Construction Cost Estimation in Data-Poor Areas Using Grasshopper Optimization Algorithm-Guided Multi-Layer Perceptron and Transfer Learning

Xuan Sha[1*], Guoqing Dong[2], Xiaolei Li[3] and Juan Sheng[4]
School of Civil and Transportation Engineering, Southeast University ChengXian College,
Nanjing, Jiangsu, 210088, China.[1,2,3,4]
College of Mechanics and Engineering Science, Hohai University, Nanjing, Jiangsu, 210024, China[1]

Abstract*—Accurate construction cost estimation is crucial for completing projects within the planned timeframe and budget. Using machine learning methods to predict construction costs has become a new trend. However, machine learning methods typically require a large amount of data for model training, which makes it particularly challenging in data-poor areas. This paper proposes a novel method, Grasshopper Optimization Algorithm-Guided Multi-Layer Perceptron with Transfer Learning (GOA-MLP-TL), specifically designed for construction cost estimation in data-poor areas. GOA-MLP-TL utilizes the global optimal search capability of the GOA to optimize the parameters of the MLP network. Additionally, an adaptation layer is added into the MLP network, using the Maximum Mean Discrepancy (MMD) measure as a regularization to bridge the gap between the source and target domains. The GOA-MLP-TL can effectively leverage the model trained on data-rich area, and transfer the knowledge to adapt the model suitable for data-poor areas. The proposed approach is verified on two datasets from different areas, and the experimental result shows that, compared to the traditional machine learning method MLP and GOA-MLP without transfer learning, the correlation coefficient (R^2) of the proposed GOA-MLP-TL is improved by 12.05% and 6.90%, respectively. This demonstrate the effectiveness of GOA-MLP-TL for the construction cost estimation task in the data-poor area.

Keywords*—Construction cost estimation; multi-layer perceptron; grasshopper optimization algorithm; transfer learning; machine learning

I. Introduction

Construction cost estimation is an essential component of project feasibility studies. A scientifically and efficiently conducted cost prediction can assist project investors in comparing multiple scenarios, and making more reasonable investment decisions [1]. In engineering practice, the information available is often limited in the early stages of a project, and investors and construction enterprises frequently rely on historical experience to determine project costs, which lacks of efficiency and accuracy.

To address this challenge, researchers have explored and developed various methods for construction cost prediction, aiming to enhance the precision and efficiency of estimating expenses. Traditional prediction approaches primarily rely on statistical analysis [2,3] and simple regression theory [4], these methods typically exhibit low accuracy , which cannot meet the demands of actual projects with numerous uncertainties. In recent two decades, with the development of computer science, many new machine learning methods, such as genetic algorithms [5,6], neural networks[7]–[10], Random Forest [11], and support vector machines[12,13], have been successfully applied to cost estimation. Jung et al. [5] introduces a novel hybrid approach that integrates Case-Based Reasoning (CBR) with a Genetic Algorithm (GA) and employs a Local Search Method. Experiments proved the applicability of the method in cost estimation. Fan and Sharma [12] explores the application of SVM and LSSVM in developing a construction cost prediction model, the results show that the prediction model based on SVM has a higher prediction accuracy and the results are robust. Ksenija et al. [14] delves into the application of artificial neural networks to road construction cost estimation, comparing the effectiveness of various types of neural networks in cost estimation. Sharma et al. [15] compares various machine learning algorithms to predict the construction cost, Findings the results demonstrate that the ensemble methods, such as gradient boosted trees, exhibit the best performance for construction cost prediction. Tayefeh et al. [16] reviews manuscripts that proposed for cost estimation with machine learning techniques for the last 30 years, categorises and summarises commonly used methods.

While the aforementioned methods have yielded promising prediction results, they still have certain limitations. Because the conventional machine learning algorithms are based on the assumption that both the training and test data are drawn from the same distribution [17], which means these methods are trained and tested using the same construction cost database. Due to the differences in design specifications, construction methods, and labor and material costs in different regions, if the models obtained by above methods are applied to other areas, the cost prediction accuracy may decrease. Additionally, the collection and processing of construction cost data in data-poor areas is also labor- and time-consuming.

The emergence of transfer learning [18] provides a new approach to solve the above problems. Transfer learning is a technique in which the knowledge learned from one task is re-used in order to boost performance on another different but related task [19]. Unlike conventional machine learning methods, which rely on one-to-one relationships between training data sets and individual models, transfer learning can effectively

*Corresponding author

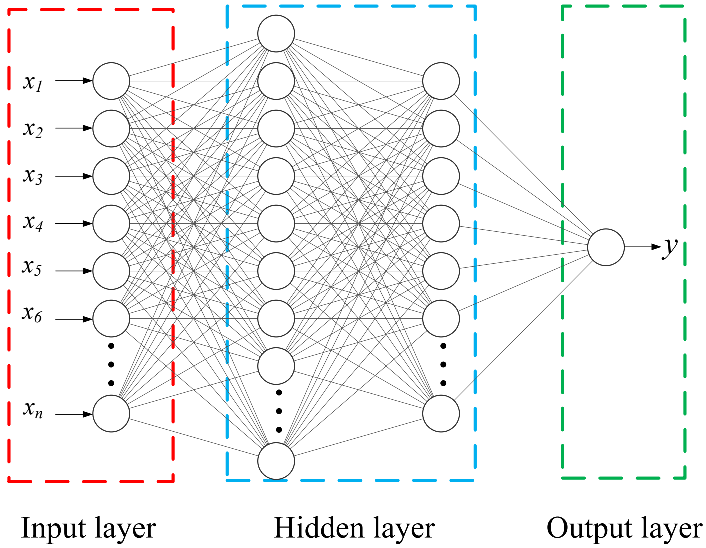

Fig. 1. The structure of a typical multi-layer perceptron.

