Report on the Phase Prediction of High Entropy Alloys using Multi-classifiers

Hao Luu,[*] Eric Oberholtz,[†] Matthew Segovia,[‡] and Aidan Rosen[§]
University of California San Diego

(Group 3)
(Dated: June 15, 2024)

In our project, we attempt to replicate the results of a paper, Phase prediction and experimental realisation of a new high entropy alloy using machine learning, using machine learning models to accurately predict the phases of High Entropy Alloys.[1]

I. INTRODUCTION

High Entropy Alloys (HEA) are a class of metals with optimal strength, structural stability, and resistance to corrosion and radiation in extreme environments, making them ideal candidates for aerospace and nuclear applications[2]. In particular, for nuclear fusion applications, HEAs can be utilized as armor tiles on the interior face of a nuclear reactor to serve as radiation shielding Fig. 1. HEAs are 5-component alloys, with atomic loadings ranging between 5 and 35 percent the atomic percentage, forming solid state solutions.

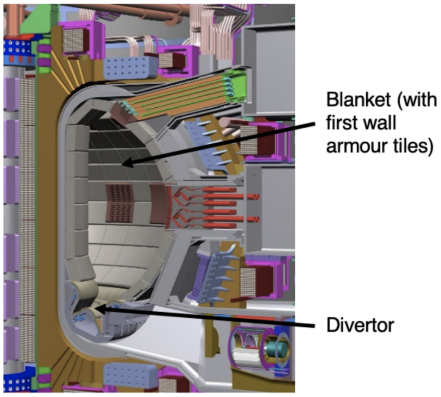


FIG. 1. Cross-section of a nuclear fusion reactor[3].

There are 10^8 HEAs producible out of 64 elements, and the experimental process for fabricating, then testing characteristics is extremely time consuming. Machine Learning models can be used to hasten that process and discover a general trend to identify alloy properties, such as through phase prediciton.

We intend to replicate a study by Singh et. al on predicting the solid phase of a novel HEA using various Machine Learning algorithms[**?**].

II. DATASET

The dataset provided by the paper was compiled from multiple resources including from: Materials for Design Open Repository. High Entropy Alloys. The total dataset has 1200 entries comprised of 30 different elements, 5 properties used to predict HEAs, and the crystal structure of each entry (our labels); each entry is itself an HEA with a chemical composition of mostly 5 elements.

There are 4 labels for the crystal structure: FCC, BCC, IM, FCC+BCC; it is worth noting that this is an imbalanced dataset. For example, MIP is almost 3 times as common compared to FCC+BCC.

The 5 properties are Mixing Enthalpy ($d\mathrm{H}_{\mathrm{mix}}$), Mixing Entropy ($d\mathrm{S}_{\mathrm{mix}}$), Valence Electron Concentration (VEC), Electronegativity Difference ($\Delta\chi$), and Atomic Size Difference (δ). These 5 physical properties are commonly used to define HEAs so we use them as our features.

The labels FCC, BCC, IM, FCC+BCC; were replaced with 0, 1, 2, 3 respectively for processing the dataframe. For preprocessing we use an imputer to replace the NaN values. Replacing the values with 0 would result in a very poor distribution of data points even if it is more physically realistic. Instead we try other methods. We used a KNN (eucliean metric) iterative imputer to replace the NaNs with the mean values from the closest neighbors. We also experimented with median and mean values across all the data points.

We also experimented with capping high or low values to minimize the effects of outlier cases. One such technique was Z-Score capping where we calculated the standard deviation and set a hard limit of 3 std from it. Datapoints that went beyond it were effectively set to this max.

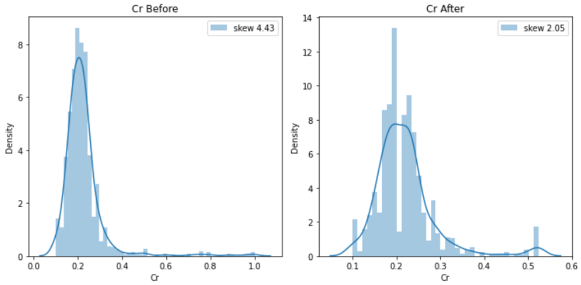


FIG. 2. Z-score capping effect on Chromium

[*] htluu@ucsd.edu
[†] eoberhol@ucsd.edu
[‡] masegovia@ucsd.edu
[§] acrosen@ucsd.edu

A common step in data preprocessing which we also took was applying a logarithmic transformation on the physical properties, save the for the enthalpy. This results in a nicer distribution and also saves on computational power by making the values more comparable especially since the data can span different orders of magnitude. The enthalpy values were already very close in value and magnitude so it was not worth applying a transformation on it.

