Sustainable Materials and Technologies 31 (2022) e00370

Contents lists available at ScienceDirect

Sustainable Materials and Technologies

journal homepage: www.elsevier.com/locate/susmat

Prediction of greenhouse gas emissions reductions via machine learning algorithms: Toward an artificial intelligence-based life cycle assessment for automotive lightweighting

Masoud Akhshik [a,*], Amy Bilton [b], Jimi Tjong [a], Chandra Veer Singh [c], Omar Faruk [a], Mohini Sain [a,b,d]

[a] Centre for Biocomposites and Biomaterials Processing, University of Toronto, 33 Willcocks St., Toronto, Ontario M5S 3B3, Canada
[b] Department of Mechanical and Industrial Engineering, University of Toronto, 5 King's College Road, Toronto, Ontario M5S 3G8, Canada
[c] Department of Materials Science and Engineering, University of Toronto, 184 College Street, Toronto, Ontario M5S 3E4, Canada
[d] Beijing University of Chemical Technology (BUCT), North Third Ring Road 15, Chaoyang District, Beijing 100029, China

ARTICLE INFO

Keywords:
Machine learning
Greenhouse gas prediction
Artificial intelligence models
Lightweighting
Natural fiber
Limited data

ABSTRACT

Within the automotive industry, there are efforts to replace glass fiber composites to a greener yet lightweighted natural fibres as they could be reducing the environmental impacts. To know if these replacements are environmentally friendlier and how much they reduce the emissions within the life cycle of the vehicle, the gold standard is a life cycle assessment (LCA) based greenhouse gas emissions. LCA is a valuable tool, However, this method is time consuming, we have to address too many details, and it could get really complicated to perform. Artificial intelligence seems to be a very promising discipline that can easily predict a complicated inquiry. In this article, we have used machine learning to compare and predict the greenhouse emissions of replacing these materials in automotive parts. This work is unique in that it processes very limited input data, in contrast to the usual machine learning dataset. This limited data usually deter researchers from solving these kinds of problems, however, it enables us to test several artificial intelligence algorithms and input matrices to quickly predict the greenhouse gas emissions for our LCA based greenhouse gas emission saving predictions. Even though this method is not conventional and needed further discussions and testing, it is showing a very promising and easy way to predict the accurate greenhouse gas saving of these materials quickly and prior to the design of the auto parts.

1. Introduction

Artificial intelligence (AI) is the field that deals with machines that are capable of showing intelligent behaviours such as problem-solving and learning [1]. This interesting field of study is re-emerging from a hibernating state, and today is a hot topic in science again, with more computation power, better algorithms, and more data collected. An AI system was able to defeat a human in the game of Go [2], which even five years ago was viewed as highly unlikely. AI has many applications: it can use natural language processing, as in Apple's Siri or Google's AI assistant, computer vision like in some self-driving cars and drones, pattern recognition for credit card fraud detection, and very important applications in modelling, simulations, and predictions like stock market predictions and weather forecasting [3]. Many of these recent applications fall into the category of AI called machine learning (ML). Like AI itself, ML is not a new field either; the word "machine learning" was coined back in 1959. ML is defined as "The ability to learn without being explicitly programmed" [4]. As the world produces more and more data every second, machine learning is helping us to process these data; in fact, with the advent of the internet of things, there is no other way to deal with these zettabytes of data. ML contains three major types of learning: supervised, unsupervised, and reinforcement learning. Supervised learning involves showing the data to the computer under supervision and letting the computer know how it is performing, and the machine is responsible for learning the rules by annotated samples [5]. On the other hand, unsupervised learning is like a school in which the student starts to see the data and tries to figure it out without supervision; sometimes, the algorithm will recognize a pattern that is

* Corresponding author.
E-mail address: Masoud.akhshik@mail.utoronto.ca (M. Akhshik).

https://doi.org/10.1016/j.susmat.2021.e00370
Received 1 November 2021; Received in revised form 25 November 2021; Accepted 27 November 2021
Available online 1 December 2021

impossible for a human to discover [6]. Reinforcement learning is the type of learning that provides feedback in terms of reward or punishment while the algorithm is learning [7].

Many algorithms exist in the domain of machine learning [8]. Some algorithms are designed as general, and some have very specific uses [9]. There is a known "no free lunch" theorem, which means that there is no single algorithm for all prediction modelling [10], and therefore a range of algorithms should be cross validated to find the best solution.

Regardless of the name, most algorithms can be categorized as performing three main tasks: regression, classification, and clustering. The current paper deals with a regression, which is a supervised modelling and prediction problem. In every regression problem, there are some known data that will help the algorithm in the prediction. As was mentioned, there are many different algorithms for regression; the most important ones are discussed briefly below.

Linear regression is the simplest and most well-known algorithm for regression. This is a powerful algorithm, especially if the relationship between the data is linear. A familiar form of this algorithm is a regression line for two types of variables. For this algorithm, overfitting could cause problems; however, there are some regularizations available to punish the high coefficients and therefore avoid overfitting. This is not a good algorithm for non-linear relationships, and as the complexity increases, this model loses its accuracy. Variations of linear regression have been developed; for example, Bayesian linear regression, which analyzes the samples based on the Bayesian inference [11] or Poisson regression, also known as a log-linear model, and has a lot of application in counting data [12].

Decision trees simply split the data into branches and maximize the information gain. This model follows several simple steps of branching, which makes it a good model for non-linear relationships. There are variations in this method, and sometimes a model will use more than one tree, which is called a decision forest or boosted decision trees. It has been reported that the decision forest has good performance on various samples [13], while boosted decision trees, although harder to adjust, can beat the decision forest in performance [14]. The tree-based techniques are very strong, and they have been a go-to algorithm for classical problems in the past [13]. A single tree is pruned for overfitting problems, and they are often inhibited by memory size or branches to avoid memorizing the samples [15].

A neural network (NN) is a family of algorithms that have been adopted based on biological neural networks (brain). There is a network of connected nodes that weight each neuron based on the provided samples. One of the major drawbacks of the NN is the fact that they may stock in local minima, and we can't be sure the solution is the best. The size of the network and the complexity of the problem will affect the performance of the NN. Designing an NN is not easy, and problems like overfitting and underfitting cause troubles [16].

A relatively newer approach is deep learning, which is a multi-layer neural network for solving complex problems. They are often used in problems by analyzing different complex data; however, these types of networks need lots of data, to begin with, and are not general-purpose algorithms. They should be fine-tuned for every purpose, and these networks are computationally expensive and harder to set up [17]. However, considering that computation power is growing, they seem to be a solution for big data analysis [18].

An emerging field of machine learning is "Information and communication technology for environmental sustainability" [19], which deals with predictions related to our environment. in this field, researchers try to model, simulate, and predict nature. Some of this research focuses on using different algorithms and even creating new algorithms to help explain and predict the environment better. An example of this work is an algorithm that has been used in global warming potential evaluations. This algorithm is able to predict emissions within 10% of the full LCA [20]. ML has been used in the prediction of other environmental factors. For example, NN models were able to predict particulate matter in the air with a diameter of 10 μm or less (PM10) reliably and outperformed the linear regression [21]. ML also has been used to expand the normalization factors for the LCA. In order to model missing toxic chemicals characterization factors, researchers used Weka, which is an open-source machine learning software package [22]. There is also research on the Organization of the Petroleum Exporting Countries (OPEC) Carbon dioxide (CO_2) emissions. In this research, an artificial neural network (ANN) was trained with the cuckoo search/particle swarm optimization algorithms. The researchers showed that the result of ANN's trained prediction is comparable with the actual real emissions data [23].

