Computational Alloy Design and Discovery Using Machine Learning

Yadu Krishna Choyi
u6728671

A report submitted for the course

Comp 4560 Advanced Computing Project

Supervised by: Prof. Nick Birbilis

The Australian National University

June 2020

Except where otherwise indicated, this report is my original work.

Yadu Krishna Choyi
June 2020

Acknowledgment

I want to express my sincerest thanks to Prof. Nick Birbilis not only for providing me with this research opportunity but also for his support and guidance throughout the past year. This research opportunity helped in giving me insight into the real-world application of machine learning, which I believe will be beneficial for my academics and future career.

Special thanks to my family and friends who's love and support helped and motivated me through this difficult time.

Abstract

Currently, the development of new materials such as alloys, using conventional research techniques such as empirical and stepwise approaches, is a lengthy, expensive process. In this project, we focus on the application of machine learning techniques on already existing experimental data relating to 5xxx series aluminum alloys to gain knowledge into the high dimensional relationship between alloy composition and its physical properties. The knowledge gained is used to screen generated alloy composition based on its corrosion resistance property. This process gives us the ability to select a handful of specific alloy composition for further experimental verification, instead of having an infinite number of possible compositions that cannot be verified through an exhaustive process. The results show us that a machine learning algorithm can accurately learn the relationship between the input alloy composition and output feature, which in this case is the material properties.

Using an algorithm to generate alloy compositions that have not yet been used in real-world applications, we generate millions of possible alloy compositions, which are then passed over though a screening stage to eliminate combinations of elements that are not applicable to our case. The resulting compositions after the screening process are then examined for validity. The screening stage is implemented as a machine learning model that has been trained on already existing 5xxx series aluminum alloys.

Table of Contents

List of Figures

Index of Tables

1 Introduction

1.1 Motivation

Alloys are central to human society. They allow for structural design and architecture, packaging, and electronics, as well as applications in the automotive and aerospace sector. Alloys have been produced so far based on a method of coming up with recipes with new compositions, followed by properties and structure testing and characterization. The design and discovery of new alloys were essential elements of advanced engineering systems for many years. However, this design method, complicated by its chemical and structural complexities, is sometimes too slow. The screening of materials that has good performance with respect to underlying material properties is a trending topic in the field of material science. The traditional method of experimenting and modeling often requires a significant amount of time and resources and also involves a considerable degree of trial and error. Only a tiny fraction of potentially interesting alloys are put into practical use, which also means that the identification of new compositions with correspondingly suited properties and microstructures can not be readily pursued by using conventional techniques.

In the last few years, through this time-consuming method, humankind has been able to accept about 100,000 different alloys. However, the feature space of possible alloys exceeds that number by many orders of magnitude, and in this vast space of possibilities are buried novel discoveries that are critical to confronting global challenges. Machine learning provides the ability to significantly accelerate the discovery of new alloys by incorporating the knowledge obtained from previous experiments. Therefore applying machine learning algorithms to the materials sector is a research area with high potential-impact and significant benefits to society.

1.2 Problem Statement and Objective

Using the data collected form 'A Survey of Sensitization in 5xxx Series Aluminum Alloys'[2], we are tasked to implement a machine learning algorithm which can then be further used to screen randomly generated alloy composition with respect to its degree of sensitization[DoS](see section 2.1.2). The average DoS value for all the available composition in the data set used for this project is 26.9. The aim here is to generate a composition which has a DoS value considerably lower(close to 0) than 26.9.

The objective of this project is to analyze the advantage of using machine learning algorithms in terms of its ability to significantly reduce the search space of discovering potentially new aluminum alloys with new material properties that can find use in various fields such as the automotive, technology and the space industry. The report also explores the ability of various machine learning algorithms, including neural networks, to accurately learn the relationships of the input alloy compositions. We also explore various techniques to generate candidate alloy composition, which is put through the screening stage to select the most optimum alloy composition.

2 Background

This section gives us a brief insight into the background knowledge related to the task in hand such as the data set used, machine learning models and neural networks.

2.1 Dataset

With the growth in the development of aluminum alloys in the welding, automotive, manufacturing, and space industry and their acknowledgment as an effective alternative to steel for several applications, there are increasing demands for those responsible for the development of aluminum alloys to become more acquainted with this group of materials. As mentioned before, the data for this project work was collected from the paper ‘A Survey of Sensitization in 5xxx Series Aluminum Alloys’ [2].

2.1.1 5xxx Series Aluminum Alloys

By adding elements to pure aluminum, an alloy is formed, creating a chemical composition which has enhanced properties. Once these compositions are developed, a 4-digit number is assigned in which the first digit represents a general series which characterizes its principal alloying elements.

The aluminum alloys, which are regarded as 5xxx series, are non-heat-treatable alloys[10] that provide good strength-to-weight ratio to solid solution strengthening and cold work. The major component in 5xxx series aluminum alloys apart from aluminum is magnesium. Magnesium is present in the alloy in the concentration range of 3%-5.6%[11]. This series of alloys has moderate to excellent mechanical properties in combination with significantly higher ductility in annealed condition, good resistance to corrosion and weldability[12]. Aluminum-magnesium alloys (5xxx series) are used for high strength foil, dump track bodies, petrol tanks, pressure cryogenic vessels, marine structures and fittings, automotive trim and architectural components.

Most passenger vessels use the 5xxx series alloy for both structural elements and hardware, ranging from 1000 tonnes per ship to 2000 tonnes. Since appearance is crucial for many ocean liners, the structures are painted, but aluminum allows a 50 percent longer period of time until refurbishing. For the construction of destroyers, 6000 tonnes of aluminum are used per year. The 5xxx series alloy's weight-saving ability allows the vessel to ensure stability with its slim hull necessary to reach high speeds. The alloy may also be used, outside the mainframe, for a range of instruments and equipment, including doors, windows, gratings, and lockers. The use of aluminum alloys has created a technological revolution for the marine industry. Although aluminum has an initial cost penalty, its ability to build lightweight vessels with lower maintenance costs guarantees that over the course of its life, the vessel would be a valuable investment.

One of the drawbacks of the 5xxx series aluminum alloys is the fact that while most 5xxx alloys are highly resistant to corrosion, a super-saturation of Mg makes these alloys susceptible to intergranular attacks, including intergranular corrosion[2]. The major component in 5xxx series aluminum alloys apart from aluminum is magnesium. Magnesium is present in the alloy in the concentration range of 3%-5.6% [11]. This presence of magnesium along with aluminum leads to the formation of beta phases[13] (Mg2Al3), which can be present in a large amount along the grain boundaries. The presence of beta phases can induce inter-granular corossion (IGC)[2], which inturn leads to the formation of cracks due to a process known as intergranular stress corrosion cracking (IGSCC)[2] in the material.

2.1.2 Degree of Sensitization (DoS)

Sensitization can be defined as the susceptibility of an alloy, to corrosion at grain boundaries[2]. DOS (or Sensitization Degree) test is performed to assess the correlation of sensitization determined by the electrochemical potentiokinetic reactivation test (EPR)[8] or nitric acid mass loss test [NAMLT][7] with the susceptibility to intergranular corrosion. NAMLT is the standard for experimental verifying the DoS value of a 5xxx series alloy and is applicable only to wrought[9] products by providing a quantitative measure of the susceptibility to intergranular corrosion. An alloy which is susceptible to the formation of beta phases (5xxx series alloys) when subjected to concentrated nitric acid, will experience dissolution. By measuring the weight of the test sample before and after being subjected to the acid, we can calculate the DoS value of the particular alloy composition. In summary, the DoS value of an alloy gives us the extent of formation of beta phases.

2.1.3 Sensitization Temperature and Time

The corrosion resistance properties of 5xxx series aluminum alloys are usually observed over a significant time frame(months-years). This time frame is not particularly useful for researchers in terms of their ability to study the corrosion resistance properties such as DoS. In order to reduce the period of corrosion and to accelerate the corrosion properties of such alloys for research purposes, the alloys are subjected to temperatures(Sensitization temperature[14]) higher than what the these alloys will experience in the real world. An alloy is subjected to such high temperatures for a period of time(hours to days), which is termed as sensitization time[14]. The properties measured form an alloy subjected to such conditions mimic the properties of alloys subjected to real world conditions over a large time frame. The acceleration of the corrosion properties allows us to measure the DoS value of a wide range of aluminum alloys with high degree of accuracy.

2.1.4 Temper Notation

Considering the various series of aluminum alloys, we observed that their characteristics and resultant application vary considerably. Aluminum has two very different forms, which are the heat-treated aluminum alloys, and the non-heat treated aluminum alloys. The 1xxx, 3xxx, and 5xxx series wrought aluminum alloys are non-heat treatable[15] and are strain hardenable only. The 2xxx, 6xxx, and 7xxx series wrought aluminum alloys are heat treatable, and the 4xxx series consists of both heat treatable and non-heat treatable alloys. The 2xx.x, 3xx.x, 4xx.x, and 7xx.x series cast alloys are heat treatable. The 5xxx series alloys acquire their optimal mechanical properties through the process of strain Hardening[16]. Strain hardening is a method which increases the strength of certain materials by introducing cold work application. The Temper Designation System[6] describes the conditions of the material, called tempers. The Temper Designation System consists of a sequence of letters and numbers that match alloy designation number. The temper designation for alloys is given below in table 1.