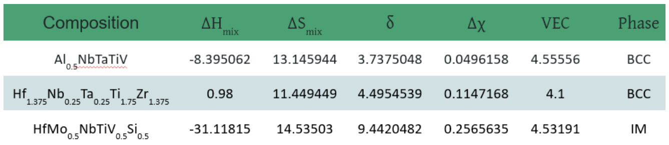
We also apply a robust scaler to fit the data to a more normal distribution. The dataset was then separated into a 20% training and 80% testing set.

III. PARAMETERS WE CONSIDERED

Each HEA is described by a chemical composition tied to the following features:

For a given composition, the phase is determined by the VEC value, per the table below.

1. BCC: VEC < 6.87
2. FCC: VEC > 8.0
3. BCC + FCC: $6.87 \leq$ VEC ≤ 8.0



Composition	ΔH_{mix}	ΔS_{mix}	δ	Δχ	VEC	Phase
$Al_{0.5}NbTaTiV$	-8.395062	13.145944	3.7375048	0.0496158	4.55556	BCC
$Hf_{1.375}Nb_{0.25}Ta_{0.25}Ti_{1.75}Zr_{1.375}$	0.98	11.449449	4.4954539	0.1147168	4.1	BCC
$HfMo_{0.5}NbTiV_{0.5}Si_{0.5}$	-31.11815	14.53503	9.4420482	0.2565635	4.53191	IM

IV. METHODS

We attempt to optimize the algorithms used in the original paper by testing the hyper-parameters of each algorithm and comparing it to the simple baseline model.

The 5 models base models were K-nearest neighbors (KNN), support vector machine (SVM), decision tree classifier (DTC), random forest classifier (RFC), and, XGBoost (XGB).

The premise of using KNN is that metallic alloys with the same phase should naturally be grouped together by the properties which characterizes HEAs. These properties make up the dimension of this space that the data points reside in. For KNN, the tested parameters were: number of neighbors from 1 to 25, uniform vs distance weighting, and the metric to determine proximity. For metrics, we tested Euclidean distancing, Manhattan (which sums the difference between each dimension), Cebyshev (which takes the max of the diffence among all dimensions), and Cosine (comparing the angular distance using dot products).

Support Vector Classifiers is a variation of Support Vector Machine offered by Sklearn for classification purposes. It uses a one-vs-one scheme for n choose 2 classes where n = 4 because we have labels. Each class then trains to label only those 2 labels. The final step is to take the most votes for a label among the classes as its prediction. For SVC, the tested parameters were: the regularization parameter C, the kernel (linear, polynomial, RBF, sigmoid), degree 2 to 5 for only the polynomial kernel, and gamma for polynomial, RBF, sigmoid kernels. Both C and gamma are constant values to be tested for influence range.

For DTC, the tested parameters were criterion (Gini vs Entropy), splitter (best vs random), tree depth from 2 to 32, minimum sample split (deciding when to stop splitting nodes), and minimum sample leaf (deciding when to stop splitting leaves).

For XGB and RFC, the tested parameters were the same as for DTC but included a parameter that determines how many trees are in the forest from 100 to 1000.

We use accuracy and create a confusion matrix to show the overall results of multiclass labeling to evaluate the performance of each model. Then, for each label we also plot the ROC curve for each individual labels to evaluate the performance on each phase and determine whether the imbalance dataset was significant in the algorithms.

All code is in the repository. [4].

V. RESULTS

Once our models were tuned and trained, the following results were obtained as shown in the ROC curves of Fig. 3, Fig. 4, Fig. 5, and, Fig. 6, and the Confusion Matrices of Fig. 7, Fig. 8, Fig. 10, and, Fig. 11. Each ROC Curve indicates we have decent model performance, with the best performance being the RFC and XGB models, and the worst performance being the SVC model. None of the models performed worse than random noise. Each of the confusion matrices represent how adept each model is at predicting the correct phase label, with a perfect fit having only nonzero values in the diagonal of the matrix. The confusion matrices indicate that the majority of our phase labels are correctly classified for the compositions within the test split.

