Prediction of Tool Wear Using Machine Learning Approaches for Machining on Lathe Machine

Ashish Kumar Srivastava
Department of Mechanical Engineering, Goel Institute of Technology and Management

Bipin Kumar Singh
Department of Mechanical Engineering, Sri Eshwar College of Engineering

Gupta, Supriya
Department of Mechanical Engineering, Goel Institute of Technology and Management

https://doi.org/10.5109/7151683

出版情報：Evergreen. 10 (3), pp.1357-1365, 2023-09. 九州大学グリーンテクノロジー研究教育センター
バージョン：

Prediction of Tool Wear Using Machine Learning Approaches for Machining on Lathe Machine

Ashish Kumar Srivastava[1], Bipin Kumar Singh[2,*], Supriya Gupta[1]

[1]Department of Mechanical Engineering, Goel Institute of Technology and Management, Lucknow-226028, Uttar Pradesh, India

[2]Department of Mechanical Engineering, Sri Eshwar College of Engineering, Coimbatore - 641202, Tamil Nadu, India

*Author to whom correspondence should be addressed:
E-mail: bipinmech2008@gmail.com

(Received January 19, 2023; Revised June 23, 2023; accepted August 2, 2023).

Abstract: In manufacturing industries, removal of material from the workpiece is the prime processes that convert raw material into finished product. During removal processes the cutting tool are incessantly deteriorated in health, which can be stated as perks and drawbacks of process. The precision and roughness of the material are directly related to the condition of the tools during the machining process. Machining analysis depends on numerous of cutting conditions when it is being performed. The likelihood of wearing increases with repeated use. So, by implementing the proposed approach for tool wear prediction can improve the quality as well as reduce the machining time. However, to maintain the healthy tool's conditions for prolong time is a major challenge for the scientific community. Hence, as a component of industry 4.0, this study explored the possibilities to monitor the tool condition using the machine learning techniques. So, an endeavor has been made to present a solution of this problem without hampering the productivity losses in terms of time, material, and tool, consequences in high productivity. For the proposed work, machine learning techniques such as k-NN, Random forest, Adaboost, k-Star, and Decision Tree are implemented and there accuracy of prediction is demonstrated. Furthermore, WEKA, open source software has been used to employ several tool learning algorithms for better understanding. The investigation noticed that the random forest algorithm has a higher accuracy of 97.30% and a root mean square error value of 0.144 among all other algorithm.

Keywords: Machine Learning, k-NN, Adaboost, k-Star, Random forest

1. Introduction

Problems and impediments are unwelcome guests in any machining process. In which the wearing of a tool that requires constant monitoring from time to time plays a prominent role. The invention of tools and associated mechanisms, are focused to increase the productivity with decreasing input effort which can only be accomplished by shortening the ideal time of run. To achieve this, several mechanisms have been developed and evolved by scientific community. In this regard, the most common material i.e. HSS (high speed steel) is developed for production of cutting tools. When a steel tool contains more than 7% molybdenum, tungsten, vanadium, and more than 0.50 % carbon is considered as HSS steel. A 5-10% molybdenum addition increases the hardness and toughness of high-speed steels. HSS retain these properties even at high temperatures generated by metal cutting. Another advantage of molybdenum is that at high temperatures, steel softens and become embrittled if the primary carbides of iron and chromium grow rapidly in size. The most useful cutting property of high speed steel (HSS) is extended for proper running and effective tool operation by applying tin, but for further increment of hardness, titanium carbides coating are recommended, which reduces friction and increases wear resistance.

Furthermore, a proper maintenance and condition monitoring of cutting tools can reduce the rejection of finished component and increase productivity. Use of lubricant is recommended to reduce friction in combination with the proper feed and speed beneficial to decrease regrinding of tool for each machining cycle[1-3]. During machining monitoring of a tool requires time and expertise that hamper the production, if done manually. However, with the advancement of technology, for monitoring of cutting tool can now be easily sorted out by involving machine computer interface, interaction, that result in better inspection model to perform in a efficient way. The expected objective of this work is to predict an appropriate maintenance condition for an industrial cutting tool based on various datasets at numerous of machining condition and their corresponding datasets. This will help to identify a well-known problem of failure occur in the cutting tool at various parameters. Hence, the objective of this work to present a solution for fault identification using machine learning analysis and to provide necessary condition monitoring alongside required steps to be taken. In the proposed work, the authors are concentrated on using machine learning techniques to develop an inspection model for a single point cutting tool. This work also proposed a model for monitoring of health condition in an efficient manner, helpful for selection of best suited maintenance option, an indeed requirement in Industry 4.0 era. The current era also direly requires a high precision prediction tool that beneficial for maintaining the high precision of dimension which is only possible through the machine learning techniques. So, to explore such techniques towards commercial application many research are still requires. In order to fulfill the objective of work, K-Nearest Neighbor, decision tree, random forest, support vector machine, and multilayer perceptron algorithms are selected for evaluation. Using various machine learning algorithms, a comparative study is also carried out to reveal an efficient algorithm having highest accuracy towards prediction of health of cutting tool.

