PAPER • **OPEN ACCESS**

Predictive Maintenance using Machine Learning Based Classification Models

IOP Conf. Series: Materials Science and Engineering **954** (2020) 012001 doi:10.1088/1757-899X/954/1/012001

Predictive Maintenance using Machine Learning Based Classification Models

Anisha Chazhoor[1]*, Mounika Y[1], Vergin Raja Sarobin M[1], M V Sanjana[1], Yasashvini R[1]

[1]School of Computer Science and Engineering, Vellore Institute of Technology, Kelambakkam - Vandalur Road, Rajan Nagar, Chennai, Tamil Nadu- 600127 India.

Email: anisha.chazhoor2017@vitstudent.ac.in

Abstract. Machine learning facilitates predictive maintenance due to the advantages it holds over traditional methods of maintaining semi-conductor devices such as preventive and breakdown maintenance. Several predictive models using machine learning on the Semiconductor Manufacturing process dataset (SECOM) will be applied in this paper. The dataset contains the information related to semiconductor manufacturing process, with the attributes corresponding to signals collected from semiconductor devices. Due to the high-dimensionality of the data and class imbalance problem in the SECOM dataset, it poses several challenges related to data pre-processing, which is an essential step incorporated in this work while applying various machine learning models. Comparison and analysis of various predictive machine learning classification models were carried out based on the performance metrics like, accuracy and Receiver Operating Characteristic (ROC) curve.

1. Introduction

Large volumes of data are generated and shared every single day. Analyzing such large volume of data to make meaningful results is a taxing job and manually impossible. Developing a method to analyze data to find patterns and to achieve meaningful interpretations is necessary for any application domain for making informed decisions. But it is possible to apply various supervised or unsupervised machine learning algorithms such as Random Forest, Logistic Regression, Support Vector Machine and clustering algorithm to do better predictions [1]. The efficiency and time taken by each algorithm can be compared to reach the expected output. SECOM dataset consists of 591 features which are the signals generated from the semiconductor device to sense anomalies. It contains only 104 fail cases. Thus, it is an example of class imbalance problem. The dataset is not very large as it has only 1567 examples. But it has high dimensionality. It could hence cause problems like the curse of dimensionality.

Hence, to address these problems in the dataset, data pre-processing [2-3] is required, which is an essential pre-requisite for the application of machine learning models. Approaches like Synthetic Minority Oversampling Technique (SMOTE) analysis, correlation, feature selection etc., could be applied to remove initial hurdles like class imbalance, correlated features etc. After this step, machine learning models are applied and compared in order to select the best possible prediction approach according to the needs. The organization of the paper is as follows. The review of related work to predictive maintenance models in semiconductor manufacturing process and handling high-dimensional, class-imbalanced datasets is presented in Section II. Then a description of various predictive machine learning models is given in Section III. A series of experimentation and results are presented in Section IV. Finally, Section V concludes the findings of our paper regarding the predictive models.

2. Related Works

In one approach, Anghel et al., [1] compare and contrast two different predictive models and apply it to SECOM dataset. Pre-processing is performed, in which null and redundant data is removed. Over-sampling is performed to overcome the problem of class imbalance. Next, feature selection is performed using two approaches: one with Multivariate Adaptive Regression Splines (MARS) algorithm and another with Support Vector Machine (SVM) algorithm. After the important features are selected, Gradient Boosting Trees (GBT) is applied on MARS and Neural Networks using Tensor Flow is applied on SVM. They conclude that the approach involving NN is more effective [2].

In another paper, Kerdprasop et al., have developed a unique algorithm for pre processing of the SECOM dataset called the MeanDiff algorithm, which creates clusters on the basis of pass or fail cases, compare value differences and then rank features according to the calculated value differences. Also, columns with more than 55% null values and redundant data are removed. The performance of the clean data on Naïve Bayes, Decision Tree, k-Nearest Neighbors (k-NN) and Logistic regression algorithms are compared. It is concluded that though Naïve Bayes provides the best performance, the false positives are very high. On the other hand, decision tree provides low false positives and false negatives [3]. In another well-known approach, Verdier et al., replace the Euclidean distance method with the Mahalanobis method. The machine learning algorithm used is k-NN. While the traditional method used involves Hotelling-T2 model, this method involving Hotelling-T2 has several disadvantages. For example, it works only on Gaussian data which is not always relevant, as mostly real-life data is non-Gaussian. Thus, the proposed method eliminates the need to convert data to Gaussian data [4].

