

Research Article

Open Access



Industrial big data analysis strategy based on automatic data classification and interpretable knowledge graph

Bingtao Ren, Chenchong Wang, Yuqi Zhang , Xiaolu Wei*, Wei Xu

State Key Laboratory of Rolling and Automation, Northeastern University, Shenyang 110819, Liaoning, China.

*Correspondence to: Dr. Xiaolu Wei, State Key Laboratory of Rolling and Automation, Northeastern University, NO. 3-11, Wenhua Road, Heping District, Shenyang 110819, Liaoning, China, E-mail: weixl@smm.neu.edu.cn

How to cite this article: Ren, B.; Wang, C.; Zhang, Y.; Wei, X.; Xu, W. Industrial big data analysis strategy based on automatic data classification and interpretable knowledge graph. *J. Mater. Inf.* 2025, 5, 20. <https://dx.doi.org/10.20517/jmi.2024.85>

Received: 10 Dec 2024 **First Decision:** 17 Jan 2025 **Revised:** 6 Feb 2025 **Accepted:** 22 Feb 2025 **Published:** 8 Mar 2025

Academic Editor: Qian Ma **Copy Editor:** Pei-Yun Wang **Production Editor:** Pei-Yun Wang

Abstract

Machine learning has emerged as a critical tool for processing the complex and large-scale datasets generated in the steel industry. However, a single machine learning model struggles to capture all relevant information owing to the variety of steel grades, thereby limiting its extensibility and broader industrial application. Furthermore, most machine-learning models are “black boxes” with low interpretability. Therefore, this paper proposes a novel strategy for industrial big data analysis. First, a data classification model was developed using unsupervised clustering techniques to automatically divide the dataset into four distinct classes. Simultaneously, key physical metallurgy (PM) variables were calculated and incorporated as input features to improve property prediction. Next, an interpretable knowledge graph was constructed for each class, connecting the relevant features with the PM variables. Using these graphs, a graph convolutional network (GCN) model was developed for each class to predict the steel properties. The results demonstrate that this approach delivers better predictions than models without automatic data classification. Furthermore, compared to traditional deep learning models, GCN models based on interpretable knowledge graphs provide superior prediction accuracy and significantly improved interpretability and extensibility.

Keywords: Industrial big data, property prediction, data classification, physical metallurgy, graph convolutional network



© The Author(s) 2025. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.



INTRODUCTION

As the steel industry faces growing pressure to reduce carbon emissions, enhancing production efficiency and improving the mechanical properties of steel have become crucial^[1]. The accurate prediction of these properties is vital because reliable models can optimize the chemical composition and processing parameters, resulting in increased efficiency. However, the complexity of steel production processes, which are influenced by various factors, generates large high-dimensional datasets, particularly for hot-rolled steel strips. Traditional analysis methods struggle to handle this complexity^[2-6], underscoring the need for more advanced approaches to predict mechanical properties based on chemical composition and processing parameters while fully extracting valuable insights from industrial big data.

The rapid advancement in machine learning has provided effective research tools in various fields^[7-11]. This emerging method excels in processing industrial data, showcasing powerful big data analysis capabilities such as high efficiency and precise prediction, thereby accelerating material design and innovation. In the field of steel, an increasing number of studies have focused on the use of machine learning algorithms for property prediction^[12-17]. Li *et al.* proposed a deep-learning-based model [convolutional network for predicting mechanical properties (CNPMP)] to predict the mechanical properties of hot-rolled strip steel^[18]. By converting one-dimensional data into two-dimensional data, the complex relationships between various influencing factors can be better expressed. The prediction accuracy of CNPMP was higher than that of various machine learning models. Xie *et al.* designed a deep neural network (DNN) to predict the yield strength, ultimate tensile strength, elongation, and impact energy of industrial steel plates and applied it online to actual steelmaking plants based on the process parameters and composition of raw steel^[19]. Yang *et al.* established a time series neural network based on long short term memory (LSTM) to predict the yield strength, ultimate tensile strength, and elongation of hot-rolled steel plates^[20]. This model can fully utilize the information contained in the input parameters and has better generalization ability. Cui *et al.* established a machine-learning model and used physical metallurgy (PM) and data-driven strategies to reduce the dimensionality of the dataset and predict the yield strength and elongation^[21]. Li *et al.* adopted a 3-dimensional continuous data sampling method for time-temperature deformation and established the gcForest framework to predict the yield strength, tensile strength, and elongation of hot-rolled strip steel^[22]. In these studies, although property prediction models were developed for various steel grades, the differences between different sample types were not considered, resulting in limited extensibility and restricting the industrial application of machine learning. To address this issue, some researchers have established big data analysis systems that combine data classification with property prediction models by introducing PM variables as inputs to guide model training. For example, Li *et al.* combined data classification strategies with multiple regression algorithms to establish a PM-guided multitype steel property prediction model^[23]. However, the analysis system mentioned above exhibited considerable subjectivity in classifying different grades of steel. Therefore, more intelligent and automatic data-classification methods are required. Although the work of Li *et al.* has introduced PM parameters to some extent, it may not accurately predict the Ac1 and Ac3 temperatures based on empirical formulas, because it cannot consider the influence of certain elements. In addition, most machine-learning models are “black boxes” with low interpretability, which may lead to unreasonable predictions. Graph neural networks (GNNs), such as graph convolutional networks (GCNs), may be an effective strategy and have been widely used in the field of materials science in recent years^[24-29]. The special structure of a knowledge graph considers the logical relationships between variables, which may provide the network with good interpretability.

Therefore, this study presents a unique combination of automatic data classification and an interpretable knowledge graph-based GCN model for yield strength prediction in steels. Unlike conventional models that

apply machine learning without distinguishing material subtypes, an automatic clustering approach segments the dataset into distinct clusters, reducing sample complexity and improving prediction accuracy. Additionally, by integrating PM parameters into the knowledge graph, domain knowledge is incorporated to enhance model interpretability. The data used in this study were sourced from the hot-rolling production line of Benxi Iron and Steel Co., Ltd., China. The k-means algorithm was used to classify the dataset automatically. The Thermo-Calc software was used to calculate the PM variables for each cluster to guide the model. Finally, a property prediction model was constructed by integrating the GCN algorithm with knowledge graphs, enabling accurate property prediction.

MATERIALS AND METHODS

Dataset and data preprocessing

In this study, based on a large amount of actual production data of hot-rolled strip steel stored from 2017 to 2020, the original dataset was obtained, and reasonable data preprocessing was performed to improve the quality of the data, providing a high-quality data foundation for establishing a property prediction model in the future. First, raw industrial data were extracted and stored in the MongoDB database, and MongoDB Compass software was used to visualize the raw industrial data. The MongoDB database contains 54,527 samples, each representing a real steel strip. The samples were aggregated using steel grade numbers. Subsequently, data reduction was performed, and samples containing null or unknown values were removed. Then, the input features were selected and the dimensionality was reduced; that is, features with low correlation to the mechanical properties were removed based on professional knowledge and feature importance analysis. Subsequently, outliers or noise were cleaned up, mainly including samples containing temperatures of 0 °C, similar inputs but different outputs, and rolling process parameters of 0. Finally, the data features were standardized to reduce the interference caused by differences in the magnitudes of different input features^[23].

After data preprocessing, the initial dataset was established using 26,165 samples. The dataset contains input features, including 12 alloy element features and 22 processing parameters, with yield strength as the output feature. The information on each feature is presented in Table 1. The initial dataset is composed of seven steel grades; the data amount for each steel grade is shown in Figure 1A. In addition, principal component analysis (PCA) is employed to identify the primary directions of change in the dataset samples. Its goal is to map high-dimensional data onto a lower-dimensional space while preserving the variance information of the samples as much as possible. When reducing the sample dimensionality from 12 to 2 using PCA, the two most significant feature vectors are selected based on sample information, corresponding to the two largest eigenvalues. These feature vectors serve as the basis vectors for the new two-dimensional space, one representing the X-axis and the other the Y-axis. The X-axis indicates the direction of the first principal component, which captures the highest variance in the data and shows the greatest dispersion. On the other hand, the Y-axis represents the direction of the second principal component, which displays the second-highest variance orthogonal to the X-axis. X and Y values were used to represent the two coordinates after dimensionality reduction, and different colors were used to represent the seven steel grades, as shown in Figure 1B. Steel I is a common carbon structural steel, representing the largest portion of the dataset and accounting for approximately 37% of the initial dataset. Steels II and III are pipeline steel and carbon structural steel, respectively. Steels IV and V are low-alloy high-strength steels. Steel VII is a low-alloy steel. Steel VI is a high-quality carbon-structured steel. The compositions of the steels are shown in Supplementary Figure 1.