2. Literature Review

Numerous of researchers dedicated their worked on the detection of faults in the cutting tools using machine learning techniques. The work carried out by Srivastava et al.[4] had proposed a machine learning approach to predict the fault detection in the gear of an industrial gearbox. Based on the results, researchers illustrated that the random forest algorithm was observed as the best algorithm, with an accuracy of 89.15% and a root mean square error of 0.172. In another study, Srivastava et al.[5] presented a different approach of machine learning for bearing datasets. The results of various machine learning algorithms such as K-NN, decision tree, random forest, support vector machine, and multilayer perceptron was thoroughly discussed. With 87.15% accuracy and 0.192 error, the random forest algorithm was observed as the best performing algorithm. Chen et al.[6] proposed tool wear prediction, in which three tool wear monitoring systems were used i.e. multiple linear regression, artificial neural network, and statistics assisted fuzzy net. The result successfully stated a new method for prediction of tool wear during milling process applicable to one tool and workpiece interaction. Deore et al.[7] proposed a concept to measure the flank and crater wear with profile projector PP-200. The flank wear was trained with the ANFIS for prediction of tool life during turning process. The research illustrated that ANFIS was more appropriate for predicting tool wear, with an accuracy of 87.87%. Siddhpura et al.[8] also investigated the flank wear during machining operation. Researchers selected three-monitoring system, signal acquisition, signal processing, and feature extraction, as well as artificial intelligence techniques, to monitor the tool condition during turning process. The findings showed beneficial effect towards reduction of overall production costs, production time, machine downtime, and material waste. Later, Srivastava et al.[9] studied different machine learning techniques such as K-Nearest Neighbor, decision tree, Ad boost, random forest, and support vector machine to compare the algorithm that precisely monitor the detection of steel tool faults during turning operation. The result illustrated random forest algorithm as best for detecting faults in steel tools with an accuracy of 79.23%. Gouarir et al.[10] also illustrated the predicting algorithm for flank wear of cutting tool based on deep learning approach based on two methods i.e. machine learning and traditional neural networks. The results demonstrated CNN (conventional neural network) as the best for identifying the faults in cutting tools, with an accuracy of 90%. Madhusudana et al.[11] explained the automation of machining system by using systematic tool condition monitoring system that result in high productivity with prolong tool life. Researchers found that the decision tree algorithm was proved to be best performing as per the dataset. Furthermore, vibration signals were used for better correlation with tool life. Researchers also demonstrated an accuracy of about 96.90%, for the face milling process. Some more works carried out by the researchers also showed systematic failure[12-14] occurred inside the cutting tools which are taken care for reference in the manuscript. Furthermore, the earlier work[15-19] that motivated the authors to carried out this novel work that cope up with the demand of Industry 4.0.

Hence, in this experiment, mild steel bar is used as workpiece material and HSS tools are used for turning operation. The machined data in terms of cutting force, surface roughness and flank wear occurred during the machining was evaluated. The collected data set at various set of machining conditions are used to train the algorithm using machine learning approach. The developed model is used to predict the condition of tool with high precision.

3. Methodology & Dataset

The proposed methodology for work for this paper is represented through flow diagram shown in Figure 1. At, first conventional lathe has been selected to machining the mild steel using HSS as cutting tool. The prior researches carried out by the authors are helpful for the selection of machining parameters. A total of eighteen experiments were carried out to train the algorithm. A detail pictorial representation of experimental set-up is shown in Figure 2. Tool maker microscope (Make: METZER) was used to evaluate the flank wear after every experiments. An average of ten readings was considered to cite the flank wear.

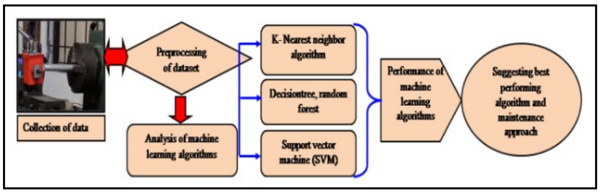

Fig. 1 Systematic representation of methodology opted

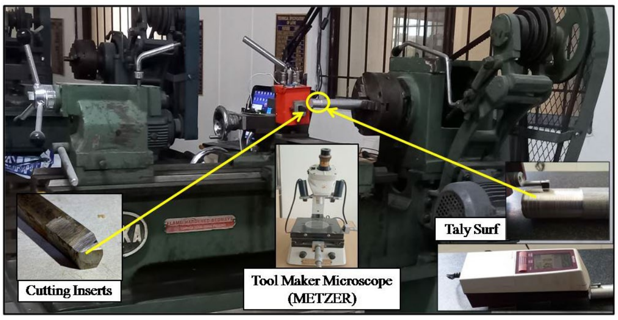

Fig. 2 Systematic representation of experimental set-up

Table 1: Parameters opted to carried out the experiments.

