Martensite Start Temperature Predictor for Steels Using Ensemble Data Mining

Ankit Agrawal[1], Abhinav Saboo[2], Wei Xiong[3], Greg Olson[4], Alok Choudhary[1]
[1]Department of Electrical and Computer Engineering, Northwestern University
[2]QuesTek Innovations
[3]Department of Mechanical Engineering and Materials Science, University of Pittsburgh
[4]Department of Materials Science and Engineering, Northwestern University
Corresponding author email: ankitag@eecs.northwestern.edu

***Abstract*—Martensite start temperature (MsT) is an important characteristic of steels, knowledge of which is vital for materials engineers to guide the structural design process of steels. It is defined as the highest temperature at which the austenite phase in steel begins to transform to martensite phase during rapid cooling. Here we describe the development and deployment of predictive models for MsT, given the chemical composition of the material. The data-driven models described here are built on a dataset of about 1000 experimental observations reported in published literature, and the best model developed was found to significantly outperform several existing MsT prediction methods. The data-driven analyses also revealed several interesting insights about the relationship between MsT and the constituent alloying elements of steels. The most accurate predictive model resulting from this work has been deployed in an online web-tool that takes as input the elemental alloying composition of a given steel and predicts its MsT. The online MsT predictor is available at http://info.eecs.northwestern.edu/MsTpredictor.**

***Keywords*-Materials informatics, supervised learning, ensemble learning, steel**

I. Introduction

The field of materials science and engineering involves experimentation and simulations to understand the processing-structure-property-performance relationships in materials and develop new materials with better properties [1]. If we look at the advances in the field of materials science and engineering over the centuries, we can identify four distinct stages of development, much like any other field of science. For a substantial part of history, materials science was purely observational. In fact, the type of material predominant at different points in history is often used to represent that "age", e.g. stone age, bronze age, iron age, and steel age. This first paradigm of empirical science (today known as the experimental branch of science) was the only method of science for a long time until calculus was invented in mid-17th century, which enabled mathematical modeling of real-world phenomena (second paradigm of model-based theoretical science). Classical examples of second paradigm in the field of materials science are the laws of thermodynamics and quantum mechanics. The invention of computers in the 20th century then allowed for (numerically) solving progressive larger and more complex systems of equations representing the theoretical models, enabling simulations (third paradigm of computational science). Density functional theory and molecular dynamics simulations are excellent examples of the third paradigm in materials science. Over the last couple of decades, the data generated and collected by such experiments and simulations has grown tremendously, motivating the use of (big) data-driven techniques (fourth paradigm of science) [2]. In the field of materials science, this has led to the rise of the new field of materials informatics [3], [4], [1], [5].

The Materials Genome Initiative (MGI) [6] was announced by the US government in 2011 to accelerate the development of advanced materials in the society. In particular, the Materials Genome Initiative envisioned the "discovery, development, manufacturing, and deployment of advanced materials at least twice as fast as possible today, at a fraction of the cost". In general, time-to-insertion of a new material after initial discovery can be a staggering 20 years or more, which MGI aims to cut by half. The Materials Genome Initiative Strategic Plan [7] specifically identified data analytics as one of the key objective to enhance the value of experimental and computational data.

Inspired by the above-described vision of MGI and several recent applications of materials informatics [8], [9], here we describe the development of an online data-driven tool for predicting martensite start temperature (MsT) of steels, which is an important property for engineering steels. The predictive model deployed in the tool is built on an experimental dataset of about 1000 steels represented by their elemental compositions and corresponding martensite start temperatures. Following are the main contributions of this work:

- Comparison of 40 supervised modeling techniques on the MsT dataset to build models to predict MsT from steel composition.
- Investigating the relative importance of individual attributes using feature ranking techniques.
- Deployment of the most accurate predictive model in an online materials informatics tool.

The online tool developed in this work is expected to be a useful resource for the materials science and engineering community to make fast and accurate predictions of this crucial property of steels, which can in turn aid in designing new improved steels. The rest of the paper is organized

978-1-7281-4493-1/19/$31.00 ©2019 IEEE
DOI 10.1109/DSAA.2019.00067

as follows: Section 2 briefly describes the materials science background relevant to this work, and Section 3 presents the data mining workflow used in this study. Experimental results, analysis, and the MsT predictor tool are presented in Section 4, and we conclude the paper with some future directions in Section 5.

II. Materials science background

In this section, we briefly describe the basic materials science concepts of steel processing and martensite start temperature, as relevant to this study.

It is well-known that almost everything in materials science and engineering depends on understanding processing-structure-property-performance (PSPP) relationships [1], where the science relationships of cause and effect go from left to right (e.g., materials processing/structure is the cause and property is the effect), and engineering relationships of goals and means go from right to left (e.g., if the goal is to optimize the performance of a material, the means to achieve it would be to modify the processing, composition, and structural aspects of the material). Note that material composition can be considered a subset of the structure information. Typically, experimentation (simulation) to measure (simulate) a certain property of a given material describes a forward PSPP relationship, and can be thought of as a single data point in mapping the high-dimensional space of possible materials to their property. A dataset of experiments or simulations could therefore be used with supervised machine learning techniques to learn this high-dimensional mapping, which can serve as a fast proxy or surrogate model (also known as forward models) for experiments and simulations, and in turn can also help in realizing the inverse models of materials discovery and design. Recent years have seen a surge in the application of machine learning techniques in materials science [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [9], [23], [24], [25], [26], [27], [28].

There has been an active development of new improved high-performance steels over the past few decades to meet the demands of important industrial applications such as construction, transport, appliances, and so on. Design of new steels and its processing optimization is greatly accelerated with the help of computational models. In accordance with the PSPP relationships described above, properties of steels are dependent on its structure, which in turn is dependent on the processing it undergoes. Structure of a steel is composed of different phases which can be predicted by modeling microstructure characteristics like transformation temperatures, precipitation kinetics, etc. Modeling these characteristics as a function of steel composition and processing parameters can enable design and optimization of new steels suitable for these demanding high performance applications.