Property prediction framework

The performance of machine-learning models is significantly affected by the complexity and diversity of the samples within a dataset. As the diversity and complexity of the samples increase, this can sometimes lead to

Table 1. Range of input and output features in the initial dataset

Inputs and output		Minimum	Maximum	Mean	Standard deviation
Inputs	Carbon (wt.%)	0.0170	0.2000	0.1144	0.0482
	Manganese (wt.%)	0.1400	1.4400	0.3651	0.2526
	Silicon (wt.%)	0.0001	0.3500	0.0781	0.0697
	Sulfur (wt.%)	0	0.0250	0.0088	0.0037
	Phosphorus (wt.%)	0	0.0400	0.0137	0.0040
	Chromium (wt.%)	0	0.3600	0.0035	0.0136
	Molybdenum (wt.%)	0	0.0700	0.0002	0.0012
	Copper (wt.%)	0	0.2900	0.0009	0.0037
	Nickel (wt.%)	0	0.0500	0.0009	0.0026
	Titanium (wt.%)	0	0.0720	0.0082	0.0135
	Vanadium (wt.%)	0	0.0220	0.0001	0.0005
	Niobium (wt.%)	0	0.0400	0.0002	0.0017
	fce_xtmp ^a (°C)	1,122	1,322	1,200	17
	rme_tmp_meas ^b (°C)	1,000	1,204	1,074	29
	etmp_surf_hd_FCE ^c (°C)	1,157	1,320	1,279	24
	etmp_avg_hd_RME ^d (°C)	1,134	1,299	1,249	23
	xtmp_avg_hd_RMX ^e (°C)	976	1,213	1,136	22
	FETAIL_tmp_avg ^f (°C)	887	1,104	999	25
	Reduction of pass no. 1 (mm)	7.88	22.20	15.07	1.98
	Reduction of pass no. 2 (mm)	5.26	13.70	8.79	1.40
	Reduction of pass no. 3 (mm)	2.25	8.42	4.57	0.99
	Reduction of pass no. 4 (mm)	1.07	5.32	2.54	0.64
	Reduction of pass no. 5 (mm)	0.50	3.65	1.56	0.46
	Reduction of pass no. 6 (mm)	0.25	2.18	0.89	0.26
	Reduction of pass no. 7 (mm)	0.08	1.35	0.49	0.16
	Force of pass no. 1 (kN)	13,997	50,898	28,039	4,641
	Force of pass no. 2 (kN)	13,533	44,012	27,130	3,841
	Force of pass no. 3 (kN)	11,994	37,877	24,546	3,099
	Force of pass no. 4 (kN)	10,838	32,190	21,121	2,664
	Force of pass no. 5 (kN)	8,035	26,160	16,639	2,113
	Force of pass no. 6 (kN)	6,392	25,046	13,913	1,829
	Force of pass no. 7 (kN)	5,011	18,827	11,437	1,610
FDT_mtmp_tail ^g (°C)	796	923	866	14	
CT_mtmp_tail ^h (°C)	357	717	589	37	
Output	Yield strength (MPa)	130	658	322	61

^afce_xtmp denotes the furnace temperature. ^brme_tmp_meas is the rough rolling inlet temperature. ^cetmp_surf_hd_FCE is the single-point temperature at the head position of the rough rolling surface. ^detmp_avg_hd_RME is the average temperature at the head position of rough rolling. ^extmp_avg_hd_RMX is the average exit temperature at the head position of rough rolling. ^fFETAIL_tmp_avg is the average temperature of the FET rolling inlet. ^gFDT_mtmp_tail is the final deformation temperature. ^hCT_mtmp_tail is the coiling temperature.

a decrease in the predictive accuracy of the model. The initial dataset obtained after preprocessing contained various steel grades. In the context of industrial data, clustering serves as a valuable analytical approach. It involves dividing an industrial dataset into multiple subdatasets, simplifying the dataset complexity and enabling the creation of more accurate prediction models. By developing performance prediction models for each subdataset separately, the efficiency and accuracy of performance prediction can be significantly improved. To ensure the accuracy of subsequent property predictions, this study proposes a property prediction framework that integrates data preprocessing, automatic data classification, and property prediction. In this framework, a preprocessed initial dataset is automatically classified into several

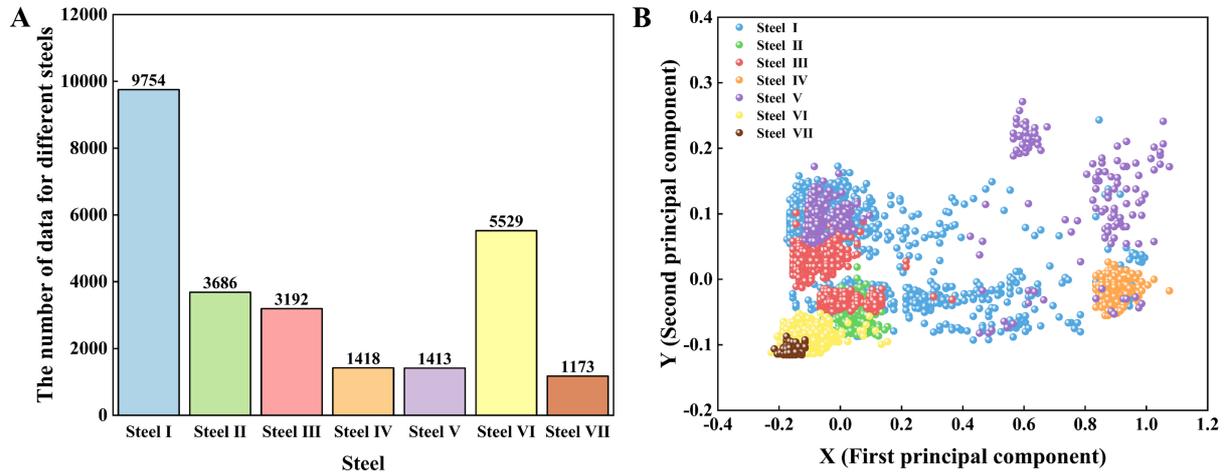


Figure 1. Number and distribution of each steel grade sample in the dataset. (A) Data amount of each steel grade; (B) Data distribution of each steel grade.

subdatasets, with property prediction models established for each subdataset. The property prediction framework is illustrated in Figure 2.

In machine learning, various algorithms are available for dataset classification, including clustering algorithms in unsupervised learning, and classification algorithms in supervised learning. However, for the samples in this study, using a classification algorithm from supervised learning requires manual labeling of the samples, which introduces excessive subjectivity. Therefore, clustering algorithms are adopted for unsupervised classification learning. Common clustering algorithms include k-means clustering, density-based spatial clustering of applications with noise (DBSCAN), and spectral clustering. This study chose the k-means algorithm, which has stable properties and wide applications, to establish the classification model^[30,31]. The k-means algorithm aims to partition an initial dataset into k clusters, ensuring that each data point belongs to the cluster represented by its nearest centroid. The process begins by randomly selecting k data points as the initial centroids. Subsequently, the algorithm calculates the distance between every other data point and each of these initial centroids. Each data point is then assigned to the cluster represented by the nearest centroid. After this assignment, the centroids of the clusters are recalculated based on the mean of the data points assigned to each cluster. This process is repeated iteratively until convergence is achieved. During each iteration, the k-means algorithm refines the cluster assignments by updating the positions of the centroids, progressively optimizing the clustering results until an optimal partition is obtained. It is important to note that the value of k must be predetermined, and selecting the optimal k typically requires the application of various evaluation techniques to ensure the best possible classification outcome^[32-34]. Compared to the K-nearest neighbor (KNN) classification algorithm, the k-means algorithm does not require training data and instead automatically clusters samples based on their feature information^[31]. The automatic data classification model based on the k-means algorithm uses 12 alloy elements as input parameters, and its information is shown in the alloy elements in Table 1. When performing dataset classification, it is necessary to determine the optimal K value (number of clusters) to obtain reasonable classification results. This study used the elbow rule and silhouette coefficient (SC) to evaluate the K values. The sum of the squared errors (SSE) of the elbow rule and the SC are respectively given in