Table 1: Temper designation for alloys

Notation	Meaning
F	As fabricated (No temper)
O	Annealed
H	Strain hardened
W	Solution heat treated
T	Thermally treated

Table 2: Temper designation for first digit of H

Notation	Meaning
H-1	Strain hardened
H-2	Strain hardened and partially annealed
H-3	Strain hardened and stabilized

The temper designation specific to 5xxx series aluminum alloys are H(strain hardened). The H temper designation is followed by two or more digits that further specify the type of temper. The first digit indicates the specific combination of basic operations, as shown in table 2. The second digit in the temper notation of H describes the degree of strain hardening as shown in table 3. The variation in the two digit temper notation of H can be represented by a third digit, for example a temper notation of H111 implies that the alloy underwent some amount of cold strain hardening after annealing but not enough for it to qualify as an H11 or H12 temper.

Table 3: Temper designation for second digit of H

Notation	Meaning
2	¼ Strain hardened
4	½ Strain hardened
6	¾ Strain hardened
8	Fall hard
9	Extra hard

2.1.5 Recrystallization

Recrystallization is the process in which the grains of a deformed alloy structure is replaced by an entirely new set of stress free grain. These new grains nucleate, grow and spread until all the old grains have been replaced. Alloys subjected to recrystallization experience an increase in ductility.

2.1.6 Criteria for classifying an alloy as 5xxx series

The major component in 5xxx series aluminum alloy is magnesium. Apart from this the presence of other elements enhance the material properties of the alloy. The addition of these elements are restricted withing certain range as shown in table 4.

Table 4: Alloy selection criteria(implies strict criteria)*

Element	Range	Element	Range
Al	No Limit*	Cu	0.0% - 0.7%
Mg	3.5% - 6.0% *	Ag	0.0% - 0.2%
Mn	0.1% - 0.6%	Si	0.0% - 0.5%
Fe	0.1% - 0.4%	Ni	0.0% - 0.2%
Cr	0.0% - 0.3%	Ca	0.0% - 0.2%
Ti	0.0% - 0.3%	Ge	0.0% - 0.4%
Sr	0.0% - 0.5%	Nd	0.0% - 0.4%
Zn	0.05% - 0.75%	Ce	0.0% - 0.4%
Zr	0.0% - 0.5%		

2.2 Machine Learning and Neural Networks

Machine-learning algorithms utilize statistics on massive amounts of data to find underlying hidden patterns and is a data analysis technique that automates the construction of analytical models. It is a branch of AI focused on the premise that systems with minimal human interaction can learn to recognize patterns from data, and make decisions. Machine learning today is not like that of the past because of new computing technologies. It was born from pattern recognition and is based on the theory that computers can learn to perform specific tasks without being specifically programmed. They learn to generate accurate, reproducible decisions and results from prior computations. Machine learning was formulated on the basis of its potential in using computers to test the structure of data, even when we do not have a theory as to what that structure is. A validation error on new data is the test for a machine learning model. Since machine learning often uses an iterative approach to learn from data, it is simple to automate the learning. This iterative aspect of machine learning when it comes to training a model is important since these models can adjust independently as they are exposed to new data.

Neural network models on the other hand have similar functionality to that of a machine learning model, but is more versatile and can solve highly complex tasks and is generally regarded as a subset of machine learning algorithms. The structure of a neural network model resembles the neuron structure in the human brain. Neural networks can adapt to varying inputs; therefore, the network generates the best possible results without redesigning the output criteria.

3 Design and Implementation

This section describes the steps taken to achieve the final objective of the project, including the various machine learning techniques that were tested in order to select the best performing model.

3.1: Data Analysis and Preprocessing

Analysis of correlation of composition elements with respect to DoS shows us that aluminum and magnesium are comparatively more correlated with DoS than any other constituent elements as is evident from figure 1. Aluminum is inversely correlated to DoS, which implies that an increase in the amount of aluminum in an alloy will lead to a decrease in DoS value(an increase in amount of aluminum means that the percentage contribution of magnesium is reduced, therefore resulting in lower DoS). Magnesium is positively correlated, which is expected as sensitization is caused due to presence of magnesium in the alloy. This knowledge can help a machine learning model to learn other underlying relationships and therefore get better result.

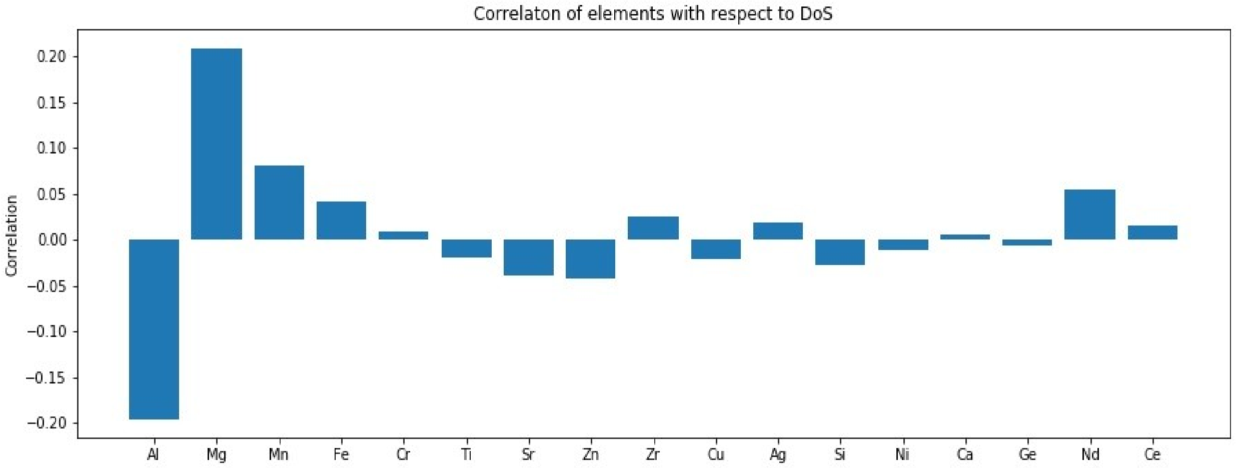


Figure 1: Correlation of elements with respect to DoS

The first step in training any machine learning model is to process or encode the input data in an appropriate manner so that the model can quickly parse the data. Data preprocessing is an essential step as any imperfection resulting in this stage can lead to drastic variation in the performance of models. Errors are usually present or are introduced during the data gathering process. These errors can be due to human error, faulty equipment such as sensors, missing values, registering fake data as genuine, and also due to corrupted files. These errors can mislead a learning algorithm into finding relationships in input data that does not exist or not finding any correlation in the input data at all, which implies that the knowledge discovery process is hampered.

3.1.1: Encoding

The data collected from [2] was unified and written over to an excel file. Each row in the dataset contains the alloy name, sensitization time, sensitization temperature, recrystalization, temper notation, alloy composition and its corresponding DoS value assessed using the NAMLT[7] process. For each of the alloy composition, it was ensured that the sum of composition was equal to 100.0. Alloys that underwent recrystalliztion was encoded as 1 and the rest were encoded as 0. No missing values were found in the dataset compiled. The temper notation for each alloy was encoded as shown in table 5.

Table 5: Temper notation encoding

Temper type	Designation
H (lab)	1
H116	2
H131	3
H321	4
O	5
Stabilised	6
n/l	7

The resulting data set was split into the input features X and the output feature y. The input to all the machine learning models in this project are the following:

Input(X): Sensitization time, sensitization temperature, encoded temper and composition.

Output(y): Degree of Sensitization(DoS).

3.1.2 Train Test Validation Split

In machine learning, we usually split the data set into two, **a)** Training set: The data that is used by the machine learning model for the knowledge discovery process, **b)** Test set: The data that has been reserved on which we test the performance of the model, which ensures that the performance is based on new unseen data. This workflow can often lead to two major problems; we might overfit or underfit our model. Overfitting means that the model learns all the intrinsic relationship of training data such that the model cannot perform on any new data, on the other hand, a model can underfit if it not complex enough to capture any valid relationship. The aim is to build a model that can generalize well with respect to new unseen data. This problem is usually solved by using the concept of k-fold cross validation.

For k-fold cross validation[17], we randomly split the dataset into k sets. K number of models are generated and evaluated for performance. For each set, the set is taken as hold out data/test data, and a machine learning model is trained on the rest of the data. The performance of the resulting model is saved, and the model is discarded. The process is repeated for all k sets, and the performance results are summarized. One of the important features of using k-fold cross validation is that a record/sample is only used once as testing data, and k-1 times as training data. The major benefit comes when the size of data set is small as is in our case, where it is hard to train a model that generalizes well The results shown in this report are based on 10-fold cross validation with a fixed random state set to 42.