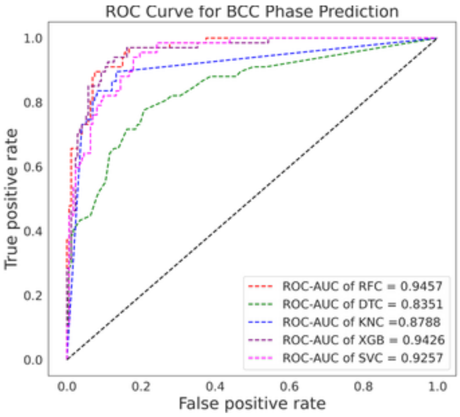


FIG. 3. ROC Curves for all models for the BCC phase label. ROC-AUC scores calculated using the one-vs-all method.

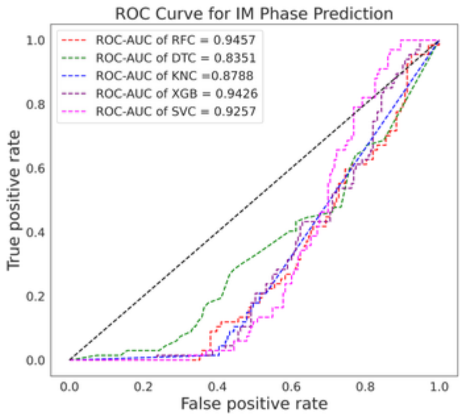


FIG. 4. ROC Curves for all models for the intermetallic phase label. ROC-AUC scores calculated using the one-vs-all method.

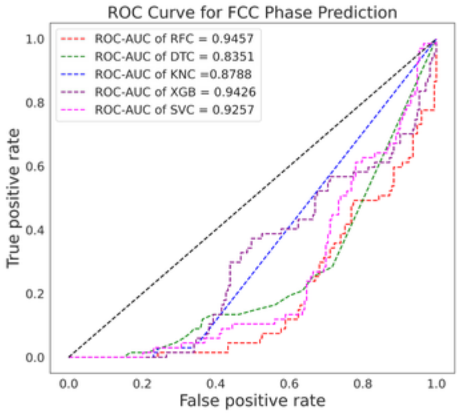


FIG. 5. ROC Curves for all models for the FCC phase label. ROC-AUC scores calculated using the one-vs-all method.

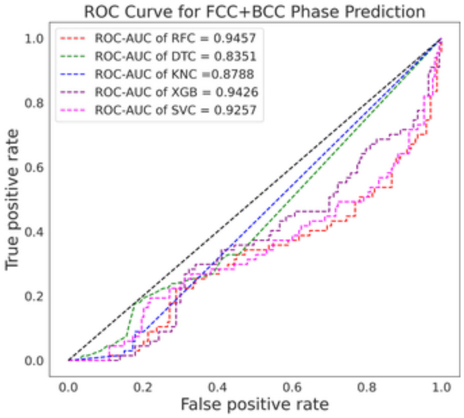


FIG. 6. ROC Curves for all models for the BCC+FCC phase label. ROC-AUC scores calculated using the one-vs-all method.