$$SSE = \sum_{i=1}^k \sum_{P \in C_i} |P - m_i|^2 \quad (1)$$

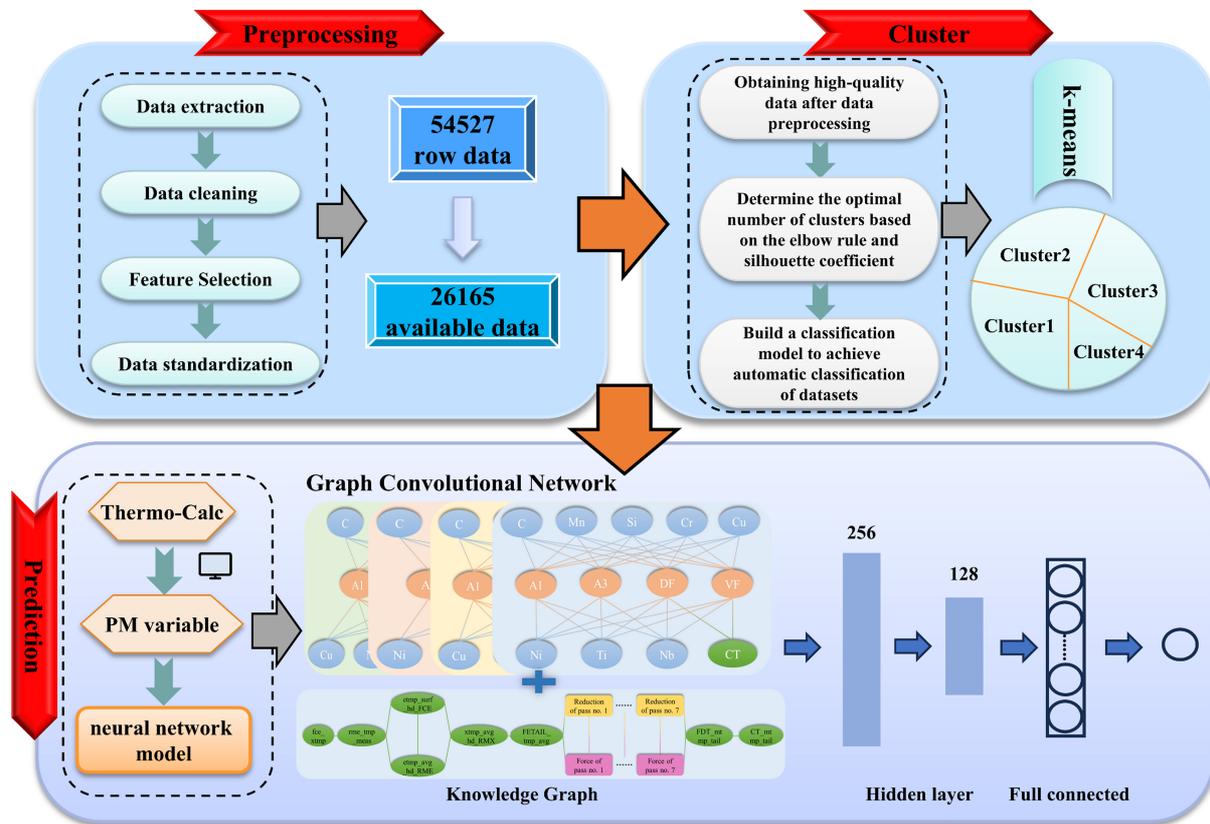


Figure 2. Property prediction framework of GCN based on k-means clustering. GCN: Graph convolutional network.

$$SC = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (2)$$

where C , represents clusters, k represents the number of clusters, P represents samples within a cluster, m represents the cluster center, $a(i)$ is the average distance between the i th sample and other samples in the same cluster, and $b(i)$ is the average distance between the i th sample and all samples in the nearest other clusters. The Euclidean distance was used in this study.

After data classification, the initial dataset was divided into four sub-datasets. Before building the machine learning models to predict the yield strength, feature engineering in each sub-dataset was performed to determine the input features for each cluster, as shown in [Supplementary Figure 2](#). The composition features of each sub-dataset are summarized in [Supplementary Table 1](#).

In addition, introducing PM variables into the model not only improves the quality of the dataset and enriches it but also effectively enhances the model's interpretability and generalization ability^[28]. The A1 and A3 temperatures are closely related to the phase transformation behavior, microstructure formation, and final property control of the alloys during rolling, and are crucial for developing rolling processes and achieving the desired properties. In this study, the final cooling process parameter was the coiling temperature. Therefore, at the coiling temperature, the driving force (DF) for the transformation of austenite to ferrite/pearlite and the volume fraction (VF) of ferrite/pearlite play decisive roles in determining the final properties. In summary, A1, A3, DF, and VF were incorporated as PM variables in the input features to guide the yield strength prediction. The above PM variables were calculated using the

TCFE9 database within the Thermo-Calc® software. Information on the PM variables for each cluster is shown in [Table 2](#).

Subsequently, a feature importance analysis was conducted on each sub-dataset by combining the composition and process features with PM variables. The importance of these features was assessed by calculating the Pearson correlation coefficient and mean decrease accuracy (MDA) values to evaluate the relationship between the PM variables and the yield strength, as shown in [Supplementary Figure 3](#).

In this study, the GCN algorithm was used to build a property prediction model using a knowledge graph as the input. Unlike traditional neural network models, GCN can capture more comprehensive graph structure information, which may be useful for processing high-dimensional and highly coupled feature information of the composition, process, and PM parameters. The architecture of the GCN is illustrated in the prediction module shown in [Figure 2](#). Initially, the input features were passed through two hidden layers to extract feature information, with output dimensions of 256 and 128 for the respective layers. Subsequently, the fully connected layer outputted the prediction results. Sigmoid and Adam are used as the activation function and optimizer, respectively, for the model training. In the GCN model, the input features first underwent feature transformation, converting the feature matrix into a sparse matrix. After symmetric normalization, the adjacency matrix was transformed into a sparse matrix. Next, through message passing, the feature and adjacency matrices were multiplied to obtain information about adjacent nodes. Finally, the graph was mapped to the predicted values using a fully connected layer. In addition, a conventional neural network (CNN) model was established for comparison to highlight the key role of knowledge graphs in performance prediction. For the CNN model, the input features were reshaped into a matrix of size 6, which served as the input. The model then processed this input using a combination of convolutional blocks, pooling layers, and fully connected layers to generate the output. In addition, all four sub-datasets used stochastic gradient descent (SGD) as the optimizer. The above models were trained using an 8:2 split between the training and testing sets, with a random split performed ten times. In this study, the server utilized is equipped with an NVIDIA GeForce RTX 2080 Ti GPU, which includes 11 GB of graphics card memory, a 100 GB hard drive, and an Intel(R) Xeon(R) CPU E5-2678 v3 @ 2.50 GHz. The software environment consists of a Pytorch 1.8.1-Horovod system image, Python 3.7, and Pytorch 1.8.1. In addition, the mean absolute error (MAE) and effective ratio (ER) were used to evaluate the accuracy and reliability of the model, as given by

$$MAE = \frac{1}{n} \sum_{i=1}^n |f(x_i) - y_i| \quad (3)$$

$$ER = \frac{N_e}{N_{all}} \times 100\% \quad (4)$$

where n is the number of samples; $f(x_i)$ and y_i are the predicted and true values of the i -th sample; N_e is the amount of data within the specified error range ($< 9\%$); and N_{all} is the total number of samples.

RESULTS AND DISCUSSION

Automatic data classification and reliability verification

As mentioned previously, an automatic data classification model based on the k-means algorithm was established. The effectiveness of k-means clustering largely depends on the selection of the K value. In this study, the optimal number of clusters was selected based on the SSE and SC to achieve the optimal classification, as shown in [Figure 3A](#). With an increase in the K value, the partition of samples in the dataset becomes finer, leading to a gradual decrease in the SSE value. In the line chart of the SSE, if the K value is lower than the optimal cluster number, increasing the K value will make the SSE decrease and the decline

Table 2. The PM variables of each subdataset

	A1/K	A3/K	DF/J	VF/%
Cluster1	973.8-996.7	1090.9-1141.6	252.7-1948.8	96.9-98.9
Cluster2	973.4-1000.9	1121.9-1169.1	316.8-2046.7	98.3-99.9
Cluster3	967.9-1000.7	1085.0-1163.6	287.2-2207.5	96.9-99.9
Cluster4	950.9-982.5	1081.1-1143.6	310.4-1662.2	96.9-99.2

PM: Physical metallurgy.