Beside from classic LCA studies [24–26] there are several studies published on LCA using a type of AI technique in the field of agriculture. For example, researchers used a Multi-Objective Genetic Algorithm to estimate energy efficiency and also reduce the global warming potential from wheat farms [27] and wetland rice production [28] and more recently, Multi-Objective optimization of energy use and environmental emissions of walnut productions [29]. Recently, ANN was used to predict yield and greenhouse gas emissions of watermelon production, and the best network topology had a correlation coefficient of 0.969 and 0.995, respectively [30]. Also, ANN has been used for energy consumption and Greenhouse gas (GHG) emissions in wheat and claimed to reach a coefficient of 0.998 for GHG emissions [31]. It has also been used for the prediction of emissions and yield for kiwifruit production [32]. In another study, researchers created a meta-model to predict the emissions of Nitrous oxide (N_2O) (as a greenhouse gas) from farm soils and showed that the correlation coefficient for this model was 0.97 for maize and 0.91 for wheat farms [33].

AI also has been used to predict rainfall run-offs. Scientists compared different AI models like ANNs, and Adaptive Neuro-fuzzy Inference Systems (ANFIS) coupled with a wavelet transform [34], or even coupling adaptive ANFIS for the environmental impact of the wheat milling factories [35]. In another study, a decision tree was compared with ANN, and the result was in favour of ANN [36]. Even the daily pan evaporation has been predicted by several machine learning methods. Comparing the results of ANN, support vector regression, fuzzy logic, and ANFIS showed that fuzzy logic and support vector regression were the best methods for this particular prediction [37]. Drought index has been forecasted using a wavelet extreme learning machine. Here, they compared several models, including ANN and support vector machines and their wavelet transformed counterparts, and proved that their model outperforms all other compared models [38]. There are also reports on hybrid models, for example, a hybrid of support vector machine and the firefly algorithm to predict global solar radiation [39]. In another study, a random forest algorithm has been used to map carbon via analyzing the remote sensed data. These researchers were able to couple spatial context to random forest and show that this has the best performance among the compared methods, including the random forest without spatial context [40].

Another group of well-studied algorithms is Bayesian principles. Bayesian networks have been reviewed in environmental modelling [41]. Also, there are case studies to compare Bayesian networks in environmental and resource management problems [42]. Bayesian networks have been studied for the prediction of fish and wildlife populations [43]. In another study, the cons and pros of Bayesian networks in environmental modelling were discussed [44]. This study compares the advantages of these networks, such as the ability to mix different sources of knowledge, fast responses, explicit treatment of uncertainty and support for decision analysis, possibility of structural learning, and ability to handle small and incomplete datasets vs. the challenges, such as discretization of continuous variables, absence of feedback loops, and difficulty of structuring expert knowledge [44].

In environmental prediction, there is outstanding research on modelling and prediction of marine environments by means of machine learning algorithms, such as genetic algorithms, which are out of the scope of this research and can be found elsewhere [45–47]. A group of researchers combined LCA and AI to predict sugarcane's environmental

impacts and output energy [48]. There is even research indicating the ability of ANN and model trees in the prediction of algal growth [49]. A similar study uses genetic programming and ANN for the prediction of harmful algal blooms [50]. Another researcher developed an aggregated boosted trees method and showed the application in ecological modelling predictions [51].

To the best of our knowledge, there has been no attempt to predict the GHG emission of fiber reinforced composites in the literature, which is the focus of this research.

The goal of this research is to estimate the greenhouse gas emissions of natural/glass fiber reinforced automotive parts via several machine learning algorithms. We have performed and find the best machine learning algorithm despite the fact that our data was limited. There are not too many LCA-based GHG emission reports in the resources, which was a limiting problem for us, and its accuracy may change as there are more LCA data published; however, this problem empowered us to do the experiments that are not possible when you have lots of data and this will be not only a guide and steppingstone for the future and more accurate predictions but also it will help us to know what type of data we need to collect more if we want to have a better predictions. While lightweighting of the car with these materials are well established, there are many details remain unknown. The result of this paper led to a better understanding of the relationship between emissions and weight and life cycle in automotive industry.

2. Materials and methods

2.1. Input matrix development and determination

All the LCA-based GHG emissions studies related to glass/natural fiber reinforced composite automotive parts (which usually result in lightweighting) were collected, and the necessary information was extracted and processed in the form of an input/output matrix (Table 1). Then we used this matrix as the source for several input/output matrices. As different resources report the GHG emission in different ways, it was not easy to compare this data; therefore, we changed the GHG emissions to a relative number indicating the percentage of the CO_2eq saved/emitted if we shifted from glass fiber to natural fiber composite. In this way, we could compare the input data regardless of the country, electricity grid mix, and any assumptions that may make the results otherwise incomparable.

At the early stage, it was obvious that every LCA has its own details and is unlikely to predict the exact number of impact categories directly; therefore, the matrix was pre-processed to show the results of LCA as the percentage of each other. In this case, even though we lose the ability to predict for single materials, we can still compare the parts as percentages of each other's emissions.

For the purpose of validation of the results of predictions, we have performed LCA-based GHG emissions for some of the automotive parts. For the validation of the model, we have also used the holdout method for cross-validation of the data sets.

To determine the best combination for input/output, several input matrices were tested to find the best matrix as an input. Among all the input data possibilities after a preliminary study, we selected three different inputs, and the analysis represented in this publication is based on these inputs. A description of the input matrices is shown in Table 1.

We also tested two different levels of skewed data to see the effect of the limited amount of data in our machine learning models, which will be discussed further. Although skewed data is not standard practice for these types of predictions, in the field of machine learning, it is really useful, especially for image recognition analysis, and it is simply skewing the original data to increase the number of samples.

Table 1
The input/output matrices for this experiment.

Input matrix 1					
Weight of lightweight resin	Weight of natural fibres	Weight of current resin	Weight of glass fibres	Mileage	GHG emission
Input matrix 2					
Weight of lightweight composite		Weight of current composite		Mileage	GHG emission *
Input matrix 3					
Lightweighting ratio		Mileage		GHG emission	*

2.2. Pre-processing data

In this paper, predictions are based on the comparison of a glass fiber reinforced composite part with the lightweight natural fiber-reinforced counterpart within the same electricity grid mix, same transportation, etc. The essential input data contains the weight of the automotive parts (both current and lightweight version), mileage of the automobile driving phase (which was after preliminary study narrowed down to 150,000 km to 290,000 km), and other data, including the energy demands, were also used in the initial development of the models.

In some of our trials, the data of the weight of resins and fibres (for both bio and current) were added together to create the general weight. We also examined the weight ratio or lightweighting percentage.

Then the data were pre-processed and normalized (feature scaled). This is an important step because the weights are usually within kilograms, and mileages are usually a six-digit number. It has been accepted as a common practice when the input data contains big numbers (the mileage in this study), the neural network can't perform well, and the effect of the bigger number will dominate the results; therefore, a normalization step is usually essential. Using the following standard formula, mileage was normalized. We have evaluated the effect of normalization on the other inputs as well. Below is the standard normalization formula used in this study:

$$Normalizedvalue = Low + \frac{Actualvalue - Min}{Max - Min} \times (High - Low) \quad (1)$$

where Min is the minimum data (150,000 km for the mileage and 0.360 kg for the weight of part)

Max is the maximum data (290,000 km for the mileage and 6.75 kg for the weight of part)

High is the new maximum (0.9)

Low is the new minimum (0.1).

The result of normalization is the new data sets, which have a distribution from 0.1 to 0.9. For example, 0.9 is the maximum for mileage and represents the driving cycle for a truck or a Sport Utility Vehicle (SUV) which is 290,000 km.