Cutting Parameters	Selected Value
Cutting Speed (m/min)	50 - 300
Feed Rate (mm/sec)	6-20
Depth of Cut (mm)	2-3

The dataset used for prediction of tool wear were discrete in eleven features and one target variable. The dataset contains total of eighteen experiments, ten of which were performed with a worn tool and the remaining eight with an unworn tool. Each experiment with a worn and unworn tool was carried out with a different feed rate and clamp pressure. In this experiments the dataset were merged with all data points according to clamp pressure and feed rate.

- X1_ActualPosition: actual x position of part (mm)
- X1_ActualVelocity: actual x velocity of part (mm/s)
- X1_ActualAcceleration: actual x acceleration of part (mm/s/s)
- Y1_ActualPosition: actual y position of part (mm)
- Y1_ActualVelocity: actual y velocity of part (mm/s)
- Y1_ActualAcceleration: actual y acceleration of part (mm/s/s)
- Z1_ActualPosition: actual z position of part (mm)
- Z1_ActualVelocity: actual z velocity of part (mm/s)
- Z1_ActualAcceleration: actual z acceleration of part (mm/s/s)
- Feed_rate: relative velocity of the cutting tool along the workpiece (mm/s)
- clamp_pressure: pressure used to hold the workpiece in the vise (bar)
- tool_condition: worn or unworn

3.1 Development of ML Algorithms

In this investigation the opted methology for machines learning algorithm are K-star, K-Nearest neighbor, Adaboost algorithm, decision tree, and random forest algorithm.

3.1.1 Development of k-NN algorithm

k-NN algorithm is an supervised machine learning algorithm, also known as instance-based machine learning algorithm. The main advantage of this algorithm is it didn’t require any training time data. On arrival of new instance, algorithm search for k nearest examples and based on voting of these nearest neighbor examples, class label of new instance determined. The value of K is positive and odd integer. Due to absence of training phase, this algorithm also knows as the lazy algorithm. There are many different metrics used to find the nearest examples. Euclidean, Manhattan, city block, Chebyshev and many other distance metric algorithms is used to find the nearest neighbors. After collecting nearest neighbor, majority voting method is used to find the final prediction of the new instance. For example, in 3-NN algorithm, for classification of new instance, 3 nearest algorithms search based on distance metric. Majority voting method applied on these 3 nearest examples and find label will be predicted.

3.1.2 K- star Algorithm

K-Star algorithm is also called as instance-based machine learning algorithm. This algorithm is enhanced version of the -NN algorithm main difference with K-NN algorithm and K-star algorithm is use of metric to find the distance between instance. K–Star algorithm uses entropy-based method as distance metric. The final prediction come from k-star is in probability number.

3.1.3 Ada-boost Algorithm

This algorithm is also known as Adaptive Boosting, a part of machine learning meta- algorithm formulated by Yoav Freund and Robert Schapire. It may be used along with many different styles of studying algorithms to enhance overall performance. The output of the alternative machine learning algorithms ('susceptible novices') is blended right into a weighted sum that represents the very last output of the boosted AdaBoost is adaptive. In this algorithm the important is to relearn the weak classifier. This algorithm uses weight-based method and for correctly classifying the instance weight increase and decrease with wrong classification.

3.1.4 Decision Tree

Another name of decision tree algorithm is supervised machine learning algorithm. This algorithm is used for mutually sorting and reversion complications. Decision tree learn training instance and build tree. This tree has one root and last child are the class labels of the problem. Nodes represent the feature and edges represent the feature value.

Decision tree are buildup of information gain or entropy value. First step is to find the best correlating feature with class label. The most correlated feature will be assigned to root node of tree and based on the next feature is selected and so on. At the time of inference, the new instance will travel the tree based on feature and its value and final class label predicted.

3.1.5 Random Forest

It is an enhanced version of the decision tree algorithm. Decision tree used information gain method to determine the place of the features in the tree whereas random forest give each tree equal opportunity and build multiple trees. If dataset consists of the n features, this algorithm builds or train n tree by giving equal opportunity to all features. At the time of inference, each tree will predict the class label and majority voting method is used to find the final predicted class label. This algorithm is one of the most popular algorithms because of getting inference from multiple trees.

4. Tool Setup

In this investigation, all algorithms were developed through WEKA tool, which is open source software used to run various machine learning algorithms. This tool accepts input files in variety of formats, including .csv, .arff, .data, and .json. WEKA is a graphical user interface (GUI) tool that does not require a code environment to perform machine learning tasks. This tool also aids in the visualization of datasets, construction of models, and the analysis of output data. Figure 3 depicts the tool's dashboard, which includes several options such as explore, experimenter, knowledge flow, workbench, and simple CLI[20]. To investigate machine learning algorithms, we must select the investigate option. Figure 4 depicts the explore GUI page and the process of selecting a dataset for analysis. The GUI view after selecting the dataset from the explorer is shown in Figure 5. The dataset summary is shown on the left side of the page, including the number of features, number of instances, and attribute/feature names. The initial data analysis, such as the number of missing values and the dataset graph, is shown on the right side of the page. Figure 6 depicts the user interface for selecting machine learning algorithms. The experimental dataset was obtained from the University of Michigan Smart Lab.