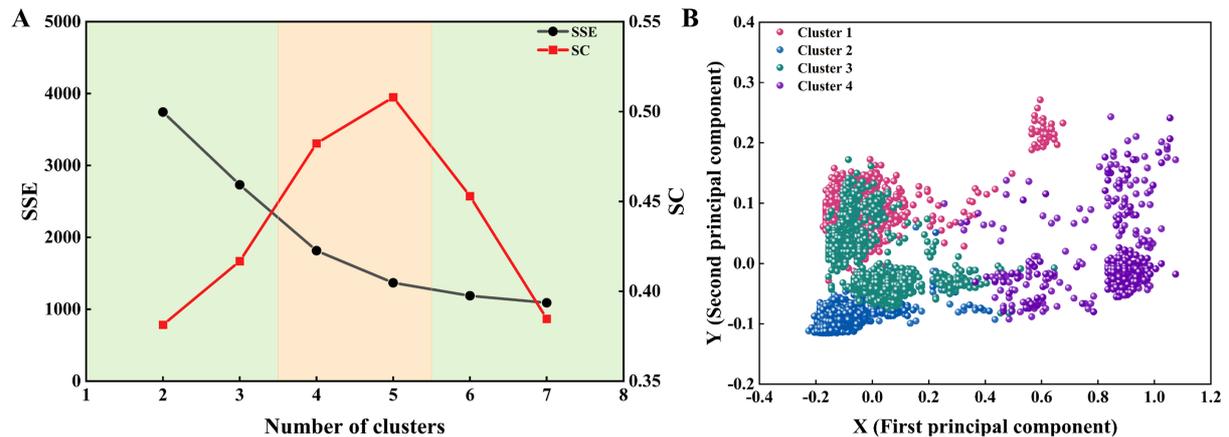


Figure 3. Classification overview based on k-means classification model. (A) Evaluation of optimal cluster number; (B) Data distribution for each category in 4-cluster classification.

speed faster; however, when the K value approaches the optimal number of clusters, the return of the aggregation degree caused by increasing the K value will decrease rapidly; therefore, the decline rate will slow down sharply and then tend to be stable with a further increase in the K value. The K value corresponding to this turning point was the true cluster number. According to the SSE curve in Figure 3A, the SSE decreases by approximately 900 when K increases from 3 to 4, whereas it only decreases by approximately 400 when K grows from 4 to 5. Therefore, when the number of clusters is four or five, the decline rate of the SSE changes from fast to slow, indicating that four or five may be the best number of clusters. The SC represents the clarity of each category after clustering. Therefore, the larger the SC value, the better the clustering effect. From the SC curve in Figure 3A, the value of SC is relatively large when the K values are 4 and 5. Based on these evaluations, it is reasonable to cluster the dataset into four or five groups. Therefore, the initial dataset was divided into four and five clusters, respectively. The data distributions for each steel grade within each cluster after classification are shown in Tables 3 and 4. To ensure a balanced data distribution between clusters and a concentrated distribution within each steel grade, a four-cluster classification method was adopted. Furthermore, a data classification model based on the hierarchical clustering algorithm was established to validate the robustness of the k-means classification model. According to the alloy composition characteristics in the initial dataset, it was divided into four subdatasets. The classification results of the hierarchical clustering model were then compared with those of the k-means model. The results revealed an 85.95% consistency in classification between the two models. This indicates that the classification outcomes of the two models are highly similar for the dataset in this study, further demonstrating that the k-means model exhibits strong stability and reliability in clustering the initial dataset. This, to a significant extent, confirms its robustness and justifies the rationality of the proposed method. PCA was used to map the 12-dimensional component feature information of each

Table 3. Data distribution of each steel grade under the 4-cluster classification

	Cluster1	Cluster2	Cluster3	Cluster4
Steel I	8452	84	966	252
Steel II		562	3124	
Steel III	155		3037	
Steel IV				1418
Steel V	58		1212	143
Steel VI		5529		
Steel VII		1173		
All Steel	8665	7348	8339	1813

Table 4. Data distribution of each steel grade under the 5-cluster classification

	Cluster1	Cluster2	Cluster3	Cluster4	Cluster5
Steel I	8446	69	967	248	24
Steel II		477	3209		
Steel III	183		3005		4
Steel IV				1418	
Steel V	58		2	143	1210
Steel VI		5529			
Steel VII		1173			
All Steel	8687	7248	7183	1809	1238

sample in each cluster to a two-dimensional feature space and observe the data distribution of each cluster. The values of X and Y were used to represent the two coordinates after dimensionality reduction, and different colors were used to represent the four subdatasets, as shown in Figure 3B. The classification results revealed that the k-means algorithm using the component features of the dataset as inputs successfully achieved automatic data classification. The samples in the dataset are divided into four clusters with clear boundaries to demonstrate effective classification outcomes. Unlike the KNN classification algorithm, there is no need to add labels to the samples, thereby ensuring a high degree of objectivity. As for steel grade, although all seven steel grades are low-alloy steels, they have different elemental compositions. Additionally, there were variations in the elemental composition, even among samples of the same steel grade. Consequently, samples of the same steel grade appeared in multiple clusters in the classification results. From the perspective of composition, the four clusters displayed distinct features. Clusters 1 and 3 had similar elemental contents, and Cluster3 had a higher Ti content. Compared to the other clusters, Cluster2 had lower C and Si contents, whereas Cluster4 exhibited higher Mn content. In summary, automatic classification of datasets was achieved through unsupervised clustering strategies, laying a good foundation for the subsequent establishment of prediction models.

To demonstrate the advantages of the proposed framework, which integrates automatic data classification with property prediction, the property prediction results of the models with and without data classification were compared. Two modeling approaches were employed for this purpose - Non-Classification and Classification Approaches. In the first, a dataset was created using all samples from seven steel grades without any data classification, and a property prediction model was built directly, referred to as "Non-Classification". In the second, a model was built using the method proposed in this study, which combined automatic data classification with property prediction, referred to as "Classification". The k-means

algorithm works by iteratively updating the cluster centers and randomly partitioning the dataset into four clusters. Its objective is to maximize intra-cluster similarity and minimize inter-cluster similarity until convergence is achieved. The same input feature selection method was used for both Non-Classification and Classification. In other words, the Pearson correlation coefficients of component features in Non-Classification were calculated, and dimension reduction was performed on features with low correlation with the target performance, such as S, P, and Cu, as shown in [Supplementary Figure 4](#). Subsequently, the property prediction models for both approaches were developed using the GCN algorithm.

The property prediction results for both the Non-Classification and Classification approaches are shown in [Figure 4](#). First, as shown in [Figure 4A](#), the MAEs for all seven steel grades in the testing set were consistently lower when using the classification approach than when using the Non-Classification approach. Second, for ER, as shown in [Figure 4B](#), the values for all seven steel grades in the Classification approach were higher than those in the Non-Classification approach. This indicates that the property prediction model based on automatic data classification demonstrates greater accuracy and reliability.

The data volumes of the seven steel grades in this study varied, with Steels IV, V, and VII having relatively small volumes [[Figure 1A](#)]. In addition, the data volume of each steel grade in the four sub-datasets also varied significantly, as shown in [Table 3](#). However, compared to Non-Classification, Classification significantly improved the prediction accuracy for the seven steel grades, indicating that the property prediction model based on automatic data classification established in this study ensured the prediction effect of steel grades in the dataset.

Property prediction based on GCN

After the initial dataset was divided into four subdatasets, the GCN algorithm was used to establish a property prediction model for each subdataset. PM variables were introduced into each sub-dataset to guide the modeling process. Because different knowledge graphs represent different logical relationships between the input features, selecting an appropriate knowledge graph is particularly important. Therefore, the best knowledge graph structure was selected for each sub-dataset. To explore the impact of the knowledge graph structure on prediction performance, the GCN was also compared with a traditional CNN algorithm.

