselben Sample berücksichtigt werden, und die maximale Anzahl von Samples in einer Nachbarschaft, damit dies als Kernpunkt betrachtet wird. Diese beiden Parameter sind so konfiguriert, dass DBSCAN ein einzelnes Cluster erkennt, wobei der eine das normale Verhalten des Prozesses neu gruppiert und alle Ausreißer die Punkte sind, die eine Anomalie erkennen.

Im nächsten Schritt, werde ich die LSTM Methode implementiert. Ich werde mit dem Schreiben der Masterarbeit fortfahren.