Serbian Tribology Society

SERBIATRIB '23

18th International Conference on Tribology

Faculty of Engineering
University of Kragujevac

Kragujevac, Serbia, 17 – 19 May 2023

SELECTION OF MACHINE LEARNING ALGORITHMS FOR NANOCOMPOSITE ZA-27 MATERIAL TRANSFER PREDICTION

Aleksandar ĐORĐEVIĆ[1], Dragan DŽUNIĆ[1], Marko PANTIĆ[2,*], Milan ERIĆ[1], Slobodan MITROVIĆ[1], Miladin STEFANOVIĆ[1]

[1]University of Kragujevac, Faculty of Engineering, Kragujevac, Serbia

[2]University of Priština in Kosovska Mitrovica, Faculty of Technical Sciences, Kosovska Mitrovica, Serbia

*Corresponding author: marko.pantic@pr.ac.rs

Abstract: *This study explores the use of machine learning algorithms in predicting material transfer in tribological contacts. The results of the analysis indicate that the machine learning models can accurately predict the occurrence of material transfer with a high degree of accuracy. The Gradient Boosting Classifier algorithm was found to outperform other algorithms in terms of predictive accuracy. The study's practical implications suggest that machine learning can be an effective tool for predicting and preventing material transfer, leading to increased system reliability and durability. The findings highlight the importance of domain-specific expertise in selecting appropriate algorithms and input features. One limitation of the study is that it focused only on material transfer and did not consider other important factors such as wear and friction. Future research could investigate the use of machine learning algorithms in predicting wear and friction in tribological systems.*

Keywords: *machine learning, data-driven analyses, machine learning hiperparameter optimization, Triboinformatics.*

1. INTRODUCTION

In the last decade, the use of "Big Data" and data-driven analysis in many scientific and engineering fields has become the prevailing trend. Tribology, the science of surface interactions, as it has been and continues to be one of the most relevant scientific and engineering disciplines is not devoid of this trend. When it comes to tribology, large amounts of data are generated as a result of a large number of experiments, with parameters such as friction and wear, whose values are constantly measured. Furthermore, since friction and wear have a complex nature, tribology has remained a highly experimental and empirical science, leading to the formation of big data on friction, wear, and surface properties of different materials, that posts an opening to perform data-driven analyses. The appearance of contemporary computer systems with significant computing power enabled the rapid development of new approaches to data-driven analysis that originated new insights. "Big Data" algorithms belonging to Artificial Intelligence (AI) and machine learning (ML) are now being used to determine new correlations within data-driven areas that cannot otherwise be discovered using traditional methods [1]. The application of new data-driven approaches paved the way for the development of new areas within tribology such as "Triboinformatics" and "Intelligent tribology" [2].

The characteristic that was used to perform data-driven analyses in this paper is material transfer from different nanocompozite of ZA-27 alloy matrix reinforced with different volume fractions of nanometric "Al2O3", i.e. "1%Al2O3", "3%Al2O3", and "5%Al2O3", respectively. These materials were used because, as a consequence of its higher chemical reactivity, aluminium bonds to steel much more quickly during sliding, leading to adhesive wear and material transfer [3].

Although many factors, including surface roughness, lubricants, sliding distance, contact pressure, and tool coating, are known to influence the occurrence of material transfer at low temperatures (i.e. room temperature), describing the mechanisms behind the initiation of material transfer remains a persistent challenge in this field [3]. The addition of nanometric Al2O3 particles to ZA-27 alloy matrix can improve its tribological properties by enhancing wear resistance, reducing friction, and minimizing material transfer [4]. However, the specific material transfer characteristics of the composite depend on the specific tribological system and the volume fraction of Al2O3 nanoparticles.

Motivation for the research is to find the adequate ML method and optimal values for the hyperparameters for the adequate ML method so that it can be used effectively with the available tribology data. Although the influence of hyperparameters in general may be understood, their precise effect on a dataset and their interactions during learning may not be known [5]. As a result, it is critical to fine-tune the method hyperparameter values. Typically, hyperparameters are tuned using naive optimization algorithms such as grid search and random search; however, the reason for this study is to also use a stochastic optimization hill climbing algorithm.