In the research work, Munirathinam et al., [5] have worked on methods based on machine learning techniques to build an accurate model for fault detection. Feature selection techniques that range from removing features with constant value and features containing missing values (above 55%) to statistical analysis such as chi-square and Principal Component Analysis (PCA) are used. To overcome problems like over fitting, more computational power and less prediction accuracy, techniques like subject matter expert knowledge, correlation analysis and variable component analysis have been used. The main disadvantage is that by duplicating of existing examples (over sampling), it makes the model prone to over fitting. To increase the prediction probability of rare classes, deep learning is required. A deep belief network is a deep learning technique that supports both supervised and unsupervised learning. The nodes and the number of layers constituting a deep belief network are different and it depends on the dataset used. It is difficult to obtain global optimization using gradient descent.

A new technique called particle swarm optimization is used to improve the performance and to carry out global optimization. The objective of the proposed technique is to exhaust the search space of a problem with the intention of finding the parameters which maximize the required objective. This algorithm imitates the particles behavior in a swarm. The position of a particle x is determined by its location in the kth step and its velocity in the (k+1)th step. The optimal position can be determined from the input parameters, which are the position of the particles after each step [6]. Computing the distances between the data points is computationally expensive and time consuming. To reduce the computation time, many fast k-NN search algorithms like the a Principal Axis search Tree (PAT) algorithm, Lower Bound search Trees (LBT), Modified principal axis search tree (MPAT), and Orthogonal Search Trees (OST) were proposed. The performances of all the above-mentioned algorithms were compared to find the best k-NN search algorithm. By conducting experiments on various datasets, it is seen that the OST algorithm performs well for most cases. For a data set from real images, the LAI algorithm is a good choice. If a data set with a huge number of dimensions and the pre-processing time is important, the MPAT method becomes the best choice. The disadvantage of this technique is that it is highly influenced by the number of dimensions, number of data points, and data distribution of a data set [7].

A multiple classifier machine learning approach for Predictive Maintenance (PdM) is proposed. Susto et al., propose the creation of dynamical decision rules to be adopted for the generation of quantitative indicators, associated with the possible problems that could damage the system in question. The relationship between these factors and the operational costs and failure risk could also be determined. This paper proposes training of multiple classification modules, to provide different trade-offs between the performance determining factors. The information gathered will be used to make an operational cost-based decision system for maintenance. This approach is suitable for semiconductor datasets, given their size, because this method works well for high-dimensional and censored datasets. This method is proved to have been working better than classical preventive maintenance (PvM) approaches and single SVM classifier [8].

SECOM dataset, being a high-dimensionality dataset, originally requires a two-step framework consisting of the dimensionality reduction and machine learning algorithm (like SVM) steps, for its analysis. But this demands high storage (of original data) and processing capacity (in the pre-processing stage) from the system, which increases the complexity. This paper proposes a unique algorithm called Incremental Projection Vector Machine (IPVM), which combines new dimensionality reduction feature in incremental format combined with feed-forward NN training simultaneously. Thus, when new samples are input in the system, the task of creating SVD becomes easier as one need not compute the full rank version on the entire dataset, instead just update the existing version. The results show that this method is far better than the two stage learning combinations like (SVD, BP). Hence, this approach is highly suitable for high-dimensional, large sample data like the SECOM dataset [9]. The SECOM dataset has class imbalance issues along with high dimensionality. The class imbalance is handled with SMOTE analysis. The algorithms used are Naive Bayes, k-NN and decision trees with the metrics Chi-square, Information gain metrics and Relief [10]. Not all data collected and features are relevant. So simple data manipulation or data transformation is required. This is often achieved through pre-processing steps like standardization, normalization, signal enhancement (smoothening and sharpening), principle component analysis and multidimensional scaling.

McCann et al., investigate various feature selection techniques and study how accurate they are in recognizing the causal effects in the SECOM semi-conductor manufacturing. After using these feature selection techniques along with simple algorithms like Naïve Bayes, it is concluded that a hybrid approach involving the appropriate feature extraction techniques and the existing business improvement techniques be developed to improve performance [11]. The semi-conductor manufacturing process is based on the concept of multi-stage manufacturing system. It refers to the system that involves more than one workstation to manufacture the final product akin to the model adopted by several automobile industries. Cascade Quality Prediction Method (CQPM) focuses on analysing the complex variable relationship in multi-stage manufacturing using a combination of multiple principle component analysis and iterative Dichotomiser 3 algorithm to extract rules. Ten-fold cross validation scores were calculated to compare the predicted models. It is observed that CQPM performs better than the other models. From the relatively low value of geometric mean, it is evident that the probability of misclassification in negative class is still high [12].

