An Enhanced XGBoost Algorithm for Mobile Price Classification

Yehang Zhang
School of Computer Science
China University of Geosciences
Wuhan, China
ZhangYeHang2002@outlook.com

Qi Ding
No. 722 Research Institute, CSIC
Hubei Siant Science & Technology CO.,LTD
Wuhan, China
dingqi722@163.com

Chao Liu*
School of Computer Science
China University of Geosciences
Wuhan, China
* corresesponding author
cs_liuchao@cug.edu.cn

2023 IEEE 6th International Conference on Big Data and Artificial Intelligence (BDAI) | 979-8-3503-0366-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/BDAI59165.2023.10256847

***Abstract*—With the popularity of mobile devices in recent years, understanding and predicting mobile prices is crucial for mobile manufacturers. Mobile price can be influenced by different hardware factors, making accurate prediction of prices complex and challenging. In this paper, a novel classification model combining Dung beetle optimizer (DBO) and XGBoost is proposed to accurately classify mobile price. In the PCA dimensionality reduction phase, we design a feature filtering strategy with consideration of feature importance. Through DBO, the optimal parameters of XGBoost are selected to build the model. The experimental results demonstrate the superiority of our proposed DBO-XGBoost model compared with the baseline models such as standard XGBoost, Decision Tree (DT), Random Forest (RF), and AdaBoost models. Our model scores 95.5% in classification accuracy with the kaggle mobile-price-classification dataset, outperforming the other models (XGBoost: 91.8%, DT: 85.3%, RF: 90.5%, AdaBoost: 71.5%). Furthermore, the performance of DBO is verified to be better than grid search for parameter optimization.**

Keywords*—*XGBoost, Dung beetle optimizer (DBO), Parameter optimization, Classification

I. INTRODUCTION

Nowadays, the number of smartphone users is increasing significantly and consumers have a large number of choices in smartphones, while the price of smartphones has a positive impact on consumers' purchase intention [1]. Therefore, predicting the price of mobile phone based on its characteristics can help consumers choose the right item and enable cell phone manufacturers to price their phones more reasonably.

Several studies have applied machine learning algorithms to address the mobile price classification problem. Reference [2] used Logistic Regression algorithm and one-vs-rest method for multiple classification and achieves 81% accuracy on the mobile price dataset. In [3], the authors utilized a Decision Tree algorithm in conjunction with feature filtering to classify mobile phone price, achieving superior results compared to the Logistic Regression algorithm. The classification accuracy reached 86.2%. In [4], a random forest algorithm with parameter pruning was utilized, resulting in a classification accuracy of 92.2%.

XGBoost is also a promising classification model. However, XGBoost has multiple parameters, including tree depth, learning rate, etc. The parameter optimization is complex due to intricate parameter interactions, coupled with the multiple factors for selecting optimal parameter values. To address these issues, we propose utilizing the DBO algorithm to optimize the parameters of XGBoost.

The study initially obtained the mobile price classification dataset from Kaggle community, which included features such as battery power, screen size, and number of processor cores, etc. Subsequently, the data was effectively cleaned, and the feature importance was calculated using a decision tree. Irrelevant features were removed based on these results. Furthermore, Principal Component Analysis (PCA) was applied to reduce the dimensionality of the data, identifying the most suitable data dimensions for classification purposes. The DBO algorithm was employed to optimize the parameters of XGBoost, resulting in a noteworthy enhancement in the accuracy of XGBoost for mobile price classification. The proposed model was compared to several baseline models. Furthermore, experiments were conducted on various datasets to compare the performance of DBO-XGBoost with standard XGBoost. The results demonstrate that our proposed algorithm effectively improves classification performance and exhibits a degree of generalizability.

