Delft University of Technology

Multi-Source Domain Adaptation Method of Mill Load Based on Common and Special Characteristics

Liu, Yiwen; Yan, Gaowei; Li, Rong; Pang, Yusong; Qiao, Tiezhu

DOI
10.23919/CCC58697.2023.10240161

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the 42nd Chinese Control Conference, CCC 2023

Citation (APA)
Liu, Y., Yan, G., Li, R., Pang, Y., & Qiao, T. (2023). Multi-Source Domain Adaptation Method of Mill Load Based on Common and Special Characteristics. In *Proceedings of the 42nd Chinese Control Conference, CCC 2023* (pp. 6682-6688). (Chinese Control Conference, CCC; Vol. 2023-July). IEEE. https://doi.org/10.23919/CCC58697.2023.10240161

Important note
To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Green Open Access added to TU Delft Institutional Repository

'You share, we take care!' - Taverne project

https://www.openaccess.nl/en/you-share-we-take-care

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

Proceedings of the 42nd Chinese Control Conference
July 24-26, 2023, Tianjin, China

Multi-source Domain Adaptation Method of Mill Load Based on Common and Special Characteristics

Yiwen Liu[1], Gaowei Yan[1,*], Rong Li[1], Yusong Pang[2], Tiezhu Qiao[3]

1. School of Electrical and Power Engineering, Taiyuan University of Technology, Taiyuan 030024
E-mail: yangaowei@tyut.edu.cn

2. Section of Transport Engineering and Logistics, Delft University of Technology, Mekelweg 2, Netherlands
E-mail: Y.Pang@tudelft.nl

3. School of Physics and Optoelectronics, Taiyuan University of Technology, Taiyuan 030024
E-mail: qiaotiezhu@tyut.edu.cn

Abstract: In the grinding industry, accurate prediction of the mill load is the key to increasing mill income and reducing mill failure. It is difficult to improve the prediction accuracy of the model due to insufficient information on single-source domain data and distribution differences among different data. A multi-source domain unsupervised domain adaptation method based on common and special features is proposed. Multi-source domain data has both common and special characteristics. If only common features are emphasized, some useful information will be discarded. If only special features are used, the model generalization is not good. To solve this problem, a common feature extraction block is used to extract the common domain invariant representation of multiple source domains and target domains, and special features are obtained through the special feature extraction block. After the features are fused and input into the common regressor, the multi-source domain predicted values are obtained. Finally, the predicted values of multiple source domains are added and averaged to get the final prediction result. The effectiveness of this method is proved by cross-experiments on the ball mill data set collected in the laboratory.

Key Words: Multi-source domain, Common feature, Special feature, Wet ball mill, Mill load

1 Introduction

The grinding process is one of the important links in the mineral processing industry [1]. The wet ball mill is widely used in the grinding process. Mill load (ML) is an important parameter in the grinding process, including charge volume ratio (CVR), material to ball volume ratio (MBVR), and pulp density (PD) [2,3]. It is directly related to the quality and quantity of production. Accurate detection of ML is the key to improving grinding efficiency, improving grinding product quality, and avoiding no-load and overload faults [4]. However, the ball mill is closed and has the characteristics of continuous rotation, so the ML cannot be measured directly [5]. Fortunately, the strong mechanical vibration and sound signals of the ball mill contain a wealth of information about ML [6]. The model of ML can be established through the soft sensor. Literature [7] used the improved long short-term time memory (LSTM) network to build a soft sensor regression model to predict ML parameters. Literature [8] adopted the partial least squares regression (PLSR) algorithm as the soft sensor model and took the hidden features extracted from the vibration signals as the input to predict the load parameters of the ball mill.

The above soft sensor method requires data to satisfy the assumption of independent and homogeneous distribution. However, in the actual grinding process, the changes in ball mill operation tasks, set values, raw materials, and environment are easy to lead to inconsistent data distribution [9]. The emergence of domain adaptation provides an effective solution to the above problems. As one of the machine learning methods, domain adaptation mainly uses labeled data in one or more source domains to solve new tasks in target domains [10]. The methods of domain adaptation can be divided into three types: sample-based method, feature-based method, and inference-based method [11]. Literature [12] introduces DARWNN to achieve the purpose of migrating a small amount of labeled data in the target domain with the source domain data, so as to solve the problem of inconsistent data distribution between the source domain and the target domain. On this basis, manifold regularization is introduced to improve the prediction effect of ML. Literature [13] proposes a network JDMAN based on deep transfer learning, which combines source domain and target domain data for training. In the network, the central moment and center loss are used to align and cluster the features learned from the network, so as to improve the precision of the model.