Research questions arise as to whether it is possible to apply ML algorithms that will be able to perform a sufficiently precise prediction of the values of tribological characteristics and thus enable the realization of extensive long-term tribological experiments to be reduced. An additional research question is whether, after selecting the most accurate algorithm from the spectrum of ML algorithms, it is possible to apply the stochastic optimization algorithm and additionally fine-tune the hyperparameter values of the selected ML algorithm.

The paper practical aim is to present a ML model that will enable adequate prediction of the occurrence of material transfer, and to aid in understanding the mechanisms of friction and wear, which will in turn lead to the development of better materials and lubricants for improved performance and durability of mechanical systems.

The theoretical goal of the paper is to present a new methodology based on the selection and application of ML models and the hill climbing optimization method for determining hyperparameters of ML algorithm in order to make the accurate predictions of the occurrence of material transfer.

The methodological contributions of ML in tribology and material transfer are focused on improving the understanding of tribological phenomena, developing predictive models, and optimizing the design and operation of tribological systems. These contributions have the potential to improve the reliability and efficiency of tribological systems in a wide range of applications.

2. LITERATURE REVIEW

The literature review is organized into four subsections, each subsection corresponds to an overview of current research in the fields devoted to the material transfer, ML and tribology and ML. The last subsection presents identified research gaps.

2.1 Triboinformatics review

Tribology research is primarily concerned with studying the friction, wear, and lubrication of interacting surfaces. As industrialization continues to expand, tribology research has broadened its scope considerably. Over time, tribology research methods have evolved from empirical science based on observation to theoretical science based on models, and finally, computational science based on simulations. Thanks to advances in information technology, more efficient and effective methods for generating, collecting, processing, and analyzing tribological data have emerged, leading to the introduction of the modern concept known as "tribo-informatics (triboinformatics)." [6]. As information technology continues to advance, the range of information processing methods has expanded beyond traditional methods such as regression, fitting, and induction. In particular, the emergence of ML and AI technologies has significantly enhanced the efficiency of information processing methods and opened up new areas of application. Thus, the concept of tribo-informatics emerged in response to the need for more efficient and effective tribology research. By establishing tribology standards, developing tribology databases, and utilizing information technology to collect, organize, store, retrieve, analyze, and share tribology information, tribo-informatics has enhanced the efficiency of the tribology research process [7].

Tribo-informatics methods currently have a wide range of applications, including regression and clustering [8]. These methods are primarily employed for the purposes of condition monitoring, behaviour prediction, and optimization of tribological systems [9-11]. The establishment of relationships between data should be the end objective of all methods used in informatics. System behaviour prediction is used to determine the relationship between tribological quantity and time quantity; system state monitoring is used to determine the relationship between tribological quantity and state quantity; and system optimization is used to determine the relationship between the system input and output. Because of this, the collaborative movement of data and model is an essential component of the research methodology utilized in tribo-informatics [6].

2.2 Tribology and ML Review

Tribo-informatics encompasses every technique ever developed for analyzing tribological data. It incorporates state-of-the-art ML techniques in addition to more conventional information processing techniques like the Gaussian regression method, linear regression method, and least squares method. ML research has yielded a variety of AI techniques [12].

To achieve full integration of informatics and tribology, tribology information processing must be founded on fundamental tribological models and concepts. Model-driven data processing, or MDDP, is a technique used in tribo-informatics to guarantee that the parameters chosen to characterize a system have some sort of physical significance. The best values for the defining characteristics can then be derived through correlation. Moreover, "data-driven model optimization" can be used to create novel tribological physical models or principles by determining optimum characteristic parameters [13].

The most popular of these techniques are those based on ML algorithms such as artificial neural networks (ANNs) [14], support vector machines (SVMs) [15], 2021), k-nearest neighbors (KNNs), and random forests (RFs).