II. RELATED WORK

A. Logistics Regression

Logistic Regression (LR) is a widely-used algorithm for linear regression and dichotomous problems. It can also be converted to multiclassification by OvO and OvR methods. Huang [5] employed LR to analyze the impact of housing location and neighborhood on rental prices, achieving good classification performance for room types by utilizing a Softmax function. Literature [6] obtained accurate real-time data through Google Maps and used the LR algorithm to make real-time predictions of house prices. Reference [7] used the LR to analyze and predict electricity prices containing time-series data, demonstrating lower mean square error compared to an artificial neural network model.

B. Decision Tree

Decision Tree (DT) is a common supervised machine learning algorithm that is based on a tree structure for decision making. In [8] , the researchers proposed a classification model based on feature extraction and decision trees for identifying plant leaf diseases. Literature [9] proposed to apply the decision tree model to mobile phone price classification by filtering the features according to feature importance selection.

C. Random Forest

Random Forest (RF) is an integrative learning algorithm based on decision tree algorithm with Bagging strategy, which has the advantage of processing highly dimensional data. Literature [10] used the RF algorithm to predict stock prices and experimentally demonstrates that the RF algorithm outperforms LR regression and Smo regression. In addition to price prediction, the Random Forest model can also be used for symptom classification diagnosis of Alzheimer's disease [11].

D. AdaBoost

Distinct from the Random Forest algorithm, the Adaboost algorithm is an integrated learning algorithm based on the Boosting method. In [12], the Adaboost model was used to the classification of house resale prices and achieved better classification accuracy than random forest and decision tree. In [13], AdaBoost was employed for the detection and classification of lung cancer cells, demonstrating superior performance compared to methods such as SVM .

E. Clustering Algorithm

Distinct from the above algorithms, the clustering algorithm is an unsupervised algorithm. Literature [14] uses a new subspace clustering method for data downscaling and classification. Literature [15] used K-means method to classify cell phone prices, but the results were not satisfactory.

In summary, considering the model performance, this paper uses the supervised integrated learning method XGBoost for improvement.

III. DBO-XGBoost Classification Model

This section first introduces the basic principles of the XGBoost and DBO algorithms, and then describes the construction process of the DBO-XGBoost model.

A. XGBoost

XGBoost is an enhanced and efficient algorithm based on GBRT, integrating a linear scale solver and a tree learning algorithm [16]. Compared with the traditional Boosting method, the XGBoost algorithm performs a second-order Taylor expansion on the loss function and introduces two regularization parameters, L1 and L2, to find the global best solution, measure the decline of the objective function and the complexity of the model as a whole, efficiently improving the generalization capacity of the model.

Suppose that for the data set consisting of p features and a total of m samples. The model uses n (n = 1, 2,..., N) regression trees and F is the set space of regression trees. $\hat{y}_i$ is predicted value and y_i is true value.The model and objective function can be defined as Eq. (1) and (2).

$$\hat{y}_i=\sum_{n=1}^{N} f_n x_i ,f_n \in F \tag{1}$$

$$Obj=\sum_{i=1}^{m} l(y_i,\hat{y}_i)+\sum_{n=1}^{N}\Omega(f_k) \tag{2}$$

XGBoost, like GBDT, uses incremental training, where each step adds a new tree to the previous step to fix the deficiencies of the previous tree, and its iterative process is defined as Eq. (3).

$$\hat{y}_i^{(t)}=\hat{y}_i^{(t-1)}+f_t(x_i) \tag{3}$$

Substituting Eq. (3) into Eq. (2), we obtain the objective function for t iterations.

$$Obj^{(t)}=\sum_{i=1}^{m} l\left(y_i,\hat{y}_i^{(n-1)}+f_t(x_i)\right)+\sum_{k=1}^{K}\Omega(f_k) \tag{4}$$

By performing a second-order Taylor expansion on the target function, the target function can be estimated as Eq. (5). In Eq. (6): T and ω are the number of leaf nodes and leaf weight values, respectively; γ is the leaf penalization factor; λ is the leaf weight penalty factor.