We have also used a software-assisted feature, scaling, which recommends the following formula to normalize the mileage data, which after the comparison of the final results did not cause any difference from the standard method:

$$y = 0.000006x - 0.7571 \quad (2)$$

To obtain better input/output data, all out-of-range data were trimmed and removed from the input files. These eliminated data were mostly very old studies that have different evaluations. Some of the trimmed data belonged to buses and big commercial trucks (with very high mileage), which should be out of the range that we are predicting. After this trimming, we normalized the original data and made them ready for the next step. Normalization also had another purpose: we wanted to scale the results from the lowest to highest globally. This study mainly predicts the parts that are made through the injection moulding process that usually weigh between 0.360 kg to 6.75 kg.

Please note that for the parts that have smaller weights, usually during the mould design, a grouping number is planned, and even though in our data there was a 5-gram auto part, it was combined into a set of 72, and the total weight was 0.361 kg when injection moulded.

Due to the limited data, we did not want to lose any data because of the missing number; therefore, we used the average for the missing data first, which caused a problem known as snooping (leaking or revealing the output to the AI). Instead of using the average for the missing data, the median was used, and in this way, the leakage of data to the output was reduced in comparison to using the mean.

2.3. Models

After carefully selecting the list of the different classes of machine learning algorithms to predict and score the data sets, we performed a preliminary study to see the general performance and then we further limited our list of algorithms which will be discussed here. These algorithms include linear regression, Bayesian regression, Poisson regression, neural network, boosted decision tree, and decision forest. The hyperparameters of the models were swept for the best combination. To fine-tune the hyperparameters, we also used the standard Taguchi design of the experience method. Then the best of each parameter was used for further experiments. The hyperparameters that were tested for each of the models are shown in Table 2.

After finding the best combinations, these parameters were used for further analysis. All the comparisons of the models were based on the root mean squared error (RMSE) and mean absolute error (MAE), which were mostly comparable for our study. Generally, one of the important factors in choosing the best model is MAE, which is an error that evaluates how good the predictions are. The data was initially split into 70% training, 15% validation, and 15% testing. Then it was only split into 80% for training and 20% for testing. All the models were trained using the training split and then evaluated by the 20% data that the model had not seen. The effect of the limited input data was studied by means of creating a skewed data set that was performed, which will be discussed in the result section; another aim for using skewed data was to study the possibility of using skewed data in these types of studies.

All the data analysis performed, including the Taguchi method for predicting the best parameter for the models, was performed in Minitab 17 [52].

3. Results and discussions

For simplicity, we compare all the models based on RMSE comparisons, which is a standard practice in the field.

3.1. Bayesian linear regression

In this very simple yet powerful linear regression model, there is only one changeable parameter, which is regularization weight. The sweep result of changing regularization weight has been shown in Fig. 1. As you can see the error has minimum around a regularization weight of 10 which is a match for our manual adjustment at 9. Bellow 0.1 the RMSE remain the same however after 10 it will increase linearly.

3.2. Neural network

With the neural network, we have four hyperparameters to change. These were the number of neurons, number of iterations, learning rate, and type of normalizer. As was mentioned, we perform a Taguchi test to evaluate the best combination throughout this research. The Taguchi test predicted that the best possible combination would be three neurons with 10,000 iterations with a learning rate of 0.01. The Gaussian normalizer has shown a significantly higher performance (Fig. 2). Generally, we found out that our neural network has a relatively reliable result, and the errors have minimum variance among the test replications.

3.3. Linear regression

As was mentioned for the linear regression, which is one of the simplest models, the available hyperparameter is the epoch, and we have measured it against the error. The result of error vs. epoch of training is shown in Fig. 3. As you can see the epoch training between 100 and 250 in our data sample resulted the lowest error. Considering the simplicity of this model comparing to other models, the error was decent and the fact that it is easy to explain this model, makes it a good go to model for future works.

3.4. Poisson

For the Poisson, we have four hyperparameters to adjust (Optimization Tolerance, L1 Weight, L2 Weight, Memory Size) after sweeping for the best combination on each of the inputs. We set the combination at 1000 different combinations and did the sweep for 10,000 replications. Then we chose the best combination by MAE and RMSE. One unique feature about this mode was that the same error could be reached by many of the combinations. Even though our data is not considered count data we kept Poisson in our modelling for the purpose of comparison.

3.5. Boosted decision tree

As it was mentioned the Taguchi method was used to have a prediction about the hyper parameters. In boosted decision tree we have also used this method to estimate the most effective hyperparameters. The Taguchi method prediction for the best error is shown in Table 3. The experience was repeated 1000 times and the data are the average of the ten lowest errors for each input. Boosted decision trees are very strong and they are usually great for the predictions, However, our limited data seems to affect this model's performance.

3.6. Decision forest

For the decision forest, we noticed that many variations of

Table 2
The machine learning algorithms and the hyperparameters that were tested in this study.

ML algorithm	Parameters				
Linear regression	Epoch training				
Bayesian	Regularization weights				
Poisson	Optimization Tolerance	L1 Weight	L2 Weight	Memory Size	
Boosted decision tree	Maximum number of leaves per tree	Minimum number of samples per leaf node	Learning rate	Total number of trees constructed	
Decision forest	Resampling method	Number of decision trees	Maximum depth of the decision trees	Number of random splits per node	Minimum number of samples per leaf node
Neural network	Number of neurons in hidden layer (3, 4, 5, 6)	Number of iteration (1,005,001,000 10,000)	Normalizer type (Min/Max, Binning, Gaussian, Not normalizing)	Learning rate	

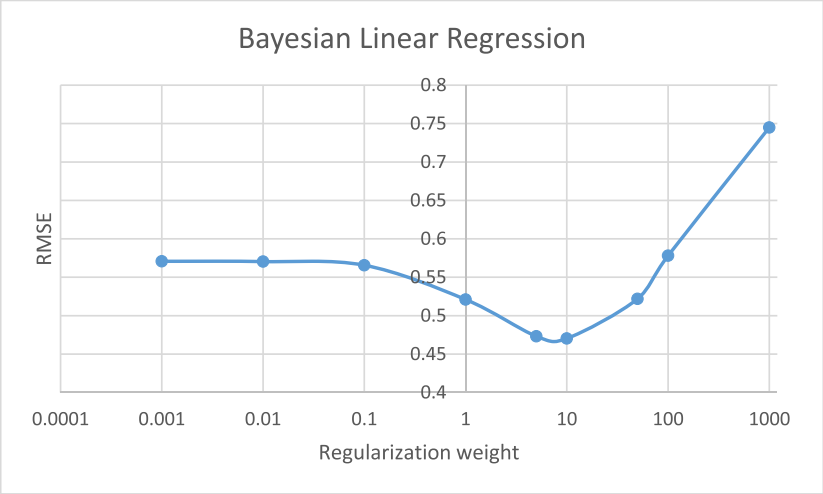

Fig. 1. The plot shows the root mean squared error for the Bayesian model with different regularization weights from 0 to 1000.