3.1.3 Outlier Detection

Outliers are data points in the data set that deviate significantly from the rest of the data. These usually implicate the presence of noise or errors and can significantly affect the performance of a model. Outliers are hard to find as these anomalies are usually multivariate which means that they exist in n-dimensional space. We try to remove these outliers as they force the learning algorithm to find unwanted relationships in the data. Various outlier detection algorithm exists which can effectively detect most of the outliers. For this project, the isolation forest algorithm was used to identify and remove outliers..

3.1.3.1 Isolation forest

Isolation forest[18] is based on the tree data structure. The main idea here is that an outlier is more perceptible to isolation than a normal data point. A typical outlier detection algorithm works by grouping together all the data points that it thinks are normal, and the remaining data points are classified as outlier. With the isolation forest algorithm, all the data points that the algorithm thinks are outliers are grouped together first and the remaining data points are classified as normal data points.

Figure 2 visualizes the intuition behind the algorithm. On the left hand side, we can observe that the outliers can be easily isolated using a single line, and on the right hand side, we can see that more number of lines are required to isolate a data point that isn't an outlier. These outliers are represented by a branching in the tree closer to its root, which means that the average path length from the root to an outlier is shorter than other observed data points. This property allows us to identify outliers in the data set.

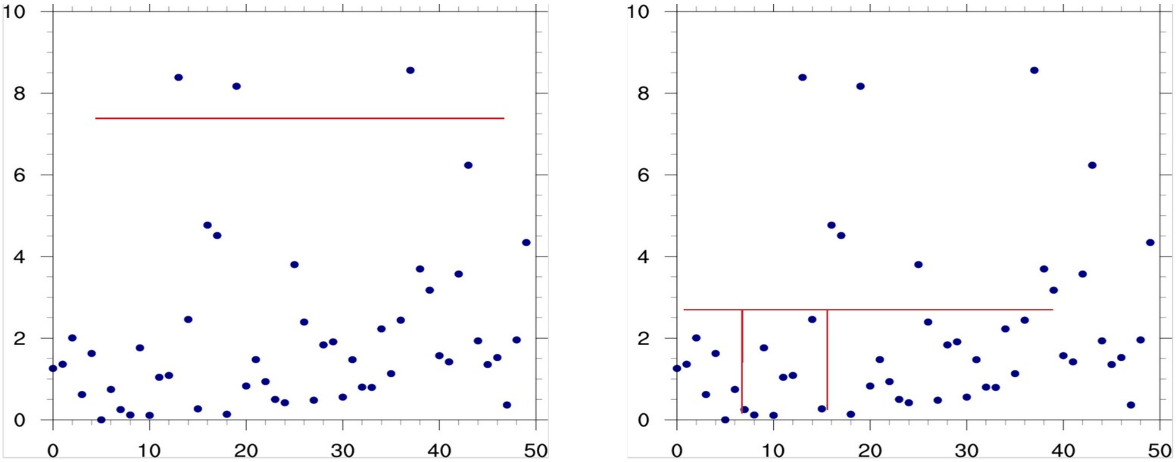


Figure 2: Isolation forest intuition. Figure on left shows that an outlier can be isolated easily, whereas a normal data point is harder to isolate as is illustrated in the right hand side figure.

3.2 Neural Networks

A neural network model was trained in order to assess its performance and viability in predicting accurate DoS value. The process involved constructing a simple feed forward regression model with one hidden layer, which was then fine tuned using genetic algorithm[19]. The network model has 21 input features corresponding to X as mentioned above and one output feature DoS(y) which was normalized using a min-max scaler. The number of hidden layers was fixed to 1, and the sigmoid[4] activation function was used for the hidden layer.

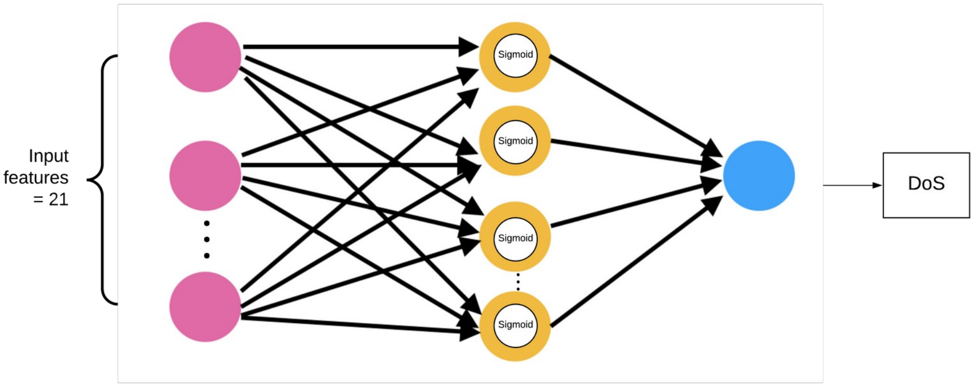


Figure 3: Network architecture

The loss function was set to MSE(Mean Squared Error) loss[20]. This loss function give us the squared error between the predicted and the actual value, and is most commonly used loss function for regression tasks. MSE loss function is very susceptible towards outliers, but this is not a problem as we have already dealt with the outliers using the isolation forest algorithm. The optimizer was set to Adam(Derived form adaptive moment estimation), which is an extension to stochastic gradient descent algorithm. The task of an optimizer is to update the weights of a neural network model based on the input and output training data. The Adam optimizer[5] was selected on the basis of its performance in terms of its accuracy, lower computational overhead required, and its ease of use. *Figure 3* visualizes the network architecture. Other parameters such as the learning rate, epochs and number of hidden layer neurons are selected by the custom genetic algorithm implementation.

3.2.1 Genetic Algorithm

The genetic algorithm[19,3] is based on the idea of natural selection, where the aim is to optimize a constrained or unconstrained problem. Here the genetic algorithm is implemented by defining an initial population of size n. The initial population consists of n number of network models, where each model has randomly assigned hyper-parameter. The fitness of each individual(model) is then calculated for the current generation. The fitness of an individual is defined by the test set accuracy of a specific model. The higher the test set accuracy of a model, the greater is the fitness. Two parents are then selected form the population to perform cross-over and mutation. The cross-over operator takes properties(hyper-parameters) from two parents and then combines them to generate an intermediate on which the mutation operator is performed. The mutation operator performs mutation by randomly flipping bits in the binary representation of the intermediate. The resulting mutated representation is converted back to decimal and is then used as hyper-parameters to train a child network model, which is then passed on to the next generation. The process continues for m number of generations, where members in each generation are expected to evolve so as to find the optimal solution.

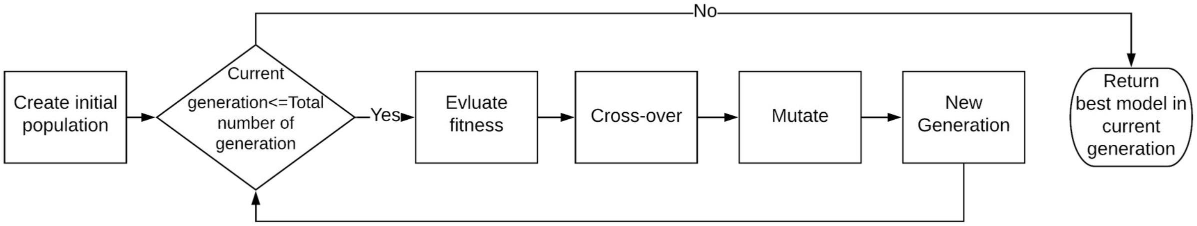

No
Create initial population
Current generation<=Total number of generation
Yes
Evluate fitness
Cross-over
Mutate
New Generation
Return best model in current generation

Figure 4: Genetic algorithm flowchart

For our case, the task of the genetic algorithm was to find the optimal value for the following hyper-parameters; 1) Learning rate: This parameter determines the extent to which the model weights are updated in response to the estimated error, 2) Number of hidden layer neurons, 3) Number of epochs for learning: This parameter determines the number of times the training data is used to update the weights of the network. These parameters are learned over time through the use of a genetic algorithm. For the purpose of this paper, the hyper-parameters of the genetic algorithm, such as the population size, mutation rate, and the number of generations was set to 100, 0.1, and 100, respectively.

The genetic algorithm was manually implemented in python. The main components of the implementation includes the following:

- A method to initialize the initial generation with random hyper-parameter values. The hyper-parameters such as the learning rate, number of hidden layer neurons and epochs for each individual in the first generation was selected randomly. The random selection of these hyper-parameters was restricted within the limit mentioned below.

Learning rate = (0.01 – 0.9)

Number of hidden layer neurons = (10 – 300)

Number of epochs = (10 – 500)

- A method to determine the fitness of all individuals in the current generation. The fitness of an individual is equal to its test set accuracy. The chances of an individual being selected as parents in a specific generation is directly proportional to its fitness score, therefore individuals that perform best has a higher chance of being selected as a parent. The aim of the evolution process is to maximize the fitness score.