The complex processes in tribological systems can be investigated and their behavior can be classified or quantified using state-of-the-art ML or AI techniques in an effective or even real-time manner [10]. As a result, their potential extends well beyond the realm of academia and into practical, commercial settings. When it comes to dealing with high-dimensional problems and data sets and adapting to changing conditions with acceptable effort and cost, ML and AI methods stand out as particularly promising and advantageous [16]. They allow for the discovery of important connections, and further growth of understanding and using previously collected information. However, the potential of ML and AI techniques for tribological issues remains surprisingly underexplored in comparison to other fields or domains. The reason for this may be derived from the fact that tribology characteristics are not representative of fixed data but rather irreversible loss quantities that vary with time and test circumstances [9]. Furthermore, there are few or no studies that deal with the use of ML algorithms to predict the frequency of material transfer. Thus, the authors of this paper represent the possibility of using ML algorithms to identify the tribological characteristics values that will not contribute to material transfer occurrence prediction.

3. MATERIALS AND METHODS

ML algorithms were formulated to predict the tribological behavior of ZA-27 alloy matrix reinforced with different volume fractions of nanometric "Al2O3", i.e. "1%Al2O3", "3%Al2O3", and "5%Al2O3" under dry conditions. In this section, data collection, data preprocessing, parameter optimization, and performance-enhancing techniques used for the ML algorithms will be discussed.

3.1 Data collection

The acquisition of data is essential to the development of an effective data-driven ML algorithm. When a large, pertinent dataset is used to train an ML algorithm, the performance of the algorithm's predictive capabilities improves. The process of conducting tribological experiments in order to generate sufficient data for an ML analysis is a process that is both time-consuming and expensive. In addition, experimental data from a particular experimental setup may transmit undesirable trends and biases during the training of a ML model. This can severely impair the predictive performance of the ML model when applied to new datasets from other sources. Having in mind these issues, authors of this paper have collected normal load, length and width of the wear track, wear rate, and material transfer occurrence data of ZA-27 alloy matrix reinforced with different volume fractions of nanometric "Al2O3", i.e. "1%Al2O3", "3%Al2O3", and "5%Al2O3" under dry condition against a steel counterpart. For fources, length and width of the wear track, wear rate and material transfer occurrence prediction, tribological data set with 192 sample points was applied.

3.2 Input and output parameters

Variables aluminium carbide content, sliding speed, length of the wear track, width of the wear track and normal load were the input parameters for the ML algorithms. Among the input parameters, aluminium carbide content were defined as categorical data for the ML algorithms while the rest of them were numerical. Material transfer occurrence was the target or output parameter.

3.3 Machine leanrning algorithms

For this study, we have trained eleven ML algorithms: Logistic Regression (LR), Linear Discriminant Analysis (LDA), K-Neighbors Classifier (KNC), Decision Tree Classifier (DTC), Gaussian Naive Bayes (GNB), C Support Vector Classification (CSVC), Extreme Gradient Boosting (EGB), Ada Boost Classifier (ABC), Gradient Boosting Classifier (GBC), Random Forest Classifier (RFC), and Extra Trees Classifier (ETC) with tribological data of alloy matrix reinforced with different volume fractions of nanometric "Al2O3" to find correlations between the input and output parameters and predict the tribological behaviour. The ML analyses were performed using Python and its built-in “scikitlearn” and “xgboost” toolkits. A brief description of the ML algorithms used in this study is presented below.

LR is a model that is used to predict the probability of an event happening based on certain input variables. LR is an abbreviation for the term "logistic regression." In ML, it is a binary classification technique, where with a given collection of variables serving as inputs, the objective is to identify the line or curve that provides the best approximation of the possibility that an event will take place. A logistic function is used to model the relationship that exists between the variables that are used as input and the binary variable that is used as output in this form of generalized linear model.