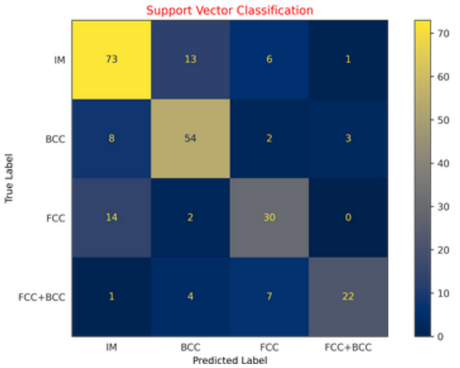


FIG. 7. Confusion Matrix for the tuned SVM model

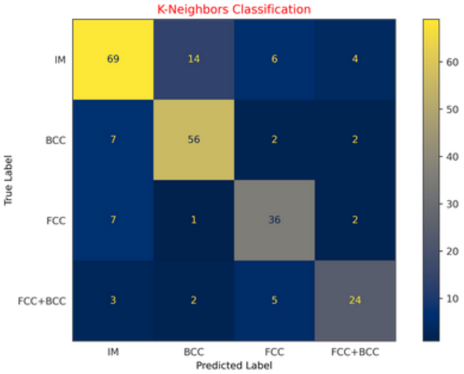


FIG. 8. Confusion Matrix for the tuned KNN model

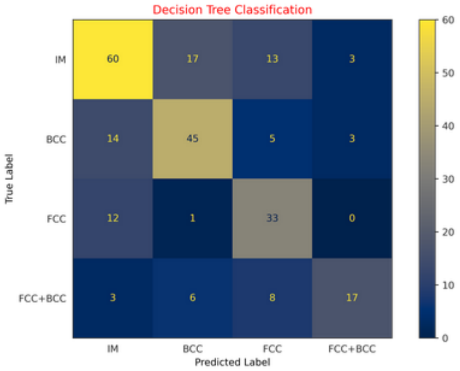


FIG. 9. Confusion Matrix for the tuned DTC model

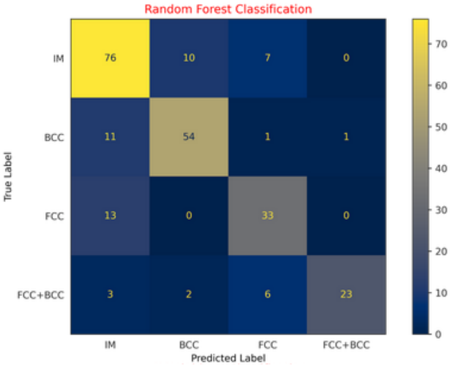


FIG. 10. Confusion Matrix for the tuned RFC model

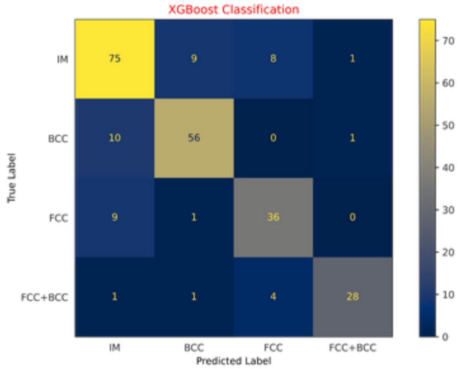


FIG. 11. Confusion Matrix for the tuned XGB model

VI. FUTURE

For future improvements we can try different normalization techniques on the data. In the original paper, the authors used a Multiple Imputation by Chained Equations (MICE) imputer that we did not have time to work with. The algorithm creates multiple imputers to iteratively improve upon. We should do more exploratory data analysis like gauging feature correlations or using dimensionality reduction along with visualizations.

When dealing with outliers, we can experiment with Winsorization where values outside a [X, Y] percent range of the data are replaced with the values at X and Y percent.

Regarding the model tuning. The range of values were chosen arbitrarily so increasing it may help find a better set of parameters. Additionally, we could choose to rank parameters in order of importance and set one parameter fixed so we can test the other parameters. The models may also be too complex so we should work to reduce the number of parameters to avoid overfitting.

Feature engineering is another step. The parameters we used are what material scientists use to characterize HEAs but that doe not mean all those physical parameters carry the same weight. It would be advantageous to isolate a few features that most effectively aid model performance and prune those that carry less weight.

In the future, we could test our model on the full dataset provided, which includes computationally generated HEAs. We should also work to gather more data from different sources to further validate our model.

VII. CONCLUSION

In this work, we tuned then train the SVM, KNN, DTC, RFC, and, XGB classification algorithms to assign one of four phase labels to a spread of HEA compositions. We compared the performance of 5 classification algorithms and determined the Random Forest Classifier to perform the best with a ROC-AUC score of 0.9457 which is slightly less than the paper's best of 0.9578.

Hyperparameter tuning and training the models, we were able to get fairly good results, comparable to that from the paper. We can conclude that RFCs are the best bet to continue future optimization, although it may be worth experimenting to see if other algorithms could achieve comparable results.

ACKNOWLEDGMENTS

Credit to the paper's authors for providing multiple data sets and their baseline models.[1]

[1] K. N. G. Singh, S. *et al.*, Phase prediction and experimental realisation of a new high entropy alloy using machine learning, in *Sci Rep 13* (2023) 4811.

[2] E. P. George, D. Raabe, and R. O. Ritchie, High-entropy alloys, Nature Reviews Materials **4**, 515 (2019).

[3] E. J. Pickering, A. W. Carruthers, P. J. Barron, S. C. Middleburgh, D. E. J. Armstrong, and A. S. Gandy, High-entropy alloys for advanced nuclear applications, Entropy **23** (2021).

[4] L. Hao, O. Eric, *et al.*, https://github.com/htluu-ucsd/phys-139-project.