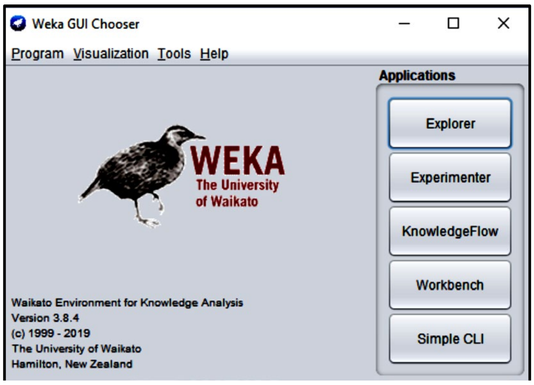

Fig. 3 WEKA Tool Wizard

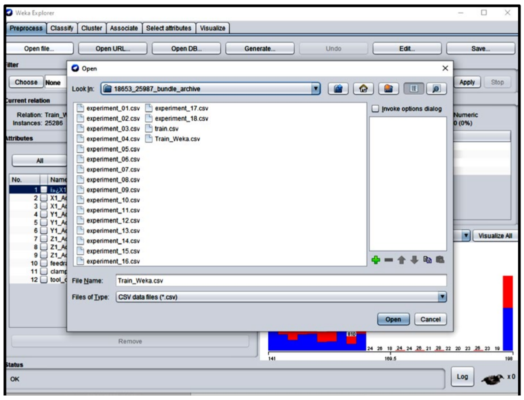

Fig. 4 Dataset Selection in WEKA Tool

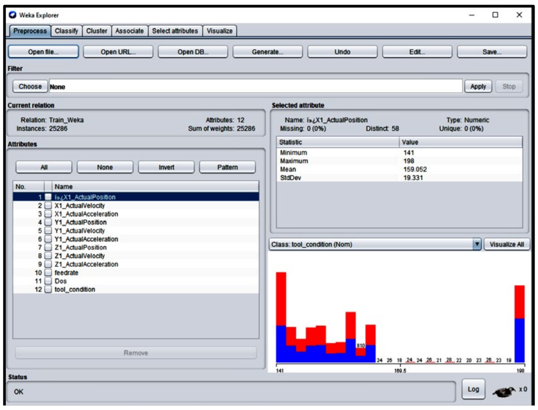

Fig. 5 WEKA Tool GUI

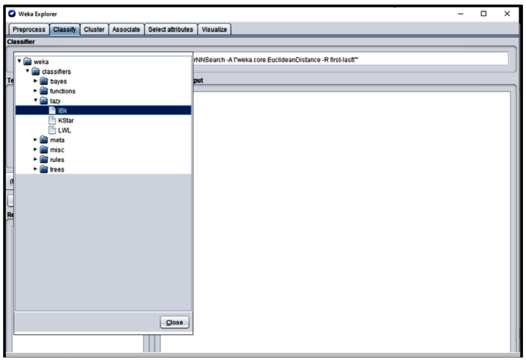

Fig .6 Algorithm selection

5. Model Evaluation and Performance Parameter

There are various methods for assessing model performance. One of the most common methods for evaluating the model performance of a machine learning algorithm is cross validation. Model evaluation is divided into two categories: holdout methods and k-fold validation methods. The dataset is divided into two sets in the Holdout method: training dataset and test dataset. The training dataset is used to train the model, while the test dataset is used to assess the model's performance[21]. When training and testing data have different variances, model performance suffers. So, in order to achieve good model performance, the dataset can be divided into three parts: train, validation, and test. Train the model on the training dataset, then test its performance on the validation dataset to minimize model error. Finally, use the trained model on the test dataset to assess the model's performance. The K-fold method is a more advanced version of the holdout method. The dataset will be divided into K parts, with (K-1) subsets used as training instances and one subset used as testing each time. So, the total model performance will be evaluated K times. Take the average of the error and accuracy of all K - models to calculate overall model performance. A crucial step is the evaluation of the machine learning model. Different parameters, such as accuracy, confusion matrix, F1score, MSE (Mean Square Error), RMSE (Root Mean Square Error), and confusion matrix, can be used to evaluate machine learning models. This experiment makes use of all of these parameters.

Accuracy: Accuracy is an important parameter to determine correct number of correctly classified instances.

$$\boldsymbol{Accuracy} = \frac{Number\ of\ corretly\ classified\ examples}{Total\ number\ of\ examples} \quad (1)$$

Root Mean Square Error (RMSE): RMSE is mean square of the prediction error. RMSE is standard method of calculating the error at the prediction time.