LDA is a technique for supervised learning that seeks to identify a linear combination of features that best separates the data into distinct classes. LDA aims to reduce the dimensionality of input data by projecting it onto a lower-dimensional space while preserving as much class-discriminatory information as feasible. The resulting lower-dimensional space can then be applied to classification problems.

As its name indicates, the KNC algorithm takes into account the data points in the training set that are the most similar to the new data point being predicted. The parameter "k" or "n" represents the amount of data points that are considered to be the most relevant in order to make a prediction about a new point. The level of difficulty of a KNC model is determined by the number of the parameter n. Therefore, in order to optimize performance, it is necessary to select the value of k in a thoughtful manner, taking into consideration the data type and the degree of difficulty of the optimization problem. A KNC model also needs to take into account the assignment of a uniform or distance-based weight to the neighboring points. This is an essential consideration.

DTC is a supervised ML algorithm used as a form of decision support tool that classifies new data using a tree-like model of decisions and their potential outcomes. Each internal node represents a feature or attribute, each branch represents a possible value of that feature or attribute, and each leaf node represents a class designation or a decision. The objective of the DTC algorithm is to discover a decision tree that can classify new instances precisely.

GNB is a probabilistic ML algorithm employed for classification tasks. It is a variant of the Naive Bayes algorithm that implies continuous and normally distributed input features. GNB is a supervised learning algorithm that calculates the conditional probability of each input feature given the class label, and then multiplies these probabilities together to derive the probability of the class label given the input features. However, if the input features are not normally distributed or if there are correlations between the input features, it may not perform well.

The CSVC is a complex ML algorithm. It is particularly well-known for its ability to handle successfully a large number of input variables with a small dataset. In order for CSVC regression algorithms to work properly, the datapoints need to be reorganized into hyperplanes in higher dimensional space.

EGB is an ML algorithm that pertains to the boosting algorithm family. It is a robust, versatile, and effective algorithm that operates by training a series of weak decision tree models iteratively and then combining them to form a robust model that can accurately predict the target variable. In each iteration, the algorithm focuses on the examples that the current model struggles to correctly classify and attempts to adapt the new model to these examples. This is accomplished by adjusting the weights of the examples so that the new model gives greater weight to the misclassified examples.

ABC is an algorithm for ML that pertains to the boosting algorithm family. Similarly to EGB, ABC operates by training a series of weak classifiers iteratively and then combining them to produce a strong classifier that can accurately predict the target variable. In each iteration, the algorithm focuses on the examples that the current model struggles to correctly classify and attempts to adapt the new model to these examples.

The GBC prediction model is constructed using the ensemble technique, which involves combining a number of different decision trees. Each decision tree works on its own independently on a portion of the dataset, and the optimization of arbitrarily differentiable loss functions is carried out for each of them individually. After that, the results from all of the decision trees are combined to produce an accurate projection.

RFC is a type of supervised ML algorithm that makes use of decision trees to learn and improve itself based on instances from training.

This algorithm is well-known for its ability to handle massive datasets and problems that involve a large number of input variables in an effective manner. The RF algorithms are made up of a collection of individually crafted decision trees. After that, the results of these decision trees are averaged, which provides RFC with an enhanced capability for predictive analysis.

The ETC is a ML algorithm that is a member of the ensemble methods family. It is similar to RFC, but creates decision trees using a different method. Similar to RFC, ETC combines the predictions of multiple decision trees to arrive at a final conclusion. However, ETC employs a random split at each node to construct the decision trees rather than the optimal split at each node.

3.4 Deterministic optimization algorithm

The assumption of the authors of this paper is that after applying the mentioned ML algorithms for predicting the occurrence of material transfer, it is additionally possible to optimize the hyperparameters of the ML algorithm that proved to be the most accurate by using a deterministic optimization algorithm,i.e. Hill Climbing Local Search (HCLS) algorithm. HCLS is a straightforward algorithm for local search optimization. It seeks the optimal solution in close proximity to the current solution, while improving a candidate solution iteratively by making minor local modifications and selecting the best neighboring solution at each iteration. The algorithm begins with an initial candidate solution and then repeatedly investigates its surrounding area to find a superior solution. It evaluates each neighboring solution and selects the one that improves the objective function the most before reiterating the procedure with that solution. The algorithm terminates when it reaches a local maximum, a solution that cannot be enhanced by any adjacent solution.