The above domain adaptation methods are all single-source domain adaptation. Generally, the knowledge of the single-source domain is not enough to predict the target task and may result in overfitting [14]. Therefore, it is considered to carry multiple source domains with rich information to complete the target task. The common method of multi-source domain adaptation (MSDA) is to combine multiple source domains directly to form the single-source domain, and then use the method of the single-source domain for prediction. As more knowledge is included in the data, the prediction accuracy is improved, but the effect may not be significant [15]. Therefore, it is necessary to find a better way to take full advantage of knowledge from multiple source domains. In recent years, some MSDA methods map multiple source domains and target domains to the same feature space through neural networks and extract their common features. Common

*This work is supported by General Program of National Natural Science Foundation (NNSF) of China under Grant 61973226, National Natural Science Foundation (NNSF) of China under Grant 62003233, and the "select the best candidates to lead" science and technology major plan in Shanxi Province under Grant 202201090301013, and is in cooperation with the ARC Linkage Projects LP190100378.

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.

features help to discover the correlation between domains, and the model built by using them is more generalized. However, data from different domains not only have common characteristics but also have special characteristics [16]. If only common features are emphasized, less information is gained from multiple source domains, and some useful information is discarded [17]. Therefore, special features can be considered. The simpler method is to set up neural networks with different parameters for each source domain and target domain data directly. However, the model established by this method not only has many parameters but also takes a long time to calculate.

To sum up, we use a neural network to build the mill load multi-source domain adaptation model based on common and special features. The model includes three parts: common feature extraction block, special feature extraction block, and common regressor. Common feature extraction block and special feature extraction block are used to extract the common feature and special feature of multiple source domains respectively. After the features are fused, the mill load of the wet ball mill is predicted by the regression of shared parameters.

2 Preliminary

2.1 Problem description

Assume there are N source domains in multi-source domain unsupervised domain adaptation. Define the source domain as $X_i^s = \{x_{i,j}^s\} \in \mathbb{R}^{n_i^s \times m}$, $j = 1, 2, \cdots, n_i^s$, where n_i^s represents the number of samples of the i th source domain, and m represents the sample feature dimension. The source field also has a continuous label $Y_i^s \in \mathbb{R}^{n_i^s \times 1}$. Define the target domain as $X^t = \{x_j^t\} \in \mathbb{R}^{n^t \times m}$, where n^t represents the number of samples to be predicted by the target domain.

2.2 LSTM

As a special recurrent neural network (RNN), the long short-term memory network (LSTM) [18] consists of a forget gate, an input gate, an output gate, and a hyperbolic tangent layer, as shown in Fig. 1.

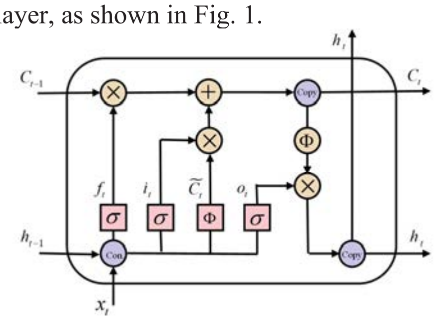


Fig. 1: LSTM structure diagram

Firstly, LSTM uses the input x_t at time t and the hidden state h_{t-1} at the previous time to obtain four states f_t, i_t, $\tilde{C}_t$ and o_t, which can be obtained by formula (1).

$$\begin{aligned} f_t &= \sigma\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right) \\ i_t &= \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \\ \widetilde{C}_t &= \phi\left(W_C \cdot \left[h_{t-1}, x_t\right] + b_C\right) \\ o_t &= \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \end{aligned} \quad (1)$$

Where σ is the sigmoid activation function, ϕ is the tanh activation function, $W = \left[W_f, W_i, W_C, W_o\right]$ is weight, and $b = \left[b_f, b_i, b_C, b_o\right]$ is the bias coefficient.

At this time, the cell state C_{t-1} and hidden state h_{t-1} of the previous moment can be updated into the new cell state C_t and hidden state h_t by formula (2).

$$\begin{aligned} C_t &= f_t * C_{t-1} + i_t * \widetilde{C}_t \\ h_t &= o_i * \phi\left(C_t\right) \end{aligned} \quad (2)$$

2.3 1D-CNN

Convolutional neural network (CNN), as a method of the neural network, is widely used in many fields [19]. CNN carries out convolution operations between the local region and the convolution kernel of input data at the convolution layer to extract local features of data and reduce complexity. One-dimensional convolution means that the convolution kernel only moves in one direction. The input is a vector and a convolution kernel, and the output is a vector.

3 Method

Our framework consists of three parts, which are the common feature extraction block, the special feature extraction block, and the common regressor. Among them, the common feature extraction block projects all source domain and target domain data into the same feature space, and extracts the common features of all domains after reducing the distribution differences. After obtaining the time sequence information through the information extraction block, the paired source domain data and target domain data are mapped to different feature spaces respectively to obtain the special features of multiple domains. Finally, the common features and special features are fused and input into the regression with shared parameters to obtain the predicted value of the target domain. The overall frame is shown in Fig. 2.