- A method to implement the cross-over operator. This allows us to combine the hyper-parameter values of two parents to generate a child. The cross-over operator selects two individuals from the current population as parents. The hyper-parameters of the parents are then represented in their binary form. The binary representation facilitates cross-over. The cross-over is implemented manually by mixing up the binary representation of parameter values of both the parents. The resulting binary number is then converted back to its decimal representation and is then used as hyper-parameters to train the child model.

- A method to implement mutator operator. The mutator operator is used to maintain genetic diversity throughout the evolution process. This operator changes the properties of the hyper-parameters of a child class by a small amount based on a random event. The hyper parameters resulting from the cross-over operator is modified slightly by adding a small value(a random value not greater than 10% of the max limit mentioned above) to it This small change ensures that the child class is not a direct copy of its parent class.

3.3 Supervised Machine Learning Models

In search of the best machine learning models, various algorithms was tested for performance in terms of its mean validation set accuracy. Five different machine learning algorithm which included a) Linear Regressor, b) K-Neighbor Regressor, c) Decision Tree Regressor, d) Support Vector Regressor e) Xgboost were tested and compared with the neural network model. The best performing model was then selected for the final screening process. Each model was tested using the cross validation technique with the random seed set to 6. The mean accuracy of all the validation set accuracy was reported and is the basis of selection for the optimum model. The accuracy was measured in terms of the models R2 score. R2(R-squared) score gives us a measurement of the variation in the predicted and the real output. In other terms, the R2 score is the statistical measurement of how close the predicted data is fitted to the regression line. The value of R2 is in the range -1 to 1. An R2 score of zero implies that the model does not explain any variance with respect to the real output. (Note: a high R2 score does not directly imply better performance. The interpretation of R2 score depends on its use in a specific task. According to Cohen(1992)[1], R2 value of 0.12 or below indicate low, between 0.13 to 0.25 values indicate medium, and any value above 0.26 indicate high effect size. Since the data we have is measured under lab conditions, where many of the variables are under control, we can sufficiently rely on the R2 score as a metric for evaluating models)

3.3.1 Linear Regression

Linear regression is a primary method in predictive analysis and is widely used. The general principle of regression is to look at two things: (1) Does a collection of input variables(independent) have any correlation with the outcome variable (dependent/DoS)(2) Which variables are significant predictors of the outcome variable in particular, and in what way do they indicate the magnitude and the sign of the beta estimate? A linear regression model tries to find the relationship between the input variables by fitting a linear equation to the data. This model is useful only in selected handful of cases and is not the best model for our project. The input data here does not have linear relationship, and therefore we expect this model to perform the worst. The purpose of training this model is to set a baseline R2 score for comparing other models.

3.3.2 K-Neighbor Regression(KNR)

K-Neighbor regression is a non-parametric algorithm based on the idea that a new unseen data point has characteristics similar to k data points that are closest to it in a multi-dimensional feature space. The main advantage here is that there is no training phase, but instead consists of storing the training data as points in higher dimensional feature space. When the model is given an input to predict the output, the algorithm finds k number of neighbors that closest to the given input point in feature space, and the predicted value is usually the mean of these k neighbors. The data set in our case can be termed as sparse(significant presence of zero values), which implies that most of the data points in the training set lies on the axis of the multi-dimensional feature space. The K-neighbor algorithm's performance is therefore hindered due to the presence of these zero values.

The k value was set to 5 with uniform weights for all input features. The distance is calculated as Minkwoski metric.

3.3.3 Decision Tree Regression(DTR)

The decision tree algorithm works by constructing a regression or classification model in a tree structure. This is done by breaking down a dataset into smaller and smaller subsets at the same time incrementally creating a related decision tree. The end result is a tree with nodes for decision and nodes for output as leaves. Decision trees are predictive models that use a set of binary rules to calculate a target value. A decision tree arrives at an estimate by asking the data a series of questions, each question narrowing down our possible values until the model becomes confident enough to make a single forecast. The prediction will be an estimate based upon the train data on which it was trained. The major disadvantage of this algorithm is its inability to predict continuous values, which is required for our purpose.

The criterion for splitting was set to MSE(Mean Squared Error), and the splitter strategy was set to ‘best’.

3.3.4 Support Vector Regression(SVR)

A support vector regression model is based on a similar principle that drives the support vector machine model used for classification. The model tries to find a hyperplane that separates the data classes. This process of finding an optimal hyperplane is done by a kernel. A kernel helps in finding a hyperplane in the higher dimensional space without drastically increasing the computational cost. The computational cost will usually increase if the number of data dimensions increases. This increase in dimension is necessary when we can not find a separate hyperplane in a given dimension, and therefore we are forced to move to a higher dimension. With respect to support vector regression, the hyperplane is used to predict the continuous output. One of the significant advantages or SVR is that the user can specify the level of tolerance in error by setting appropriate hyper-parameters.

The kernel was set to rbf(Radial Basis Function) due to its performance in terms of accuracy. The kernel coefficient gamma was set to ‘auto’, regularization parameter C was fixed at 50 and the epsilon value was fixed at 5. The epsilon value determines the tolerance in error.

3.3.5 Xgboost Regression

Xgboost[21] (Extreme Gradient Boosting) is an ensemble learning model and is a variant of the popular GBM(Gradient Boosting Machine) algorithm. Often times, depending on the characteristics of only one machine learning model may not be enough to produce well generalized accurate output. Ensemble learning provides a structured approach for integrating the multiple learners' predictive ability. The effect is a single model that has the power of multiple models' ability to aggregate performance. The xgboost algorithm combines techniques from models such as decision trees, bagging, random forest, and gradient boosting. The evolution of xgboost can be summarized as follows: Decision trees allows for the graphical representation of all possible solutions to a decision. While decision trees are used as-is for many classification and regression tasks, it is limited by its performance as a single stand-alone model. This drawback is addressed through the idea of bagging. Bagging (Bootstrap Aggregation) is an ensemble meta-algorithm that combines the output of multiple models. In the bagging process, a group of models is trained on the same data. The output is finalized by the majority voting system. This technique facilitates the model to be more generalized, robust, and stable. Bagging also helps avoid overfitting and reduces variance. The introduction of bagging was followed by the development of the random forest ensemble model. Instead of training multiple models on the same data, the random forest model trains a specific model with randomly sampled data. This also implies that some of the training data will be used by multiple models. The idea is that by training each tree on different samples, even if each tree has high variance with respect to a specific collection of training data, the entire forest would generally have lower variance overall, but not at the cost of raising the bias. The output predictions are made by averaging the predictions of each decision tree. This algorithm was followed by the introduction of boosting. Boosting is similar to bagging in the sense that multiple models are trained for a single task. The difference here is that instead of training all the models independently, irrespective of the performance of each other, here the training is done sequentially. A model is first trained, and a new generation of models is trained based on the performance of its predecessor such that the new model overcomes the weakness of the previous model. The performance of a boosting algorithm can be improved by using the gradient boosting algorithm. This technique is similar to boosting except for the fact that the gradient boosting algorithm uses a loss function that describes the criteria to be minimized. The model relies on the intuition that, when the current model is combined with all the previous models, the next best possible model minimizes the overall error in prediction. The main idea is to set the target results for this next model so as to minimize the error. The xgboost algorithm is a variant of such a gradient boosting algorithm with system optimization and algorithmic enhancements. This optimization includes parallelization, tree pruning, and improved memory allocation and usage.

The xgboost model was trained similar to all other models mentioned above, using 10 fold cross validation. The parameters are set as follows: max_depth = 6 (set to low value to prevent overfitting), n_estimator = 1000(Maximum number of decision trees), colsample_bytree = 0.8 (Subsample ratio of columns during tree construction).

3.4 Data Generators and Screening

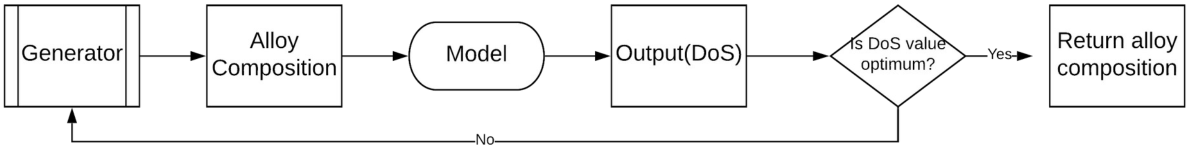
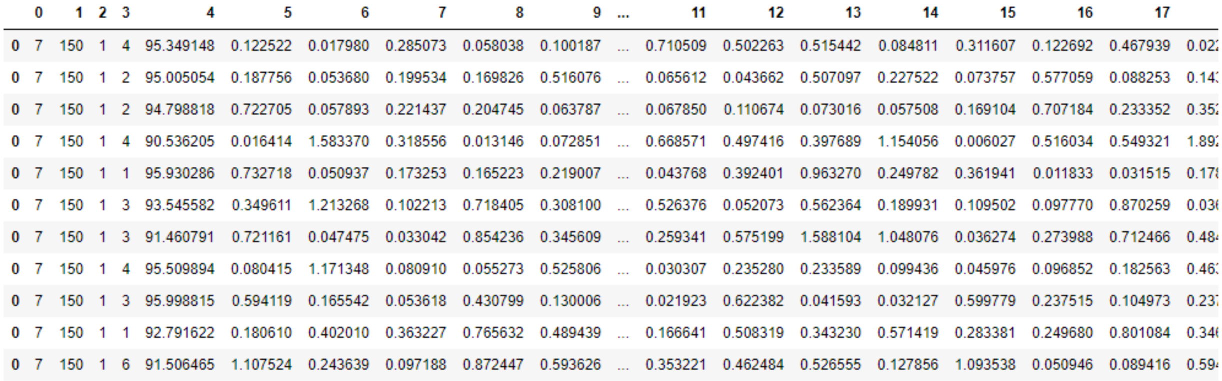


Figure 5: Flowchart for screening candidate alloy compositions

Data generators are required to generate candidate test data/ compositions that can then be passed over to the trained machine learning model for the screening stage. The resulting output from the machine learning model(DoS value) will be checked for validity. The aim is to select the input candidate with the lowest output value(DoS) screening through an average of 10^5 randomly generated test compositions. These random composition are restricted based on the range mentioned in *table 4*. The best composition is then marked as a viable output.