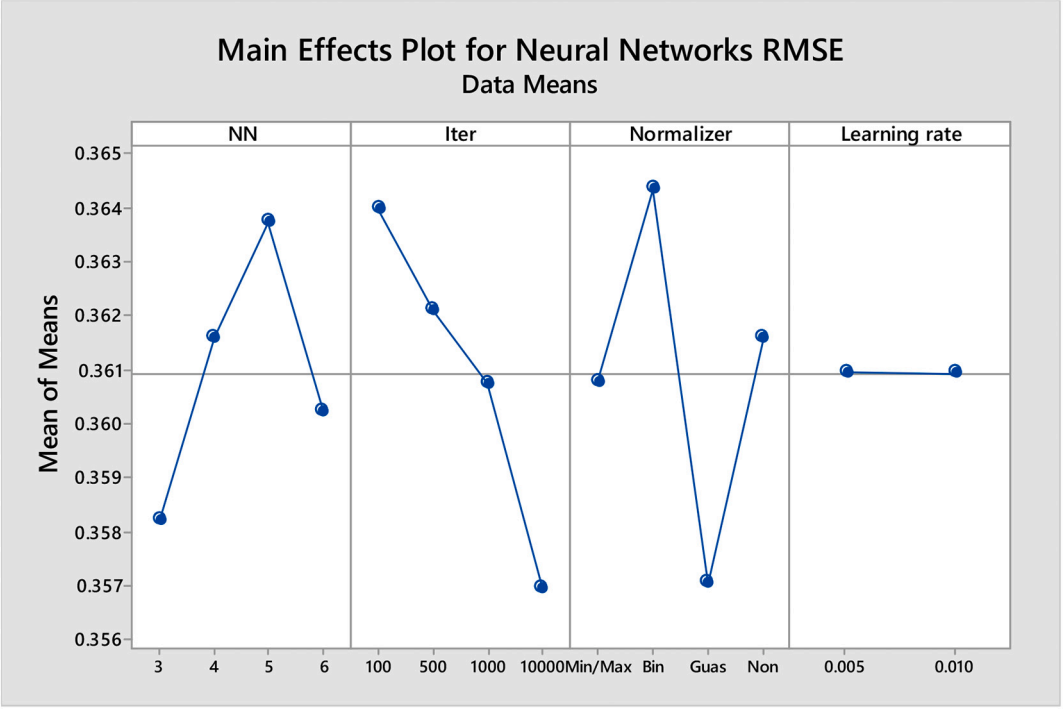

Fig. 2. The plot shows the result of the Taguchi analysis of hyperparameters. We have studied 3–6 neurons, 100–10,000 iterations, two typical learning rates 0.005 and 0.01 and different normalizers including (not normalizing, Min/Max, Bin and Gaussian. The Root mean squared errors for the neural network model has been shown at the left bar.

hyperparameters result in the lowest with MAE (0.09), and RMSE fluctuates between 0.12 and 0.13. The averages for these were: 1.79 for the minimum number of samples per leaf node, 367.5 for the number of random splits per node, 17.17 for the maximum depth of decision trees, and 8.54 for the number of decision trees. As you have noticed, these averages are not telling us the whole story. What was seen was most of the samples per node were 1 or 2 with a few exceptions of 3 and 4. Therefore, we focused on these two numbers, and the averages for them were rounded to a true number here: The best-chosen parameters were 2, 133, 10, 9 and 1, 475, 21, 11, which was very similar for inputs 1 and 2 and for input 3. These results were compatible with the Taguchi experiment that we performed (Table 4) again here the numbers are an average of ten lowest error and the experiment repeated 1000 times. As an ensemble method as it was expected the decision forest performed better than our other ensemble model boosted decision tree.

Fig. 4 shows a comparison for the lowest RMSE for different machine learning models. Except for linear regression and the Bayesian model, every other model responded better with inputs 1 and 2 and had a higher error with input 3. Based on these graphs, we can see that most of the models performed well with inputs 1 and 2; however, the Bayesian model did not perform as well as others. One noticeable thing was that the Bayesian provided a significantly better result for input 3, and this could be showing that this model can handle complex inputs better than the others.

For a better understanding of the models after setting the hyperparameters, we created different combinations of the input data vs.

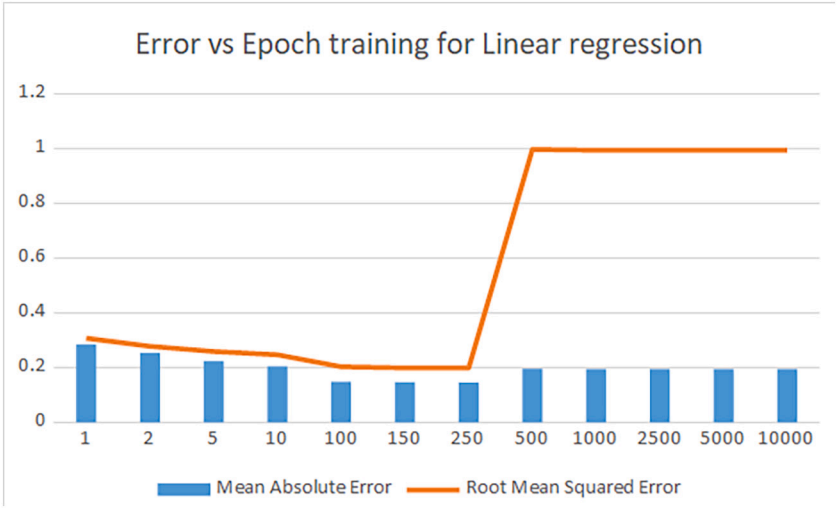

Fig. 3. Figure shows error versus epoch for linear regression as you can see an epoch training of 100 to 250 has the lowest error.

Table 3
shows Taguchi predicted parameters for the boosted decision trees.

	Number of leaves	Minimum leaf instances	Learning rate	Number of threes
Input 1	5	3	0.35	99
Input 2	5	2	0.22	100
Input 3	4	2	0.10	64

Table 4
The predicted parameters for the decision forest models.

	Minimum number of samples per leaf node	Number of random splits per node	Maximum depth of decision trees	Number of decision trees
Input 1 and 2	1	475	21	11
Input 1 and 2	2	133	10	9
Input 3	1	295	14	10

validation data (which will not be seen by the model). Then we tested our models with these data and recorded the best and the worst performances. Fig. 5 shows the best and the worst errors for these combinations.

The performance of the machine learning algorithms can be seen in Fig. 6. As is shown, Bayesian regression is the worst model among all regardless of input matrix; however, other models' performance was good. Choosing the best model is not straightforward, as the worst-performing models with higher standard deviation led to the lowest possible RMSE. A higher population standard deviation means that our result could have a higher error in some cases; this error will undermine the reliability and the accuracy of the models' predictions. The model that has the lowest deviation is the neural network, followed by the decision forest for input 1 and Poisson regression for input 2. Generally speaking, the neural network, Poisson, and decision forest can all be used to make predictions. The boosted decision trees, in some cases, predict with an error and standard population deviation higher than was expected from the general performance of the model; even though forest models usually perform well in prediction, it is not the case here.

3.7. Use of skewed data

The skewed data was produced based on the random generation of numbers; however, the model was forced to keep the means and standard population deviations and the distribution of the original inputs to reduce the bias in the system. These random numbers (namely 84 and 168) were mixed with the 28 original samples and then used for the purpose of training and evaluation. In one instance, the training performed on the skewed data and the evaluation performed solely on the original data, and the difference in the RMSE was not statistically significant, even though the latter tend to have lower RMSE. Having more data, has the potential to enhance the model performance (Fig. 7).

3.8. Input matrices

As was mentioned, to see a better picture of the effect of the input matrices, we used three different input matrices with different levels of detail. As you can see in Fig. 8, generally, with the exception of the Bayesian regression, inputs 2 and 1 are comparable, and input 1 is slightly better in most cases. It can also be inferred that some of the models here are more sensitive to the input; for example, the methods that use trees (random forest and boosted decision tree) predict similarly under the different types of inputs. Input 3 seems to lose some of the important details, and except for the linear regression, other models' prediction power shows a reduced accuracy.

3.9. Cross-validation of the models

To check the general performance of the models in the prediction of unseen data, we used real-life data from the actual LCA results. In this cross-validation experiment, we calculated both RMSE and MSE and their relevant standard deviation (Fig. 9).

As you can see, the results are comparable with previous calculations, and in terms of predictions' accuracy, Bayesian regression is not an impressive model for this kind of prediction. Linear regression, on the other hand, shows an outstanding performance for these data.