leverage existing data resources across different domains for predictive modeling. In this work, we propose an approach that combines Grasshopper Optimization Algorithm-Guided Multi-Layer Perceptron with Transfer Learning(GOP-MLP-TL) for cost estimation in data-poor areas. Specifically, GOA-MLP-TL utilizes the global optimal search capability of the GOA to optimize the parameters of the MLP network.Additionally, an adaptation layer is incorporated into the network before the output layer, and regularization is introduced to reduce the distribution mismatch between data from different areas at this layer. With our method, utilizing existing models and data, only a small amount of data from data-poor areas is required to train new models for cost estimation in these areas.

The rest of the paper is organized as follows. Section II explains the principles of the multi-layer perceptron, grasshopper optimization algorithm and transfer learning, meanwhile presents the framework of our approach. Section III describes the experimental setup, datasets, results, and discussion. Finally, Section IV concludes the paper.

II. Methodology

The estimation model in this work is based on a multi-layer perceptron and transfer learning. In this section, we first introduce the structure of the MLP network and the MLP parameter optimization method based on the Grasshopper Optimization Algorithm (GOA). Following that, we provide the implementation details of transfer learning using an adaptation layer. Finally, we present the flowchart of the training and estimation process for our approach.

A. GOA-Guided Multi-Layer Perceptron

Multi-layer perceptron (MLP) is one of the most widely used neural networks. It has one input layer, one output layer and one or more hidden layers of neurons. Multiple layers of neurons within the MLP network enhance the input-expected output mapping capability [20]. In this work, MLP network is used to learn the estimation model based on the existing data from data-rich areas.

The structure of a typical MLP is shown in Fig. 1 MLP learning mainly includes the following two stages:

1) Forward propagation: Assuming the number of hidden layers in the MLP is K , the output vector of the i-th hidden layer is denoted as $\mathbf{h}^{(i)}$, and the output value of the output layer is denoted as y, then:

$$\mathbf{h}^{(i)} = \begin{cases} \sigma\left(\mathbf{x}\mathbf{W}^{(1)} + \mathbf{b}^{(1)}\right), & i = 1 \\ \sigma\left(\mathbf{h}^{(i-1)}\mathbf{W}^{(i)} + \mathbf{b}^{(i)}\right), & i \in \{2, 3, \ldots, K\} \end{cases} \tag{1}$$

$$y = \mathbf{h}^{(K)}\mathbf{W}^{(O)} + b^{(O)} \tag{2}$$

where, $\mathbf{W^{(i)}}$and $\mathbf{b}^{(i)}$ are the weight matrix and bias vector for the i-th hidden layer respectively, $\mathbf{x}$ represents the input layer vector, $\mathbf{W}^{(O)}$and $b^{(O)}$ are the weight matrix and bias for the output layer. $\sigma(\cdot)$in Eq. (2) is the activation function, which is used to achieve non-linear mapping between neuron inputs and outputs. Common activation functions include Sigmoid, hyperbolic tangent function (Tanh), rectified linear unit (ReLU), etc. In this work, Sigmoid is used as the activation function, which is defined as follows:

$$\sigma\left(x\right) = \frac{1}{1 + e^{-x}} \tag{3}$$

2) Back propagation: Assuming the number of input samples is N, y_n and $y_n{}'$ represent the actual data and the predicted output of the n-th sample. The error between y_n and $y_n{}'$ is measured by mean square error (MSE), and the MSE is used as the loss function, which is defined in Eq. (4). The MSE loss is minimized by adjusting the weights in the MLP network through the back propagation algorithm.

$$L_{MSE} = \frac{1}{N}\sum_{n=1}^{N}\left(y_n - y_n{}'\right)^2 \tag{4}$$

Traditional MLP optimization methods are sensitive to initial parameter values, which may lead to issues such as getting stuck in local optima. To mitigate this challenge and enhance the performance of the MLP model, this study employs the Grasshopper Optimization Algorithm (GOA) to optimize the MLP network parameters. The GOA was introduced by Saremi et al. in 2017 as a new type of swarm intelligence algorithm [21]. It simulates the foraging behavior of grasshopper swarms to search for optimal solutions. In this algorithm, adult grasshoppers perform global searches in the early stages, while nymph grasshoppers conduct detailed exploitation in the vicinity of specific areas during the later stages. The position model of the i-th grasshopper in the d-th dimension is as follows:

$$X_i^d = c\left(\sum_{j=1, j\neq i}^{N} c\frac{\mathrm{UB}_d - \mathrm{LB}_d}{2} s\left(\left|x_j^d - x_i^d\right|\right)\frac{\left(x_j^d - x_i^d\right)}{d_{ij}}\right) + \hat{T}_d \tag{5}$$