The two technique used to generate the random candidate compositions are mentioned below:

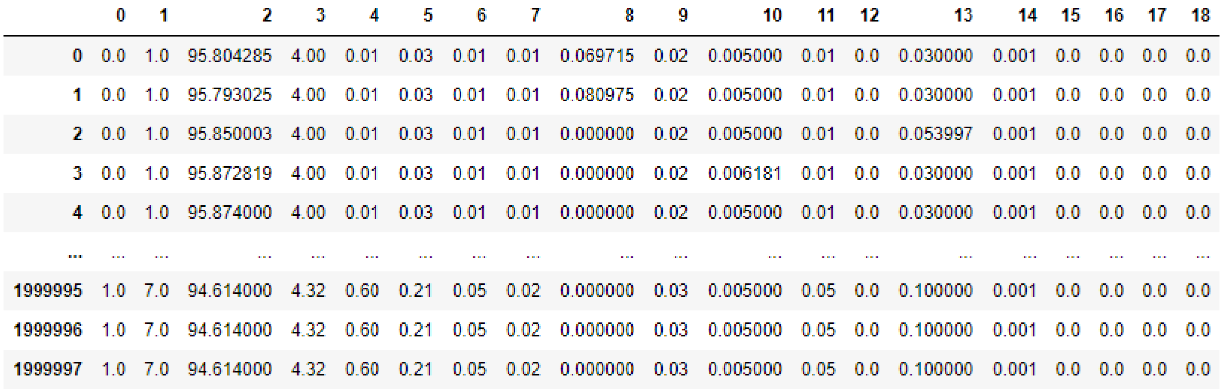
a) Stochastic Generator: This generator randomly selects values in the range mentioned in *table 4* and sends it as a list to the machine learning model. A random continuous value was selected for all the constituent elements in the alloy, and the categorical values(Temper notation) were selected randomly form a list.



	0	1	2	3	4	5	6	7	8	9	...	11	12	13	14	15	16	17	
0	7	150	1	4	95.349148	0.122522	0.017980	0.285073	0.058038	0.100187	...	0.710509	0.502263	0.515442	0.084811	0.311607	0.122692	0.467939	0.02
0	7	150	1	2	95.005054	0.187756	0.053680	0.199534	0.169826	0.516076	...	0.065612	0.043662	0.507097	0.227522	0.073757	0.577059	0.088253	0.14
0	7	150	1	2	94.798818	0.722705	0.057893	0.221437	0.204745	0.063787	...	0.067850	0.110674	0.073016	0.057508	0.169104	0.707184	0.233352	0.35
0	7	150	1	4	90.536205	0.016414	1.583370	0.318556	0.013146	0.072851	...	0.668571	0.497416	0.397689	1.154056	0.006027	0.516034	0.549321	1.89
0	7	150	1	1	95.930286	0.732718	0.050937	0.173253	0.165223	0.219007	...	0.043768	0.392401	0.963270	0.249782	0.361941	0.011833	0.031515	0.17
0	7	150	1	3	93.545582	0.349611	1.213268	0.102213	0.718405	0.308100	...	0.526376	0.052073	0.562364	0.189931	0.109502	0.097770	0.870259	0.03
0	7	150	1	3	91.460791	0.721161	0.047475	0.033042	0.854236	0.345609	...	0.259341	0.575199	1.588104	1.048076	0.036274	0.273988	0.712466	0.48
0	7	150	1	4	95.509894	0.080415	1.171348	0.080910	0.055273	0.525806	...	0.030307	0.235280	0.233589	0.099436	0.045976	0.096852	0.182563	0.46
0	7	150	1	3	95.998815	0.594119	0.165542	0.053618	0.430799	0.130006	...	0.021923	0.622382	0.041593	0.032127	0.599779	0.237515	0.104973	0.23
0	7	150	1	1	92.791622	0.180610	0.402010	0.363227	0.765632	0.489439	...	0.166641	0.508319	0.343230	0.571419	0.283381	0.249680	0.801084	0.34
0	7	150	1	6	91.506465	1.107524	0.243639	0.097188	0.872447	0.593626	...	0.353221	0.462484	0.526555	0.127856	1.093538	0.050946	0.089416	0.59

Figure 6: Snapshot of data generated using Stochastic generator

b) SMOTE(Synthetic Minority Oversampling Technique) Generator: This algorithm is generally used for addressing class imbalance problems. The SMOTE[22] generator synthesizes new minority instances for the already existing minority classes in the data set. This algorithm samples points from the high dimensional feature space according to the neighborhood of each example of the minority class. Applying the SMOTE generator on the data set directly will yield a new data set that is very much similar to the original data set, but this is not what we want. The aim is to generate data that has high variance and is only slightly similar to the original data and also conforms to the criteria mentioned in *table 4.* In order to solve this problem, we only sample a small portion of the original data set as a reference for the SMOTE algorithm, this ensures that the generated data set is more random and does not match the original data set.



	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	0.0	1.0	95.804285	4.00	0.01	0.03	0.01	0.01	0.069715	0.02	0.005000	0.01	0.0	0.030000	0.001	0.0	0.0	0.0	0.0
1	0.0	1.0	95.793025	4.00	0.01	0.03	0.01	0.01	0.080975	0.02	0.005000	0.01	0.0	0.030000	0.001	0.0	0.0	0.0	0.0
2	0.0	1.0	95.850003	4.00	0.01	0.03	0.01	0.01	0.000000	0.02	0.005000	0.01	0.0	0.053997	0.001	0.0	0.0	0.0	0.0
3	0.0	1.0	95.872819	4.00	0.01	0.03	0.01	0.01	0.000000	0.02	0.006181	0.01	0.0	0.030000	0.001	0.0	0.0	0.0	0.0
4	0.0	1.0	95.874000	4.00	0.01	0.03	0.01	0.01	0.000000	0.02	0.005000	0.01	0.0	0.030000	0.001	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1999995	1.0	7.0	94.614000	4.32	0.60	0.21	0.05	0.02	0.000000	0.03	0.005000	0.05	0.0	0.100000	0.001	0.0	0.0	0.0	0.0
1999996	1.0	7.0	94.614000	4.32	0.60	0.21	0.05	0.02	0.000000	0.03	0.005000	0.05	0.0	0.100000	0.001	0.0	0.0	0.0	0.0
1999997	1.0	7.0	94.614000	4.32	0.60	0.21	0.05	0.02	0.000000	0.03	0.005000	0.05	0.0	0.100000	0.001	0.0	0.0	0.0	0.0

Figure 7: Snapshot of data generated using SMOTE algorithm(Note that the sensitization time and temperature is not displayed here as it was fixed to 7 and 150 respectively)

Results and Discussion

This section describes the results of all the models tested and evaluated. The best performing model was then used for the screening stage and the final output composition is obtained.

4.1 Neural Network Evaluation

A neural network model was trained using genetic algorithm to fine tune three basic hyper-parameters(Learning rate, Number of hidden neurons, and Number of epochs). The best performing model in the last generation was then evaluated for performance in terms of R2 score. The mutation rate for the genetic algorithm was set to 0.2 and the number of generations was set to 100 with each generation having a population size of 100. The hyper-parameters and the R2 score of the best model after every 10 generation was then recorded and visualized in figures 8, 9, 10, and 11(Note that each increment in generation on the figure represents 10 generations).