In the research work [13], Narul Afiqah A. Majid et al., presents the use of mechatronics technology that can help to implement predictive maintenance in mechanics by combining intelligent and predictive maintenance instrument. This paper uses the above concept and shows the choice of the appropriate strategy in the vibration of diagnostic process of the mechanical system. It discusses the role of mechatronics as a prediction tool behind the use of signature analysis of rotary machines. It proposes a method called Vibration Fault Simulation System (VFSS), a simulation system for detecting the unique vibration faults signatures of a rotating machine. Each time, the VFSS system is tested on various types of faults such as vibration, imbalanced rotor disc or ball, mechanical looseness, etc and signature of each fault is noted. The paper concludes that vibration is the most reliable parameter to be monitored for predictive maintenance purposes.

In the research work [14], Tayaba Abbasi et al., discusses the use of Recurrent Neural Network (RNN) using long short-term memory (LSTM) to carry out predictive maintenance of Air booster compressor (ABC) motor, an essential equipment in the oil and gas industry. Two most popular training algorithm i.e., Levenberg Marquardt and Bayesian Regularization were compared for training the network. Levenberg Marquardt has proven to be the fastest, generating lower root mean square error. The optimal number of epochs and hidden layers are 750 and 15 respectively. The main advantage of this algorithm is that it does not require any prior expertise or feature engineering. It can also perform multistep ahead prediction with high prediction accuracy. In the research work[15], Xiang Li et al., uses Random Forest Regression model to implement IoT equipment maintenance predictive model. This machine learning algorithm was chosen since it performs well on the data compared to other algorithms. The "equipment life prediction system" runs under the Ubuntu system. The algorithm successfully passes the unit testing.

This paper combines the actuality of equipment maintenance forecasting work, based on the engine life prediction data, and systematically discusses the construction processes and presents a Python implementation for the same.

3. Machine Learning Models

Random forest is a type of ensemble learning, which uses the approach of generating multiple trees simultaneously and vote for the most accurate tree classification is presented in Algorithm1.

Algorithm 1: Random forest

1: **for** *i*←0 to number of trees, *n*
2: **for** *i*←0 to number of nodes, *l*
3: Select *k* out of *m* features.
4: Calculate the node *d* using best split point based on highest information gain.
5: Split the node into daughter nodes.
6: **end for**
7: **end for**
8: **for** *i*←0 to *n*
9: Use generated rules to predict output
10: Calculate votes for each predicted output.
11: **end for**
12: **return** highest voted predicted target as the final prediction.

Logistic regression is one of the simplest and most efficient algorithms to implement binary dependent variable problems. It is a statistical approach with the main objective being to calculate the weights associated with the variables is explained in Algorithm 2.

Algorithm 2: Logistic regression

1: Randomly generate *w*=(*w[0],w[1],..,w[n-1],1)*
2: **while** $\nabla E_{in} > \epsilon$

3: **for** *i*←0, 1, 2...
4: Compute gradient :
5: $\nabla E_{in} = -\frac{1}{N}\sum_{n=1}^{N}\frac{y_n x_n}{1+e^{y_n w^T(t)x^n}}$
6: Compute gradient
7: $w(t+1) \leftarrow w(t) - \eta \nabla E_{in}$
8: **end for**
9: **end while**
10: **return** w

Algorithm 3: Decision tree

1: Set training instances to root
2: Current *node*←*root*
3: **for** *i*←*c*
4: Calculate information gain for every attribute

$E(S) = \sum_{i=0}^{c} -p_i \log_2 p_i$
where c is the number of classes and p_i is the probability of randomly selecting class i.
$E(T,X) = \sum_{C \in X} P(c)E(c)$ where E(T,X) is the average entropy of the child attributes.
Information Gain=E(S)-E(T,X)
6: **end for**
7: Find feature with greatest information gain.
8: Set this feature to be the splitting criterion at the current node.
9: **if** *information gain*←0
10: **return** current *node*←*leaf*
11**: end if**

Decision tree classifies the dataset into classes based on the attribute value. It utilizes the concept of attribute selection. There are several methods of performing attribute selection, with the most popular ones in practice being using information gain and Gini index. The given algorithm is implemented for information gain methodology is explained in Algorithm 3. The multilayer perceptron as its name suggests has multiple layers with the lower most layer being the input layer followed by a number of hidden layers and the output layer at the top. All neurons in one layer are fully connected to the ones in adjacent layer. It utilizes the back-propagation method which is shown in Algorithm 4.