3.1 Common feature extraction block

The common feature extraction block maps all data in the original feature space into the same feature space using a network of shared parameters. In this way, the common domain invariant representation of all domains can be extracted. In this paper, LSTM is used as a common feature extraction block to capture dynamic features of data while extracting common features.

3.2 Special feature extraction block

Special feature extraction block includes information extraction block and N subnetworks that do not share network parameters. Each subnetwork is trained separately by source domain data. Firstly, all the source domain data and target domain data are input into the information extraction block to obtain the features containing timing information. Then, the source domain features and target domain features are mapped into different feature Spaces in pairs. By minimizing the divergence criterion, the special characteristics of the source domain are obtained. In this paper, LSTM is used as the information extraction module, and the two-layer 1D-CNN is used as the subnetwork to

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.

automatically extract effective features while reducing the feature dimension.

3.3 Common regressor

After obtaining common features and special features through the above two modules, weight is given to the features to obtain the fused features. The fusion formula is shown in formula (3).

$$f = \alpha f_c + (1-\alpha) f_e \tag{3}$$

Where, f is the feature after fusion, f_c is the common feature, f_e is the special feature, α is the weight.

After the fusion features are obtained, the predicted values of multiple source domains are obtained by inputting them into the common regressor with shared network parameters. The average of the multiple source domain predictions is the final prediction result. The regressor is constructed from the fully connected layer and its activation function is relu.

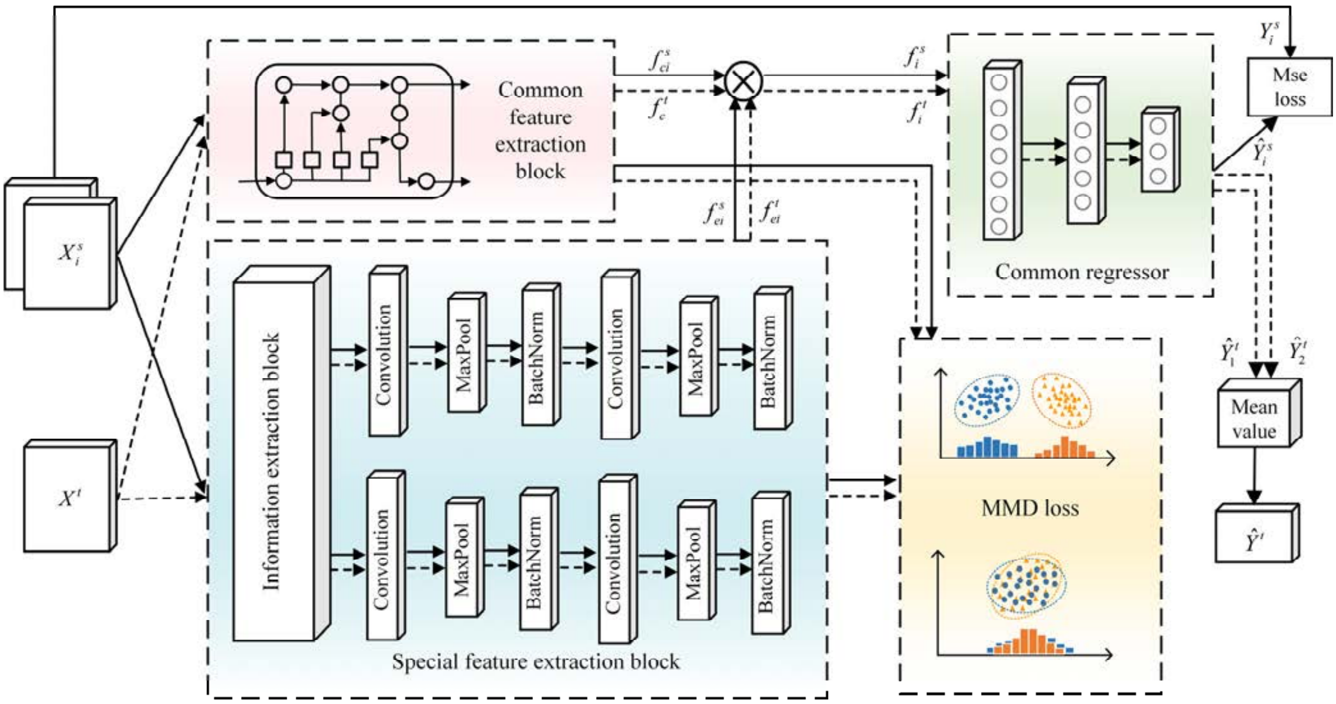


Fig. 2: Overall frame diagram

3.4 Loss function

To improve the prediction accuracy, we use regression loss L_m. Common regression losses include mean square error (MSE) loss, mean absolute error (MAE) loss, Huber loss, etc. The formula used in this paper is MSE loss, and its calculation formula is shown as formula (4).