3.10. Knowledge extraction

The next step here is to reverse engineer the algorithms to see how they are calculating and extracting the knowledge they have learned during the training and see how this model performs the prediction. Table 5 shows the extracted knowledge from the machine learning models. This table simply shows the formula that the AI algorithm uses

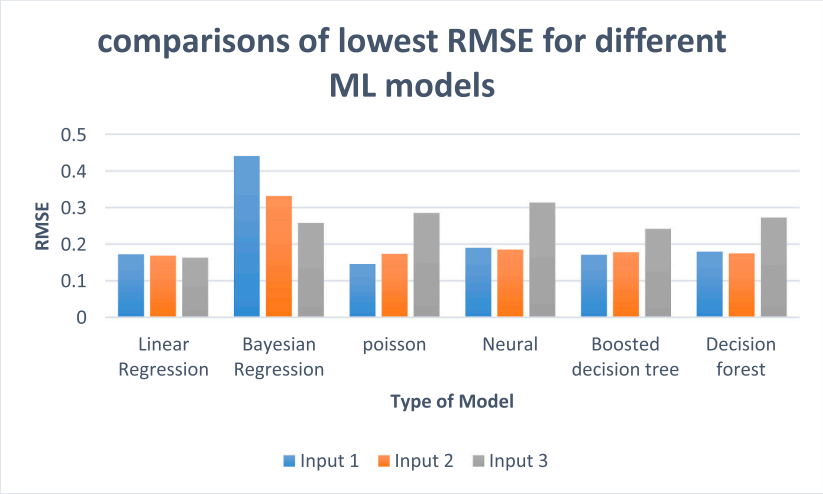

Fig. 4. Diagram of the machine learning models based on the root mean squared error. This diagram also contains information on the three main inputs and provides a visual comparison of the different input matrices.

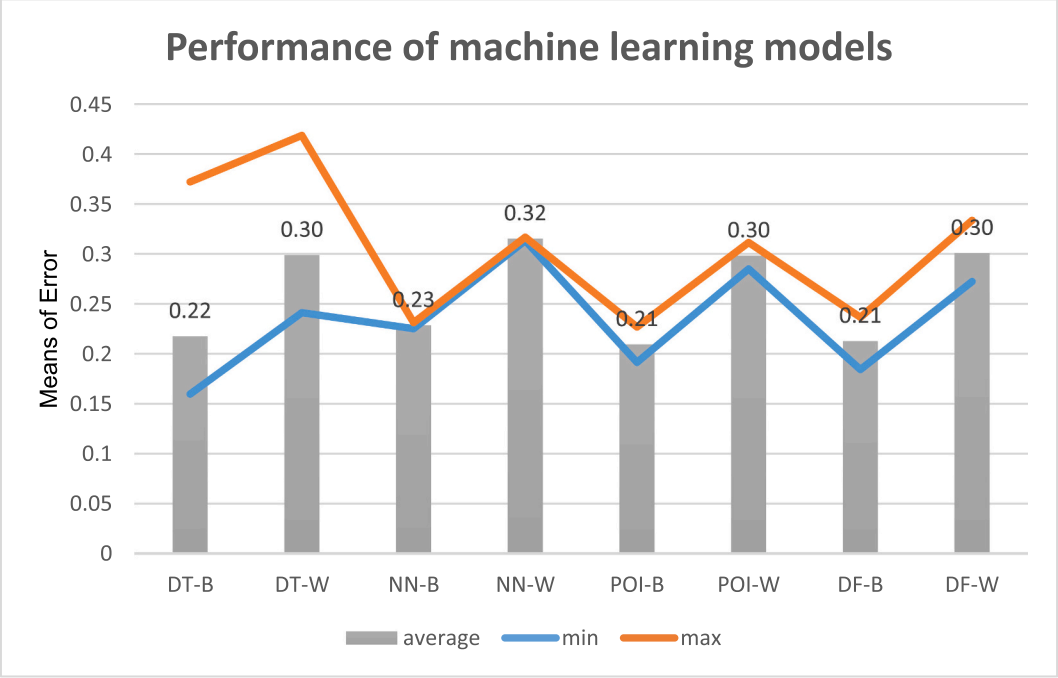

Fig. 5. Performance of the machine learning models. The diagram shows different models' best and worst results and the means of the errors for the models: DT-B (Decision Tree Best), DT-W (Decision Tree Worst), NN-B (Neural Network Best), NN-W (Neural Network Worst), POI-B (Poisson Best), POI-W (Poisson Worst), DF-B (Decision Forest Best), DF-W (Decision Forest Worst).

to predict our emissions.

As you can see the standard deviation of errors was the lowest in the neural network model, and the proofed to be a reliable model for our predictions. Please note that these table will change if we have different data sets and as more data become available a comparison of this table with the new one has the potential to show a trend in our future.

3.11. Skewed data analysis

The skewed analysis showed that having more data will help the accuracy of the system. In our experience, the RMSE is directly affected by the amount of input data. As more and more LCA is performed and published, the input file will grow, and this system will become more and more accurate. Even though skewed data is not a standard method for prediction studies we have used this as a test of our system with the increased number of inputs.

4. Conclusions

Machine learning can be really helpful in many fields; however, there are some fields that can't use the advantage of machine learning like others. One of these fields is the prediction of greenhouse gases, which was studied here. The main reason that machine learning has not been developed in this field is the lack of sufficient data. After studying the literature, we figured out that there are less than 30 published papers that can be potentially used in this research, and out of that, half was old