Algorithm 4: Multi-layered perceptron

Procedure MLP
1: Take input layer, forward propagate the patterns of training data and generate an output.
2: Calculate error, ϵ
3: Minimize ϵ using cost function f
4: Call Back_propagate()
5: Find derivative with respect to each weight *w[i]* in the network, and update model.
Procedure Back_propagation
1:**for** every node in the output layer
2: calculate error signal
3: **end for**
4: **for** all hidden layers
5: **for** every node in layer
6: Calculate signal error
7: Update each node's weight in the network
8: **end for**
9: **end for**

4. Implementation and Results

The dataset has been fitted to several machine learning models such as k-NN, decision tree, logistic regression and multi-layer perceptron. Univariate analysis has also been implemented. As the number of features is very high, several dimensionality reduction techniques such as principal component analysis and Linear Discriminant Analysis have been implemented. Accuracy and cross-validation scores have been noted. Boosting algorithms such as AdaBoost have been implemented and the accuracy scores are compared with k-NN algorithm. The efficiency of the models can be judged using their ROC curves. The first step in pre-processing is to find the ratio between the pass and fail cases. As we can see in the Figure 1, the ratio of pass: fail is almost 14: 1. This makes it a case of class imbalance which needs to be dealt separately.

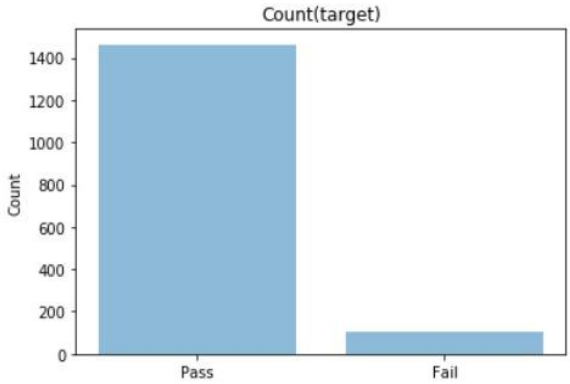

Figure 1. Count of pass and fail cases

The models that have been implemented are k-NN, logistic regression, MLP, random forest, AdaBoost and decision trees. Various Dimensionality reduction methods like PCA, LDA are used. The percentage of accuracy scores have been noted and compared simultaneously in Table 1. It could be observed that Logistic Regression with False Discovery Rate is the best performing algorithm and Random Forest with LDA is the least performing algorithm with respect to accuracy.

Table 1. Comparison of various algorithms with PCA and LDA

Algorithm	With PCA	With LDA	With False Discovery Rate
Random Forest	93.87	85.20	94.13
MLP	91.63	89.03	91.32
Logistic Regression	90.56	88.77	94.64
AdaBoost	91.07	89.03	91.58

Decision trees have also been implemented with dimensionality reduction techniques and the accuracy scores have been noted. The comparative study of the algorithms has been shown in Table 2. The dimensionality reduction with respect to decision trees is in terms of pruning to avoid over fitting. All values are in %.

Table 2. Comparison of accuracy various algorithms with and without dimensionality reduction

Algorithm	Without Dimensionality	With Dimensionality
Random Forest	76.588	94.13
MLP	90.051	91.63
Logistic regression	88.52	94.64
AdaBoost	92.09	92.602
Decision trees	86.48	93.622

The class imbalance problem is solved by SMOTE analysis. The accuracy before and after SMOTE sampling are recorded in Table 3. All the accuracy values are in %. The cv is set as 5.

Table 3. Comparison of mean cross validation scores and accuracy scores after SMOTE sampling on various algorithms

Algorithm	Mean cross validation score	Using SMOTE
Random Forest	92.40	94.3877
MLP	77.82	88.9285
Logistic regression	81.70	84.693
k-NN	92.46	55.357

The dependencies of the attributes with each other and their influence in determining the predicted output can be visualized using a correlogram as given below in Figure 2. PCA has been used as a pre-processing method with the intention of performing dimensionality reduction. The Explained Variance Ratio is defined as the fraction of variance of the current principal component with the total variance. As one can notice in the graph, the Explained Variance Ratio increases largely at the initial components and stabilizes after around 150-200 components. The said graph is shown as Figure 3.