The two models were compared by calculating the MAE and ER values of the testing set. The property prediction results are shown in [Figure 5](#). Across all four subdatasets, the GCN model consistently outperformed the CNN model in terms of both MAE and ER, thus achieving higher accuracy and reliability. In addition, both the GCN and CNN showed similar trends in the property prediction results. Compared with other clusters, the MAE of Cluster1 and Cluster2 was larger and the ER was smaller, especially in Cluster2, whose MAE and ER were the worst among the four clusters. To highlight the significance of PM parameters, a GCN model excluding PM parameters was constructed to predict yield strength, and its predictive performance was worse than that of the GCN model with PM parameters incorporated, as depicted in [Supplementary Figure 5](#). Subsequently, the correlation between PM parameters and yield strength was further investigated for samples with higher yield strength in the dataset, with the results aligning with physical metallurgical principles^[35], as illustrated in [Supplementary Figure 6](#). Moreover, to further substantiate the superiority of the GCN model, a performance prediction model based on MLP was employed to predict yield strength. As shown in [Supplementary Figure 7](#), the GCN model showed better performance than the MLP model in terms of both MAE and ER. In addition, the training times for the GCN models of the four subdatasets were approximately 90, 80, 90, and 30 mins, respectively, demonstrating the efficiency of the GCN model. Additionally, the GCN model exhibits rapid real-time application capabilities, with the trained model requiring only about ten seconds to complete real-time

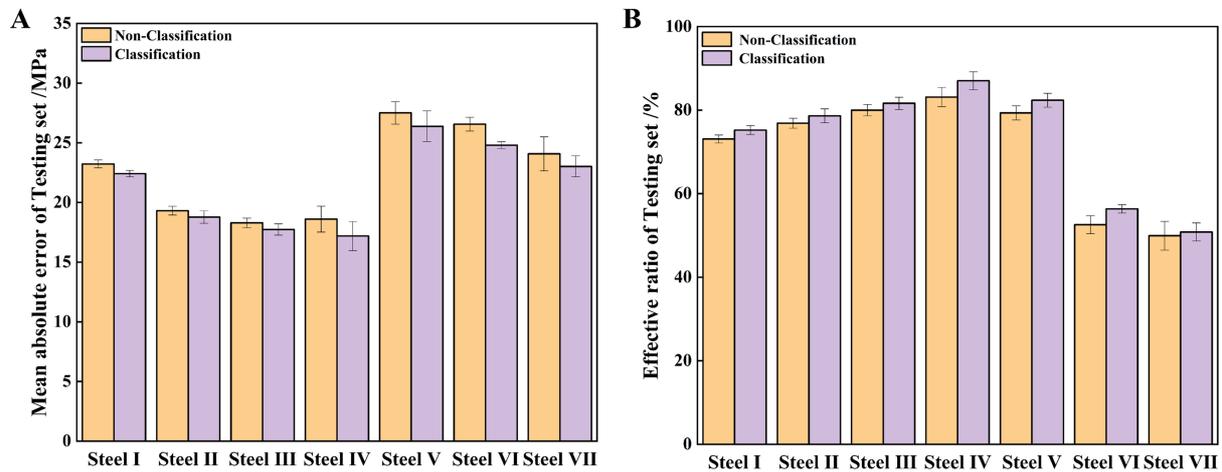


Figure 4. Reliability verification results of classification model. (A) Results of MAE; (B) Results of ER. MAE: Mean absolute error; ER: effective ratio.

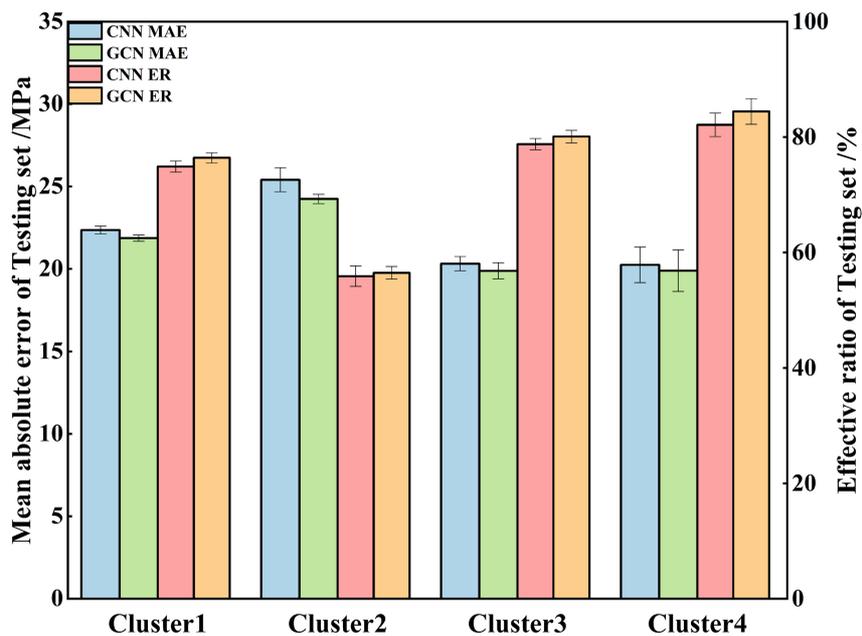


Figure 5. Comparison of prediction results for different neural network models.

prediction tasks. This further highlights its efficiency and practicality in real-world applications. In addition, a comprehensive analysis of the input features was conducted using the SHapley Additive exPlanations (SHAP) method to elucidate their respective contributions, as presented in [Supplementary Figure 8](#). Moreover, the prediction results were further analyzed using statistical methods (confidence intervals), as presented in [Supplementary Table 2](#). The analysis revealed that the GCN model significantly outperformed the CNN model in terms of MAE and ER.

The above results show that the strategy of combining automatic classification and the GCN achieves a significant improvement in predictive ability. In this section, the predictive ability of this strategy for each steel grade in the dataset is further investigated.

The MAE and ER for each steel grade were then calculated using both the GCN and CNN models, as shown in [Figure 6](#). The GCN model demonstrated a reduction in MAE for six of the seven steel grades compared with the CNN model, except for Steel V. However, for the ER, the GCN model yielded higher values across all seven grades compared with the CNN model. The sample points of Steel V were scattered, as shown in [Figure 1B](#). This indicates that there are significant differences among the components of the Steel V samples, which may hinder the effective learning of the relationship between nodes in the knowledge graph using the GCN model, resulting in a higher MAE of Steel V. In addition, to explore the reasons for the relatively low performance prediction results of the GCN model for Steel V, SHAP and MDA were used for feature importance analysis, as shown in [Supplementary Figure 9](#). Compared with the SHAP and MDA results of Cluster3 [[Supplementary Figure 8C and 3F](#)], the SHAP values of A1 and A3 in Steel V are lower, while the MDA score of Ti in Cluster3 is higher. This may be due to the fact that the outliers in Steel V have led to changes in the contribution degrees of some features.

Similarly, the GCN and CNN show similar trends in predicting the properties of different steel grades. Compared with other steel grades, the MAE values of Steels I, V, VI, and VII were higher. However, the ER values of Steels VI and VII were lower. From [Table 3](#), we can see that Steel I was mainly distributed in Cluster1, Steel V was mainly distributed in Cluster3, and Steels VI and VII were distributed in Cluster2. Therefore, Cluster1 and Cluster2 had higher MAE values and Cluster2 has lower ER values. Because the proportion of steel V in Cluster3 was small, the prediction result for Cluster3 was still good. This is consistent with the predicted results shown in [Figure 5](#).

The above results show that when considering individual steel grades or the entire subdataset, the GCN model combined with the knowledge graph outperformed the CNN model in predicting properties and offered higher interpretability. Therefore, the knowledge graph structure, which considers the complex logical relationships between the composition, processes, and PM variables, plays a significant role in property prediction. In particular, regarding the introduction of PM variables into the CNN model, the numerical values of the PM variables were introduced into the input features to guide model prediction. For the GCN model, PM variables were incorporated in the form of knowledge graphs, which not only included the numerical values of PM variables but also the logical relationships between PM variables, composition, and process features. This approach makes it easier for the model to recognize the importance of the PM variables and better guide the prediction process. Additionally, different knowledge graphs represented various ways of introducing PM variables, which helped reveal the inherent relationships between PM variables and other input features, thereby improving the interpretability of the model. Overall, compared with the traditional CNN model, the GCN model offers better prediction accuracy and interpretability.

In addition, designing the composition and processing parameters based on the trained model is crucial for achieving the desired properties. Therefore, a performance prediction model was established using the GCN algorithm and high yield strength grade steels to predict the yield strength and study the influence of alloying elements on the mechanical properties. This lays a foundation for achieving excellent properties in the future, as shown in [Supplementary Figure 10](#).