4. RESULTS AND DISCUSSION

In the present study, ML algorithms were utilized to predict the occurrence of material transfer in tribological contacts. Using a dataset of experimental observations of material transfer from prior studies, we trained and evaluated a eleven of ML algorithms and employed HCLS optimization method to determine combinations of input features and algorithm parameters, for the ML algorithm thta showed best results.

4.1 Experimental Results

After loading, summarizing and preparing the data, data were used to train the considered ML algorithms. After training the considered algorithms, the values of precision were determined, which are shown in Table 1. The precision of considered ML algorithms is expressed in the form of mean and standard deviation for 10 k-folds of training data set.

Table 1. Applied algorithms parameters and training accuracy scores

No	ML algorithm	Algorithm hyperparameters	Training accuracy scores for 10 k-folds (mean and standard deviation)
1.	LR	penalty = l2, dual = False, tolerance = 0.0001, C = 1, fit_intercept = True, intercept_scaling = 1, solver = lbfgs, max_iterations = 100	0.849167 (0.093408)
2.	LDA	solver = svd, store_covariance = False, tolerance = 0.0001	0.815833 (0.088207)
3.	KNC	n_neighbors = 5, weights = uniform, leaf_size = 30, p = 2, metric = minkowski	0.828750 (0.068121)
4.	DTC	criterion = gini, splitter = best, min_samples_split = 2, min_samples_leaf = 1, min_weight_fraction_leaf = 0, min_impurity_decrease = 0, ccp_alpha = 0	0.882083 (0.056336)

5.	GNB	var_smoothing = 1e-9	0.777083 (0.172856)
6.	CSVC	C = 1, kernel = rbf, degree = 3, gamma = scale, coef0 = 0, shrinking = True, probability = False, tolerance = 0.001, cache_size = 200, verbose = False, max_iterations = -1, decision_function_shape = ovr, break_ties = False	0.868333 (0.066750)
7.	EGB	learning_rate=0.1, n_estimators=100, subsample=1, max_depth=3	0.894583 (0.061068)
8.	ABC	n_estimators = 50, learning_rate = 1, algorithm = SAMME.R	0.802917 (0.083209)
9.	GBC	learning_rate=0.1, n_estimators=100, subsample=1, max_depth=3	0.901250 (0.061295)
10.	RFC	n_estimators = 100, criterion = gini, min_samples_split = 2, min_samples_leaf = 1, max_features: str = "auto",	0.868333 (0.059722)
11.	ETC	n_estimators = 100, criterion = gini, min_samples_split = 2, min_samples_leaf = 1	0.815417 (0.093728)

The data from the table 1 shows that the GBC algorithm had the highest accuracy score (0.90) when it comes to predicting the material transfer occurrence.

Similarly, in order to additionally display the results of training algorithms, boxplot based on k-fold cross-validations were created (Figure 1). Box plot showing the distribution of accuracy score for all considered ML algorithms.