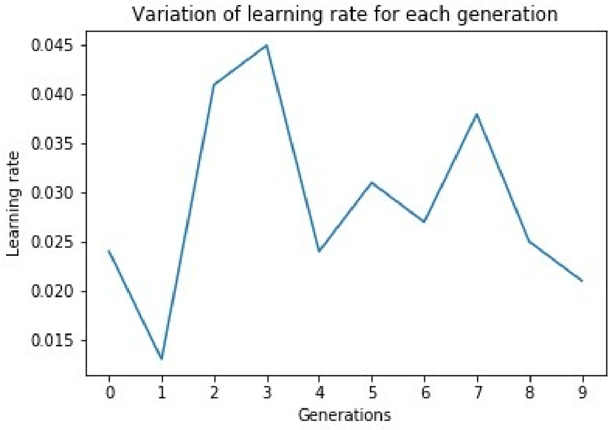


Figure 8: Evolution of Learning Rate

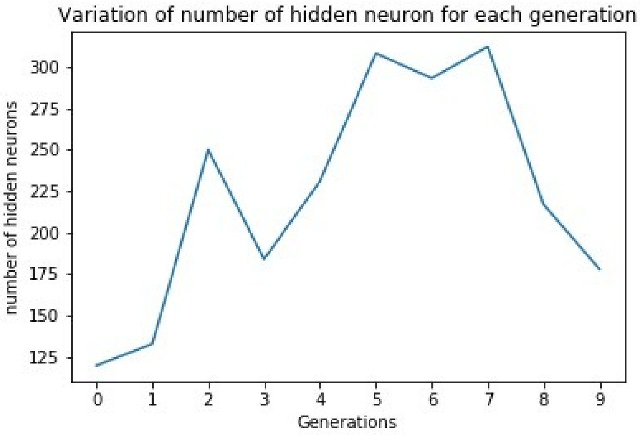


Figure 9: Evolution of number of hidden neurons

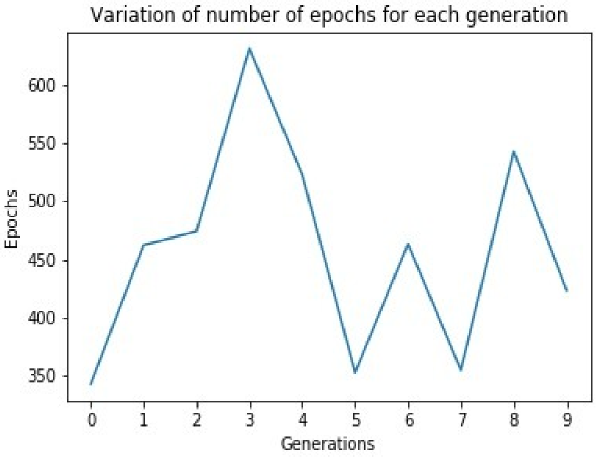


Figure 10: Evolution of number of epochs

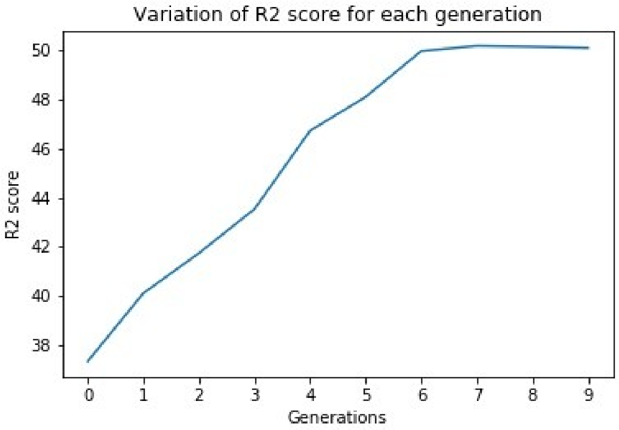


Figure 11: R2 score(scaled b/w 0 and 100) of best model after every 10 generations

The genetic algorithm successfully modifies the hyper-parameter such that the resulting best model in the final generation has much higher R2 score when compared to the best model in the first generation. Upon testing for improvement in performance by increasing the number of generations, varying the mutation rate, and size of the population, it was seen that the accuracy of the best performing model did not go beyond 0.52. The hyper-parameters for the best performing model after 100 generations are as follows:

- Learning rate = 0.02315
- Number of hidden layer neurons = 186
- Epoch = 439

The genetic algorithm was run four times with the same settings(generations = 100, population size = 100, mutation rate = 0.1) and the average results recorded is shown in table 7.

Table 6: Results for separate consecutive runs of the Genetic algorithm

S.no	Learning Rate	Number of Hidden neurons	Epochs	R2 score
1	0.02467	211	572	0.5178
2	0.02189	193	682	0.5031
3	0.02384	118	450	0.5206
4	0.02264	256	427	0.5189

Table 7: Average hyper-parameter settings and R2 score for the Neural Network model

Average Learning Rate	Average number of hidden neurons	Average Epochs	Average R2 score
0.02362	193	532	0.52

4.2 Evaluation of Supervised Machine Learning models

Each of the machine learning models(Linear, KNR, DTR, SVR and Xgboost) was run with the hyper-parameters set as mentioned in the method section and evaluated on the basis of its R2 score. Figure 12 shows the comparison of R2 score of all the models.

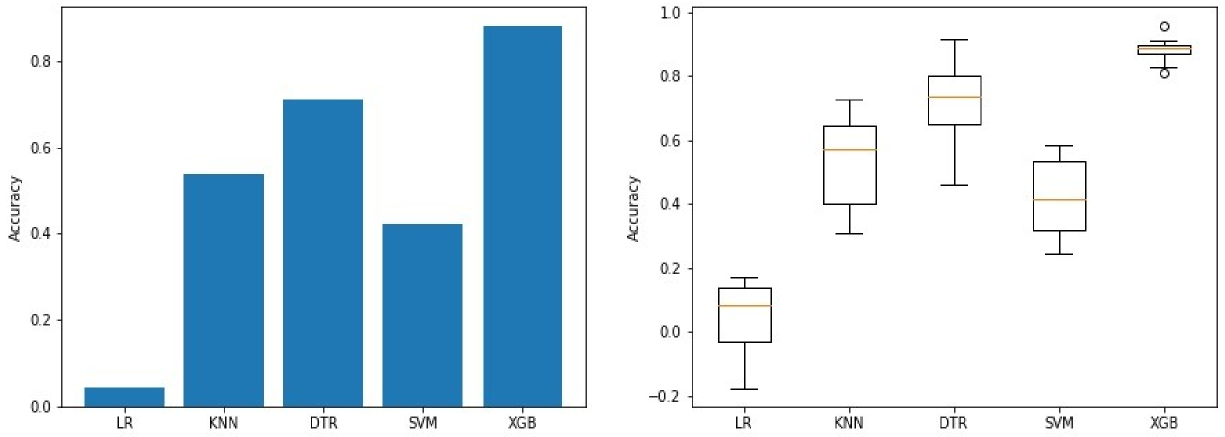


Figure 12: Figure on the left shows the average 10-fold CV R2 score of all models, Figure on the right shows the box plot of all models with respect to the cross validation R2 scores

The xgboost model outperforms all other models by a large margin. It has the least variance and the highest R2 score. The linear regression model performed the worst with an average R2 score of 0.04. The decision tree regressor has the second best performance, which was unexpected as the output of this model is not continuous. Based on these results, the xgboost model was selected for the final screening stage.

The xgboost model was tested for change in performance before and after the removal of outliers. It was observed that the R2 score increased by an average of 2% after removing outliers. The average R2 score and mean squared error for the xgboost model was 0.889 and 21.13 respectively.