Different knowledge graphs

The knowledge graph can clearly illustrate the relationships among alloy compositions, processing parameters, and PM variables, thereby enhancing the interpretability of the predictions. In addition, the advantage of knowledge graphs is not only in introducing the logical relationship between the input features and PM variables in the model, but also in enabling the selection of different knowledge graphs as inputs for property prediction based on the same dataset. To clarify the importance of knowledge graphs that combine

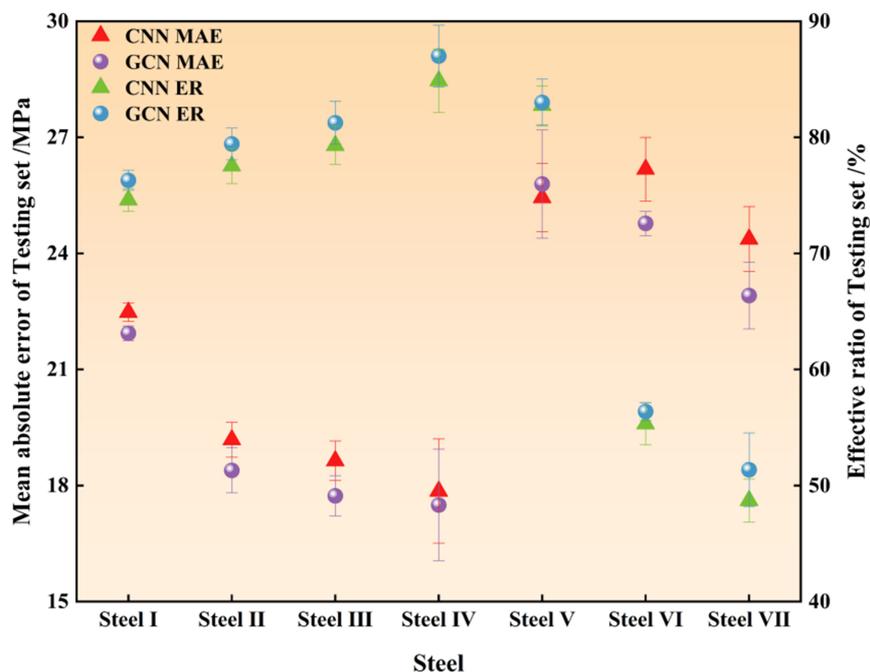


Figure 6. Comparison of property prediction results for different steel grades between CNN and GCN. CNN: Convolutional neural network; GCN: graph convolutional network.

accuracy and logical relationships, multiple knowledge graph-trained GCN models were constructed, and the property prediction of steel grades under different knowledge graphs was discussed in detail. For this purpose, property predictions were made for steel grades in four subdatasets, and different knowledge graphs were used for comparison. Four distinct knowledge graphs were constructed for the four subdatasets to perform performance prediction, as illustrated in Figure 7. The criteria for selecting edges in the knowledge graphs were based on a combination of domain expertise and Pearson correlation coefficients. Initially, all alloy composition features involved in the calculation of PM parameters were connected to their respective PM parameters, resulting in PM-Graph1. Subsequently, the edges in the knowledge graphs were further refined based on Pearson correlation coefficients. Specifically, for each subdataset, the Pearson correlation coefficients between alloy composition features and each PM parameter were calculated, as shown in Supplementary Figure 11. For Cluster1, the top-three, top-two, and top-one alloy composition features most correlated with A1 and VF were selected and connected accordingly. Notably, due to the strong correlation between Mn and DF in Cluster1, only Mn was connected to DF. Following these selections, PM-Graph2 to PM-Graph4 were constructed, as depicted in Figure 7A. For Clusters 2-4, the same approach was used: the top-three, top-two, and top-one alloy composition features most correlated with each PM parameter were selected to construct PM-Graph2 to PM-Graph4, as shown in Figure 7B-D. Additionally, CT_tmp_tail (CT), the temperature parameter used to calculate DF and VF, was consistently connected to both DF and VF in all knowledge graphs across the four subdatasets. Notably, the knowledge graph contains three colored edges: green, red, and black. The green edge is associated with the alloy element composition characteristics that have the highest absolute Pearson correlation coefficient with the PM variable. The red edge is associated with the second - highest correlation, and the black edge is associated with the third - highest correlation. DF and VF are calculated under CT conditions and are connected to CT through a green edge. This approach was applied to all four sub-datasets. Additionally, the composition features that were not connected to the PM variables were treated as individual nodes, whereas the process features were connected based on the order of production. A knowledge graph of the process

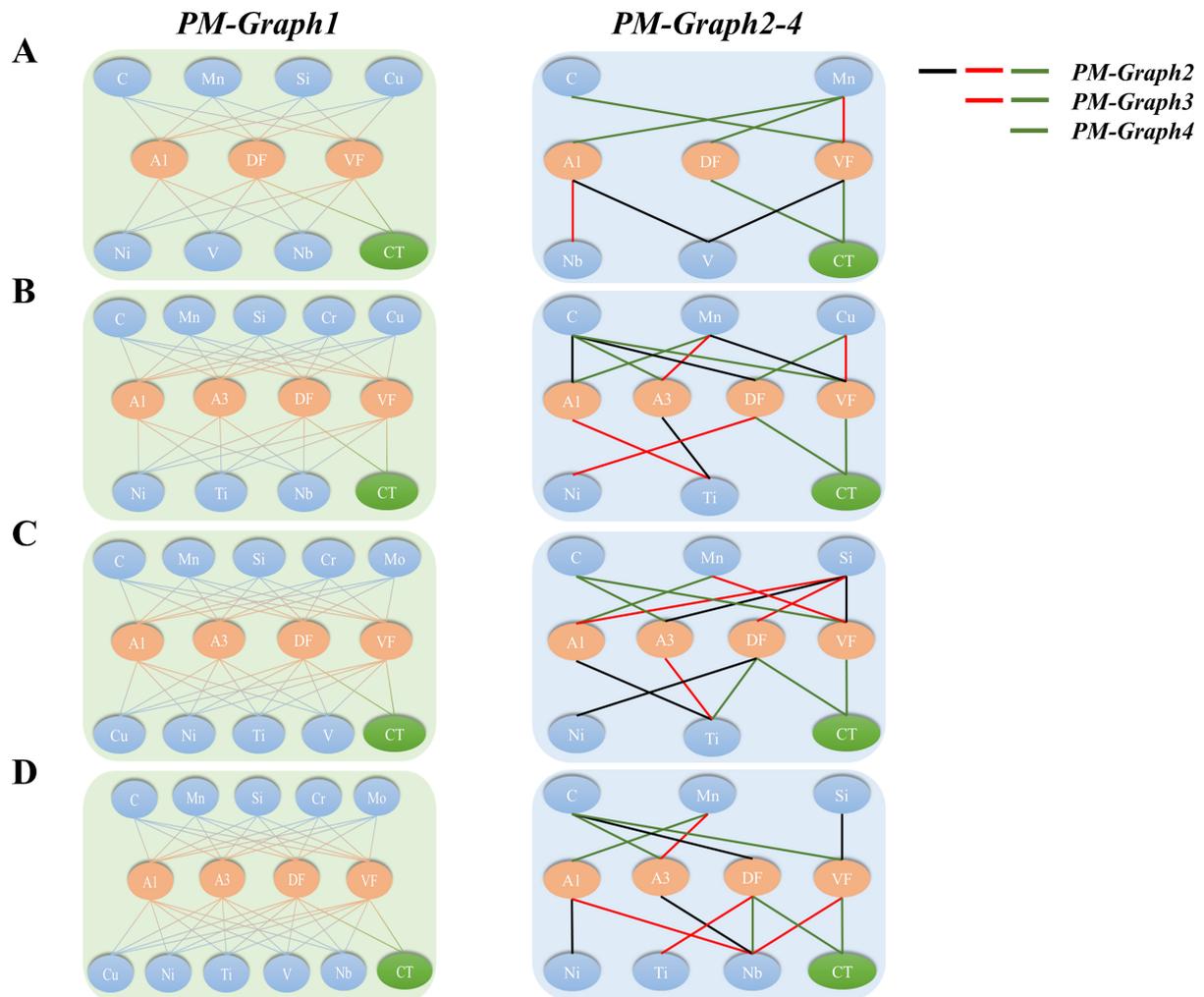


Figure 7. Knowledge graph depicting each cluster obtained through professional knowledge and Pearson correlation coefficient analysis. (A) GCN-PM-Graph1-4 of Cluster1; (B) GCN-PM-Graph1-4 of Cluster2; (C) GCN-PM-Graph1-4 of Cluster3; (D) GCN-PM-Graph1-4 of Cluster4. GCN: Graph convolutional network; PM: physical metallurgy.

features is presented in [Supplementary Figure 12](#). Each knowledge graph was used to train the GCN model ten times, with an 8:2 random split between the training and testing sets. The MAE and ER values computed for different knowledge graphs are shown in [Figure 8](#).