where, the subscript d represents the d-dimensional space, UB_d and LB_d respectively represent the upper bound and lower bound of grasshopper search, $\left|x_j^d - x_i^d\right|$ is the Euclidean distance from the i-th grasshopper to the j-th grasshopper, $\frac{(x_j^d - x_i^d)}{d_{ij}}$ is the unit vector from the i-th grasshopper to the j-th grasshopper, $\hat{T}_d$ represents the best solution (target) attained so far, s is a factor representing the range and strength of social interaction within the population and c is a linear decreasing factor. the c factor is attained as follows:

$$c = c_{\max} - l\frac{c_{\max} - c_{\min}}{L} \tag{6}$$

where, l represents the number of iterations, L denotes the upper bound of the iterations, $c_{\max}$ and $c_{\min}$ are respectively the upper and lower bounds of the decreasing factor c, with $c_{\max}$ set to 1 and $c_{\min}$ set to 10^{-5} in this article.

Using the GOA can effectively optimize the weights and biases in each layer of an MLP network, thereby improving the predictive performance of the MLP network. Specifically, the main steps of GOA-MLP include:

- Define the MLP Structure: Determine the architecture of the MLP, including the number of layers, number of neurons in each layer, activation functions, and other parameters.
- Initialize the Population of grasshoppers: Determine the basic parameters of the GOA, such as the grasshopper population size and the number of iterations. Initialize a random set of grasshoppers. Each grasshopper individual represents a potential MLP configuration, including the network's weights and biases. Assuming the MLP network has N weights and bias parameters, one grasshopper can be represented as an N-dimensional single vector.
- Fitness evaluation: For each grasshopper, maps its position value to the corresponding weights or biases in the MLP network. Use Mean Squared Error (MSE) as the fitness function, and evaluation the fitness on the training dataset.
- Update the best position: If the fitness of a grasshopper's current position is superior to its historical best position, then update the best position.
- Update the positions of the grasshoppers: Taking into account the interactions between individuals and the influence of the target position, update the grasshopper positions based on Eq. 5.
- Repeat the steps 3–5 until the specified number of iterations or until a stopping criterion is met
- Termination and testing: Finally, the process is terminated and the MLP with the minimum MSE should be tested on the test/validation datasets.

The overall steps of the GOA-MLP are demonstrated in Fig. 2.

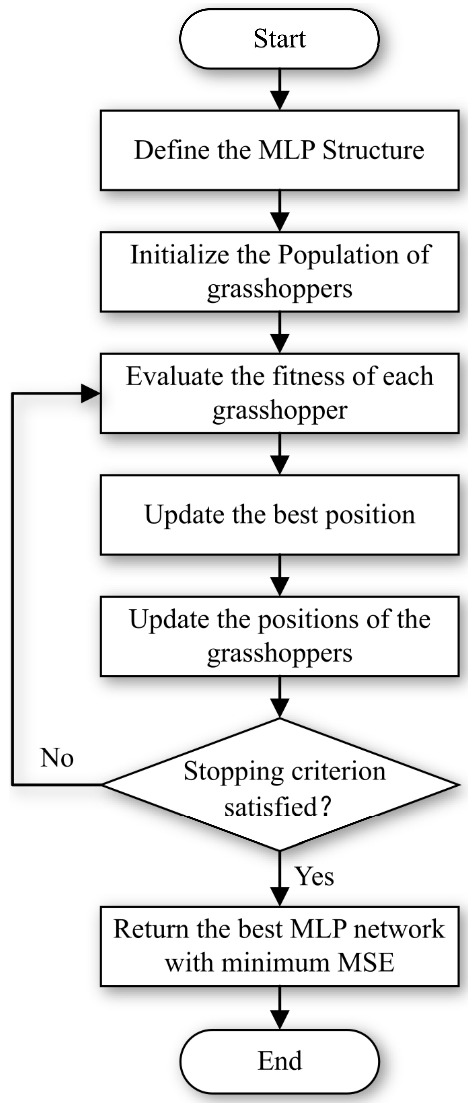

Fig. 2. The flowchart of the GOA-MLP.