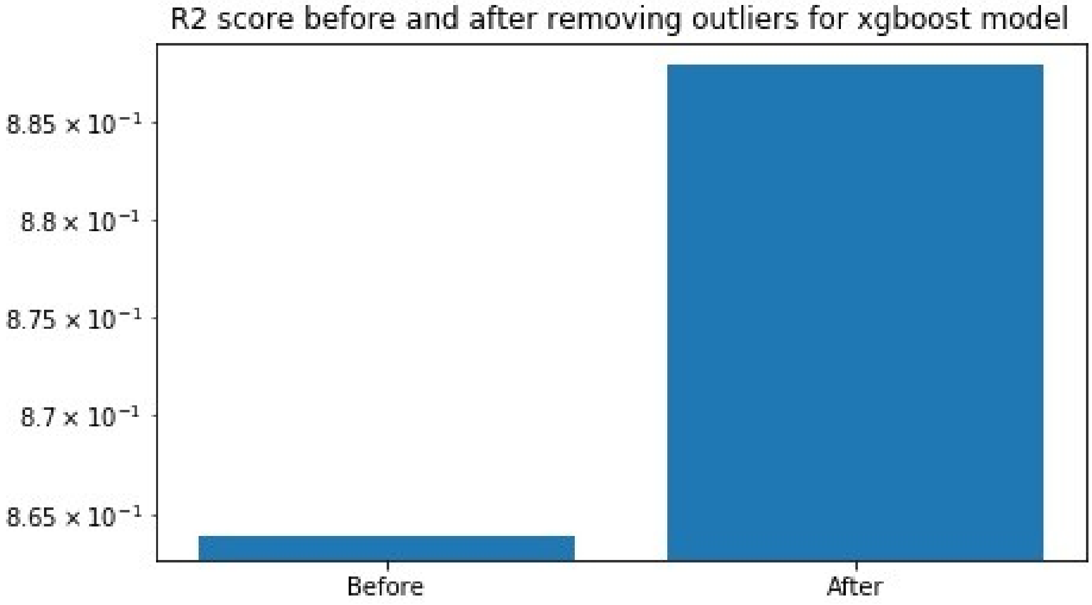


Figure 13: Comparison of R2 score before and after removing outliers

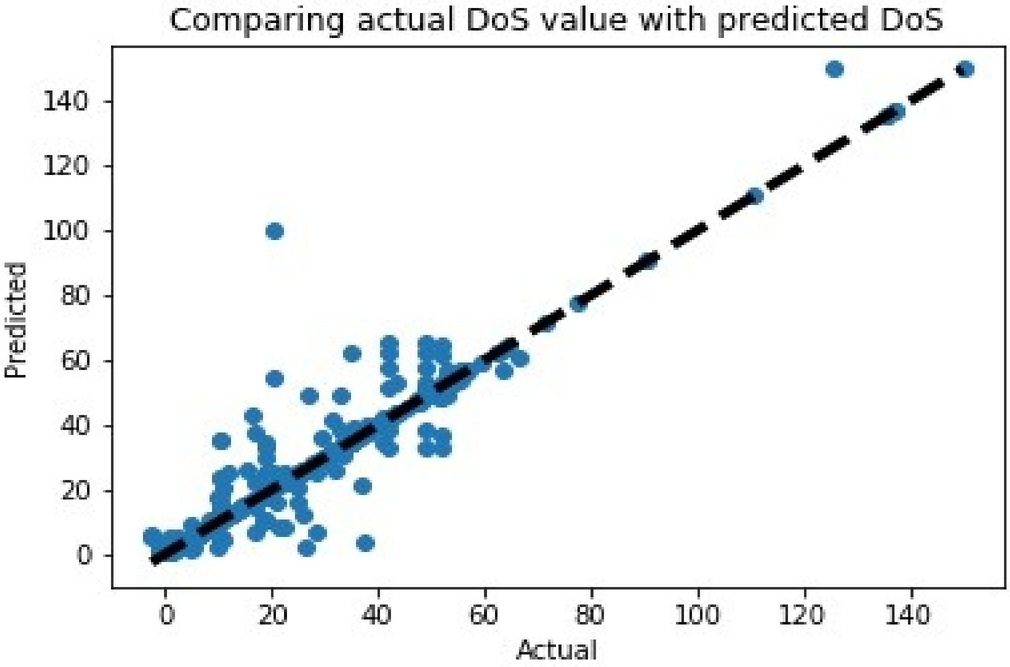


Figure 14: Comparing the prediction of xgboost with the actual DoS value

4.3 Analysis of Screened Alloy Composition

The restrictions on percentage contribution of each alloy forces us to screen a large number of candidate alloy compositions as this restriction also constricts the search space. Screening an average of 10^5 randomly generated inputs with no restrictions on composition limits, and selecting the input with the least DoS, the results shown in table 8 were obtained.

Table 8: Composition with least DoS value obtained without any restrictions on compositions using stochastic generator.

Input	Value	DoS	Remarks
Sensitization time	7	0.0018	The DoS value for this particular set of input is almost close to zero, which is what we need, but the composition does not meet the requirements mentioned in *table 4*. The percentage contribution of magnesium is only 0.22, but it should range between 3.5% and 6.0%. The low value of magnesium is the reason why the DoS value is lower(Note that percentage contribution of magnesium is positively correlated to Dos [*figure 1*]).
Sensitization temperature	150		
Recrystallized	1		
Temper	5		
Al	95.67		
Mg	0.22		
Mn	0.11		
Fe	0.33		
Cr	0.05		
Ti	0.17		
Sr	1.22		
Zn	0.12		
Zr	0.21		
Cu	0.26		
Ag	0.06		
Si	0.35		
Ni	0.02		
Ca	0.06		
Ge	0.29		
Nd	0.6		
Ce	0.25		

Table 9: Composition with least DoS value obtained with restrictions on compositions using SMOTE generator

Input	Value	DoS	Remarks
Sensitization time	7	1.045	The DoS value for this particular set of input is higher than that shown in *table 8*. This is what is expected as the percentage contribution of magnesium in this case is also higher than the previous output. The inputs also conforms to the criteria and the DoS value is also low considering the fact that the average DoS value of the original dataset is 26.9. This composition can therefore be considered as valid.
Sensitization temperature	150		
Recrystallized	0		
Temper	1		
Al	94.74		
Mg	4.01		
Mn	0.49		
Fe	0.36		
Cr	0.01		
Ti	0.02		
Sr	0		
Zn	0.03		
Zr	0.01		
Cu	0.1		
Ag	0		
Si	0.22		
Ni	0.01		
Ca	0		
Ge	0		
Nd	0		
Ce	0		

Using the stochastic generator without any restrictions on criteria leads to generation of composition that has low mg value. The low value of magnesium intrinsically conveys that the resulting DoS value is also low. Using the same trained xgboost model used for the stochastic generator, a new set of random input generated using the SMOTE generator was fed to the model. The SMOTE generator was only given a portion of the original data set[100 random records form original data set] for reference. This process increases the variance of the resulting data set. The SMOTE algorithm also alleviates the need for checking if the results match the criteria, as these properties are learned from the reference input passed to the SMOTE function. Similar to the previously mentioned process, each record was screened to obtain the input that has the least DoS value. The resulting input is shown in *table 9*.

4.4 Discussion

Based on the performance of the neural network model, it seems that the architecture was not able to capture the full relationship of the input variables, leading to a lower performance metric. This may be due to insufficient input data. The size of available data is small, as we are mainly focused on aluminum alloys. This constraint means that obtaining more data that has been experimentally verified is difficult. On the other hand, models such as DTR and xgboost seems to perform better with respect to the neural network model. One of the reasons behind this is that the data set used here has intrinsic mathematical foundations with respect to the alloy crystal structure, bonds, and material properties. These properties help a machine learning model to identify underlying hidden patterns of interest, thereby allowing them to perform better.

We were able to successfully generate inputs/composition which ticks all checkboxes when it comes to input criteria and DoS value. By repeating the process of screening random data, more ideal compositions can be obtained. The generation of these ideal compositions is the first stage in the development of an alloy that can be put to use in the real world. The next stage would be to experimentally verify the DoS value of the ideal composition, and thereby allowing us to check the validity of the model trained. Experimentally verified compositions can then be used as input for training or updating a model, further improving its performance.

Future Work

The results from this project work gave us insight into the use of machine learning in the field of material science. The results showed a machine learning model can learn to identify underlying patterns and gain knowledge and perform with high accuracy. The output of these models if verified experimentally can allow us to reduce the development time of various alloys with different material properties. The project works focuses only on aluminum alloys, but can be extended for use with other alloys with different criteria and material properties, for example: deriving compositions for magnesium alloys with better tensile strength and reduced weight for use in automotive industry, development of specialized alloys which requires extremely specific material properties for use in the tech industry. More over new and advanced data generation techniques can be researched upon which takes into consideration the stoichiometric relationship of the constituent elements in the alloy, and other such properties.

References

1. Cohen, J. (1992). Statistical Power Analysis. Current Directions in Psychological Science, 1(3), 98–101. https://doi.org/10.1111/1467-8721.ep10768783
2. Birbilis, Nick & Zhang, Ruifeng & Knight, Silent & Holtz, Ronald & Goswami, Ramasis & Davies, Chris. (2015). A Survey of Sensitization in 5xxx Series Aluminum Alloys. Corrosion. 72. 150903122215004. 10.5006/1787.
3. Goldberg, David E. "Genetic and evolutionary algorithms come of age." Communications of the ACM, vol. 37, no. 3, Mar. 1994, p. 113+. Accessed 30 May 2020
4. Karlik, Bekir, and A. Vehbi Olgac. "Performance analysis of various activation functions in generalized MLP architectures of neural networks." International Journal of Artificial Intelligence and Expert Systems 1.4 (2011): 111-122.
5. Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In ICLR, 2015
6. Cayless RBC. Alloy and temper designation systems for aluminum and aluminum alloys. In: Metals handbook. Properties and selection: nonferrous alloys and special-purpose materials, vol. 2. Ohio: ASM International; 1990. p. 15–28.
7. American Society for Testing and Materials, 2004. *Standard Test Method for Determining the Susceptibility to Intergranular Corrosion of 5XXX Series Aluminum Alloys by Mass Loss After Exposure to Nitric Acid (NAMLT Test)*. ASTM International.
8. Was, G.S. and Rajan, V.B., 1987. On the relationship between the epr test, sensitization, and igscc susceptibility. *Corrosion*, *43*(9), pp.576-579.
9. Substech.com. 2020. *Wrought Aluminum-Magnesium Alloys (5Xxx) [Substech]*. [online] Available at: <https://www.substech.com/dokuwiki/doku.php?id=wrought_aluminum-magnesium_alloys_5xxx> [Accessed 8 June 2020].
10. Sanders Jr, R.E., Hollinshead, P.A. and Simielli, E.A., 2004. Industrial development of non-heat treatable aluminum alloys. In *Materials Forum* (Vol. 28, pp. 53-64).
11. Cáceres, C.H. and Rovera, D.M., 2001. Solid solution strengthening in concentrated Mg–Al alloys. *Journal of Light Metals*, *1*(3), pp.151-156.
12. Bray, J.W., 2013. Aluminum mill and engineered wrought products.
13. Zhao, M.C., Liu, M., Song, G. and Atrens, A., 2008. Influence of the β-phase morphology on the corrosion of the Mg alloy AZ91. *Corrosion Science*, *50*(7), pp.1939-1953.
14. Oguocha, I.N.A., Adigun, O.J. and Yannacopoulos, S., 2008. Effect of sensitization heat treatment on properties of Al–Mg alloy AA5083-H116. *Journal of Materials Science*, *43*(12), pp.4208-4214.
15. Polmear, I.J. and Couper, M.J., 1988. Design and development of an experimental wrought aluminum alloy for use at elevated temperatures. *Metallurgical transactions A*, *19*(4), pp.1027-1035.
16. Bird, J.E. and Duncan, J.L., 1981. Strain hardening at high strain in aluminum alloys and its effect on strain localization. *Metallurgical Transactions A*, *12*(2), pp.235-241.
17. Shao, J., 1993. Linear model selection by cross-validation. *Journal of the American statistical Association*, *88*(422), pp.486-494.