One such transformation temperature that characterizes the phase constitution of a steel is the martensite start temperature (MsT). As the name suggests, it is the temperature at which the austenite-to-martensite transformation begins during cooling of steels. One typical martensitic transformation in steels is the one with body-centered tetragonal (BCT) structure generated upon cooling (usually rapidly) from high temperature where the stable parent phase is austenite which has a face-centered cubic (FCC) structure. This martensitic phase transformation has several unique aspects that make it exhibit better performance than the parent austenite phase in certain applications involving high strength steels. Knowing the MsT of a given steel alloy can help in optimizing the processing and composition to obtain the desired microstructure. For example, it is known that addition of several alloying elements like carbon (C), nickel (Ni) and chromium (Cr) in steel could decrease the MsT to even below the room temperature. For such high alloyed steels, design of a special low temperature cryogenic treatment is required to get a martensitic microstructure and meet the high strength targets. Thus, it is important to be able to predict MsT for steels of interest in advance. Another purpose of MsT prediction could be to avoid the formation of martensite phase in steels like in the case of austenitic steels that are known to have better low temperature toughness than martensitic steels.

There are several prior works for predicting MsT. They can be broadly classified into two categories: i) mechanistic or those based on thermodynamics [29], [30], [31], [32]; and ii) empirical or data-driven [33], [34], [35], [36], [37], [38], [39], [40]. A critical assessment of these models is available in the literature [41]. We compare the results of the proposed ensemble model with some of the above existing works in the experiments and results section.

III. Methods

The overall data mining workflow is depicted as a block diagram in Figure 1. Next, we describe the data and the different stages of the workflow.

A. Dataset

MsT is most commonly measured experimentally using a quenching dilatometer which measures the volume change of a sample as it is cooled down. Austenite-to-martensite transformation results in a volume expansion which can be measured using a dilatometer as a function of temperature to give the starting temperature where this volume change starts to happen, thus giving the MsT. This experimental technique has been used for decades, and MsT of nearly all common steel grades can be found in atlases or similar publications. Sourmail et al. [42] have compiled experimental MsT measurements (in Kelvin) and corresponding steel composition (in wt %) for 1091 steels. The steels in this dataset consist of the following alloying elements: carbon (C), manganese (Mn), silicon (Si), chromium (Cr), nickel (Ni), molybdenum (Mo), vanadium (V), cobalt (Co), aluminium (Al), tungsten (W), copper (Cu), niobium (Nb), titanium (Ti), boron (B), and nitrogen (N). This is one of the most comprehensive dataset of experimental martensite start temperatures of steels.

B. Preprocessing

The collected data was found to have duplicate records in terms of steel composition, i.e., multiple observations of

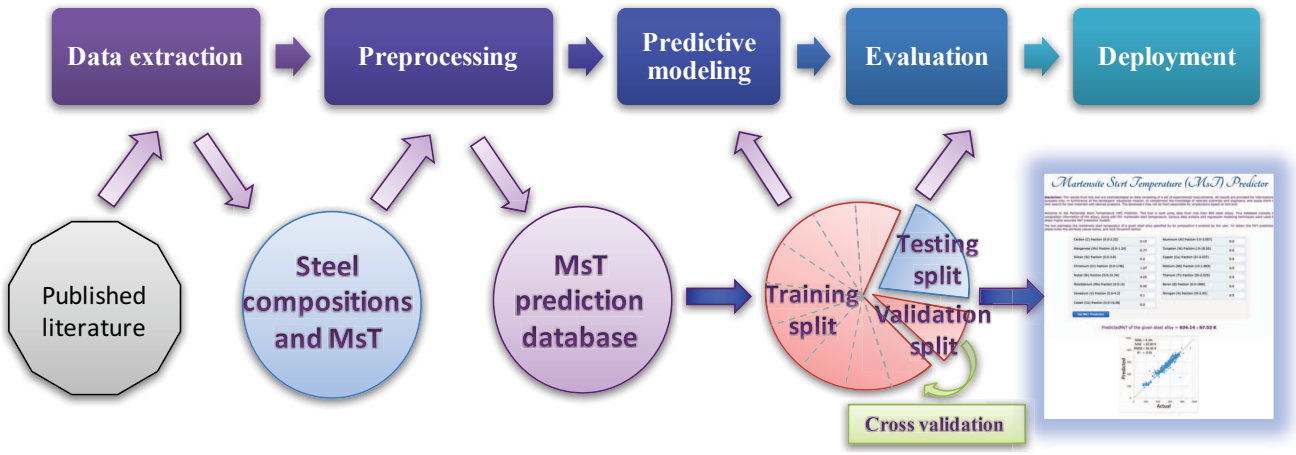

Fig. 1. The data mining workflow used in this work. Experimental data extracted from published literature was preprocessed to get the MsT prediction database, which was then analyzed with supervised learning techniques to learn predictive models for MsT. The models were evaluated using standard validation techniques, and the most accurate models were subsequently deployed in an online user-friendly web-tool that can predict MsT of arbitrary steel compositions.

identical steel compositions but different MsT values. This is possible since there could be other material factors in addition to the composition that also influence the martensite start temperature of a steel alloy, which are not accounted for in this dataset. In addition, multiple measurements taken even for the same material can have some experimental variance and noise. The machine learning techniques we use in this work need to be fed de-duplicated data, otherwise there could be highly similar data points in the training and testing sets, which would overestimate the model accuracy. As part of data preprocessing, therefore, we reduce multiple instances of identical steel compositions to one instance and assign it a new MsT value which is the average of the MsT values of the duplicates.

C. Predictive modeling

Following the methodology of [9], we investigated 40 regression schemes in this study, resulting from relevant compatible combinations of various regression and ensembling techniques. The 12 regression techniques explored in this study include: linear regression, nearest-neighbor (IBk), nearest-neighbor (KStar), artificial neural networks (ANNs), Gaussian process, support vector machines, decision table, decision stump, M5 model trees, random tree, reduced error pruning tree, and random forest. The 6 ensemble modeling methods include: boosting, bagging, random committee, random subspace, rotation forest, and voting. Since not all regression techniques and ensembling techniques are compatible with each other, and due to some other reasons (such as large model size, large training/testing time, etc.) [9], only few combinations of the regression and ensembling techniques were explored, in addition to all the 12 base regression techniques. This resulted in a total of 40 distinct modeling configurations. Here we briefly describe only those regression and ensembling techniques that ended up featuring in the final ensemble model.