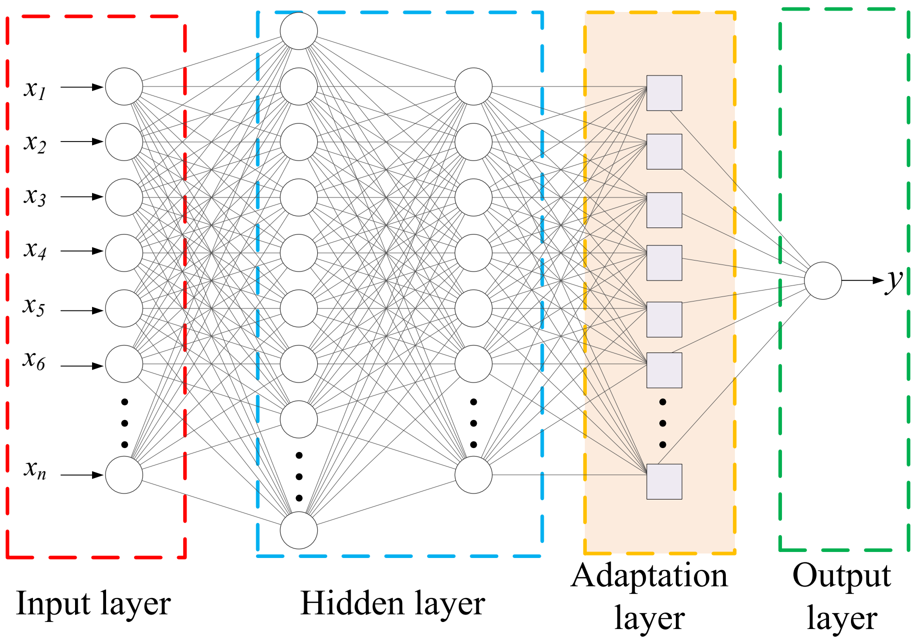

Fig. 3. The structure of MLP with an adaptation layer.

B. Adaptation Layer for Transfer Learning

Transfer learning involves a source domain and a target domain, each characterized by distinct yet related distributions. The process of transfer learning is the process of transferring the knowledge from the source domain to the target domain, so as to solve the problems of insufficient knowledge in the target domain and insufficient accuracy of the model. In this work, existing construction cost dataset from the data-rich area belongs to the source domain, and the target domain corresponds to the data-pool areas, which lack of the sufficient construction cost data.

In order to achieve knowledge transfer, an adaptation layer is added in the MLP network before the output layer, the modified MLP network is shown in Fig. 3 The distribution mismatch between data from source and target domains is minimized on this adaptation layer. Specifically, the Maximum Mean Discrepancy (MMD) [22] is utilized to measure the distribution mismatch, and the MMD is used as a regularization embedded in the back propagation training process. During the process of training adaptation layer, the parameters of the hidden layers are fixed, so the original model learned from source domain can be re-used.

The Maximum Mean Discrepancy (MMD) serves as a measure of the difference between two probability distributions based on their samples. This criterion proves to be effective in comparing distributions without the need for an initial estimation of their density functions.

Let $\left\{\mathbf{x}_s^i\right\}_{i=1,\ldots,n_s}$ and $\left\{\mathbf{x}_t^i\right\}_{i=1,\ldots,n_t}$ be data vectors drawn from distributions of source domain and target domain, respectively, MMD can be defined as:

$$MMD\left(\mathbf{x}_s, \mathbf{x}_t\right) = \left\| \sum_{i=1}^{n_s} f\left(\mathbf{x}_s^i\right) - \sum_{i=1}^{n_t} f\left(\mathbf{x}_t^i\right) \right\|_H \tag{7}$$

In this equation, H represents the Reproducing Kernel Hilbert Space(RKHS), $f(\cdot)$ is a mapping function used to map the original variables into the RKHS. Expanding the equation into a matrix multiplication form, one can rewrite the kernelized empirical estimate of MMD as:

$$\begin{aligned} MMD_e\left(\mathbf{x}_s, \mathbf{x}_t\right) = & \left(\frac{1}{n_s(n_s-1)} \sum_{i=1}^{n_s} \sum_{j=1}^{n_s} k\left(\mathbf{x}_s^i, \mathbf{x}_s^j\right) + \right. \\ & \frac{1}{n_t(n_t-1)} \sum_{i=1}^{n_t} \sum_{j=1}^{n_t} k\left(\mathbf{x}_t^i, \mathbf{x}_t^j\right) - \\ & \left. \frac{2}{n_s n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} k\left(\mathbf{x}_s^i, \mathbf{x}_t^j\right) \right)^{\frac{1}{2}} \end{aligned} \tag{8}$$

where $k(\cdot, \cdot)$ is a Gaussian kernel function.

After adding the adaptation layer for transfer learning, the total loss function of the entire network can be expressed as follow:

$$L = L_{MSE} + \lambda {MMD_e}^2\left(\mathbf{x}_s, \mathbf{x}_t\right) \tag{9}$$

where L_{MSE} denotes the standard MSE loss over the available labeled data from both source and target domains, and λ is a constant controlling the weight of MMD contribution to the total loss function. The loss function simultaneously optimizes the estimation output error and the distribution mismatch on the adaptation layer, makes the model more suitable for the data from the target domain.

C. Framework of the Method

The framework of our method is depicted in Fig. 4. During the training phase, we initially train the MLP network model