18. Liu, F.T., Ting, K.M. and Zhou, Z.H., 2008, December. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 413-422). IEEE.

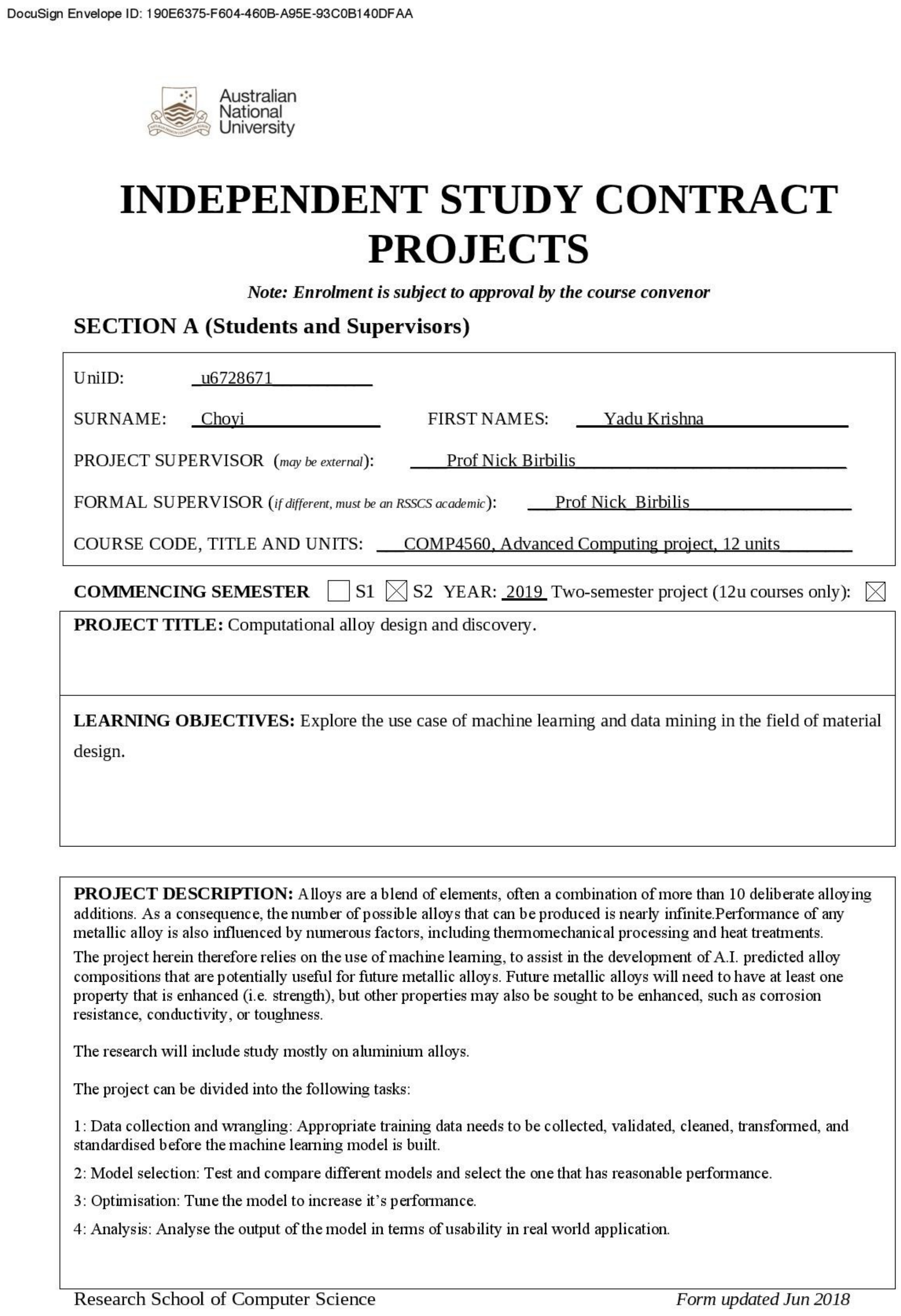
19. Whitley, D., 1994. A genetic algorithm tutorial. *Statistics and computing*, *4*(2), pp.65-85.

20. Christoffersen, P. and Jacobs, K., 2004. The importance of the loss function in option valuation. *Journal of Financial Economics*, *72*(2), pp.291-318.

21. Chen, T. and Guestrin, C., 2016, August. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

22. Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, pp.321-357.

Appendix



DocuSign Envelope ID: 190E6375-F604-460B-A95E-93C0B140DFAA

Australian National University

INDEPENDENT STUDY CONTRACT PROJECTS

Note: Enrolment is subject to approval by the course convenor

SECTION A (Students and Supervisors)

UniID: _u6728671___________

SURNAME: _Choyi________________ FIRST NAMES: ___Yadu Krishna________________

PROJECT SUPERVISOR (*may be external*): ____Prof Nick Birbilis____________________________

FORMAL SUPERVISOR (*if different, must be an RSSCS academic*): ___Prof Nick Birbilis__________________

COURSE CODE, TITLE AND UNITS: ___COMP4560, Advanced Computing project, 12 units________

COMMENCING SEMESTER □ S1 ☒ S2 YEAR: _2019_ Two-semester project (12u courses only): ☒

PROJECT TITLE: Computational alloy design and discovery.

LEARNING OBJECTIVES: Explore the use case of machine learning and data mining in the field of material design.

PROJECT DESCRIPTION: Alloys are a blend of elements, often a combination of more than 10 deliberate alloying additions. As a consequence, the number of possible alloys that can be produced is nearly infinite.Performance of any metallic alloy is also influenced by numerous factors, including thermomechanical processing and heat treatments.

The project herein therefore relies on the use of machine learning, to assist in the development of A.I. predicted alloy compositions that are potentially useful for future metallic alloys. Future metallic alloys will need to have at least one property that is enhanced (i.e. strength), but other properties may also be sought to be enhanced, such as corrosion resistance, conductivity, or toughness.

The research will include study mostly on aluminium alloys.

The project can be divided into the following tasks:

1: Data collection and wrangling: Appropriate training data needs to be collected, validated, cleaned, transformed, and standardised before the machine learning model is built.

2: Model selection: Test and compare different models and select the one that has reasonable performance.

3: Optimisation: Tune the model to increase it's performance.

4: Analysis: Analyse the output of the model in terms of usability in real world application.

Research School of Computer Science *Form updated Jun 2018*

Australian
National
University

ASSESSMENT (as per the project course's rules web page, with any differences noted below).

Assessed project components:	**% of mark**	**Due date**	**Evaluated by:**
Report: style: Research report (e.g. research report, software description...,)	(min 45, def 60)		(examiner)
Artefact: kind: ______________________ (e.g. software, user interface, robot...,)	(max 45, def 30)		(supervisor)
Presentation :	(10)		(course convenor)

MEETING DATES (IF KNOWN):

STUDENT DECLARATION: I agree to fulfil the above defined contract:

DocuSigned by:

9BA1EF77E1954AC

…………………………………………………………………….. 7/27/2019 ………………………..

Signature Date

SECTION B (Supervisor):

I am willing to supervise and support this project. I have checked the student's academic record and believe this student can complete the project. I nominate the following examiner, and have obtained their consent to review the report (via signature below or attached email)

…………………………………………………………………….. 28 July 2019 ………………………..

Signature Date

Examiner:

Name: ……Dr Liang Zheng…………………… Signature ……………………

(Nominated examiners may be subject to change on request by the supervisor or course convenor)

REQUIRED DEPARTMENT RESOURCES:

SECTION C (Course convenor approval)

…………………………………………………….. ………………………..

Signature Date