1) **Artificial neural networks**: ANNs consist of multiple layers of interconnected artificial neurons between inputs and outputs, and have been shown to be quite powerful for modeling complex and non-linear relationships. Neurons are connected to each other with weighted edges that are learned during the training process by trying to minimize the error (loss function) between the predictions and ground truth values, and backpropagating it to allow the weights to be iteratively updated using gradient descent. Multilayer perceptron for regression with one hidden layer was used in this work. A lot of excellent descriptions of neural networks are available in literature [43], [44].
2) **M5 model trees**: M5 Model Trees [45] are a reconstruction of Quinlan's M5 algorithm [46] for learning trees of regression models, which essentially learns a decision tree with linear regression functions at the leaf nodes. It first performs hierarchical data segmentation to partition the training data into clusters with minimal intra-cluster variation in the class values, and subsequently learns a linear regression function for each cluster. In addition, it is also capable of dealing with missing values using the CART techniques [47].
3) **Additive regression (Boosting)**: Additive regression is a boosting-based ensembling technique designed to iteratively enhance the performance of a base regressor or classifier. First the base model is used to build and evaluate a model, and each subsequent iteration fits a new model to the residuals left by the model in the previous iteration [48]. Eventually, the predictions of all

individual learners are added to get the overall prediction for a given test instance.

4) **Random Committee**: This is a another ensembling technique for combining multiple randomizable base models, i.e., models that take a random seed as input. Multiple base models are built on the same data using different random seeds, and the final prediction is simply an average of the individual predictions.
5) **Voting**: Voting is a popular ensemble technique for combining multiple classifiers. It is well known that ensemble classifiers using voting can outperform the individual classifiers [49]. Predictions from multiple classifiers can be combined using simple functions such as mean, maximum, minimum, median, etc.

D. Evaluation

The models are evaluated in terms of how close their predictions are to ground truth. The metrics used for this purpose include the coefficient of correlation (R), Mean Absolute Error (MAE), Root Mean Squared Error ($RMSE$), Relative Absolute Error (RAE), and Root Relative Squared Error ($RRSE$).

$$R = \frac{\sum_{i=1}^{N}(y_i - \overline{y})(\hat{y}_i - \overline{\hat{y}})}{\sqrt{\sum_{i=1}^{N}(y_i - \overline{y})^2 \sum_{i=1}^{N}(\hat{y}_i - \overline{\hat{y}})^2}} \quad (1)$$

$$MAE = \overline{e} = \frac{1}{N}\sum_{N}|y - \hat{y}| \quad (2)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{N}(y - \hat{y})^2} \quad (3)$$

$$RAE = \frac{\sum_N |y - \hat{y}|}{\sum_N |y - \overline{y}|} \quad (4)$$

$$RRSE = \sqrt{\frac{\sum_N (y - \hat{y})^2}{\sum_N (y - \overline{y})^2}} \quad (5)$$

$$MAE_f = \overline{e_f} = \frac{1}{N}\sum_{N}\left|\frac{y - \hat{y}}{y}\right| \quad (6)$$

where y denotes the actual MsT (K), $\hat{y}$ denotes the predicted MsT (K), $\overline{y}$ denotes the average MsT across the dataset, and N is the number of instances in the dataset.

Of the above, we used MAE as the primary evaluation metric in this work, since it is intuitive and has been used in prior works for modeling MsT.

E. Attribute ranking

We used two metrics for attribute ranking. Correlation coefficient between each attribute and the target attribute is used to see whether MsT increases or decreases w.r.t. a given composition attribute. But correlation alone does not tell much about the predictive potential of an attribute. Information gain ratio is used for this purpose, which is an entropy-based metric that independently evaluates each attribute by measuring the ratio of the information gain with respect to the target attribute to the entropy of the given attribute:

$$IG(Class, Attrib) = \frac{H(Class) - H(Class|Attrib)}{H(Attrib)} \quad (7)$$

where $H(.)$ denotes the information entropy.

IV. Experiments and results

As mentioned earlier, the data from [42] was found to have duplicate entries. So as part of data preprocessing, the data was deduplicated, which reduced the data size from 1091 down to 843 steel compositions. We call this dataset as the MsT prediction database consisting of 843 steels (rows), 15 composition elements (attributes), and 1 target attribute (MsT). A random subset of 20% of this data was set aside as the test set, to evaluate the final model and compare it with existing methods. The remaining 80% data was used for training and validation to identify the best model. 10-fold cross validation was used for this purpose, which randomly divides the dataset into 10 parts, uses 9 parts as training set and 1 part for validation, and repeats the process 10 times with different validation sets before aggregating the results together. Further, each modeling configuration was run 10 times with different random seeds for cross-validation splits, thereby effectively training and validating 100 models for each modeling configuration on different random subsets of the data, to aid in statistical significance testing. All modeling experiments were performed using WEKA software [50] version 3.8.3 with default parameters, unless otherwise stated. Next, we present the comparison results of different modeling techniques on the training-validation set, followed by the comparison of the best model with existing works on the test set.

A. Comparison of various modeling configurations

Table I presents the results from top 10 modeling configurations found using 10-fold cross on the training-validation set. The best technique in terms of minimum MAE was found to be AdditiveRegression_M5P, and its MAE was found to be statistically indistinguishable (at p=0.05) from that of the second best technique, which was RandomCommittee_MLP. Predictions from these top two models were therefore further combined using an ensemble voting scheme, the results of which are also included in Table I. Not surprisingly, the MAE of the ensemble model was found to be significantly better (at p=0.05) than both of its constituent models. Training the ensemble voting model took less than 1 s, while running it takes less than 1 ms, implying that the model is both fast and accurate.

B. Comparison of proposed ensemble model with existing models

Next, we evaluate the ensemble voting model on data that the modeling process has not seen even once. A new ensemble model was thus created on the entire 80% training-validation set and tested against the 20% held-out test set. Table II presents the performance of the proposed ensemble model as

TABLE I

RESULTS OF TOP 10 MODELING CONFIGURATIONS IDENTIFIED BY 10-FOLD CROSS-VALIDATION ON THE TRAINING-VALIDATION SET (SORTED BY MAE). THE STANDARD DEVIATION ASSOCIATED WITH EACH MEASUREMENT IS ALSO REPORTED. THE MAE OF TOP TWO OF THESE 10 MODELS WERE FOUND TO BE STATISTICALLY INDISTINGUISHABLE AT P=0.05 (MAE VALUES BOLDFACED), AND WERE THUS COMBINED TOGETHER USING AN ENSEMBLE VOTING SCHEME, RESULTS FROM WHICH ARE ALSO INCLUDED IN THE LAST ROW