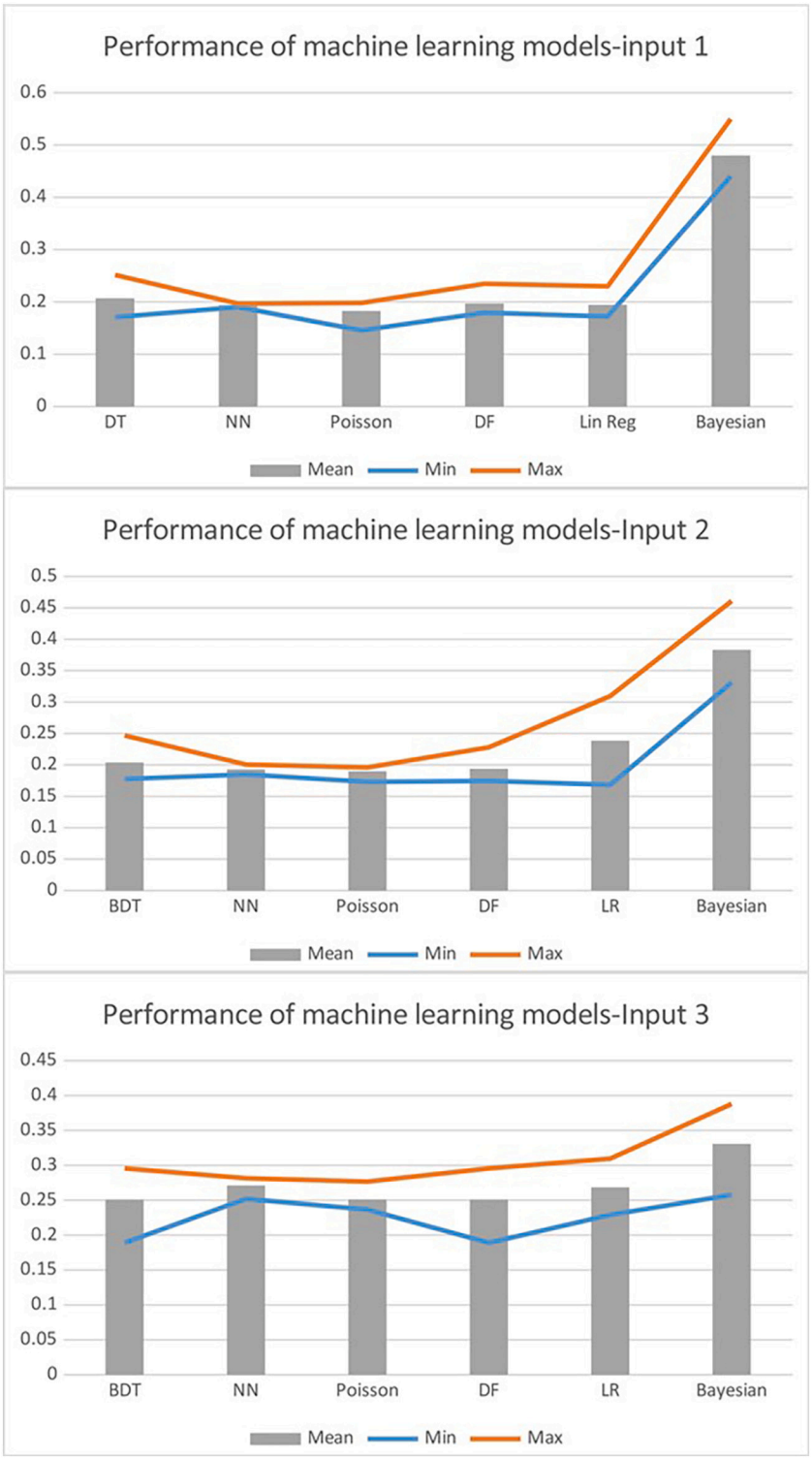

Fig. 6. The RMSE for different machine learning models for different inputs: BDT = Boosted decision tree, NN = Neural network, DF = Decision Forest, LR = Linear regression. The lines show the minimum and the maximum error of the prediction for the model.

and unrelated to the current automotive industry.

Even though with this amount of data, machine learning may not be accurate, here we lay a foundation of the studies that can bring emission prediction to real life.

Having limited input data also helped us to do analysis, which is normally impossible to do in the field of AI due to the large quantities of data. We have tested all major machine learning algorithms, analyzed them, and extracted the learned knowledge from them. These models mostly performed as expected with the exception of Bayesian, which underperformed, and linear regression, which overperformed.

We showed that with all its limitations and scarce input data, machine learning can still be beneficial, can predict with an acceptable error, and will help to shape the future of emission research.

Part of this paper includes a very unconventional method of skewed data in order to increase the number of input data and see the performance of machine learning. It could be seen that by having more data,

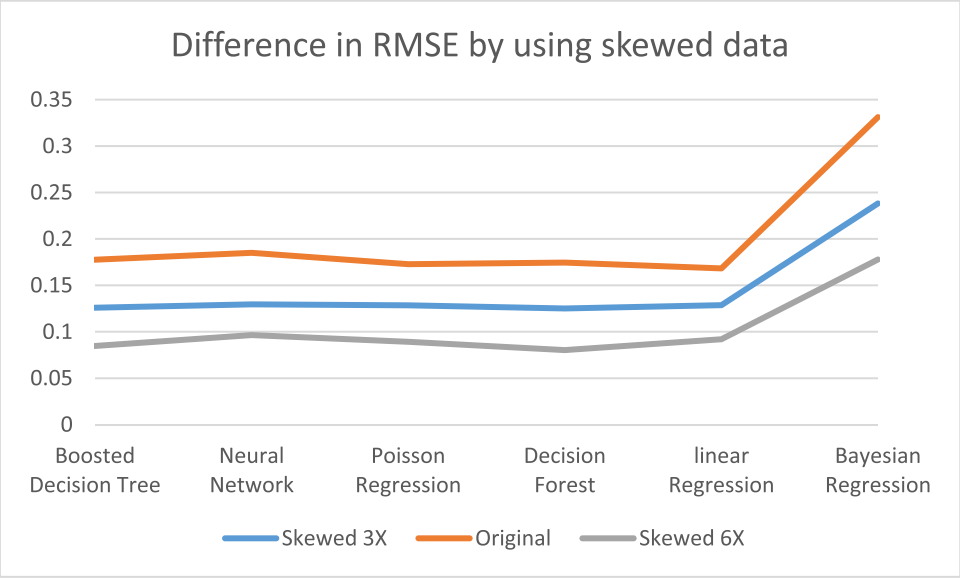

Fig. 7. The difference in RMSE between the models by using skewed data. We have used 3× and 6× data.

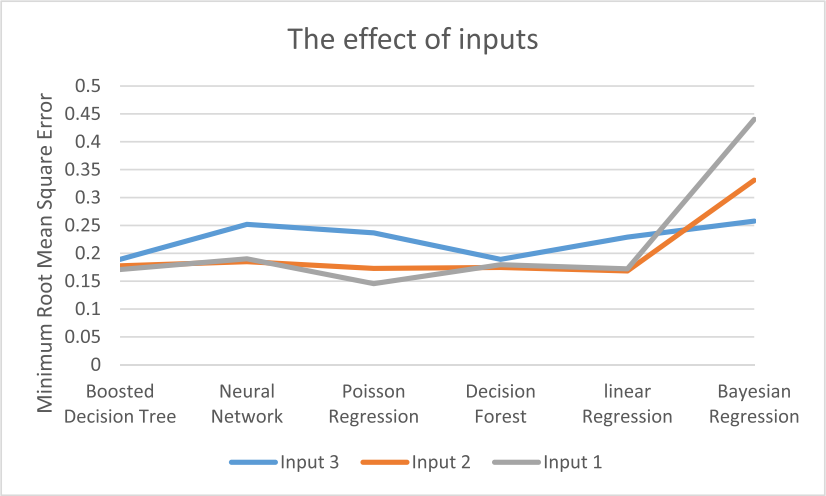

Fig. 8. The effect of the input matrix in different machine learning models. Generally, input with more details produces a better prediction except for the Bayesian regression model.

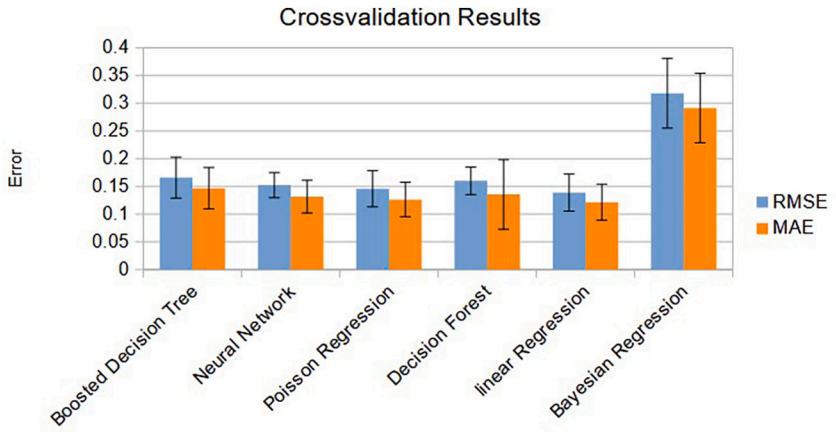

Fig. 9. The result of the cross-validation of the machine learning models. The Bayesian model error was the highest, and the linear regression error was the lowest. The error bars are standard deviations.

Table 5
Extracted knowledge from the AI algorithms. Weights are in kilograms, and mileages are in kilometres.