$$L_m = \frac{1}{N}\frac{1}{n_i^s}\sum_{i=1}^{N}\sum_{j=1}^{n_i^s}(\hat{y}_{i,j}^s - y_{i,j}^s)^2 \tag{4}$$

Where $\hat{y}_{i,j}^s$ is the predicted value of the model and $y_{i,j}^s$ is the true value

Since the distribution of source and target domain data is different, the difference between them needs to be reduced. In general, a divergence-based approach can be used to reduce distribution differences. Common divergence methods include maximum mean difference (MMD) [20], covariance alignment (CORAL) [21], and Wasserstein distance [22]. Among them, MMD is the most commonly used divergence method in transfer learning. Compared with MMD, the calculation of CORAL is simpler, but this also leads to poor adaptability[23]. Wasserstein distance also has advantages in solving the problem of gradient disappearance and explosion and is often used in adversarial networks[24]. Since the research focus of this paper is on the multi-source migration framework, the adaptive divergence criterion is more conducive to the framework of this paper. Therefore, this paper chooses to minimize MMD loss to reduce the distribution difference between the source domain and the target domain. In MMD, we define the distance between the two distributions as the distance between the average embedded features. By finding the continuous function on the sample space, the mean of the function values of the samples with different distributions on the continuous function is obtained. The mean discrepancy is obtained by diverting the mean of the two data. MMD is the maximum discrepancy in the mean. The MMD loss is calculated by formula (5).

$$L_d = \frac{1}{N}\sum_{i=1}^{N}\left\| \frac{1}{n_i^s}\sum_{j=1}^{n_i^s}\phi(x_{i,j}^s) - \frac{1}{n^t}\sum_{j=1}^{n^t}\phi(x_{i,j}^t) \right\|_H^2 \tag{5}$$

Where H representing reproducing kernel Hilbert space (RKHS), $\phi(\cdot)$ represents the mapping function from the original feature space to RKHS.

Since there is a difference in the distribution of the input data of the common feature extraction block and the special feature extraction block, MMD loss is added to both modules. The calculation formula of the total loss function is shown in formula (6).

$$Loss = L_m + \beta L_d^c + \gamma L_d^e \tag{6}$$

Where, L_d^c is the MMD loss used in the common feature extraction block, L_d^e is the MMD loss used in the special feature extraction block, β and γ are the weight coefficients.

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.

The pseudocode of the overall algorithm is shown as Algorithm 1.

Algorithm 1：MSDA algorithm pseudocode

Input: Multiple ball mill data $\{X_1^s, Y_1^s\}$, $\{X_2^s, Y_2^s\}$, $\{X_3^s, Y_3^s\}$, Target ball mill data X^t, Weight coefficient α, β, γ.

Output: ML prediction value $\hat{Y}^t$

1. Input multi-source domain data X_1^s, X_2^s, X_3^s and target domain data X^t into common feature extraction block.
2. According to formula (1) and formula (2) to obtain the common feature f_{c1}^s, f_{c2}^s, f_{c3}^s and f_c^t.
3. Input common features into formula (5) to calculate MMD loss L_d^c.
4. Input multi-source domain data X_1^s, X_2^s, X_3^s and target domain data X^t into the information extraction block to obtain features containing timing information.
5. Input paired source domain features and target domain features into special feature extraction block respectively to obtain special features f_{e1}^s, f_{e2}^s, f_{e3}^s, f_{e1}^t, f_{e2}^t and f_{e3}^t.
6. Input special features into formula (5) to calculate MMD loss L_d^e.
7. According to formula (3), common features and special features are fused to obtain source domain features f_1^s, f_2^s, f_3^s and target domain features f_1^t, f_2^t, f_3^t.
8. Input the fused features into the regressor to obtain the predicted value $\hat{Y}_1^s$, $\hat{Y}_2^s$, $\hat{Y}_3^s$ of each source domain.
9. Input the real and predicted values of the source domain into formula (4) to calculate the MSE loss L_m.
10. Calculate the total loss according to formula (6) to get the training model.
11. Input the sample of the target domain into the trained model to obtain the predicted value $\hat{Y}_1^t$, $\hat{Y}_2^t$, $\hat{Y}_3^t$.
12. Find the average of $\hat{Y}_1^t$, $\hat{Y}_2^t$, $\hat{Y}_3^t$ to get the final predicted value $\hat{Y}^t$.

4 Experiment

To verify the validity of the method used in this paper, the experiment was carried out on the data collected from the laboratory ball mill as shown in Fig. 3.