Modeling Scheme	*R*	*MAE (K)*	*RMSE (K)*	*RAE (%)*	*RRSE (%)*
AdditiveRegression_M5	0.9486 ± 0.0552	**21.9527 ± 3.6544**	35.4755 ± 14.0547	25.2238 ± 4.3941	29.8159 ± 12.7291
RandomCommittee_MLP	0.9463 ± 0.0496	**22.4044 ± 3.6094**	36.3996 ± 13.3623	25.7738 ± 4.5229	30.5557 ± 11.7993
RotationForest_MLP	0.9458 ± 0.0483	22.4987 ± 3.7699	36.5997 ± 12.8438	25.8706 ± 4.6180	30.7219 ± 11.4353
Bagging_M5	0.9415 ± 0.0517	22.8616 ± 4.2257	38.3590 ± 14.6020	26.2373 ± 4.7531	32.1660 ± 12.6500
M5 ModelTrees	0.9443 ± 0.0467	23.1894 ± 3.5929	37.2988 ± 12.3603	26.6608 ± 4.3509	31.3414 ± 11.0554
Bagging_MLP	0.9365 ± 0.0663	23.2617 ± 4.1981	39.2367 ± 15.7974	26.7633 ± 5.1806	33.0120 ± 14.4039
NeuralNetworks (MLP)	0.9340 ± 0.0774	24.8646 ± 4.5520	40.5872 ± 18.9065	28.6163 ± 5.7269	34.0609 ± 16.5726
RotationForest_RandomTree	0.9433 ± 0.0343	25.1803 ± 3.7629	38.7369 ± 9.3684	28.8701 ± 4.0892	32.4207 ± 8.1862
RandomForest	0.9430 ± 0.0331	25.4917 ± 4.1126	39.4521 ± 9.8040	29.1910 ± 4.1471	32.9077 ± 8.0169
AdditiveRegression_MLP	0.9172 ± 0.0811	26.2108 ± 5.3421	45.7042 ± 19.4068	30.2227 ± 6.8432	38.6174 ± 17.8222
EnsembleVoting	0.9517 ± 0.0457	21.2061 ± 3.4484	34.3727 ± 12.6705	24.3747 ± 4.1824	28.8816 ± 11.3498

TABLE II

COMPARISON OF PROPOSED ENSEMBLE MODEL WITH EXISTING METHODS ON THE TEST SET. * INDICATES THE USE OF A DIFFERENT TEST SET THAN USED IN THE PRESENT STUDY.

Model	*R*	*MAE (K)*	*RMSE (K)*	*MAE_f (%)*
Ghosh et al. [32]	0.88	44.13	61.27	8.07
Payson et al. [33]	0.80	44.75	84.12	9.07
Grange et al. [34]	0.77	52.48	107.51	10.41
Nehrenberg et al. [35]	0.83	38.59	72.82	7.78
Steven et al. [36]	0.76	42.85	98.80	9.81
Andrews et al. [37]	0.82	37.35	80.37	8.17
Andrews et al. [37]	0.63	53.94	136.78	11.26
Kunitake et al. [38]	0.83	38.61	81.45	7.94
Capdevilla et al. [39]	0.85	35.87	62.23	7.49
Sourmail et al. [40]*	NA	22 [40]	NA	NA
EnsembleVoting	**0.97**	**18.02**	**25.52**	**3.56**

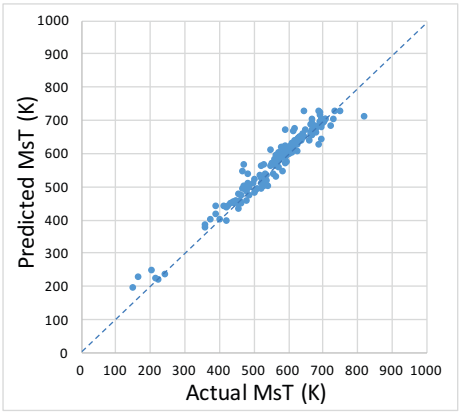

Fig. 2. Parity plot of predicted MsT values from the ensemble voting model on the test set vs. actual MsT values.

well as of some of the existing works on MsT prediction [32], [33], [34], [35], [36], [37], [38], [39], [40] on the test set. Note that the mechanistic model from [32] failed on 5 out of 169 instances of the test set, so the evaluation metrics for [32] in Table II were calculated on the remaining 164 test instances. Clearly, the proposed ensemble voting model outperforms all other models across all available evaluation metrics.

All prior models in Table II were evaluated on the same test set with the exception of [40], so its results are not directly comparable. The MAE for [40] in the table is the one reported by the authors themselves in [40], using a random but unknown subset of the same dataset used in the present study [42] as their test set. Since their training and testing splits are unavailable, it is not possible to do an exact comparison. But because they used the same data as in this work, we include their reported accuracy in the same table. Moreover, as indicated before, the dataset from [42] has duplicate entries, but it is not clear from [40] whether the authors performed any preprocessing to deal with it. Presence of duplicate data points can potentially lead to overlap between training and testing sets, thereby overestimating the model accuracy. Therefore, based on the comparison results in Table II, we observe that the proposed ensemble voting model produces the most accurate predictions of MsT on the test set.

Another interesting thing to note from Table II is that the accuracy of ensemble voting is apparently even better than the cross validation accuracy in Table I. Although little surprising, this is not impossible, since the cross validation accuracy metrics were indeed found to have some variation (indicated by corresponding standard deviation values in Table I). Moreover, all the accuracy metrics on the test set for the ensemble voting model are within one standard deviation of their corresponding cross validation accuracy metrics.

The parity plot for MsT predictions from the ensemble voting model vs. ground truth MsT values on the test set is depicted in Figure 2.

C. Attribute ranking

It would also be interesting to identify if any attributes exhibit correlation with MsT, and which ones are most influential in predicting MsT. We used correlation and information gain ratio respectively for this purpose. Since information gain ratio can be calculated only with discrete target attributes, we first

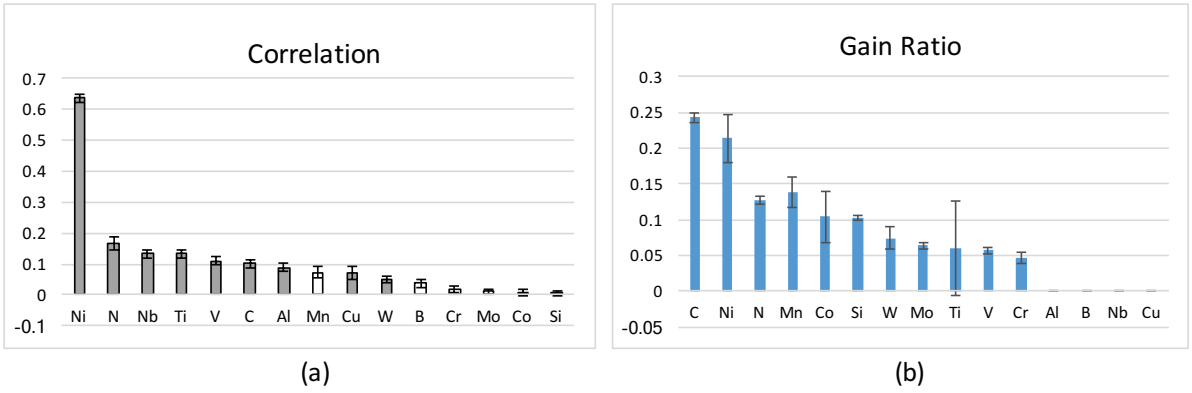

Fig. 3. Attribute rankings in terms of a) correlation with MsT (grey bars denote negative correlation and white bars denote positive correlation) and b) information gain ratio for predicting MsT. Standard deviation of all measurements indicated as error bars.

