

Detail Project Report

wafer fault detection

Dinesh Kumar Naik | Wafer Sensors Fault Identifier | 03-Aug-2021

## Problem Statement:

Most of the industrial process, company use sensor to automate their process. But to predict the quality of the wafer manually is very difficult. So we wanted to build the machine learning model, which can predict the wafer quality before it got damaged completed.

## Value of the Project:

Wafer are electronic device having with photo voltaic cells, this is used in production line to automate the process. If any sensor fail then entire production line needs to stop and then change the wafer manually.

We build the machine learning model to identify the faulty wafer and give alarm, which any issue detects on wafer, so only that unit can be stopped to change the wafer and other unit runs without any hassle.

## System Architecture:



**Storage:** Mongodb

**Cloud: IBM Watson Cloud:**

**API:**

**Core tech Stack:** In wafer detection model, we get data in multiple sets of csv file as input in specific format, data will contain wafer names and 590 columns of different sensors, last column have +1 and -1 value, which denotes +1 wafer means bad wafer and -1 wafer means good wafer.

In these files, we perform different type of data validation and based on the validation we segregate bad data or good data. We move these files in different folder based on the good data and bad data, similarly store the data in different folders on the basis of good and bad.

The data stored in database is exported as a CSV file to be used for pre-processing and model training.

KMeans algorithm is used to create clusters in the preprocessed data. The optimum number of clusters is selected by plotting the elbow plot, and for the dynamic selection of the number of clusters, we are using "KneeLocator" function. The idea behind clustering is to implement different algorithms.

After clusters are created, we find the best model for each cluster. We are using two algorithms, "Random Forest" and "XGBoost". For each cluster, both the algorithms are passed with the best parameters derived from GridSearch. We calculate the AUC scores for both models and select the model with the best score. Similarly, the model is selected for each cluster. All the models for every cluster are saved for use in prediction.

**Data Pre-Processing:**

* **Data Validation & Preprocessing:**
  + **We perform different sets of validation & preprocessing on the given set of training files.**
    - CSV File Name
    - No of Column
    - Name of Columns
    - Datatype of Columns
    - Null Value imputation
    - Remove columns, which doesn’t impact the model result during training(Zero Std. Deviation)
* **Database Insertion:**
  + Insert Bad Data
  + Insert Good Data
* **Algorithms:**
  + KMeanse Clustering
  + Random Forest
  + XGBoost

**Hosting:**

**Dashboarding:** Flask

**Monitoring:**

**Hypercare:**

**Question Answers:**

**Q1. Tell me about your current project.**

In this POC, we build the automation process to predict the wafer, if it is working fine are any issue detected. We build the machine learning model to identify the faulty wafer and give alarm, which has any issue detects on wafer, so only that unit can be stopped to change the wafer and other unit runs without any hassle.

We have used Kmeans Clustering for clustering the data and execute that clustering on Random Forest and XGBoost model to get the better performance.

**Q3. What was the data type.**

All the data is numeric (Int. Float)

**Q3. What technique for data preprocessing, could you please explain the method?**

We perform different sets of validation & preprocessing on the given set of training files.

* + - CSV File Name
    - No of Column
    - Name of Columns
    - Datatype of Columns
    - Null Value imputation
    - Remove columns, which doesn’t impact the model result during training(Zero Std. Deviation)

**Q4. What is clustering and why it is needed?**

Clustering is group of datasets, which has similar kind of characteristics. So we do execute to divide the datasets in cluster. These cluster dataset could better performance on different machine learning model, So get the better performance we do clustering on datasets.

**Q5. Please tell us other clustering model?**

* DBSCAN (Density based Spatial Clustering of Application with Noise)
* Hierarchical Clustering
* KMeanse Clustering

**Q6. What is KMeanse clustering and how it works?**

KMeanse clustering is unsupervised machine learning algorithms. K-means algorithm identifies k number of centroids and then allocate every datapoint to the nearest cluster.

**Q7. In your project how you identify the no of cluster.**

To identify the K, we run the loop from k=1 to k<=40 and pass the K value to the KMeanse algorithms. The optimum number of clusters is selected by plotting the elbow plot, and for the dynamic selection of the number of clusters, we are using "KneeLocator" function.

**Q8. What is elbow plot and what is the purpose?**

Elbow plot is graph, which is used to see the deviation of the data point move through graph.

**Q9. What is kneeLocator in KMeanse clustering?**

The knee point is loosely defined as the point of maximum curvature in as system. In machine learning, it is used to select appropriate value of K in K-means clustering.

**Q10. What is random forest and how it works?**

A random forest is a machine learning technique that’s used to solve regression and classification problems. It utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems.

A random forest algorithm consists of many decision trees. The ‘forest’ generated by the random forest algorithm is trained through bagging. Bagging is an ensemble meta-algorithm that improves the accuracy of machine learning algorithms.

**Q12. What is the feature of random forest algorithm?**

* It’s more accurate than the decision tree algorithm.
* It provides an effective way of handling missing data.
* It can produce a reasonable prediction without hyper-parameter tuning.
* It solves the issue of overfitting in decision trees.
* In every random forest tree, a subset of features is selected randomly at the node’s splitting point.

**Q13. What is XGBoost?**

XGBoost is supervised machine learning algorithm, implementation of gradient boosted decision trees designed for speed and performance