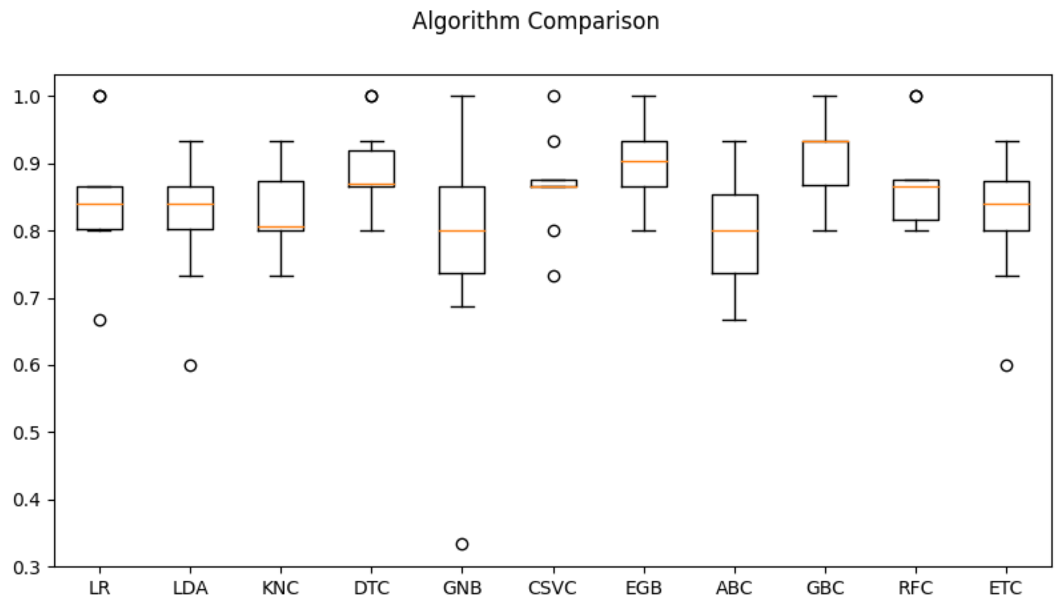

Figure 1. ML algorithm comparisons boxplots

The box represents the accuracy scores (AS) from the first quartile (Q1) to the third quartile (Q3), and the line inside the box represents the median for the k-fold cross-validations of all considered ML algorithms. The whiskers extend to the minimum and maximum values within 1.5 times the AS, and any data points beyond the whiskers are considered outliers. Presented boxplots show that the median line within boxes are closest to accuracy score of 1 for the k-fold cross-validations of GBC algorithm and that it coincides with upper whisker. This indicates that the accuracy score data for k-fold cross-validations of GBC algorithm are skewed towards higher values, with no extreme values that are considered outliers.

Optimization process for parameters values of the GBC that had the highest accuracy score has been performed with the HCLS algorithm. HCLS optimization algorithm may be used to tune the hyperparameters of the GBC model. There are many hyperparameters that we may want to optimize for the GBC model. We have focused on four key hyperparameters; and they are:

- Learning Rate (learning_rate)
- Number of Trees (n_estimators)
- Subsample Percentage (subsample)
- Tree Depth (max_depth)

The learning rate controls the contribution of each tree to the ensemble. Sensible values are less than 1.0 and slightly above 0.0 (e.g. 10−8). The number of trees controls the size of the ensemble, and often, more trees is better to a point of diminishing returns. Sensible values are between 1 tree and hundreds or thousands of trees. The subsample percentage define the random sample size used to train each tree, defined as a percentage of the size of the original dataset. Values are between a value slightly above 0.0 (e.g. 10−8) and 1.0. The tree depth is the number of levels in each tree. Deeper trees are more specific to the training dataset and perhaps overfit. Shorter trees often generalize better. Sensible values are between 1 and 10 or 20. By applying the HCLS optimization algorithm through iterations (Table 2), the optimal values of the GBC algorithm hyper parameters were determined

Table 2 shows that the accuracy score has been improved through iterations. The accuracy score prior the optimization and after the optimization of hyperpatameters for the overall data set are presented in Table 3.