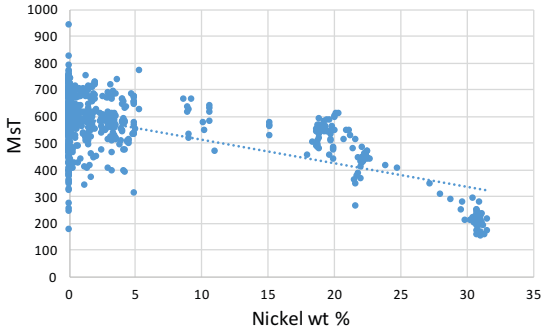

Fig. 4. Scatter plot of MsT and Nickel (Ni) wt %. The presence of multiple observations of steel alloys with high Ni content at low MsT leads to a significantly more negative correlation of Ni with MsT as compared to other elements.

discretized it into two equal-width bins. The resulting attribute rankings are depicted in Figure 3.

From the correlation ranking plot in Figure 3a, we find that all the elements are either negatively correlated or uncorrelated (with very weak positive/negative correlation) w.r.t. MsT. As mentioned before, it is well known that alloying a steel decreases its MsT in general, so the correlation findings are in accordance with domain knowledge. Interestingly, while most of the element-MsT correlations are weak, Nickel (Ni) is relatively more negatively correlated with MsT. A closer look at the data reveals that compared to other elements, Nickel is significantly more well represented at higher concentrations in the dataset (e.g., more than 100 steels with Ni $>$ 15 wt % and more than 30 with Ni $>$ 30 wt%), and the corresponding high-Ni steels have significantly low MsT (see Figure 4).

The information gain ratio ranking plot in Figure 3b reveals the elements that have the most influence on MsT. Carbon (C) and Nickel (Ni) top the list, which is again consistent with domain knowledge. Other elements with significant influence on MsT (in decreasing order of influence) were found to be nitrogen (N), manganese (Mn), cobalt (Co), silicon (Si), tungsten (W), molybdenum (Mo), titanium (Ti), vanadium (V), and chromium (Cr). Other elements like aluminium (Al), boron (B), niobium (Nb), and copper (Cu) were quite less represented in the dataset, which is probably why they did not show up as influential in the above analysis.

D. Model interpretability

Data mining models are sometimes criticized for being too much like a black box and not interpretable, even if they are very accurate. In other words, it is possible to get a high predictive accuracy with a data mining model without understanding why the model is making a certain decision or more broadly what the model has learned. While interpretability may not be very consequential for some applications like movie recommendations, it is indeed important for engineering applications, where the cost of a wrong prediction can be tremendously large. Therefore, it is important to make sure that the model is not learning something known to be incorrect, and thus, interpretable models are highly desirable. There are many notions of model interpretability and it is increasingly gaining attention in the data mining research community [51].

It is not surprising that usually there is a tradeoff between model accuracy and interpretability. Our proposed model for MsT prediction is a voting ensemble of multiple ensembling methods with different base regressors, and thus quite complex to be directly interpretable. But since one of the base learners in our ensemble model is based on interpretable decision trees (M5 model tree), we decided to look into what the decision tree model learns. So we built a M5 model tree on the MsT prediction database, and it is depicted in Figure 5. There are 10 attribute comparison nodes (oval-shaped, non-leaf nodes) and 11 decision nodes (rectangle-shaped, leaf nodes). Each leaf node corresponds to a different linear regression function learned for the specific subset of training data that reaches that leaf node on going through the tree. There are several interesting things that can be observed from this decision tree:

- Carbon (C) and nickel (Ni) are the very initial attributes the model looks at, which is very much in accordance

with domain knowledge, and reaffirms the dependence of MsT on these elements, as was also found by attribute ranking in terms of information gain ratio.

- Four out of 10 attribute comparison nodes are of carbon (C), implying that the model tried to segment the data by varying carbon content, and learn a different function for each segment. This is also supported by domain knowledge, as we know carburization is one of the most important processing step for steels to absorb carbon and make them harder. Steels with different carbon content can vary a lot in their properties, and are in fact formally classified as low carbon, medium carbon, high carbon, and very high carbon steels. It is thus quite remarkable that the model automatically learned this in a data-driven fashion without explicitly being provided that knowledge. The way the tree has been constructed indicates that carbon content increases from left to right in the tree.
- Most of the attribute coefficients of the linear models are negative, which is expected as alloying tends to reduce the MsT.
- The increasing carbon content from left to right in the tree is also reflected in the corresponding linear models. The carbon coefficients are more negative for the initial linear models, since the carbon content for those steels is lower.

These observations suggest that the M5 model tree was successfully able to segment the data into meaningful segments, which allowed it to fit specific regression functions for each segment to obtain high accuracy. Since the M5 model tree is one of the two base learners that went into the ensemble model, we believe that the above observations provide (limited) interpretability of the model, thereby making it both accurate and trustable.

E. Martensite start temperature predictor

We have created an online MsT predictor that can take as input a steel alloy represented by its alloying composition, and generate a prediction of its martensite start temperature. The ensemble voting model built on the entire MsT database has been deployed in this tool. Figure 6 shows the screenshot of the MsT predictor, and the tool is available online at http://info.eecs.northwestern.edu/MsTpredictor. The same webpage also provides the data used to build the model, including exact training-validation and testing splits, so as to facilitate reproducibility and comparison with other methods.