In Cluster1, as the knowledge graph changed, the property prediction results of Steel I changed slightly, whereas those of Steels III and V changed significantly. After a comprehensive analysis of the property prediction results of Steels III and V, the optimal knowledge graph for Cluster1 was PM-Graph3, as shown in [Figure 8A](#). This indicates that the PM variables in Cluster1 were mainly influenced by C, Mn, and Nb. In Cluster2, as the knowledge graph changes, the property prediction results of Steels VI and VII change slightly, whereas those of Steels I and II change significantly. After a comprehensive analysis of the property prediction results of Steels I and II, the optimal knowledge graph for Cluster2 was PM-Graph1, as shown in [Figure 8B](#). This indicates that the reduction in the number of edges in Cluster2 leads to a decrease in the model prediction accuracy. In Cluster3, as the knowledge graph changed, the property prediction results of Steels II and III changed slightly, whereas those of Steels I and V changed significantly. After a comprehensive analysis of the property prediction results for Steels I and V, the optimal knowledge graph

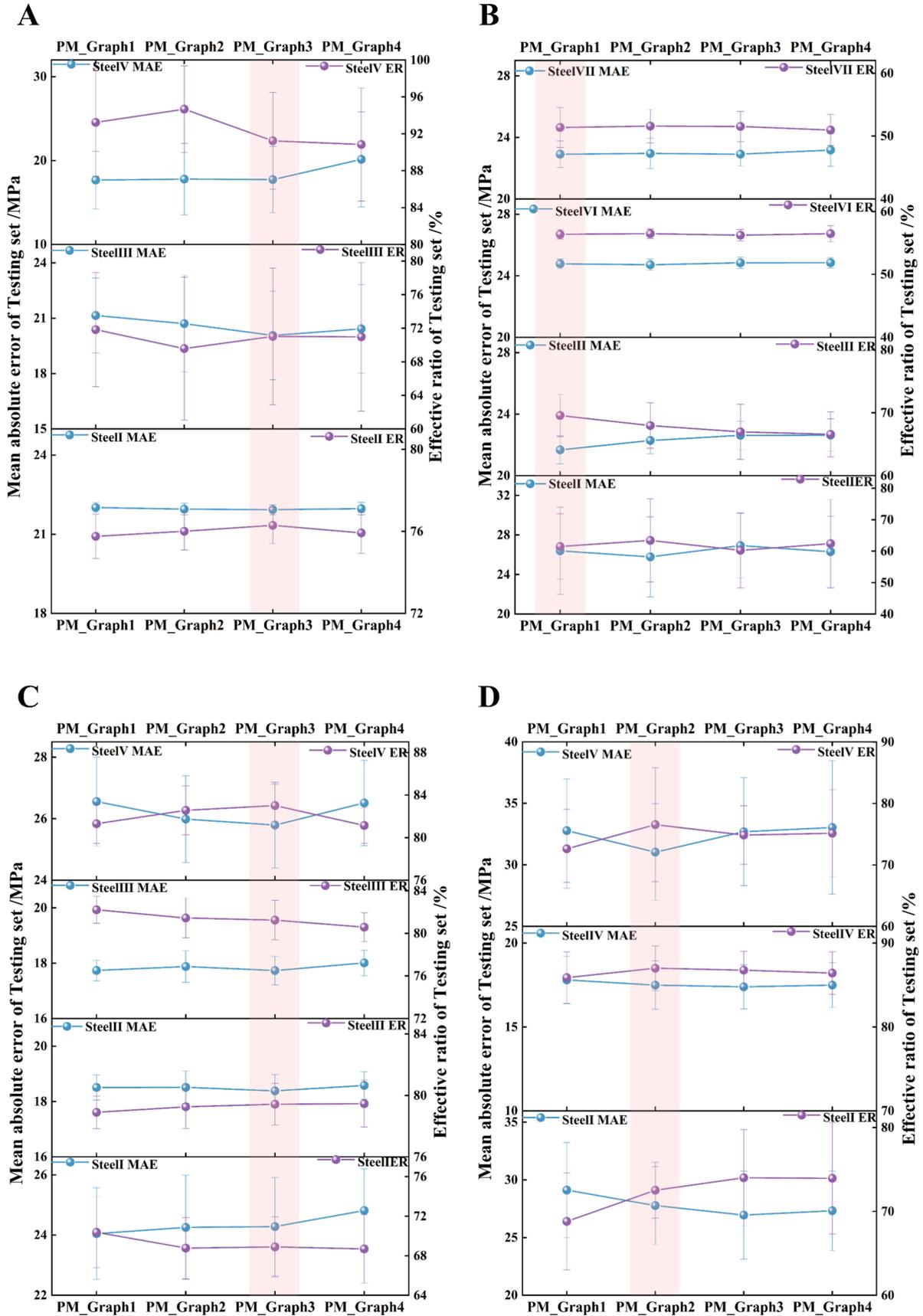


Figure 8. Comparison of property prediction results for different knowledge graphs of different steel grades. (A) MAE and ER of Cluster1; (B) MAE and ER of Cluster2; (C) MAE and ER of Cluster3; (D) MAE and ER of Cluster4. MAE: Mean absolute error; ER: effective ratio.

for Cluster3 was PM-Graph3, as shown in [Figure 8C](#). This indicates that the PM variables in Cluster3 were mainly influenced by C, Mn, Si, and Ti, and the increase or decrease in edges reduced the prediction accuracy of the model. In Cluster4, as the knowledge graph changed, the property prediction results of Steel IV changed slightly, whereas those of Steels I and V changed significantly. After a comprehensive analysis of the property prediction results for Steels I and V, the optimal knowledge graph for Cluster4 was PM-Graph2, as shown in [Figure 8D](#). This indicates that the PM variables in Cluster4 were mainly influenced by C, Mn, Si, Ni, Ti, and Nb, and the reduction in edges lowered the prediction accuracy of the model. Therefore, the optimal knowledge graphs selected for the four sub-datasets were PM-Graph3, PM-Graph1, PM-Graph3, and PM-Graph2.

To further demonstrate the superiority of the optimal knowledge graph selected in each subdataset in this study, a unified knowledge graph was constructed for the four subdatasets for performance prediction, as shown in [Supplementary Figure 13](#). Common alloy composition features and PM parameters, such as C, Mn, Si, Cu, Ni, A1, DF, and VF, were selected as input features for the four subdatasets. These selected features were interconnected to establish a unified knowledge graph for the subdatasets, as depicted in [Supplementary Figure 13A](#). DF and VF were also linked to CT in the knowledge graph. Performance predictions were then conducted using the GCN model and compared with the predictions from the optimal knowledge graph selected for each subdataset. The comparison results are displayed in [Supplementary Figure 13B](#). The research results indicate that the property prediction results of the optimal knowledge graph surpass those of the unified knowledge graph. This underscores the importance of constructing suitable knowledge graphs based on the characteristics of each subdataset.

In summary, for the steel grades in the four sub-datasets, incorrect connections between nodes reduced the predictive ability of the model. Therefore, it is particularly important to construct an accurate and reasonable knowledge graph. A suitable knowledge graph can not only improve the prediction accuracy and interpretability of machine learning but can also reverse analyze the logical relationship between features and PM variables in the prediction process.

CONCLUSIONS

(1) This study establishes an automatic data classification model based on the k-means algorithm using the compositional features of the initial dataset after data preprocessing as input features. The optimal number of clusters was determined through evaluation indicators (SSE and SC) and data distribution, thus achieving automatic classification of the industrial big dataset and successfully dividing it into four clusters. In addition, compared to the unclassified model, the model based on automatic data classification had a higher prediction accuracy and reliability.

(2) Compared with the CNN model, the property prediction model based on interpretable knowledge graphs and GCN had higher prediction accuracy and interpretability. The PM variables were introduced into the model in the form of knowledge graphs, incorporating not only the values of the PM variables but also the logical relationships between the PM variables, composition, and process features. This approach enhances the model's ability to recognize the importance of PM variables, thereby better guiding the model's properties.

(3) A reasonable knowledge graph is crucial for training the model. Incorrect connections between the nodes in a knowledge graph can reduce the predictive ability of the model. A suitable knowledge graph can not only improve the prediction accuracy and interpretability of machine learning but can also reverse analyze the logical relationship between features and PM variables in the prediction process.

DECLARATIONS

Authors' contributions

Made substantial contributions to conception and design of this review, writing and editing: Ren, B.; Wei, X.; Wang, C.; Zhang, Y.; Xu, W.

Made substantial contributions to collation of literature, figure preparation, and writing: Ren, B.; Wei, X.

Performed data analysis, discussion and review writing: Ren, B.; Wang, C.; Wei, X.

Provided administrative, technical, and material support: Wang, C.; Wei, X.; Zhang, Y.; Xu, W.

Availability of data and materials

Research data are not shared.