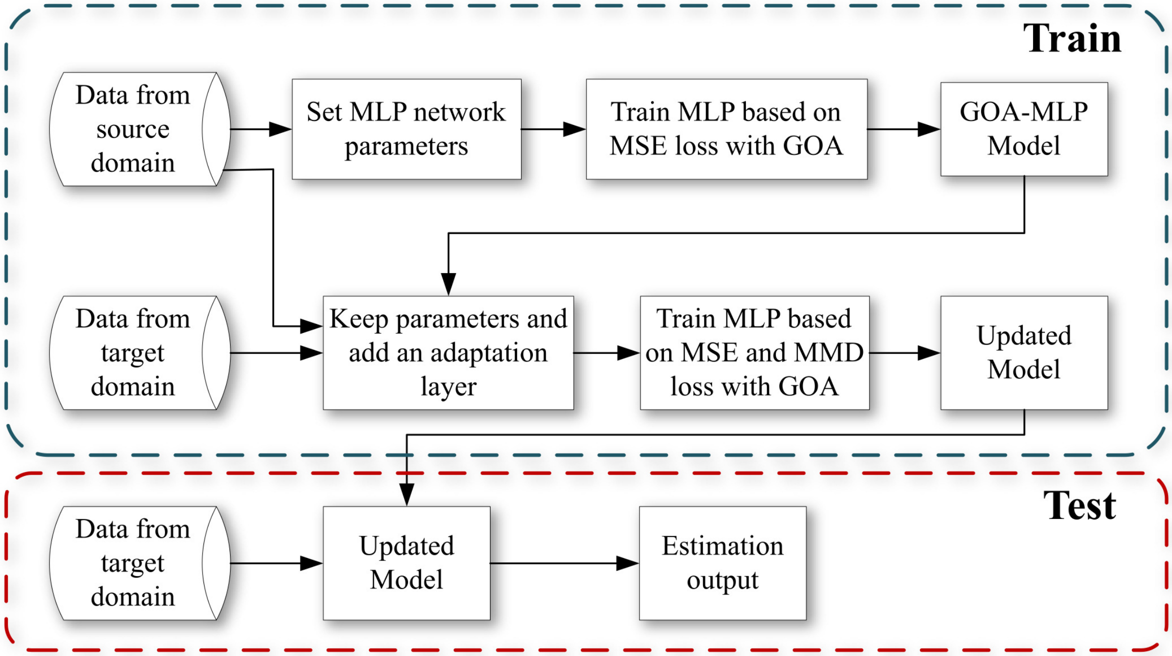

Fig. 4. The framework of the proposed method.

using the source domain data.The GOA is employed for MLP model optimization, utilizing the Mean Squared Error (MSE) loss function. Subsequently, we introduce the adaptation layer, maintaining the original parameters, and optimize the network model based on both MSE and Maximum Mean Discrepancy (MMD) loss functions. The resulting updated model is then utilized during the testing phase to estimate costs for the target domain data.

III. Experiment

A. Data Description

To verify the effectiveness of the proposed cost estimation method, two groups of construction cost data from different areas were utilized to simulate data from source domain and target domain respectively.

Construction cost data collected from RSMeans Online [23] during the period (1998–2018) was used as the source domain data. RSMeans data is North America's leading source of construction cost information, delivering reliable, locally relevant, and up-to-date cost data. RSMeans Online is a widely used construction cost estimating and project management software. It provides the construction industry with a comprehensive set of detailed and accurate cost data and related tools, helping users make precise cost predictions and manage projects at various stages. We gathered 5400 samples from RSMeans Online, and we selected eight variables for the experiments, which are commonly used and highly relevant to cost estimation. Table I lists and explains these variables.

TABLE I. List of Variables

Variable	Description
y	Actual construction costs
x_1	Building type
x_2	Structure type
x_3	Total floor area of the building
x_4	Numerical number of floors
x_5	Total height of the building
x_6	Formwork area
x_7	Concrete volume
x_8	Per square meter cost

For the target domain, data came from the construction projects from 2016 to 2022 on the Guanglianda Index Network [24]. Guanglianda is one of China's leading providers of software and information technology services for the construction industry. It offers comprehensive construction cost information to assist businesses in decision-making. We collected a total of 320 samples from it, and we chose the same eight variables as those in the source domain data setting.

B. Experimental Setup

Considering the sample size, we employed an MLP network with two hidden layers in this work. The input layer consisted of 8 nodes, corresponding to the number of input variables. The output layer had 1 node, representing the cost estimation results. The number of nodes in the hidden layers was determined through a trial-and-error process, resulting in two hidden layers with 15 and 10 nodes, respectively. The major parameters of GOA were based on reference [25] and were set as follows, The population size of grasshoppers was set to 20, the maximum number of iterations was set to 30, and the upper and lower bounds of the decreasing factor c were set to 1 and 10^{-5}, respectively. For the transfer learning part, the number of nodes in the adaptation layer was set to 10, the same size as the hidden layer 2, and the parameter for the weight of MMD loss was set to 0.35.

The performance of the proposed cost estimation method is evaluated and compared with the following commonly used statistical metrics: Correlation coefficient (R^2); Root Mean Square Error (RMSE); and Mean Average Percentage Error (MAPE). These metrics are defined as follows:

$$R^2 = 1 - \frac{\sum_{n=1}^{N} (y_n - y_n')^2}{\sum_{n=1}^{N} (y_n - mean(y_n))^2} \tag{10}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N} (y_n - y_n')^2} \tag{11}$$

$$MAPE = \frac{1}{N}\sum_{n=1}^{N} \left| \frac{y_n - y_n'}{y_n'} \right| \tag{12}$$

Here y_n represents the actual data provided in the dataset, and $y_n{}'$ corresponds to the predicted output of the n-th sample. The experimental environment is based on MATLAB software, running on a PC with an Intel i5 12400 CPU, 32G RAM, and the Windows 10 operating system.