V. Conclusion and future work

In this applied data science paper, we compared 40 different modeling techniques for predicting martensite start temperature (MsT) of steel alloys using a database of experimental MsT values. The most accurate models were combined with an ensemble voting scheme and deployed in an online web-tool called the MsT predictor. Various analyses presented in this paper shows that the final ensemble model for MsT prediction is quite fast, accurate, and trustable. The primary advantage of this tool is the capability of quickly and accurately predicting martensite start temperature of a steel, which is an important property for the design and engineering of advanced high performance steels. The deployed tool is expected to be a useful resource for researchers and practitioners in the materials science and engineering community.

Future work includes making attempts to further improve the model accuracy by using/deriving more relevant attributes. It would also be interesting to integrate such predictive models with thermodynamic models such as Computer Coupling of Phase Diagrams and Thermochemistry (CALPHAD), as well as with other data-driven models for predicting other important properties and performance metrics of steel, such as fatigue strength. In the long-term, such accurate and fast forward models of different steel properties can be gainfully combined to realize the inverse models for data-driven discovery and design of advanced steels.

Acknowledgment

This work was performed under the following financial assistance award 70NANB19H005 from U.S. Department of Commerce, National Institute of Standards and Technology as part of the Center for Hierarchical Materials Design (CHiMaD). Partial support is also acknowledged from DOE awards DE-SC0014330, DE-SC0019358.

References

[1] A. Agrawal and A. Choudhary, "Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science," *APL Materials*, vol. 4, no. 053208, pp. 1–10, 2016.

[2] T. Hey, S. Tansley, and K. Tolle, *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, 2009. [Online]. Available: http://research.microsoft.com/en-us/collaboration/fourthparadigm/

[3] S. R. Kalidindi and M. D. Graef, "Materials data science: Current status and future outlook," *Annual Review of Materials Research*, vol. 45, no. 1, pp. 171–193, 2015. [Online]. Available: http://dx.doi.org/10.1146/annurev-matsci-070214-020844

[4] K. Rajan, "Materials informatics: The materials "gene" and big data," *Annual Review of Materials Research*, vol. 45, no. 1, pp. 153–169, 2015. [Online]. Available: http://dx.doi.org/10.1146/annurev-matsci-070214-021132

[5] A. Agrawal and A. Choudhary, "Deep materials informatics: Applications of deep learning in materials science," *MRS Communications*, pp. 1–14, 2019.

[6] Materials Genome Initiative for Global Competitiveness, June 2011; OSTP 2011.

[7] Materials Genome Initiative Strategic Plan, National Science and Technology Council Committee on Technology Subcommittee on the Materials Genome Initiative, June 2014.

[8] A. Agrawal and A. Choudhary, "A fatigue strength predictor for steels using ensemble data mining," in

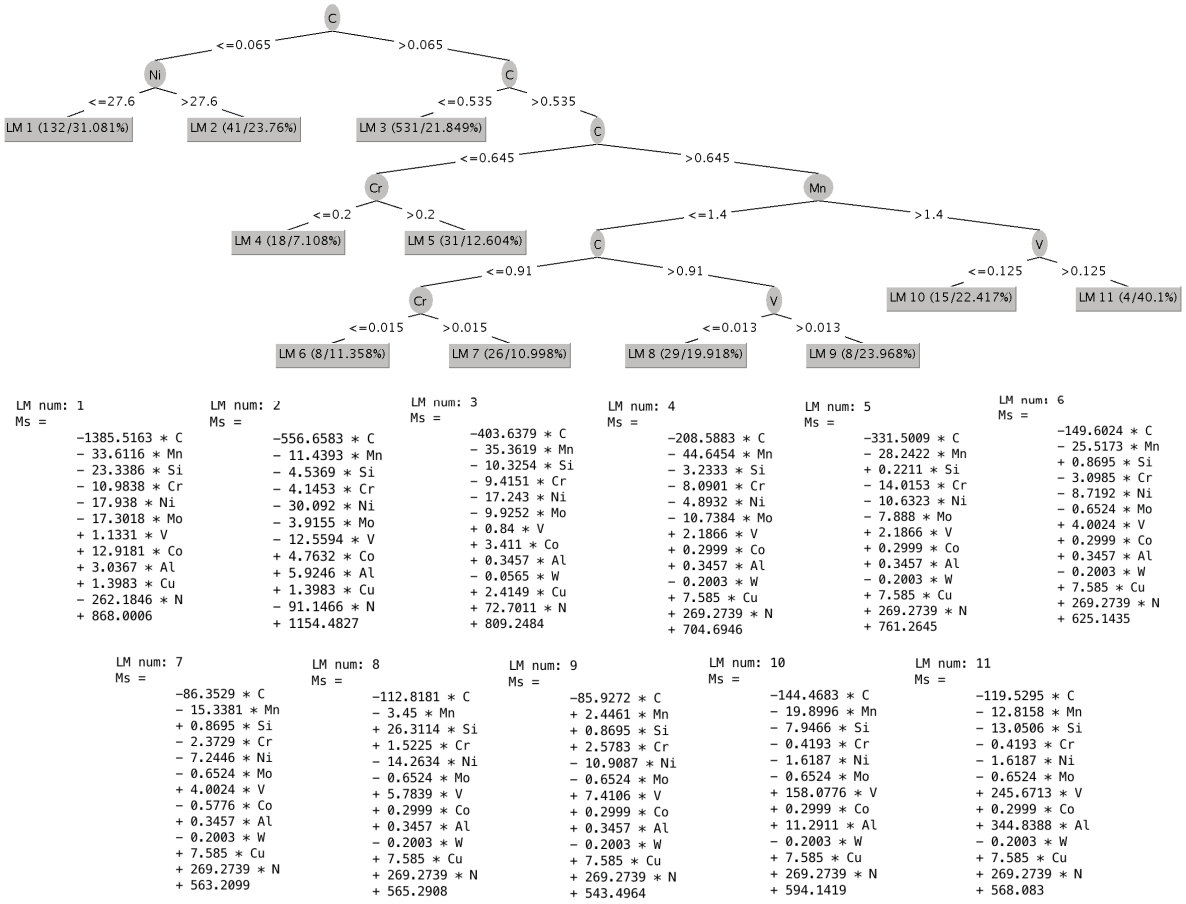

Fig. 5. The M5 model tree learned on MsT prediction database. For a given test instance (in this case steel composition), the model starts at the root node on the top, performs a series of attribute comparisons (in oval-shaped nodes) and goes down the tree till it reaches a leaf node (rectangle-shaped nodes), and subsequently uses the linear model (LM) specified in the leaf node to make the MsT prediction. The two numbers in the leaf node represent the number of instances in the training data that reach that leaf node, and the RRSE % error corresponding to the linear model of that leaf node.