Confusion Matrix: Confusion matrix describe the performance of machine learning algorithm in table format. Fig. 5 shows the format of the confusion matrix. This matrix is combination of predicted value and actual value.

- True positive (TP): example belongs to positive class and predicted also as the positive class example.
- True negatives (TN): example belongs to negative class and predicted also as the negative class example
- False positives (FP): example that belongs to positive class but is classified as negative class.
- False negatives (FN): example that belongs to negative class but is classified as positive class.

Predicted value

Actual value		
	True Positive (TP)	False Negative (FP)
	False positive (FP)	True Negative (TP)

Fig. 7 Confusion Matrix

Recall: Recall parameter is the ratio of true positive and sum of true positive and false negative. It addresses the question how many actual positive instances correctly classified.

$$\boldsymbol{Recall} = \frac{\boldsymbol{True\ positive\ examples}}{\boldsymbol{Sum\ of\ true\ positive\ and\ false\ negative\ examples}} \quad (2)$$

Precision: Precision is ratio of True positive and sum of true positive and false positive. It addresses the question how many actual positive instances correctly classified.

$$\textbf{Precision} = \frac{\textbf{True positive examples}}{\textbf{Sum of true positive and false positive examples}} \quad (3)$$

F-measure: F – measure also known as the F1 parameter. This parameter is used when dataset is unbalanced. This parameter is harmonic mean of Precision and Recall. The F-Measure value should always be similar Precision or Recall value.

$$\textbf{F1 score} = \mathbf{2} * \frac{\textbf{Precision}*\textbf{Recall}}{\textbf{Precision}+\textbf{Recall}} \quad (4)$$

6. Machine Learning Approaches on Lathe Machine Dataset

The Lathe machine dataset, which contains eleven features and one target variable called tool condition, is the subjected to develop model through machine learning algorithms[22-23]. The graph of feature distribution is shown in Figure 8. The unworn tool condition is represented by the blue colour in these graphs, while the worn tool condition is represented by the red colour. This section describes the outcomes of machine learning algorithms used on the lathe dataset. All experiments were carried out on a 10-fold cross validation dataset.

The machine learning algorithms induced for predicting the condition are K-nearest neighbor, K-Star, AdaBoost, Decision Tree and Random Forest[24-25]. In comparison to all other algorithms, the random forest algorithm has a higher accuracy of 97.30% and a root mean square error value of 0.144. Furthermore, the random forest algorithm achieved the highest class level accuracy. From comparative analysis the Adaboost algorithm and decision tree also achieved nearly identical accuracy, but the Random forest algorithm has a lower RMSE value than both algorithms.

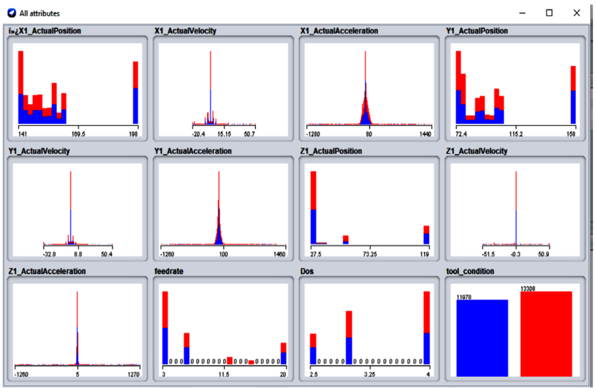

Fig. 8 Feature distribution of Lathe dataset

6.1 K-Nearest Neighbor algorithm

All experiments have been performed ten-fold cross validation dataset. Table 2 shows the decision parameters of the K-NN algorithm. This algorithm achieved 68.78% accuracy in 0.05 second. Precision Recall and F-measure all values are 0.688 and RMSE value is 0.435.

Table 2 Decision parameter of K-Nearest Neighbor algorithm on Lathe Dataset

Parameters	**Results**
Accuracy	68.78%
Time (sec)	0.050
Precision	0.688
Recall	0.688
F- measure	0.688
RMSE	0.435

6.2 K-Star algorithm

Table 3 shows the basic decision parameter of the k-star algorithm. This algorithm achieved 80.28% accuracy in 0.20 secs. Precision recall and F-measure is 0.803. Compare to K-NN algorithm, K-star algorithm is performing better. K-star algorithm achieved 80.28% accuracy whereas K-NN achieved only 68.78% accuracy. These results show that entropy method used distance metric is performing best between K-NN and K-star algorithms. Also, K-star algorithm has less RMSE value compare to K-NN algorithm.