Table 2. Iterations of the hill climbing optimization algorithm

Iteration	Learning Rate	Number of Trees	Subsample Percentage	Tree Depth	Accuracy score
1	0.09530291179573293	237	0.7793838317124095	1	0.87781
46	0.10122603449958356	252	0.7340932083423659	1	0.87956
53	0.09339199355325446	229	0.792218296926685	1	0.88307
122	0.07730787333501404	202	0.5903714816137815	1	0.88482

Table 3. Gradient Boosting Classifier parameters and accuracy scores

Algorithm with best peorformance	Parameters with values prior optimization	Overall accuracy score	Parameters with values after optimization	Overall accuracy score after optimization
Gradient Boosting Classifier	learning_rate=0.1, n_estimators=100, subsample=1, max_depth=3	0.871	learning_rate=0.077, n_estimators=202, subsample=0.59, max_depth=1	0.897

In this case, we can see that the best result involved using a learning rate of about 0.01, 202 trees, a subsample rate of about 59 percent, and a large depth of 1 level. This configuration resulted in a mean accuracy of about 89.7 percent, better than the default configuration that achieved an accuracy of about 87.1 percent. Below (Table 4) is the confusion matrix showing the number of correct and incorrect predictions made by the by default and optimized configuration of the GBC model compared to the actual outcomes.

Table 4. Gradient Boosting Classifier confusion matrix prior and after optimization

Confusion matrix prior optimization			Confusion matrix table after optimization		
	Predicted Negative	**Predicted Positive**		**Predicted Negative**	**Predicted Positive**
Actual Negative	7	2	**Actual Negative**	7	2
Actual Positive	3	27	**Actual Positive**	2	28

A confusion matrix shows that the number of positive instances that are correctly predicted by the default configuration of the GBC model is 27 and by the optimal configuration of the GBC model is 28. The number of positive instances that are incorrectly predicted as negative by the default configuration of the GBC model is 3 and by the optimal configuration of the GBC model is 2. Also, a confusion matrix shows the number of negative instances that are correctly predicted by the default configuration of the GBC model is 7 and by the optimal configuration of the GBC model is 7. The number of negative instances that are incorrectly predicted as positive by the default configuration of the GBC model is 2 and by the optimal configuration of the GBC model is 2. Having this in mind, it could be said that the confusion matrix shows that for a given case the accuracy of the ML algorithm is improved by optimizing its hyper parameters.

4.2 Discussion

The results of our analysis indicate that the ML models were able to accurately predict the occurrence of material transfer in tribological contacts, with a high degree of accuracy. In particular, we found that the GBC algorithm outperformed other algorithms in terms of predictive accuracy, with an overall accuracy score of 0.87.

Overall, our results suggest that ML algorithms can be a powerful tool for predicting the occurrence of material transfer in tribological contacts, and that the performance of the models can be improved by carefully selecting input features and optimizing algorithm parameters. These findings have important implications for the design and optimization of tribological systems, as they provide a means to predict and prevent material transfer, which can lead to increased system reliability and durability.

4.3 Practical Implications

The practical implications of the results suggest that the use of ML algorithms can be an effective tool for predicting material transfer in tribological contacts with a high degree of accuracy. This can help in designing and optimizing tribological systems by predicting and preventing material transfer, which can lead to increased reliability and durability of the system.

Furthermore, the finding that the GBC algorithm outperformed other algorithms in terms of predictive accuracy highlights the importance of careful algorithm selection and parameter optimization in ML models. This emphasizes the need for domain-specific expertise in choosing the appropriate algorithms and input features for the given problem.

Overall, the results demonstrate the potential of ML in predicting and preventing material transfer in tribological systems, which has important implications for industries such as automotive, manufacturing, and aerospace. This can lead to cost savings and improved performance, as well as increased safety and reliability of the systems.

5. CONCLUSIONS

The study's main novelty lies in its use of ML algorithms to predict material transfer in tribological contacts. The study showed that the GBC algorithm outperformed other algorithms in terms of predictive accuracy, and this highlights the advantages of using ML algorithms in predicting and preventing material transfer in tribological systems. The results also showed that the performance of the models can be improved by selecting appropriate input features and optimizing algorithm parameters.

The practical implications of the study are significant for industries such as automotive, manufacturing, and aerospace. Predicting and preventing material transfer in tribological systems can lead to increased system reliability and durability, resulting in cost savings and improved performance. The use of ML algorithms can be an effective tool in designing and optimizing tribological systems. The study emphasizes the need for domain-specific expertise in selecting the appropriate algorithms and input features for the given problem.