Financial support and sponsorship

The research was financially supported by the National Key R&D Program (Grant Number 2022YFB3304805) and the Postdoctoral Fellowship Program of CPSF under Grant Number GZC20240222. The financial support provided by the China Postdoctoral Science Foundation (2024M750370) is gratefully acknowledged.

Conflicts of interest

All authors declared that there are no conflict of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

© The Author(s) 2025.

REFERENCES

1. Zhang, Q.; Wang, Y. Research on mechanical property prediction of hot rolled steel based on lightweight multi-branch convolutional neural network. *Mater. Today. Commun.* **2023**, *37*, 107445. DOI
2. Liu, S.; Long, M.; Zhang, S.; et al. Study on the prediction of tensile strength and phase transition for ultra-high strength hot stamping steel. *J. Mater. Res. Technol.* **2020**, *9*, 14244-53. DOI
3. dos Santos, A. A.; Barbosa, R. Model for microstructure prediction in hot strip rolled steels. *Steel. Res. Int.* **2010**, *81*, 55-63. DOI
4. Han, H. N.; Lee, J. K.; Kim, H. J.; Jin, Y. A model for deformation, temperature and phase transformation behavior of steels on run-out table in hot strip mill. *J. Mater. Process. Technol.* **2002**, *128*, 216-25. DOI
5. Zhang, X.; Jiang, Z.; Tieu, A.; Liu, X.; Wang, G. Numerical modelling of the thermal deformation of CVC roll in hot strip rolling. *J. Mater. Process. Technol.* **2002**, *130-1*, 219-23. DOI
6. Militzer, M.; Hawbolt, E. B.; Meadowcroft, T. R. Microstructural model for hot strip rolling of high-strength low-alloy steels. *Metall. Mater. Trans. A.* **2000**, *31*, 1247-59. DOI
7. Wei, X.; van, Z. S.; Jia, Z.; Wang, C.; Xu, W. On the use of transfer modeling to design new steels with excellent rotating bending fatigue resistance even in the case of very small calibration datasets. *Acta. Mater.* **2022**, *235*, 118103. DOI
8. Lee, J.; Kim, M.; Lee, Y. Design of high strength medium-Mn steel using machine learning. *Mater. Sci. Eng. A.* **2022**, *843*, 143148. DOI
9. Abd-Elaziem, W.; Elkattatny, S.; Sebaey, T. A.; Darwish, M. A.; Abd, E. M. A.; hamada, A. Machine learning for advancing laser

- powder bed fusion of stainless steel. *J. Mater. Res. Technol.* **2024**, *30*, 4986-5016. DOI
10. Geng, X.; Wang, F.; Wu, H.; et al. Data-driven and artificial intelligence accelerated steel material research and intelligent manufacturing technology. *MGE. Advances.* **2023**, *1*, e10. DOI
 11. Zhu, L.; Luo, Q.; Chen, Q.; et al. Prediction of ultimate tensile strength of Al-Si alloys based on multimodal fusion learning. *MGE. Advances.* **2024**, *2*, e26. DOI
 12. Shi, Z.; Du, L.; He, X.; et al. Prediction model of yield strength of V-N steel hot-rolled plate based on machine learning algorithm. *JOM.* **2023**, *75*, 1750-62. DOI
 13. Xu, G.; He, J.; Lü, Z.; Li, M.; Xu, J. Prediction of mechanical properties for deep drawing steel by deep learning. *Int. J. Miner. Metall. Mater.* **2023**, *30*, 156-65. DOI
 14. He, X.; Zhou, X.; Tian, T.; Li, W. Prediction of mechanical properties of hot rolled strips with generalized RBFNN and composite expectile regression. *IEEE. Access.* **2022**, *10*, 106534-42. DOI
 15. Xu, Z.; Liu, X.; Zhang, K. Mechanical properties prediction for hot rolled alloy steel using convolutional neural network. *IEEE. Access.* **2019**, *7*, 47068-78. DOI
 16. Jiang, X.; Jia, B.; Zhang, G.; et al. A strategy combining machine learning and multiscale calculation to predict tensile strength for pearlitic steel wires with industrial data. *Scr. Mater.* **2020**, *186*, 272-7. DOI
 17. Guo, S.; Yu, J.; Liu, X.; Wang, C.; Jiang, Q. A predicting model for properties of steel using the industrial big data based on machine learning. *Comput. Mater. Sci.* **2019**, *160*, 95-104. DOI
 18. Li, W.; Xie, L.; Zhao, Y.; Li, Z.; Wang, W. Prediction model for mechanical properties of hot-rolled strips by deep learning. *J. Iron. Steel. Res. Int.* **2020**, *27*, 1045-53. DOI
 19. Xie, Q.; Suvarna, M.; Li, J.; Zhu, X.; Cai, J.; Wang, X. Online prediction of mechanical properties of hot rolled steel plate using machine learning. *Mater. Design.* **2021**, *197*, 109201. DOI
 20. Yang, Z.; Wang, Y.; Xu, F.; et al. Online prediction of mechanical properties of the hot rolled steel plate using time-series deep neural network. *ISIJ. Int.* **2023**, *63*, 746-57. DOI
 21. Cui, C.; Cao, G.; Li, X.; Gao, Z.; Liu, J.; Liu, Z. A strategy combining machine learning and physical metallurgical principles to predict mechanical properties for hot rolled Ti micro-alloyed steels. *J. Mater. Process. Technol.* **2023**, *311*, 117810. DOI
 22. Li, F.; He, A.; Song, Y.; et al. Deep learning for predictive mechanical properties of hot-rolled strip in complex manufacturing systems. *Int. J. Miner. Metall. Mater.* **2023**, *30*, 1093-103. DOI
 23. Li, H.; Li, Y.; Huang, J.; et al. Physical metallurgy guided industrial big data analysis system with data classification and property prediction. *Steel. Res. Int.* **2022**, *93*, 2100820. DOI
 24. Thomas, A.; Durmaz, A. R.; Alam, M.; Gumbsch, P.; Sack, H.; Eberl, C. Materials fatigue prediction using graph neural networks on microstructure representations. *Sci. Rep.* **2023**, *13*, 12562. DOI PubMed PMC
 25. Dai, M.; Demirel, M. F.; Liang, Y.; Hu, J. Graph neural networks for an accurate and interpretable prediction of the properties of polycrystalline materials. *npj. Comput. Mater.* **2021**, *7*, 574. DOI
 26. Sadeghpour, E.; Nonn, A. Data-driven models for structure-property prediction in additively manufactured steels. *Comput. Mater. Sci.* **2022**, *215*, 111782. DOI
 27. Karimi, K.; Salmenjoki, H.; Mulewska, K.; et al. Prediction of steel nanohardness by using graph neural networks on surface polycrystallinity maps. *Scr. Mater.* **2023**, *234*, 115559. DOI
 28. Li, Y.; Wang, C.; Zhang, Y.; et al. Thermodynamically informed graph for interpretable and extensible machine learning: martensite start temperature prediction. *Calphad* **2024**, *85*, 102710. DOI
 29. Shi, X.; Zhou, L.; Huang, Y.; Wu, Y.; Hong, Z. A review on the applications of graph neural networks in materials science at the atomic scale. *MGE. Advances.* **2024**, *2*, e50. DOI
 30. Celebi, M. E.; Kingravi, H. A.; Vela, P. A. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert. Syst. Appl.* **2013**, *40*, 200-10. DOI
 31. Liu, Y.; Wu, J.; Wang, Z.; et al. Predicting creep rupture life of Ni-based single crystal superalloys using divide-and-conquer approach based machine learning. *Acta. Mater.* **2020**, *195*, 454-67. DOI
 32. Pourbahrami, S.; Balafar, M. A.; Khanli, L. M.; Kakarash, Z. A. A survey of neighborhood construction algorithms for clustering and classifying data points. *Comput. Sci. Rev.* **2020**, *38*, 100315. DOI
 33. Hussain, S. F.; Haris, M. A k-means based co-clustering (kCC) algorithm for sparse, high dimensional data. *Expert. Syst. Appl.* **2019**, *118*, 20-34. DOI
 34. Tzortzis, G.; Likas, A. The MinMax k-Means clustering algorithm. *Pattern. Recogn.* **2014**, *47*, 2505-16. DOI
 35. Tan, X.; Lu, W.; Rao, X. Effect of ultra-fast heating on microstructure and mechanical properties of cold-rolled low-carbon low-alloy Q&P steels with different austenitizing temperature. *Mater. Charact.* **2022**, *191*, 112086. DOI