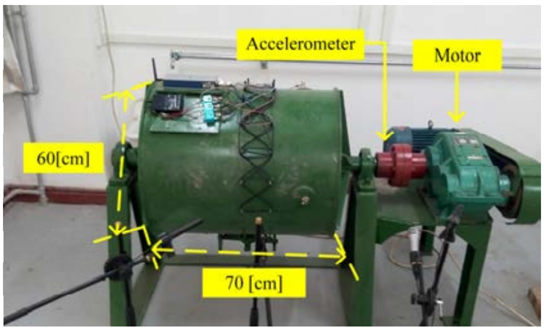


Fig. 3: Ball mill equipment used in the experiments

4.1 Data Preparation

The data set was collected on a small experimental ball mill with a volume of 200L. Using the speed of 1440r/min three-phase asynchronous motor drive equipment. The experimental material is iron ore powder with a density of 2.3t/m^3. The grinding medium is a steel ball with a diameter of 30mm. In order to simulate different situations in real industry, the mass of the ball and the mass of water in the cylinder were fixed in the experiment, and different ML were obtained by changing the medium filling rate (MFR). The experiment was carried out in five groups. The specific parameters of each group were shown in Table 1, including MFR, steel ball mass (SBM), water mass (WM), starting material weight (SMW), termination material weight (TMW), and material change times (MCT).

Table 1: Experimental Parameter

Experiment	MFR	SBM	WM	SMW	TMW	MC
1	0.30	292.00	35	25.5	174.0	139
2	0.35	340.69	40	29.7	170.1	103
3	0.40	389.36	40	34.2	157.5	88
4	0.45	483.02	35	23.4	151.2	95
5	0.50	486.70	40	15.3	144.9	102

In order to ensure the accuracy of the data, sufficient experiments were carried out under different conditions. The ML under different conditions is obtained by continuously adding materials. Taking Experiment 1 as an example, the material amount in the ball mill barrel started at 25.5 kg and gradually increased to 174 kg. A total of 139 times materials were added to the experiment. After the vibration signals are collected, they are divided into 28 samples. Non-overlapping sliding is performed on each sample with w=1024 windows, and Fast Fourier Transform (FFT) is performed on each window. Then, the average of the transformation results of each window is taken to get the final characteristics. Finally, 512-dimensional features were obtained.

Principal component analysis (PCA) was used to reduce the dimension of the data under five working conditions. Five data sets were defined as M1-M5, and data visualization was carried out after dimensionality reduction, as shown in Fig. 4. As can be seen, the distribution of the five data sets collected is inconsistent.

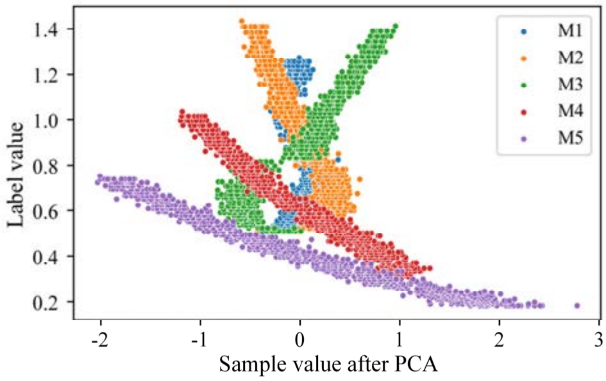


Fig. 4: Probability density plots of different data sets

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.

4.2 Performance Metrics

To quantify the prediction performance of various methods, R-square (R^2) and Root Mean Square Error (RMSE) are used as the evaluation criteria of the algorithm, and the calculation formula is shown in formula (7) and (8).

$$R^2 = 1 - \frac{\sum_{j=1}^{n^t}\left(\hat{y}_j^t - y_j^t\right)^2}{\sum_{j=1}^{n^t}\left(\overline{y}_j^t - y_j^t\right)^2} \tag{7}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n^t}\sum_{j=1}^{n^t}\left(y_j^t - \hat{y}_j^t\right)^2} \tag{8}$$

Where, y_j^t, $\hat{y}_j^t$ and $\overline{y}_j^t$ respectively represent the real value, predicted value and mean value at the j th sample.

4.3 Description of the comparison method

To prove the effectiveness of the proposed method, we use a variety of methods to carry out comparative experiments. Abbreviations and descriptions of these methods are as follows:

(1) SSDA-C: Single-source domain LSTM model, that is, only common feature extraction blocks.

(2) SSDA-E: Single-source domain LSTM+CNN model, that is, only special feature extraction blocks.

(3) MSDA-C: Multi-source domain LSTM model, which only considers common features, but does not consider special features.

(4) MSDA-E: Multi-source domain LSTM+CNN model, which only considers special features without considering common features.

(5) MSDA: The model proposed in this paper. Table 2 lists the specific network parameters.

Table 2: MSDA Network Parameter

MSDA parameter name	Variable value
Weight coefficient α	0.47
Weight coefficient [β , γ]	[0.001,0.001]
Batch size of MSDA	64
The activation function of LSTM	Tanh
The size of the convolution kernel of the first 1D-CNN	5
The size of the convolution kernel of the second 1D-CNN	3
The number of hidden layer neurons in the fully connected layer	[128,64,32,1]
Learning rate	0.001

4.4 Research on the performance of multi-source domain

To verify the prediction ability of the multi-source domain adaptation method on ML, two single-source domain methods were selected as comparison models for experiments. The regressor and network parameters used by SSDA-C and SSDA-E are consistent with those in Table 2. The load parameters PD and CVR of the mill were predicted by the experiment. As the experimental conditions of the single-source domain and multi-source domain are different, the optimal value of each method is used as the prediction result of the target domain. PD prediction results are shown in Table 3.