Neural network	Input 1	0.819942–0.260798 Resin weight (lightweight) + 0.791235 Fiber weight (lightweight) + 0.021359 Resin weight (Current) - 0.488466Fiber weight (Current) - 0.040064 Mileage
	Input 2	0.867160 + 0.29675 Weight (lightweight) - 0.276144 Weight (Current) - 0.100659 Mileage
	Input 3	0.44885 + 0.5937 Lightweighting - 0.26025 Mileage
DF	Input 1	−0.84 + 4.0 Resin weight (lightweight) + 21.2 Fiber weight (lightweight) - 7.0 Resin weight (Current) - 3.4Fiber weight (Current) + 0.207 Mileage
	Input 2	0.753 + 3.01 Weight (lightweight) - 2.28 Weight (Current) - 0.061 Mileage
	Input 3	0.289 + 0.584 Lightweighting - 0.034 Mileage
Poisson	Input 1	0.0129 + 2.929 Resin weight (lightweight) + 20.362 Fiber weight (lightweight) - 5.286 Resin weight (Current) - 9.431Fiber weight (Current) - 0.1050 Mileage
	Input 2	0.85092 + 1.4909 Weight (lightweight) - 1.2159 Weight (Current) - 0.12336 Mileage
	Input 3	0.3792 + 0.6363 Lightweighting - 0.1795 Mileage
DT	Input 1	1.288–6.88 Resin weight (lightweight) + 1.94 Fiber weight (lightweight) + 10.32 Resin weight (Current) - 9.41Fiber weight (Current) - 0.238 Mileage
	Input 2	0.782 + 0.82 Weight (lightweight) - 0.79 Weight (Current) - 0.015 Mileage
	Input 3	0.854–0.069 Lightweighting - 0.522 Mileage
Lin reg	Input 1	0.750883–0.288261 Resin weight (lightweight) + 4.32249 Fiber weight (lightweight) - 0.517907 Resin weight (Current) - 2.64599Fiber weight (Current) - 0.060447 Mileage
	Input 2	0.861149 + 1.96585 Weight (lightweight) - 1.57342 Weight (Current) - 0.161267 Mileage
	Input 3	0.392452 + 0.669591 Lightweighting - 0.242807 Mileage
Bayesian	Input 1	−0.000011 + 0.161135 Resin weight (lightweight) + 0.146636 Fiber weight (lightweight) + 0.175462 Resin weight (Current) + 0.147620Fiber weight (Current) + 0.617075 Mileage
	Input 2	0.000002 + 2.67713 Weight (lightweight) - 0.891383 Weight (Current) + 0.516348 Mileage
	Input 3	−0.000001 + 1.17834 Lightweighting - 0.090107 Mileage

we can generally expect a better performance in all the models. Some models, like the Bayesian, could benefit a little more, and some, like linear regression, tend to be less sensitive.

We have a long way ahead of us to have a general AI that can look at some scarce data, find a trend, and predict. A good predictive AI in the complex field of emissions will help us to make better environmental decisions.

Funding

This work was supported by the NSERC-Automotive Partnership Canada Program [grant number APCPJ 433821 – 12]; Mitacs Accelerate Program [grant number IT04834] and Ontario Research Fund – Research Excellence [grant number ORF-RE07-041].

Declaration of Competing Interest

None.

Acknowledgement

The authors would like to thank Ford Motor Company of Canada, especially the Powertrain Engineering Research & Development Centre (PERDC) for providing the in-kind support for this project. We would like to say thanks to the EPICentre, University of Windsor, and its epic team for their kind support. Also, we would like to appreciate Microsoft Canada Corp., IBM Canada Ltd. and Advanced Hi-Tech Centre Ltd. for their access to their AI platforms and hardware. The authors also want to thank Dr. Arash Akhshik and Dr. Birat KC, for their kind supports. We also express our deep gratitude for the researchers who helped us with this paper.

Credit statement

Masoud Akhshik: Conceptualization, Methodology, Visualization, Investigation, Writing- Original draft preparation.
Omar Faruk: Data collection, Writing and Reviewing and Editing.
Jimi Tjong: Supervision, Data collection, Reviewing and Editing.
Amy Bilton: Methodology, Editing and Reviewing.
Chandra Veer Singh: Conceptualization, Reviewing.
Mohini Sain: Supervision, Conceptualization, Reviewing and Editing.

We certify that all persons who meet authorship criteria are listed as authors, and also all authors certify that they have participated sufficiently in the work to take public responsibility for the content, including participation in the concept, design, analysis, writing, or revision of the manuscript. Furthermore, each author certifies that this material or similar material has not been and will not be submitted to or published in any other publication before its appearance in the Journal of sustainable materials and technologies.