C. Results and Discussion

1) Comparison with other methods: To validate the effectiveness of our cost estimation method (GOA-MLP-TL), we compared it with four other methods. The first three methods are traditional machine learning approaches, which are: Linear Regression (LR), Support Vector Regression (SVR) with the RBF kernel and a single multi-layer perceptron (MLP) network. The MLP shares the same network structure as our proposed method but lacks the adaptation layer for transfer learning and does not utilize GOA for optimization. The fourth method is GOA-MLP, which employs the same network structure and optimization method as our proposed approach but excludes the adaptation layer for transfer learning.

For all four comparative methods, the models were trained using data solely from the source domain and subsequently tested on both the source and target domains. For GOA-MLP-TL, when testing in the source domain, the model was trained only with data from the source domain, making it identical to GOA-MLP. However, when testing in the target domain, the model used some target domain data for knowledge transfer. The estimation performance of these five methods has been compared in Table II. To facilitate a more intuitive comparison, we have also plotted the comparison of estimation results using different evaluation metrics in Fig. 5, Fig. 6, and Fig. 7, respectively.

From Table II and Fig. 5, it can be observed that in the evaluations on the source domain data, GOA-MLP and GOA-MLP-TL achieved the highest R^2 values (as previously explained, these two methods are identical in the source domain tests), followed by MLP and SVR, with LR obtaining the lowest R^2 value. GOA-MLP shows a 3.2% improvement compared to MLP. In the tests conducted on the target domain data, it is evident that the R^2 values for all methods decreased.

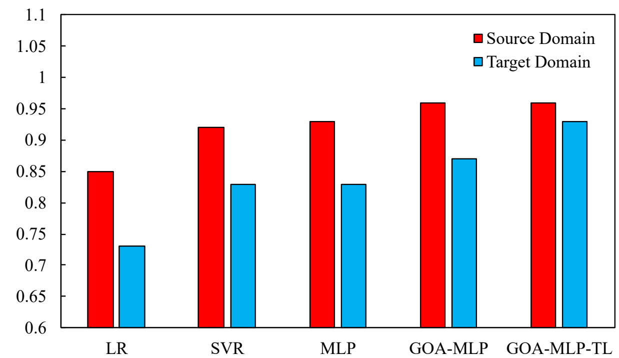

Fig. 5. Comparison of estimation performance of various methods using R^2.

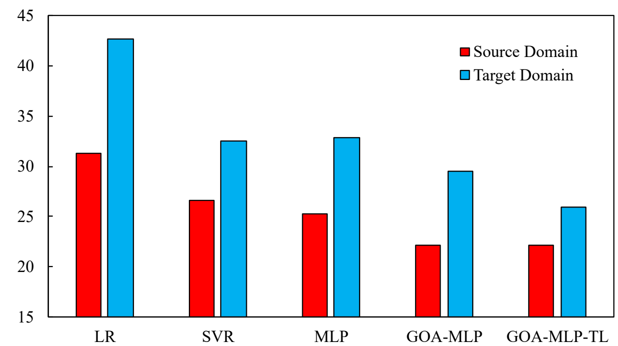

Fig. 6. Comparison of estimation performance of various methods using RMSE.

The three traditional machine learning methods exhibited an average decrease of 11.55%, GOA-MLP showed a decrease of 9.37%, whereas GOA-MLP-TL, which employs transfer learning, only demonstrated a decrease of 3.13%. GOA-MLP-TL shows improvements of 12.05% and 6.90% compared to MLP and GOA-MLP, respectively. The RMSE and MAPE values exhibit similar trends which have been illustrated in Fig. 6 and Fig. 7. it is worth noting that lower values are preferred for both RMSE and MAPE, which is contrary to R^2.

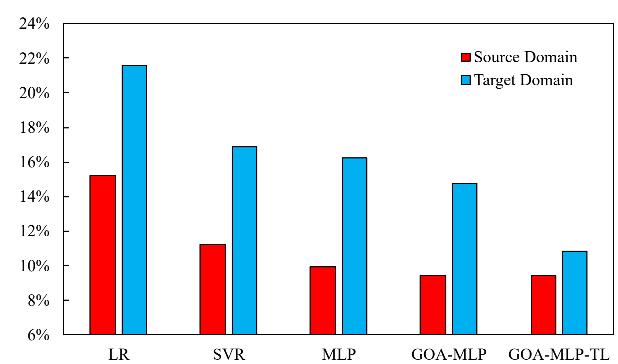

Fig. 7. Comparison of estimation performance of various methods using MAPE.

TABLE II. COMPARISON OF ESTIMATION PERFORMANCE OF VARIOUS METHODS

	LR		SVR		MLP		GOA-MLP		GOA-MLP-TL	
	source domain	target domain	source domain	target domain	source domain	target domain	source domain	target domain	source domain	target domain
R^2	0.85	0.73	0.92	0.83	0.93	0.83	0.96	0.87	0.96	0.93
RMSE	31.26	42.64	26.59	32.55	25.32	32.84	22.14	29.55	22.14	25.95
MAPE	15.21%	21.54%	11.20%	16.87%	9.91%	16.23%	9.43%	14.75%	9.43%	10.85%

The experimental results indicated that, first, the proposed GOA-MLP demonstrates enhanced predictive accuracy compared to the three traditional machine learning methods. Second, in contrast to these compared methods, the proposed GOA-MLP-TL significantly improves estimation performance on data from the target domain. This improvement is achieved by establishing a correlation between the source domain and the target domain through knowledge transfer. Consequently, this approach effectively addresses the challenge of constructing cost estimation models in data-poor areas.