Proceedings of 25th ACM International Conference on Information and Knowledge Management (CIKM), 2016, pp. 2497–2500.

[9] ——, "An online tool for predicting fatigue strength of steel alloys based on ensemble data mining," *International Journal of Fatigue*, vol. 113, pp. 389–400, 2018.

[10] G. Hautier, C. C. Fischer, A. Jain, T. Mueller, and G. Ceder, "Finding nature's missing ternary oxide compounds using machine learning and density functional theory," *Chemistry of Materials*, vol. 22, no. 12, pp. 3762–3767, 2010.

[11] K. Gopalakrishnan, A. Agrawal, H. Ceylan, S. Kim, and A. Choudhary, "Knowledge discovery and data mining in pavement inverse analysis," *Transport*, vol. 28, no. 1, pp. 1–10, 2013.

[12] A. Agrawal, P. D. Deshpande, A. Cecen, G. P. Basavarsu, A. N. Choudhary, and S. R. Kalidindi, "Exploration of data science techniques to predict fatigue strength of steel from composition and processing parameters," *Integrating Materials and Manufacturing Innovation*, vol. 3, no. 8, pp. 1–19, 2014.

[13] A. G. Kusne, T. Gao, A. Mehta, L. Ke, M. C. Nguyen, K.-M. Ho, V. Antropov, C.-Z. Wang, M. J. Kramer, C. Long *et al.*, "On-the-fly machine-learning for high-throughput experiments: search for rare-earth-free permanent magnets," *Scientific reports*, vol. 4, 2014.

[14] B. Meredig, A. Agrawal, S. Kirklin, J. E. Saal, J. W. Doak, A. Thompson, K. Zhang, A. Choudhary, and C. Wolverton, "Combinatorial screening for new materials in unconstrained composition space with machine learning," *Physical Review B*, vol. 89, no. 094104, pp. 1–7, 2014.

[15] F. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, "Crystal structure representations for machine learning models of formation energies," *International Journal of Quantum Chemistry*, 2015.

[16] R. Liu, Y. C. Yabansu, A. Agrawal, S. R. Kalidindi, and A. N. Choudhary, "Machine learning approaches for

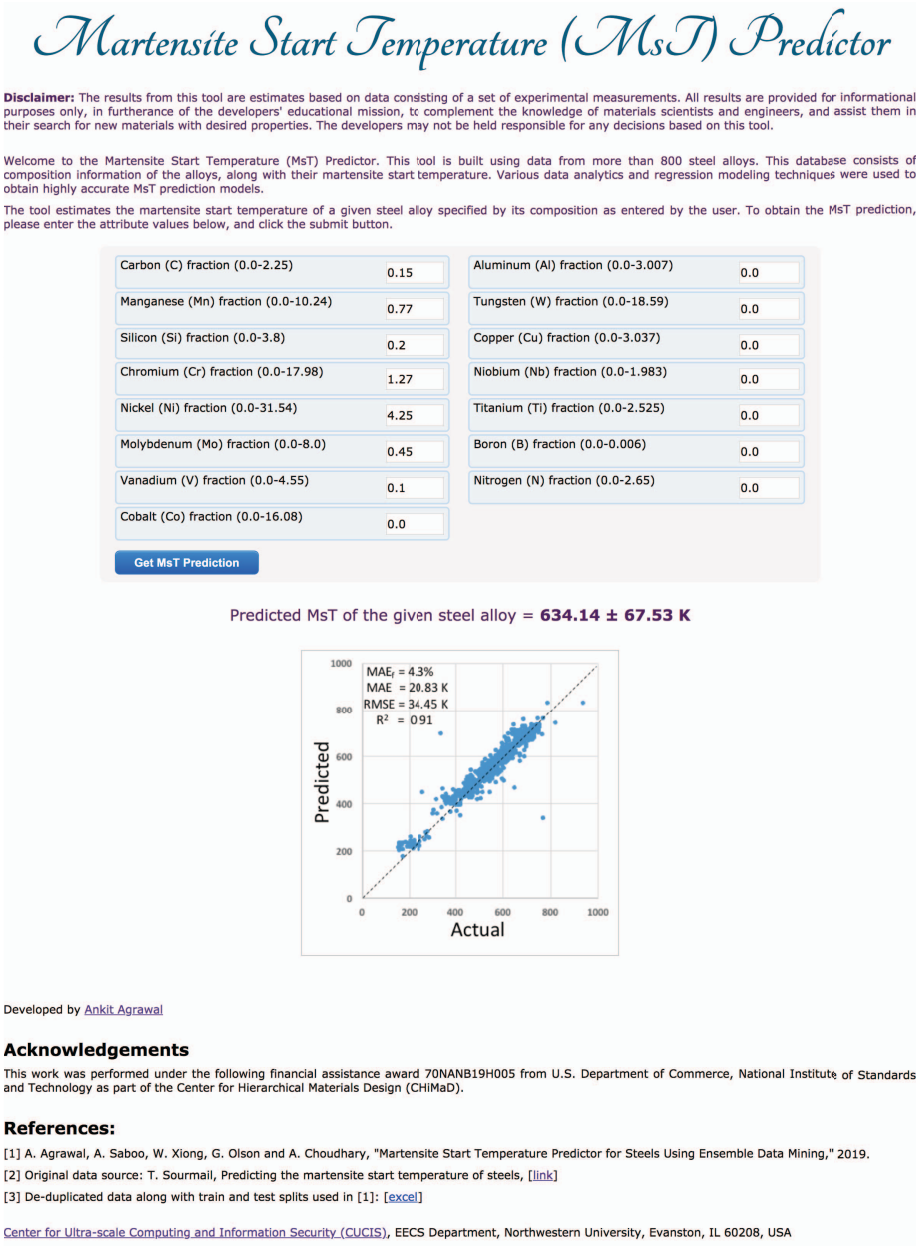

Fig. 6. A screenshot of the deployed MsT predictor.

elastic localization linkages in high-contrast composite materials," *Integrating Materials and Manufacturing Innovation*, vol. 4, no. 13, pp. 1–17, 2015.

[17] R. Liu, A. Kumar, Z. Chen, A. Agrawal, V. Sundararaghavan, and A. Choudhary, "A predictive machine learning approach for microstructure optimization and materials design," *Nature Scientific Reports*, vol. 5, no. 11551, 2015.

[18] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.

[19] L. Ward, R. Liu, A. Krishna, V. I. Hegde, A. Agrawal, A. Choudhary, and C. Wolverton, "Including crystal structure attributes in machine learning models of formation energies via voronoi tessellations," *Physical Review B*, vol. 96, no. 2, p. 024104, 2017.