$$Obj^{(n)}\cong\sum_{i=1}^{m}\left[\partial_{\hat{y}_i^{(n-1)}} l(y_i,\hat{y}_i^{(n-1)})f_t(x_i)+\frac{1}{2}\partial^2_{\hat{y}_i^{(n-1)}} l(y_i,\hat{y}_i^{(n-1)})f_t^2(x_i)\right] + \Omega(f_k)+\sigma \tag{5}$$

$$\Omega(f_k)=\gamma T+\frac{1}{2}\lambda \parallel \omega^2 \parallel \tag{6}$$

B. Dung beetle optimizer

Dung beetle optimizer (DBO) is proposed by Xue and Shen, which mainly simulates dung beetle's rolling, dancing, foraging, stealing and reproduction behaviors [17]. It combines global and local search strategies, offering fast convergence and high accuracy.

1) Rolling ball: The path of the dung beetle is affected by the light source, and during its movement, and the update of the individual position is defined by Eq. (7). In the following equation n denotes the number of iterations, x denotes the position of the jth dung beetle in generation n and k is the deflection coefficient. b is the position update weight, μ is a natural coefficient taking the value -1 or 1, X^{pw} is the global worst position, and Δx simulates the change in light intensity.

$$\begin{gathered} x_j(n+1)=x_j(n)+\mu\times k\times x_j(n-1)+b\times\Delta x \\ \Delta x=\left|x_j(n)-X^{pw}\right| \end{gathered} \tag{7}$$

2) Dancing: The dung beetle reorients itself by dancing when it encounters an obstacle that hinders its progress. Reference [17] used a tangent function to model this behavior. The positional simulation of the dancing behavior is defined as Eq. (8).

$$\begin{gathered} x_j(n+1)=x_j(t)+\tan(\theta)\left|x_j(n)-x_j(n-1)\right| \\ \theta\in(0,\Pi) \end{gathered} \tag{8}$$

3) Reproduction: Dung beetles lay their eggs in a safe range, for which a restrict selection strategy is defined to simulate the dung beetle spawning [17]. In Eq. (9), X^{cg} denotes the current global optimal position, Lp^r and Up^r denote the maximum and minimum limits of the optimization problem, respectively.

$$\begin{aligned} \mathrm{Lp^r} &= \max(\mathrm{X^{cg}} \times (1-\mathrm{R}), \mathrm{Lp}) \\ \mathrm{Up^r} &= \min(\mathrm{X^{cg}} \times (1+\mathrm{R}), \mathrm{Up}) \\ \mathrm{x_i(n+1)} &= \mathrm{X^g} + \mathrm{b_1} \times \left(\mathrm{B_j(n)} - \mathrm{Lb^r}\right) \\ &+ \mathrm{b_2} \times \left(\mathrm{B_j(n)} - \mathrm{Up^r}\right) \end{aligned} \tag{9}$$

4) Foraging: Some dung beetles in the population engage in foraging behavior, and the optimal foraging area boundary is defined as Eq. (10), where X^g denotes the global optimal position, Lp^b and Up^b denote the maximum and minimum limits of the optimal foraging area, respectivel. D_1 is a random number obeying normal distribution, and D_2 is a vector of random variables ranging from 0 to 1.

$$\begin{aligned} \mathrm{Lp^b} &= \max(\mathrm{X^g} \times (1-\mathrm{R}), \mathrm{Lp}) \\ \mathrm{Up^b} &= \min(\mathrm{X^g} \times (1+\mathrm{R}), \mathrm{Up}) \\ \mathrm{x_j(n+1)} &= \mathrm{x_j(n)} + \mathrm{D_1} \times \left(\mathrm{x_j(n)} - \mathrm{Lp^b}\right) \\ &+ \mathrm{D_2} \times \left(\mathrm{x_j(n)} - \mathrm{Up^b}\right) \end{aligned} \tag{10}$$

5) Stealing: In addition to foraging behavior, some dung beetles will compete with other dung beetles for food, and with more "optimal" food locations like moving, this thieving dung beetle's location update is defined by Eq. (11), where V is a random vector obeying a normal distribution of size D and S is a constant.

$$\mathrm{x_j(n+1)} = \mathrm{X^g} + \mathrm{S} \times \mathrm{V} \times \left(\left|\mathrm{x_j(n)} - \mathrm{X^n}\right| + \left|\mathrm{x_j(n)} - \mathrm{X^g}\right|\right) \tag{11}$$

Suppose a population of dung beetles of size 30 performs rolling, breeding, predation, and stealing behaviors with numbers of individuals 6, 6, 7, and 11, respectively.