One of the limitations of the study is that the analysis was based on a limited dataset. Further studies could be conducted with larger datasets to validate the results. Additionally, the study focused only on predicting material transfer in tribological contacts and did not consider other important factors such as wear and friction. Future research could investigate the use of ML algorithms in predicting wear and friction in tribological systems. Finally, the study's findings emphasize the importance of domain-specific expertise in selecting algorithms and input features. Future studies could investigate the development of automated methods for selecting algorithms and input features based on the characteristics of the data.

REFERENCES

[1] M. S. Hasan, A. Kordijazi, P. K. Rohatgi, M. Nosonovsky: Triboinformatic modeling of dry friction and wear of aluminum base alloys using ML algorithms. Tribology International, Vol. 161, 107065, 2021.

[2] A. Kordijazi, H. M. Roshan, A. Dhingra, M. Povolo, P. K. Rohatgi, M. Nosonovsky: Machine-learning methods to predict the wetting properties of iron-based composites. Surface Innovations, Vol. 9, No. 2–3, 111-119, 2020.

[3] G. Macêdo: Material Transfer Mechanisms during Interaction of Aluminium Alloy and Tool Steel at Elevated Temperatures, Master thesis, Luleå University of Technology, Department of Engineering Sciences and Mathematics, Luleå, 2020.

[4] S. Singh, S. Gangwar, S. Yadav: A review on mechanical and tribological properties of micro/nano filled metal alloy composites. Materials Today: Proceedings, Vol. 4, No. 4, 5583-5592, 2017.

[5] J. Brownlee: (2021). Optimization for ML Finding Function Optima with Python (1st ed., p. 383). available at: MachineLearningMastery.com, accessed: 21.03.2023.

[6] N. Yin, Z. Xing, K. He, Z. Zhang: Tribo-informatics approaches in tribology research: A review. Friction, Vol. 11, No. 1, 1-22, 2023.

[7] Z. Zhang, N. Yin, S. Chen, C. Liu: Tribo-informatics: Concept, architecture, and case study. Friction, Vol. 9, 642-655, 2011.

[8] Y. Ji, J. Bao, Y. Yin, C. Ma: Applications of artificial intelligence in tribology. Recent Patents on Mechanical Engineering, Vol. 9, No. 3, 193-205, 2016.

[9] P. Kügler, M. Marian, B. Schleich, S. Tremmel, S. Wartzack: tribAIn—Towards an explicit specification of shared tribological understanding. Applied Sciences, Vol. 10, No. 13, 4421, 2020.

[10] A. Rosenkranz, M. Marian, F. J. Profito, N. Aragon, R. Shah: The use of artificial intelligence in tribology—A perspective. Lubricants, Vol. 9, No. 1, 2, 2020.

[11] I. Argatov: Artificial Neural Networks (ANNs) as a novel modeling technique in tribology. Frontiers in Mechanical Engineering, Vol. 5, 30, 2019.

[12] V.K. Mahakur, S. Bhowmik, P. K. Patowari, Tribo-informatics evaluation of dry sliding friction of silanized jute filler reinforced epoxy composites using ML techniques. Tribology International, 108388, 2023.

[13] P. Hulipalled, V. Lokesha: Optimized ML Algorithms to predict wear behavior of Tribo-Informatics, 2022.

[14] U. M. R. Paturi, S. Cheruku, N. S. Reddy: The role of artificial neural networks in prediction of mechanical and tribological properties of composites—a comprehensive review. Archives of Computational Methods in Engineering, Vol. 29, No. 5, 3109-3149, 2022.

[15] M. Marian, S. Tremmel: Current trends and applications of ML in tribology—a review. Lubricants, Vol. 9, No. 9, 86, 2021.

[16] T. Wuest, D. Weimer, C. Irgens, K. D. Thoben: ML in manufacturing: advantages, challenges, and applications. Production & Manufacturing Research, Vol. 4, No. 1, 23-45, 2016.