As can be seen from the experimental results in Table 3, when PD is predicted by the single-source domain preadaptation method, the prediction result of the SSDA-E model is better than that of SSDA-C. But the prediction effect of the two single-source domains is worse than that of MSDA. RMSE of MSDA can be reduced by 4.1% to 6.5% and 2.1% to 4.3% compared with SSDA-C.

Table 3: Predicted Results Of PD

Target domain	Evaluation index	SSDA		MSDA
		SSDA-C	SSDA-E	
M1	RMSE	0.119	0.116	0.078
	R^2	0.069	0.075	0.567
M2	RMSE	0.110	0.088	0.049
	R^2	0.202	0.297	0.731
M3	RMSE	0.097	0.073	0.041
	R^2	0.326	0.453	0.739
M4	RMSE	0.116	0.072	0.051
	R^2	0.098	0.524	0.742
M5	RMSE	0.108	0.089	0.046
	R^2	0.457	0.630	0.912

Table 4 shows the experimental results of CVR prediction using SSDA-C, SSDA-E, and MSDA. According to the experimental results, the prediction accuracy of SSDA-C is higher than that of SSDA-E when predicting CVR. It shows that SSDA-C and SSDA-E have their own advantages and disadvantages in predicting different ML. However, MSDA has the best prediction effect for CVR, and its RMSE can be reduced by 0.1%~1.1% and 0.8%~2.6% compared with SSDA-C and SSDA-E.

Table 4: Predicted Results Of CVR

Target domain	Evaluation index	SSDA		MSDA
		SSDA-C	SSDA-E	
M1	RMSE	0.087	0.103	0.085
	R^2	0.382	0.249	0.402
M2	RMSE	0.063	0.085	0.059
	R^2	0.550	0.172	0.610
M3	RMSE	0.055	0.069	0.054
	R^2	0.551	0.312	0.571
M4	RMSE	0.046	0.069	0.045
	R^2	0.715	0.342	0.718
M5	RMSE	0.069	0.066	0.058
	R^2	0.373	0.422	0.557

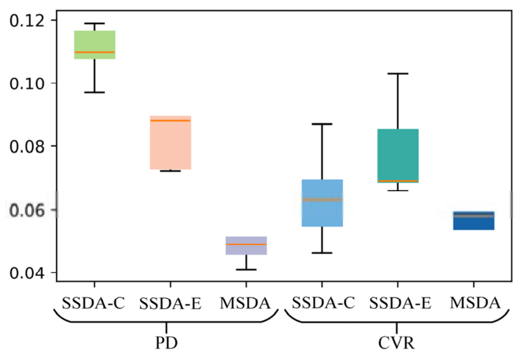


Fig. 5 Bar chart of prediction results by different methods

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.

Fig. 5 visualizes the RMSE of the predicted results of the above model in the form of a boxplot. Intuitively, MSDA predicted the best results. When the multi-source domain preadaptation method is used, the prediction effect is better than that of the single-source domain preadaptation method due to the increase of information.

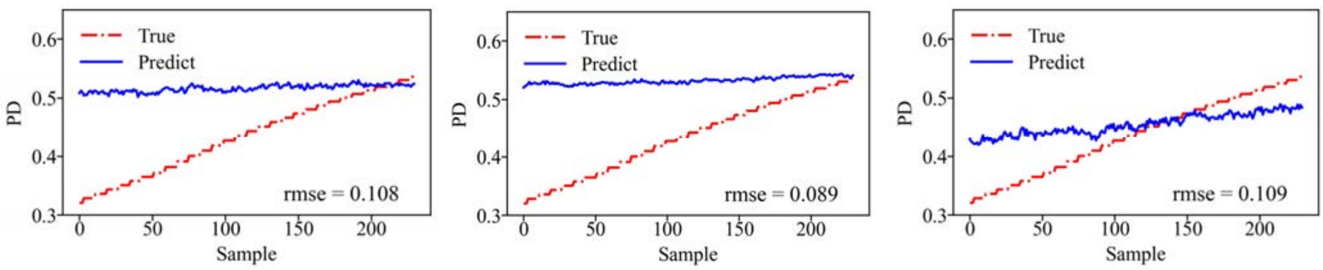
4.5 Research on the performance of common feature and special feature

In order to show that simultaneous extraction of common features and special features is beneficial to improve the model prediction effect, several variants of MSDA are used to predict PD and CVR. MSDA-C only extracts the common features of multi-source domains, and MSDA-E only extracts the special features of multi-source domains. The regressors and parameters used are consistent with MSDA. The experimental method is to predict one target domain with three source domains, and the prediction results are shown in Table 5. The optimal prediction effect of the target domain is highlighted in bold in the table. As can be seen from the table, only the common features of multiple source domains are extracted, that is, the prediction effect is poor when the MSDA-C model is used for prediction. If only the special features of multiple source domains are extracted, the prediction results of partial ML are worse than those of MSDA-C. When MSDA was used, ML predicted best. It shows that the combination of common features and special features can help improve the prediction effect of the model.