Declaration of Competing Interest

All the authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Barr, E. Feigenbaum, C. Roads, The Handbook of Artificial Intelligence Vol. 1, 1982.
[2] E. Gibney, Google AI algorithm masters ancient game of Go, Nat. News *529* (7587) (2016) (p.445).
[3] U. Fiore, A. De Santis, F. Perla, P. Zanetti, F. Palmieri, Using generative adversarial networks for improving classification effectiveness in credit card fraud detection, Inf. Sci. 479 (2019) 448–455.
[4] A.L. Samuel, Some studies in machine learning using the game of checkers, IBM J. Res. Dev. 3 (3) (1959) 210–229.
[5] K.C. Fu (Ed.), Sequential Methods in Pattern Recognition and Machine Learning, Academic Press, 1968.
[6] E.M. Darling Jr., J.G. Raudseps, Non-parametric unsupervised learning with applications to image classification, Pattern Recogn. 2 (4) (1970) 313–335.
[7] J.M. Mendel, R.W. McLaren, 8 reinforcement-learning control and pattern recognition systems, in: Mathematics in Science and Engineering Vol. 66, 1970, pp. 287–318.
[8] S. Ray, A quick review of machine learning algorithms, in: 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon), IEEE, 2019, February, pp. 35–39.
[9] C. Kim, R. Batra, L. Chen, H. Tran, R. Ramprasad, Polymer design using genetic algorithm and machine learning, Comput. Mater. Sci. 186 (2021) 110067.
[10] D.H. Wolpert, The lack of a priori distinctions between learning algorithms, Neural Comput. 8 (7) (1996) 1341–1390.
[11] S. Rong, Z. Bao-wen, The research of regression model in machine learning field, in: *MATEC Web of Conferences.* EDP Sciences Vol. 176, 2018 (p. 01033).
[12] A. Colin Cameron, Pravin K. Trivedi, Regression Analysis of Count Data Vol. 53, Cambridge University Press, 2013.
[13] L. Rokach, Decision forest: twenty years of research, Inform. Fusion 27 (2016) 111–125.
[14] H.K. Sok, M.P.L. Ooi, Y.C. Kuang, S. Demidenko, Multivariate alternating decision trees, Pattern Recogn. 50 (2016) 195–209.
[15] T. Dietterich, Overfitting and undercomputing in machine learning, ACM Comput. Surv. 27 (3) (1995) 326–327.
[16] A.D. Gavrilov, A. Jordache, M. Vasdani, J. Deng, Preventing model overfitting and underfitting in convolutional neural networks, Int. J. Softw. Sci. Comput. Intell. 10 (4) (2018) 19–28.
[17] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[18] M.M. Najafabadi, F. Villanustre, T.M. Khoshgoftaar, N. Seliya, R. Wald, E. Muharemagic, Deep learning applications and challenges in big data analytics, J. Big Data 2 (1) (2015) 1–21.
[19] L.M. Hilty, B. Aebischer, ICT for sustainability: an emerging research field, in: ICT Innovations for Sustainability, 2015, pp. 3–36.
[20] A. Bala, M. Raugei, G. Benveniste, C. Gazulla, P. Fullana-i-Palmer, Simplified tools for global warming potential evaluation: when 'good enough'is best, Int. J. Life Cycle Assess. 15 (5) (2010) 489–498.
[21] G. Grivas, A. Chaloulakou, Artificial neural network models for prediction of PM 10 hourly concentrations, in the Greater Area of Athens, Greece, Atmos. Environ. 40 (7) (2006) 1216–1229.
[22] M. Slapnik, D. Istenič, M. Pintar, A. Udovč, Extending life cycle assessment normalization factors and use of machine learning–a Slovenian case study, Ecol. Indic. 50 (2015) 161–172.
[23] H. Chiroma, S. Abdul-Kareem, A. Khan, N.M. Nawi, A.Y.U. Gital, L. Shuib, A. I. Abubakar, M.Z. Rahman, T. Herawan, Global warming: predicting OPEC carbon dioxide emissions from petroleum consumption using neural network and hybrid cuckoo search algorithm, PLoS One *10* (8) (2015) (p.e0136140).
[24] F. Hosseini-Fashami, A. Motevali, A. Nabavi-Pelesaraei, S.J. Hashemi, K.W. Chau, Energy-life cycle assessment on applying solar technologies for greenhouse strawberry production, Renew. Sust. Energ. Rev. 116 (2019) 109411.
[25] M. Akhshik, S. Panthapulakkal, J. Tjong, M. Sain, The effect of lightweighting on greenhouse gas emissions and life cycle energy for automotive composite parts, Clean Techn. Environ. Policy 21 (3) (2019) 625–636.
[26] H. Ordikhani, M.G. Parashkoohi, D.M. Zamani, M. Ghahderijani, Energy-environmental life cycle assessment and cumulative exergy demand analysis for horticultural crops (case study: Qazvin province), Energy Rep. 7 (2021) 2899–2915.
[27] A. Nabavi-Pelesaraei, H. Hosseinzadeh-Bandbafha, P. Qasemi-Kordkheili, H. Kouchaki-Penchah, F. Riahi-Dorcheh, Applying optimization techniques to improve of energy efficiency and GHG (greenhouse gas) emissions of wheat production, Energy 103 (2016) 672–678.
[28] S. Elsoragaby, A. Yahya, M.R. Mahadi, N.M. Nawi, M. Mairghany, S.M.M. Elhassan, A.F. Kheiralla, Applying multi-objective genetic algorithm (MOGA) to optimize the energy inputs and greenhouse gas emissions (GHG) in wetland rice production, Energy Rep. 6 (2020) 2988–2998.
[29] M. Khanali, A. Akram, J. Behzadi, F. Mostashari-Rad, Z. Saber, K.W. Chau, A. Nabavi-Pelesaraei, Multi-objective optimization of energy use and environmental emissions for walnut production using imperialist competitive algorithm, Appl. Energy 284 (2021) 116342.
[30] A. Nabavi-Pelesaraei, R. Abdi, S. Rafiee, Neural network modeling of energy use and greenhouse gas emissions of watermelon production systems, J. Saudi Soc. Agric. Sci. 15 (1) (2016) 38–47.
[31] B. Khoshnevisan, S. Rafiee, M. Omid, M. Yousefi, M. Movahedi, Modeling of energy consumption and GHG (greenhouse gas) emissions in wheat production in Esfahan province of Iran using artificial neural networks, Energy 52 (2013) 333–338.
[32] A. Nabavi-Pelesaraei, S. Rafiee, H. Hosseinzadeh-Bandbafha, S. Shamshirband, Modeling energy consumption and greenhouse gas emissions for kiwifruit production using artificial neural networks, J. Clean. Prod. 133 (2016) 924–931.
[33] J. Perlman, R.J. Hijmans, W.R. Horwath, A metamodelling approach to estimate global N2O emissions from agricultural soils, Glob. Ecol. Biogeogr. 23 (8) (2014) 912–924.
[34] V. Nourani, Ö. Kisi, M. Komasi, Two hybrid artificial intelligence approaches for modeling rainfall–runoff process, J. Hydrol. 402 (1) (2011) 41–59.
[35] A. Nabavi-Pelesaraei, S. Rafiee, S.S. Mohtasebi, H. Hosseinzadeh-Bandbafha, K. W. Chau, Comprehensive model of energy, environmental impacts and economic in rice milling factories by coupling adaptive neuro-fuzzy inference system and life cycle assessment, J. Clean. Prod. 217 (2019) 742–756.
[36] D.P. Solomatine, K.N. Dulal, Model trees as an alternative to neural networks in rainfall—runoff modelling, Hydrol. Sci. J. 48 (3) (2003) 399–411.
[37] M.K. Goyal, B. Bharti, J. Quilty, J. Adamowski, A. Pandey, Modeling of daily pan evaporation in sub tropical climates using ANN, LS-SVR, Fuzzy Logic, and ANFIS, Expert Syst. Appl. 41 (11) (2014) 5267–5276.
[38] R.C. Deo, M.K. Tiwari, J.F. Adamowski, J.M. Quilty, Forecasting effective drought index using a wavelet extreme learning machine (W-ELM) model, Stoch. Env. Res. Risk A. 31 (5) (2017) 1211–1240.
[39] L. Olatomiwa, S. Mekhilef, S. Shamshirband, K. Mohammadi, D. Petković, C. Sudheer, A support vector machine–firefly algorithm-based model for global solar radiation prediction, Sol. Energy 115 (2015) 632–644.
[40] J. Mascaro, G.P. Asner, D.E. Knapp, T. Kennedy-Bowdoin, R.E. Martin, C. Anderson, M. Higgins, K.D. Chadwick, A tale of two "forests": random forest machine learning aids tropical forest carbon mapping, PLoS One 9 (1) (2014) (p. e85993).
[41] P.A. Aguilera, A. Fernández, R. Fernández, R. Rumí, A. Salmerón, Bayesian networks in environmental modelling, Environ. Model. Softw. 26 (12) (2011) 1376–1388.
[42] D.N. Barton, S. Kuikka, O. Varis, L. Uusitalo, H.J. Henriksen, M. Borsuk, A. de la Hera, R. Farmani, S. Johnson, J.D. Linnell, Bayesian networks in environmental and resource management, Integr. Environ. Assess. Manag. 8 (3) (2012) 418–429.
[43] B.G. Marcot, R.S. Holthausen, M.G. Raphael, M.M. Rowland, M.J. Wisdom, Using Bayesian belief networks to evaluate fish and wildlife population viability under land management alternatives from an environmental impact statement, For. Ecol. Manag. 153 (1) (2001) 29–42.
[44] L. Uusitalo, Advantages and challenges of Bayesian networks in environmental modelling, Ecol. Model. 203 (3) (2007) 312–318.
[45] E.O. Wiley, K.M. McNyset, A.T. Peterson, C.R. Robins, A.M. Stewart, Niche modeling perspective on geographic range predictions in the marine environment using a machine-learning algorithm, Oceanography 16 (3) (2003) 120–127.
[46] T. Deng, K.W. Chau, H.F. Duan, Machine learning based marine water quality prediction for coastal hydro-environment management, J. Environ. Manag. 284 (2021) 112051.
[47] A.P. Michel, A.E. Morrison, V.L. Preston, C.T. Marx, B.C. Colson, H.K. White, Rapid identification of marine plastic debris via spectroscopic techniques and machine learning classifiers, Environ. Sci. Technol. 54 (17) (2020) 10630–10637.
[48] A. Kaab, M. Sharifi, H. Mobli, A. Nabavi-Pelesaraei, K.W. Chau, Combined life cycle assessment and artificial intelligence for prediction of output energy and environmental impacts of sugarcane production, Sci. Total Environ. 664 (2019) 1005–1019.
[49] N.C. Jung, I. Popescu, P. Kelderman, D.P. Solomatine, R.K. Price, Application of model trees and other machine learning techniques for algal growth prediction in Yongdam reservoir, Republic of Korea, J. Hydroinf. 12 (3) (2010) 262–274.
[50] N. Muttil, K.W. Chau, Machine-learning paradigms for selecting ecologically significant input variables, Eng. Appl. Artif. Intell. 20 (6) (2007) 735–744.
[51] G. De'Ath, Boosted trees for ecological modeling and prediction, Ecology 88 (1) (2007) 243–251.
[52] Minitab 17 Statistical Software, Computer software, Minitab, Inc., State College, PA, 2010. www.minitab.com.