2) *Influence analysis of target domain sample size:* To analyze the influence of target domain sample size, we conducted experiments using the proposed GOA-MLP-TL with varying numbers of samples from the target domain for transfer learning, and evaluated its performance on the target domain. The correlation coefficient (R^2) was used as the key metric. For comparison, we also employed GOA-MLP, training and testing on the target domain with different training sample sizes. Note that the number of training samples for GOA-MLP was incremented from 40 because training process cannot converge with too few samples. The results were plotted in Fig. 8.

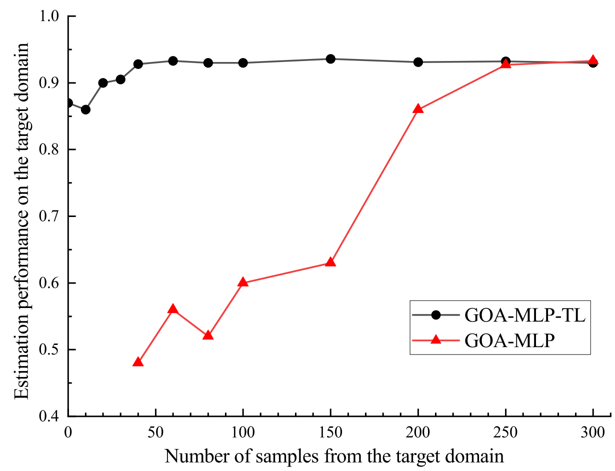

Fig. 8. Influence analysis of target domain sample size using R^2.

From the Fig. 8, it can be noted that as the number of samples in the target domain increases, the R^2 value of GOA-MLP-TL gradually improves. When the number of samples exceeds a certain threshold, the R^2 value stabilizes. In this experiment, when the number of samples in the target domain exceeds 40, the estimation model becomes quite robust. Conversely, for GOA-MLP trained directly on the target domain data, a sample size of over 250 is required to achieve gradual stabilization and attain R^2 values comparable to those of GOA-MLP-TL.

These empirical findings suggest that training a stable model directly in the target domain requires a large number of data samples. Conversely, with the adoption of transfer learning, which utilizes models pre-trained in the source domain, only a small number of target domain samples are needed to achieve comparable prediction accuracy. In this experiment, the required number of target domain samples for GOA-MLP-TL is approximately 76% less compared to GOA-MLP.

IV. CONCLUSION

In this work, we propose a novel method called the Grasshopper Optimization Algorithm-Guided Multi-Layer Perceptron with Transfer Learning (GOA-MLP-TL) for construction cost estimation in areas with limited data. GOA-MLP-TL is based on a Multi-Layer Perceptron (MLP) network, with two key improvements to enhance predictive ability. First, we utilize the Grasshopper Optimization Algorithm (GOA) to optimize the parameters of the MLP network. Second, we incorporate an adaptation layer into the network to facilitate knowledge transfer between domains. The Maximum Mean Discrepancy (MMD) measure is used as a regularization technique to reduce distributional differences between domains. GOA-MLP-TL effectively leverages models trained on data-rich areas and transfers the knowledge to adapt the model for data-poor areas. We simulate samples from data-rich and data-poor areas using two datasets and test the method on these datasets. Experimental result shows that, compared to the traditional machine learning method MLP and GOA-MLP without transfer learning, the R^2 value of the proposed GOA-MLP-TL is improved by 12.05% and 6.90%, respectively. This demonstrate the effectiveness of GOA-MLP-TL for the construction cost estimation task in the data-poor area.

Due to limitations in the availability of experimental data, this study selected only eight variables for experiment, which does not cover all aspects of construction cost estimation. Future work could explore incorporating additional relevant variables to improve prediction accuracy. Additionally, future research could employ deep transfer learning methods to enhance the algorithm presented in this study.

ACKNOWLEDGMENT

The research of this article is supported by Jiangsu Provincial University Philosophy and Social Science Research Project (Grant No. 2022SJYB0707) and Research and Development Fund for young teachers of Southeast University ChengXian College (Grant No. Z0057).

REFERENCES

[1] S. Demirkesen and B. Ozorhon, "Impact of integration management on construction project management performance," *International Journal of Project Management*, vol. 35, no. 8, pp. 1639–1654, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263786317300029

[2] D. S. Tejale, S. Khandekar, and J. Patil, "Analysis of construction project cost overrun by statistical method," *International Journal*, vol. 3, no. 5, pp. 349–355, 2015.

[3] A. A. A. Hammad, S. M. A. Ali, G. J. Sweis, and R. J. Sweis, "Statistical analysis on the cost and duration of public building projects," *Journal of Management in Engineering*, vol. 26, no. 2, pp. 105–112, 2010.