Fig. 6 shows the experimental results of PD prediction in M5 by different methods. Fig. 6(a) and Fig. 6(b) respectively show the prediction results of the single-source domain method. Fig. 6(c) shows the prediction results for extracting only common features. Fig. 6(d) shows the results of extracting only special features. Fig. 6(e) shows the prediction results after blending common and special features.

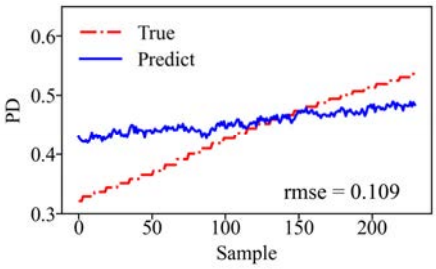
Table 5: Prediction Results Of PD And CVR By Different Models (RMSE)

Target domain	Source domain	CVR			PD		
		MSDA-C	MSDA-E	MSDA	MSDA-C	MSDA-E	MSDA
M1	M2、M3、M4	**0.115**	0.121	**0.085**	**0.107**	0.096	0.084
	M2、M3、M5	0.127	0.123	0.114	0.122	**0.084**	**0.078**
	M2、M4、M5	0.132	**0.118**	0.115	0.123	0.099	0.086
	M3、M4、M5	0.139	0.126	0.120	0.121	0.093	0.095
M2	M1、M3、M4	**0.073**	0.099	**0.059**	**0.067**	0.073	**0.049**
	M1、M3、M5	0.088	0.096	0.066	0.104	0.074	0.050
	M1、M4、M5	0.092	0.093	0.067	0.105	0.070	0.052
	M3、M4、M5	0.104	**0.092**	0.081	0.112	**0.067**	0.069
M3	M1、M2、M4	0.079	0.094	0.070	**0.054**	0.044	0.043
	M1、M2、M5	0.092	0.090	0.067	0.080	0.056	0.047
	M1、M4、M5	0.086	0.088	0.062	0.078	0.052	0.043
	M2、M4、M5	**0.072**	**0.074**	**0.054**	0.085	**0.041**	**0.041**
M4	M1、M2、M3	0.079	0.091	0.065	0.094	0.073	0.061
	M1、M2、M5	0.084	0.088	0.060	0.066	0.058	0.053
	M1、M3、M5	0.076	0.084	0.056	**0.064**	**0.051**	**0.051**
	M2、M3、M5	**0.057**	**0.073**	**0.045**	0.071	0.053	0.052
M5	M1、M2、M3	0.111	0.088	0.076	0.130	0.104	**0.046**
	M1、M2、M4	0.102	0.085	0.076	**0.109**	0.077	0.058
	M1、M3、M4	0.103	0.088	0.072	0.113	0.088	0.046
	M2、M3、M4	**0.092**	**0.075**	**0.058**	0.117	**0.056**	0.052

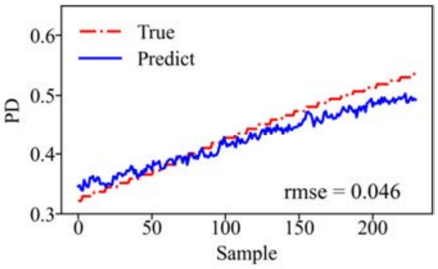


(a) SSDA-C (b) SSDA-E (c) MSDA-C

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.



(d) MSDA-E



(e) MSDA

Fig. 6: The experimental results of M5 prediction of PD by different methods

5 Conclusion

Combined with deep learning, a multi-source domain adaptive model of mill load based on common features and special features is established in this paper. Among them, the common feature extraction block is used to extract the common domain invariant features of multiple source domains and target domains, and the special features are extracted through the special feature extraction block. Considering the distribution difference between the source domain and the target domain, the method of minimizing MMD loss was adopted to improve the prediction accuracy of the model. The experiment was carried out on data collected by a laboratory ball mill. In the experimental part, the prediction results of PD and CVR by the single-source domain model and multi-source domain model are compared. Experimental data show that the prediction accuracy of the multi-source domain model is higher than that of the single-source domain model. In addition, to demonstrate the effectiveness of simultaneous extraction of common features and special features, two variants of the model in this paper are used as comparison models for experiments. The experimental results show that the prediction effect of the model is the best when the common feature and the special feature are extracted simultaneously.