Table 3 Decision parameter of K-Star Algorithm on Lathe Dataset

Parameters	**Results**
Accuracy	80.28%
Time (sec)	0.20
Precision	0.803
Recall	0.803
F- measure	0.803
RMSE	0.377

6.3 Adaboost algorithm

Table 4 shows the decision parameter of the Adaboost algorithm. This algorithm achieved 96.65% accuracy in 1.99 secs. This algorithm report 0.174 RMSE value and 0.967 precision, recall and F-measure value. Compare to previously discussed results. This algorithm is performing better than both K-NN and K-star algorithm with 96.65% accuracy. And, also AdaBoost algorithm has least RMSE 0.174 value till now whereas K-N and K-star both algorithms have greater than 0.3 RMSE value.

Table 4 Decision parameter of AdaBoost on Lath machine Dataset

Parameters	**Results**
Accuracy	96.65%
Time (sec)	1.99
Precision	0.967
Recall	0.967
F- measure	0.67
RMSE	0.174

6.3 Decision tree algorithm

Table 5 shows the decision parameters of the decision tree algorithm. Decision tree algorithm achieved 96.74% accuracy in time 1.83 seconds. This algorithm accomplished 0.967 precision, recall and F-measure and 0.166 of root mean square value. As compared to previously discussed results of the classifiers, decision tree algorithm is the outperforming algorithm. Decision tree algorithm and Adaboost algorithm achieved almost similar accuracy but RMSE value of decision tree is lower than AdaBoost algorithm.

Table 5 Decision parameter of Decision Tree Algorithm on Lath machine Dataset

Parameters	**Results**
Accuracy	96.74 %
Time (sec)	1.83
Precision	0.967
Recall	0.967
F- measure	0.967
RMSE	0.166

6.3 Random forest algorithm

Table 6 shows the decision tree algorithm of the random forest algorithm. This algorithm achieved 97.30% accuracy in 9.64 seconds and RSE value is 0.144. Compare to all algorithm, random forest algorithm has achieved the higher accuracy of 97.30% and root mean square error value of 0.144. Also, random forest algorithm achieved highest class level accuracy. Adaboost algorithm and decision tree achieved almost similar accuracy but RMSE value of Random forest is lower than both algorithms.

Table 6 Decision parameter of Random Forest Algorithm on Lath machine Dataset

Parameters	**Results**
Accuracy	97.30%
Time (sec)	9.64
Precision	0.973
Recall	0.973
F- measure	0.973
RMSE	0.144

7. Results & Discussion on Machine Learning Approaches

The accuracy of all machine learning algorithms using Lathe dataset is shown in Figure 9. The random forest algorithm is observed as best accuracy with 97.30%. AdaBoost and Decision tree algorithms are also performing well, with 96.65% and 96.74% accuracy, respectively. This means that tree-based algorithms are more accurate. Tree-based algorithms include AdaBoost, Decision Tree, and Random Forest. The random forest algorithm, among these tree-based algorithms, has the best performance. Figure 10 depicts the RMSE value of the ML algorithms. This value of error should be lower. According to the graph, the Random forest algorithm has the lowest error. Similarly, Figure 11 depicts the precision, recall, and F-measure value, all of which should be close to one, with the Random forest algorithm having the highest value. Figure 12 depicts the class level accuracy of the ML algorithms. In both classes, the accuracy value should be higher. For instance, the Decision tree algorithm achieved 99.64% in the Unworn class and 96.90% in the Worn class. In the Random Forest algorithm, the Unworn class achieved 96.85% accuracy and the Worn class achieved 97.72% accuracy.