[20] Z. Yang, X. Li, L. C. Brinson, A. Choudhary, W. Chen, and A. Agrawal, "Microstructural materials design via deep adversarial learning methodology," *Journal of Mechanical Design*, vol. 140, no. 11, p. 10, 2018.

[21] D. Jha, L. Ward, A. Paul, W.-k. Liao, A. Choudhary, C. Wolverton, and A. Agrawal, "Elemnet: Deep learning the chemistry of materials from only elemental composition," *Nature Scientific Reports*, vol. 8, no. 17593, 2018.

[22] D. Jha, S. Singh, R. Al-Bahrani, W.-k. Liao, A. N. Choudhary, M. D. Graef, and A. Agrawal, "Extracting grain orientations from ebsd patterns of polycrystalline materials using convolutional neural networks," *Microscopy and Microanalysis*, vol. 24, no. 5, pp. 497–502, 2018.

[23] Z. Yang, Y. C. Yabansu, D. Jha, W.-k. Liao, A. N. Choudhary, S. R. Kalidindi, and A. Agrawal, "Establishing structure-property localization linkages for elastic deformation of three-dimensional high contrast composites using deep learning approaches," *Acta Materialia*, vol. 166, pp. 335–345, 2019.

[24] A. Paul, P. Acar, W.-k. Liao, A. N. Choudhary, V. Sundararaghavan, and A. Agrawal, "Microstructure optimization with constrained design objectives using machine learning-based feedback-aware data-generation," *Computational Materials Science*, vol. 160, pp. 334–351, 2019.

[25] D. Jha, A. Gilad Kusne, R. Al-Bahrani, N. Nguyen, W.-k. Liao, A. Choudhary, and A. Agrawal, "Peak area detection network for directly learning phase regions from raw x-ray diffraction patterns," in *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, 2019, p. 8 pages.

[26] D. Jha, L. Ward, Z. Yang, C. Wolverton, I. Foster, W.-k. Liao, A. Choudhary, and A. Agrawal, "Irnet: A general purpose deep residual regression framework for materials discovery," in *Proceedings of 25th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*, 2019, p. 9 pages.

[27] A. Paul, D. Jha, R. Al-Bahrani, W.-k. Liao, A. Choudhary, and A. Agrawal, "Transfer learning using ensemble neural nets for organic solar cell screening," in *Proceedings of International Joint Conference on Neural Networks (IJCNN 2019)*, 2019, p. 8 pages.

[28] Z. Yang, R. Al-Bahrani, A. Reid, S. Papanikolaou, S. Kalidindi, W.-k. Liao, A. Choudhary, and A. Agrawal, "Deep learning based domain knowledge integration for small datasets: Illustrative applications in materials informatics," in *Proceedings of International Joint Conference on Neural Networks (IJCNN 2019)*, 2019, p. 8 pages.

[29] G. Ghosh and G. B. Olson, "Kinetics of fcc->bcc heterogeneous martensitic nucleation-i. the critical driving force for athermal nucleation," *Acta Metallurgica et Materialia*, vol. 42, no. 10, pp. 3361–3370, 1994.

[30] ——, "Kinetics of fcc->bcc heterogeneous martensitic nucleation-ii. thermal activation," *Acta metallurgica et materialia*, vol. 42, no. 10, pp. 3371–3379, 1994.

[31] T. Cool and H. Bhadeshia, "Prediction of martensite start temperature of power plant steels," *Materials science and technology*, vol. 12, no. 1, pp. 40–44, 1996.

[32] G. Ghosh and G. B. Olson, "Computational thermodynamics and the kinetics of martensitic transformation," *Journal of Phase Equilibria*, vol. 22, no. 3, p. 199, 2001.

[33] P. Payson and C. Savage, "Martensite reactions in alloy steels," *Trans. ASM*, vol. 33, pp. 261–281, 1944.

[34] R. Grange and H. Stewart, "The temperature range of martensite formation," *Trans. AIME*, vol. 167, pp. 467–490, 1945.

[35] A. Nehrenberg, "Discussion," *Trans. AIME*, vol. 167, pp. 494–498, 1945.

[36] W. Steven and A. Haynes, "The temperature formation of martensite and bainite in low-alloy steels, some effects of chemical composition," *J. Iron Steel Inst*, vol. 183, pp. 349–359, 1956.

[37] K. Andrews, "Empirical formulae for the calculation of some transformation temperatures," *J. Iron Steel Inst.*, pp. 721–727, 1965.

[38] T. Kunitake, "Predictionof ac1, ac3 and ms temperature of steels by empirical formulas," *Japan Society for Heat Treatment*, vol. 41, no. 3, pp. 164–169, 2001.

[39] C. Capdevila, C. FG, and C. G. De Andres, "Determination of ms temperature in steels: A bayesian neural network model," *ISIJ international*, vol. 42, no. 8, pp. 894–902, 2002.

[40] T. Sourmail and C. Garcia-Mateo, "A model for predicting the ms temperatures of steels," *Computational Materials Science*, vol. 34, no. 2, pp. 213–218, 2005.

[41] ——, "Critical assessment of models for predicting the ms temperature of steels," *Computational materials science*, vol. 34, no. 4, pp. 323–334, 2005.

[42] "Url: Predicting the martensite start temperature (ms) of steels, http://www.thomas-sourmail.net/martensite.html, accessed may 17, 2019."

[43] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford: University Press, 1995.

[44] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[45] Y. Wang and I. Witten, "Induction of model trees for predicting continuous classes," in *Proc European Conference on Machine Learning Poster Papers*, Prague, Czech Republic, 1997, pp. 128–137.

[46] J. R. Quinlan, "Learning with continuous classes." World Scientific, 1992, pp. 343–348.

[47] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Monterey, CA: Wadsworth and Brooks, 1984.

[48] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics and Data Analysis*, vol. 38, pp. 367–378, 1999.

[49] J. Kittler, "Combining classifiers: A theoretical framework," *Pattern Analysis & Applications*, vol. 1, no. 1, pp. 18–27, 1998.

[50] M. Hall, E. Frank *et al.*, "The weka data mining software: An update," *SIGKDD Explorations*, vol. 11, no. 1, 2009.

[51] Z. C. Lipton, "The mythos of model interpretability," *CoRR*, vol. abs/1606.03490, 2016. [Online]. Available: http://arxiv.org/abs/1606.03490