References

[1] Yang L, Cai J. A method to identify wet ball mill's load based on CEEMDAN, RCMDE and SRNN classification, *Minerals Engineering*, 2021, 165(1): 106852.

[2] Tang J, Chai T, Yu W, et al. Modeling load parameters of ball mill in grinding process based on selective ensemble multisensor information, *IEEE Transactions on Automation Science and Engineering*, 2013, 10(3): 726-740.

[3] Liu Z, Tang J, Chai T Y, et al. Selective Ensemble Modeling Approach for Mill Load Parameter Forecasting Based on Multi-modal Feature Sub-sets, *Acta Automatica Sinica*, 2021, 47(8): 1921−1931.

[4] Si G, Cao H, Zhang Y, et al. Experimental investigation of load behaviour of an industrial scale tumbling mill using noise and vibration signature techniques. *Minerals Engineering*, 2009, 22(15): 1289-1298.

[5] Mulenga F K, Mkonde A A, Bwalya M M. Effects of load filling, slurry concentration and feed flowrate on the attainable region path of an open milling circuit, *Minerals Engineering*, 2016, 89: 30-41.

[6] Tang J, Qiao J, Liu Z, et al. Modeling mill load parameter based on LASSO using multi-scale high dimensional frequency spectra data, in *Proceedings of 14th IEEE International Conference on Information and Automation*, 2017: 909-914.

[7] Paul, Guo X G, Qiao T Z, et al. Application of improved LSTM neural network in soft sensing of mill load Parameters , *China Mine Engineering*, 2017 (3): 66-69.

[8] Tang J, Chai T, Liu Z, et al. Selective ensemble modeling based on nonlinear frequency spectral feature extraction for predicting load parameter in ball mills, *Chinese Journal of Chemical Engineering*, 2015, 23(12): 2020-2028.

[9] Yan G W, He M, Tang J, et al. Soft sensor of wet ball mill load based on maximum mean discrepancy multi-source domain transfer learning, *Control and Decision*, 2018, 33(10): 1795-1800.

[10] Zhao S, Yue X, Zhang S, et al. A review of single-source deep unsupervised visual domain adaptation, *IEEE Transactions on Neural Networks and Learning Systems*, 2020, 33(2): 473-493.

[11] Kouw W M, Loog M. A review of domain adaptation without target labels, *IEEE transactions on pattern analysis and machine intelligence*, 2019, 43(3): 766-785.

[12] He M, Tang J, Guo X, et al. Soft sensor for ball mill load using DAMRRWNN model, *Acta Automatica Sinica*, 2019, 45(2): 398-407.

[13] Huang P, Guo J, Sang G, et al. Soft measurement of ball mill load under variable working conditions based on deep transfer learning, M*easurement Science and Technology*, 2022, 33(7): 075009.

[14] Li K, Lu J, Zuo H, et al. Multi-source contribution learning for domain adaptation, *IEEE Transactions on Neural Networks and Learning Systems*, 2022 33(10): 5293-5307.

[15] Zhu Y, Zhuang F, Wang D. Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources, in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, 33(01): 5989-5996.

[16] Qiao L, Jing Z, Pan H, et al. Private and common feature learning with adversarial network for RGBD object classification, *Neurocomputing*, 2021, 423: 190-199.

[17] Zhang X, Lu W, Ding Y, et al. A Mixed Method for Feature Extraction Based on Resonance Filtering, *Intelligent Automation & Soft Computing*, 2023, 35(3): 3141-3154.

[18] Hochreiter S , Schmidhuber J . Long Short-Term Memory, *Neural Computation*, 1997, 9(8):1735-1780.

[19] Ying Z, Xing Z, Jian C, et al. Processor Free Time Forecasting Based on Convolutional Neural Network, in *Proceedings of 37th Chinese Control Conference*, 2018: 9331-9336.

[20] Gretton A, Borgwardt K, Rasch M, et al. A kernel method for the two-sample-problem, *Advances in neural information processing systems*, 2007: 513-520.

[21] Sun B, Feng J, Saenko K. Return of frustratingly easy domain adaptation, in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, 30(1): 2058-2065.

[22] Arjovsky M, Chintala S, Bottou L. Wasserstein generative adversarial networks, in *International conference on machine learning*, 2017: 214-223.

[23] Zeng M G, Li S M, Li R R, et al. A multi-target domain adaptive method for intelligent transfer fault diagnosis, *Measurement*, 2023 207: 112352.

[24] Wang Y, Sun X, Li J, et al. Intelligent fault diagnosis with deep adversarial domain adaptation , *IEEE Transactions on Instrumentation and Measurement*, 2021(70): 1-9.

Authorized licensed use limited to: TU Delft Library. Downloaded on February 10,2025 at 13:51:12 UTC from IEEE Xplore. Restrictions apply.