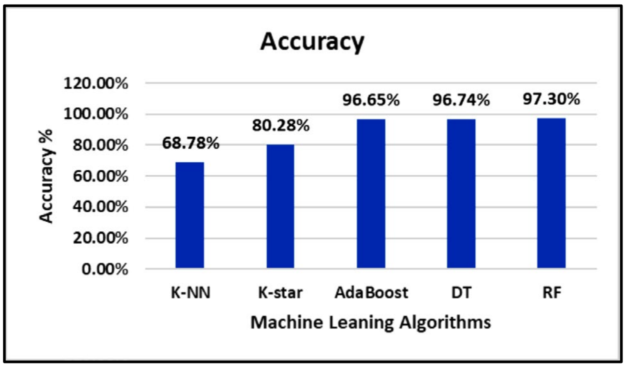

Fig. 9 Accuracy of Machine Learning Algorithms on Lathe Dataset

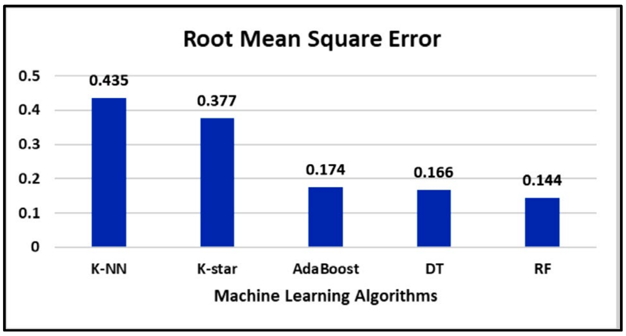

Fig. 10 RMSE value of Machine Learning Algorithms on Lathe Dataset

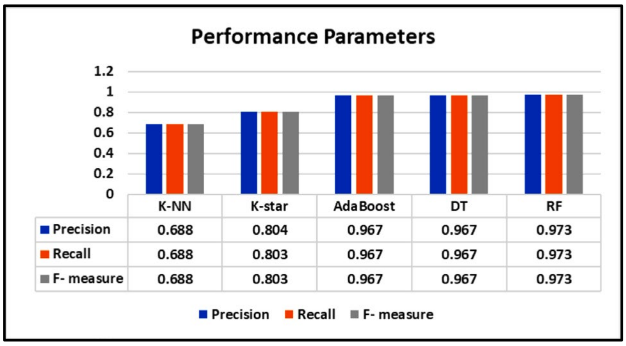

Fig. 11 Performance parameter of Machine Learning Algorithms on Lathe Dataset

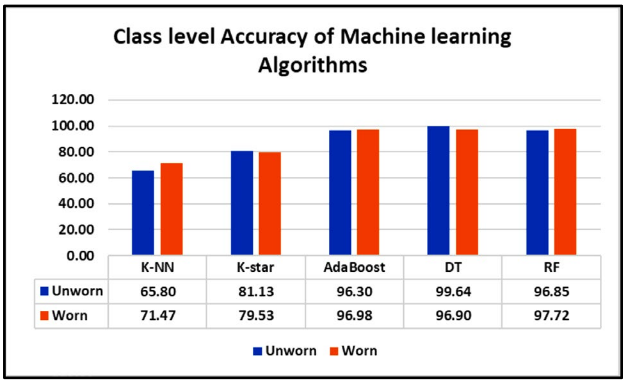

Fig. 12 Class level accuracy of Machine Learning Algorithms on Lathe Dataset

High accuracy values and low RMSE values are necessary to identify the best performing algorithm.

According to Figures 10 and 11, the Random Forest algorithm performs best with an accuracy value of 97.30% and an RMSE value of 0.144. Along with the Random forest algorithm, the Adaboost and decision tree algorithms perform well, with accuracy values of 96.65% and 96.74%, respectively. The best performing techniques in the Lathe dataset are tree-based algorithms. The algorithm that performs the best in this situation is random forest algorithm.

8. Conclusions and Future Scope

The investigated successfully showed the application of various machine learning algorithms such as K-NN (K-nearest neighbour), K-star, AdaBoost, Decision tree, and Random forest towards prediction of condition of tool during machining. The analysis showed Random Forest is the best performing algorithm, with a 97.30% accuracy and RMSE of 0.144. These algorithms aid in the prediction of tool faults.

Deep learning algorithms can also be investigated in order to further this research and find the most appropriate machine learning approach. Another area for improvement is determining the best subset of attributes that result in higher accuracy. This work may also result in the development of a simulation model that is precise and accurate and that can forecast component health based on a number of time-limited parameters. The proposed work can also be expanded to create a simulation model that can be used to create a real-time environment for performing inspections and providing early prediction for maintenance strategy planning. This will help to reduce the amount of time, error, and money spent on inspection.