[4] D. J. Lowe, M. W. Emsley, and A. Harding, "Predicting construction cost using multiple regression techniques," *Journal of Construction Engineering and Management*, vol. 132, no. 7, pp. 750–758, 2006.

[5] S. Jung, J.-H. Pyeon, H.-S. Lee, M. Park, I. Yoon, and J. Rho, "Construction cost estimation using a case-based reasoning hybrid genetic algorithm based on local search method," *Sustainability*, vol. 12, no. 19, 2020. [Online]. Available: https://www.mdpi.com/2071-1050/12/19/7920

[6] L. Zhao, W. Zhang, and W. Wang, "Construction cost prediction based on genetic algorithm and bim," *INTERNATIONAL JOURNAL OF PATTERN RECOGNITION AND ARTIFICIAL INTELLIGENCE*, vol. 34, no. 7, JUN 30 2020.

[7] S. Saeidlou and N. Ghadiminia, "A construction cost estimation framework using dnn and validation unit," *Building Research & Information*, vol. 52, no. 1-2, pp. 38–48, 2024. [Online]. Available: https://doi.org/10.1080/09613218.2023.2196388

[8] S. Yun, "Performance analysis of construction cost prediction using neural network for multioutput regression," *APPLIED SCIENCES-BASEL*, vol. 12, no. 19, OCT 2022.

[9] D. Ye, "An algorithm for construction project cost forecast based on particle swarm optimization-guided bp neural network," *SCIENTIFIC PROGRAMMING*, vol. 2021, OCT 21 2021.

[10] Q. hui Liu and Q. Cao, "Dynamic control method of construction cost based on fuzzy neural network," *Automatika*, vol. 65, no. 4, pp. 1339–1349, 2024. [Online]. Available: https://doi.org/10.1080/00051144.2024.2372747

[11] Z. Zheng, L. Zhou, H. Wu, and L. Zhou, "Construction cost prediction system based on random forest optimized by the bird swarm algorithm," *MATHEMATICAL BIOSCIENCES AND ENGINEERING*, vol. 20, no. 8, pp. 15 044–15 074, 2023.

[12] M. Fan and A. Sharma, "Design and implementation of construction cost prediction model based on svm and lssvm in industries 4.0," *International Journal of Intelligent Computing and Cybernetics*, vol. 14, no. 2, pp. 145–157, Jan 2021. [Online]. Available: https://doi.org/10.1108/IJICC-10-2020-0142

[13] T. Lin, T. Yi, C. Zhang, and J. Liu, "Intelligent prediction of the construction cost of substation projects using support vector machine optimized by particle swarm optimization," *MATHEMATICAL PROBLEMS IN ENGINEERING*, vol. 2019, SEP 25 2019.

[14] K. Tijanić, D. Car-Pušić, and M. Šperac, "Cost estimation in road construction using artificial neural network," *Neural Computing and Applications*, vol. 32, no. 13, pp. 9343–9355, Jul 2020. [Online]. Available: https://doi.org/10.1007/s00521-019-04443-y

[15] V. Sharma, M. Zaki, K. N. Jha, and N. M. A. Krishnan, "Machine learning-aided cost prediction and optimization in construction operations," *ENGINEERING CONSTRUCTION AND ARCHITECTURAL MANAGEMENT*, vol. 29, no. 3, pp. 1241–1257, MAR 24 2022.

[16] S. Tayefeh Hashemi, O. M. Ebadati, and H. Kaur, "Cost estimation and prediction in construction projects: a systematic review on machine learning techniques," *SN Applied Sciences*, vol. 2, no. 10, p. 1703, Sep 2020. [Online]. Available: https://doi.org/10.1007/s42452-020-03497-1

[17] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006.

[18] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021.

[19] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, p. 9, May 2016. [Online]. Available: https://doi.org/10.1186/s40537-016-0043-6

[20] J. Isabona, A. L. Imoize, S. Ojo, O. Karunwi, Y. Kim, C.-C. Lee, and C.-T. Li, "Development of a multilayer perceptron neural network for optimal predictive modeling in urban microcellular radio environments," *Applied Sciences*, vol. 12, no. 11, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/11/5713

[21] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: Theory and application," *Advances in Engineering Software*, vol. 105, pp. 30–47, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0965997816305646

[22] K. M. Borgwardt, A. Gretton, M. J. Rasch, H.-P. Kriegel, B. Schölkopf, and A. J. Smola, "Integrating structured biological data by Kernel Maximum Mean Discrepancy," *Bioinformatics*, vol. 22, no. 14, pp. e49–e57, 07 2006. [Online]. Available: https://doi.org/10.1093/bioinformatics/btl242

[23] RSMeans, "Rsmeans online," accessed 27 Mar. 2024. [Online]. Available: https://www.rsmeansonline.com/

[24] Guanglianda, "Guanglianda index network," accessed 27 Mar. 2024. [Online]. Available: https://www.gldzb.com/

[25] S. Moghanian, F. B. Saravi, G. Javidi, and E. O. Sheybani, "Goamlp: Network intrusion detection with multilayer perceptron and grasshopper optimization algorithm," *IEEE Access*, vol. 8, pp. 215 202–215 213, 2020.