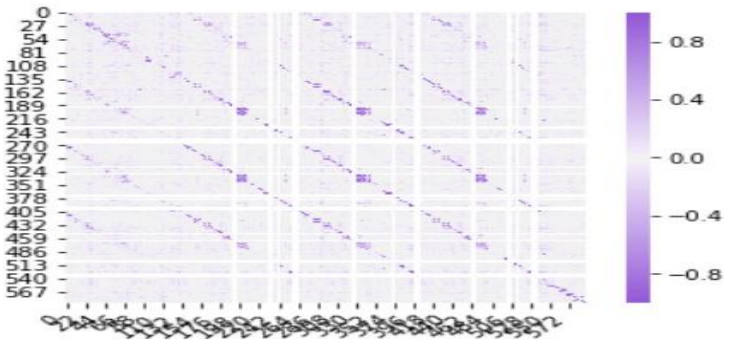

Figure 2. Correlogram showing dependencies between attributes

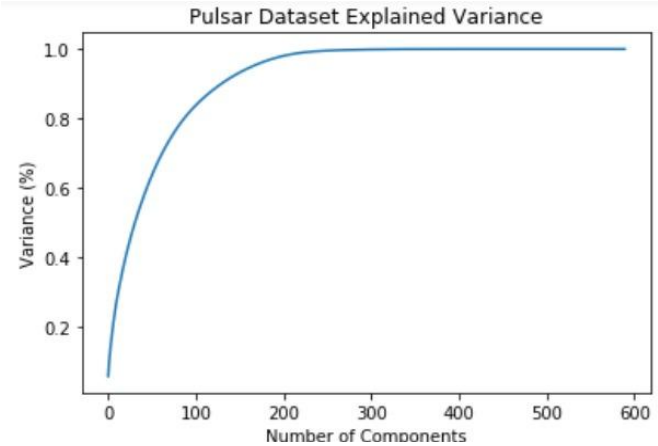

Figure 3. Explained variance ratio for PCA analysis

The ROC curves have been implemented for four machine learning models. The Area Under the ROC Curve(AUC) values describe the efficiency with which a certain parameter can be used to distinguish between two groups. Initially, the ROC curve has been implemented for the simplest algorithm, logistic regression. The results are shown in Figure 4.

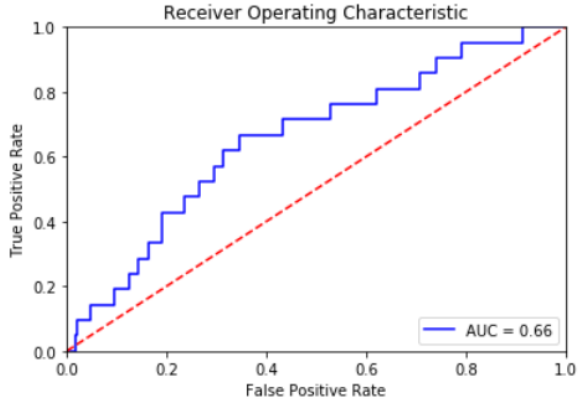

Figure 4. ROC curve for logistic regression

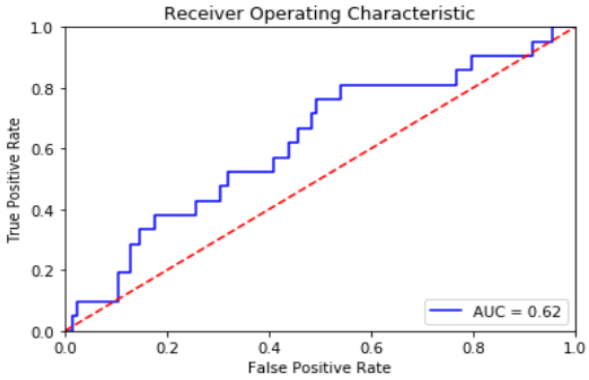

Figure 5. ROC curve for MLP

The ROC curve of MLP has been implemented in the Figure 5 given below. The ROC curve for k-NN has been implemented in Figure 6.