References

1) B. K. Singh, S. Goswami, K. Ghosh, H.Roy,N. Mandal, "Performance evaluation of self lubricating CuO added ZTA ceramicinserts in dry turning application" International Journal of Refractory Metals and HardMaterials, **98** *(105551)* (2021).
2) B. K. Singh, H Roy, B Mondal, S S Roy andN Mandal, "Measurement of Chip Morphology and Multi criteria optimization of Turning Parameters forMachining of AISI 4340 Steel using Y-ZTA Cutting Insert" Measurement, **142** 181-194 (2019).
3) B. K. Singh, H Roy, B Mondal, S.S.Roy, Nilrudra Mandal, "Development and machinability evaluation of MgO doped Y-ZTA ceramic inserts forhigh speed machining of steel", Machining Science and Technology **22** *(6)* 899-913 (2018).
4) Ashish Kumar Srivastava, Abhishek Singh Gangwar. "Fault Detection and Maintenance Prediction for Gear of an Industrial Gearbox Using Machine Learning Approaches" International Research Journal of Engineering and Technology (IRJET) **6** *(6)* 2125–28 (2019).
5) Ashish Kumar Srivastava, AbhishekSingh Gangwar. "Detection of Fault in Bearing of an Industrial Gearbox Using Machine Learning Approaches" International Journal Of Scientific Progress And Research (Ijspr) **61** *(161)* 1–4 (2019)
6) J. C. M. Chen,. "In-process tool wear prediction system development in end milling operations" (2003).
7) Bhagyashri DDeore, and Dr. P S Desale. "Modelling of Turning Process for Prediction of Tool Wear." International Advanced Research Journal in Science Engineering and Technology **3** *(7)* 113–16 (2016).
8) A Siddhpura,. and R. Paurobally, "A review of flank wear prediction methods fortool condition monitoring in a turning process" The International Journal of Advanced Manufacturing Technology, **65** *(1-4)*, 371-393 (2013).
9) A. K. Srivastava, "Comparison analysis of machine learning algorithms for steelplate fault detection" International Research Journal of Engineering and Technology (IRJET), **6***(5)*, 1231-1234 (2019).
10) A. Gouarir, G. Martínez-Arellano, G. Terrazas, P. Benardos and S. J. P. C.Ratchev "In-process tool wear prediction system based on machine learning techniques andforce analysis" Procedia CIRP, **77**, 501-504 (2018).
11) C. K.Madhusudana, S. Budati, N. Gangadhar, H. Kumar, and S. Narendranath, "Fault diagnosis studies of face milling cutter using machine learning approach" Journal of Low Frequency Noise, Vibration and Active Control, **35** *(2)* 128-138 (2016).
12) A. Pratap, B. K. Singh, N. Sardana, "Fracture in self-lubricating inserts: A case study" Materials Today: Proceedings. **66** *(9)* 3738-3742 (2022).
13) S. Ganeshkumar, B. K. Singh, , S. D. Kumar, , S. Gokulkumar, S. Sharma, K. Mausam, C. Li, Y. Zhang, and E.M. Tag Eldin, "Study of Wear, Stress and Vibration Characteristics of Silicon Carbide Tool Inserts and Nano Multi-Layered Titanium Nitride-Coated Cutting Tool Inserts in Turning of SS304 Steels" Materials, **15** *(22)*, 7994 (2022).
14) B. K. Singh, A. A. Yaduvanshi, A. K. Mishra, "Comparative Study of CuO/ZTA and ZTA Composites in Terms of Functional Properties" In: Mukherjee, K., Layek, R.K., De, D. (eds) Tailored Functional Materials. Springer Proceedings in Materials, vol 15. Springer, Singapore (2022). https://doi.org/10.1007/978-981-19-2572-6_28
15) Gde Dharma Nugraha, Budi Sudiarto, Kalamullah Ramli, "Machine Learning-based Energy Management System for Prosumer" EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy 7(2) 309-313, (2020). https://doi.org/10.5109/4055238
16) Ajay Kumar Verma1, Parveen Kumar Saini,

"Vibration Model to Detect Local Defect Characteristics of Deep Groove Ball Bearing", EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy, **9***(4)* 1056-1066, (2022). https://doi.org/10.5109/6625791

17) Fauzal Naim Zohedi, Mohd Shahrieel Mohd Aras, Hyreil Anuar Kasdirin, Mohd Bazli Bahar, "A new tuning approach of Single Input Fuzzy Logic Controller (SIFLC) for Remotely Operated Vehicle (ROV) Depth Control", EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy, 8(3) 651-657 (2021). https://doi.org/10.5109/4491657
18) Nawaharsh Weake Meena Pant, Ankita Sheoran, Abid Haleem, and Harish Kumar, "Optimising Parameters of Fused Filament Fabrication Process to Achieve Optimum Tensile Strength Using Artificial Neural Network" EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy, **7***(3)* 373-381, (2020). https://doi.org/10.5109/4068614
19) Matrix Shinji Hirata, Michitaka Ohtaki, "Simultaneous Enhancement in the Electrical Conductivity and Reduction in the Lattice Thermal Conductivity Leading to Enhanced Thermoelectric ZT Realized by Incorporation of Metallic Nanoparticles into Oxide", EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy, **7***(1)* 1-6 (2020). https://doi.org/10.5109/2740934
20) V. P. Astakhov, "The assessment of cutting tool wear" International Journal ofMachine Tools and Manufacture, **44** *(6)* 637-647 (2004).
21) X. Lu, F. Wang, , Z. Jia, L.Si, , C. Zhang, and S. Y. Liang, "A modifiedanalytical cutting force prediction model under the tool flank wear effect in micromillingnickel-based superalloy" The International Journal of AdvancedManufacturing Technology, **91** *(9)* 3709-3716 (2017).
22) https://www.kaggle.com/datasets/shasun/tool-wear -detection -in-cnc-mill
23) Wo Jae Lee, Gamini P. Mendis, John W. Sutherland, "Development of an Intelligent Tool Condition Monitoring System to Identify Manufacturing Tradeoffs and Optimal Machining Conditions", Procedia Manufacturing, **33** 256-263 (2019)
24) S. Cho, S. Binsaeid, S. Asfour, "Design of multisensor fusion-based tool condition monitoring system in end milling" The International Journal of Advanced Manufacturing Technology, **46** *(5)* 681-694 (2010).
25) Sonali. S. Patil, Sujit S. Pardeshi, Nikhil Pradhan, Abhishek D. "Patange Cutting Tool Condition Monitoring using a Deep Learning-based Artificial Neural Network" Int J Performability Eng, **18** *(1)* 37-46 (2022).