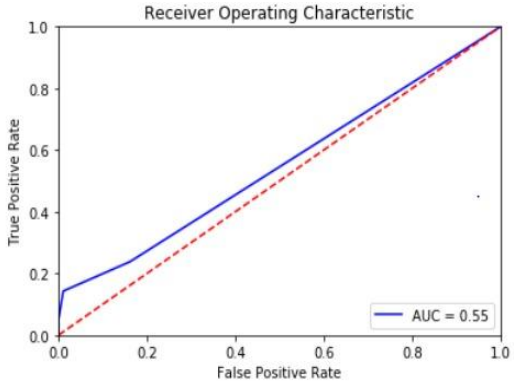

Figure 6. ROC curve for k-NN

The ROC curve for random forest is given in Figure 7.

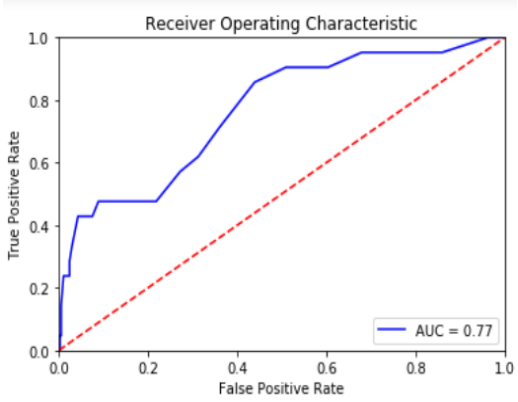

Figure 7. ROC curve for random forest

5. Conclusions

With the extensive comparative study performed, it can be concluded that, random forest performs poorly on SECOM dataset due to class imbalance problem. Logistic regression performs considerably better with a commendable accuracy with PCA. The model fitting most accurately to the dataset is Logistic Regression with False Positive Rate. Due to the class imbalance problem, certain algorithms do not perform up-to the mark and special hybrid algorithms like evolutionary machine learning could be integrated. SMOTE technique helped in the enhancement of the accuracy in Random Forest algorithm. The mean cross-validation score is the highest when cross-validation technique is used with k-NN algorithm.

6. References

[1] Gondalia, A., Dixit, D., Parashar, S., Raghava, V., Sengupta, A. and Sarobin, V.R., 2018. IoT-based healthcare monitoring system for war soldiers using machine learning. Procedia computer science, 133, pp.1005-1013.
[2] Anghel, I., Cioara, T., Moldovan, D., Salomie, I. and Tomus, M.M.,*2018 IEEE 16th International Conference on Embedded and Ubiquitous Computing (EUC) (pp. 29-36*).
[3] Kerdprasop, K. and Kerdprasop, N., *2011. Int. J. Mech, 5(4), pp.336-344.*
[4] Verdier, G. and Ferreira, A., *2010,IEEE Transactions on Semiconductor Manufacturing, 24(1*), pp.59-68.
[5] Munirathinam, S. and Ramadoss, B., *2016,IACSIT International Journal of Engineering and Technology, 8(4), pp.273-285.*

[6] Kim, J.K., Han, Y.S. and Lee, J.S., *2017. Concurrency and Computation: Practice and Experience, 29(11), p.e4128.*
[7] Liaw, Y.C., *2011. ,International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV) (p. 1).*
[8] Susto, G.A., Schirru, A., Pampuri, S., McLoone, S. and Beghi, A., *2014, IEEE Transactions on Industrial Informatics, 11(3), pp.812-820.*
[9] Zheng, Q., Wang, X., Deng, W., Liu, J. and Wu, X., *2010, December, Australasian Joint Conference on Artificial Intelligence (pp. 132-141). Springer, Berlin, Heidelberg.*
[10] Bach, M. and Werner, A., *2017, September, International Conference on Information Systems Architecture and Technology (pp. 182-194). Springer, Cham.*
[11] McCann, M., Li, Y., Maguire, L. and Johnston, A., *2010, Causality: Objectives and Assessment (pp. 277-288).*
[12] Arif F., Suryana, N., Hussin, B., *2013.,Ieri Procedia, 4, pp.201-207*
[13] Nurul Afiqah A. Majid, Asan G.A Muthalif, *2017, IOP Conf. Ser.: Mater. Sci. Eng. 260 012006*
[14] Tayaba Abbasi *, King Hann Lim, Ke San Yam,*2019 IOP Conf. Ser.: Mater. Sci. Eng.495 012067*
[15] Xiang Li,Li Wei , Jianfeng He, *2018, IOP Conf. Ser.: Mater. Sci. Eng.466 012001*