C. Construction of DBO-XGBoost model

Based on the principle of XGBoost and the theory of DBO algorithm, the cell phone price classification with DBO optimized XGBoost parameters is constructed with the following process.

1) Data pre-processing: remove missing values in the data set, normalize the data, filter the features according to the importance of the features calculated by the decision tree, and reduce the dimensionality of the filtered data by PCA principal component analysis.

2) Dividing the dataset: 70% of the dataset is used as training set, 15% as validation set and 15% as test set.

3) Model initialization: Initialize the DBO population, set the parameters in XGBoost as the surrogate parameters, and use the XGBoost classification accuracy of the model in the validation set as the fitness function value.

4) DBO algorithm iteration： The population position is updated according to the DBO algorithm process, its fitness value is calculated, the individual optimal value and the global optimal value are updated, and the iteration ends when the termination condition is satisfied. Selecting the optimal combination of parameters to construct a DBO-XGBoost classification model.

The detailed model training process is illustrated in Fig.1.

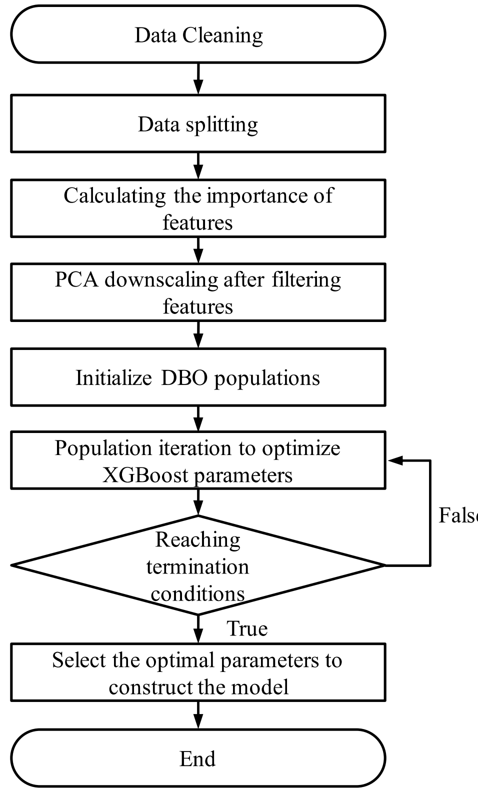

Fig. 1. DBO-XGBoost model training process.

IV. Experiment and Analysis

A. Model Evaluation Metrics

The parameters used in the evaluation of model effectiveness include accuracy, recall, accuracy and F1 score.

- Accuracy: The percentage of correct predicted results to the total sample.
- Precision: The percentage of the number of correctly classified positive samples (TP) to the total samples predicted to be positive (TP+FP).
- Recall: The number of samples correctly classified as positive class (TP) as a proportion of the actual number of positive class samples (TP+FN).
- F1 score: The summed mean of Accuracy and Recall.

B. Data Preprocessing

The dataset comes from the Kaggle community and the publisher wants to use the dataset to find the relationship between the hardware features and the price of the phone [18]. The dataset includes predicted price ranges instead of exact values, indicating the relative magnitude of prices. It comprises 2000 data entries with 20 attributes, including six discrete features, 13 continuous features, and one label representing price categories. Price range labels are divided into four classes (0, 1, 2, 3).

After sorting and confirming the data types and absence of missing values, the data are trained using Gini decision trees and the feature importance ranking results are plotted as shown in

Fig. 2. Features such as blue, dual_sim, and wifi can be removed to decrease the number of features and the overfitting phenomenon.

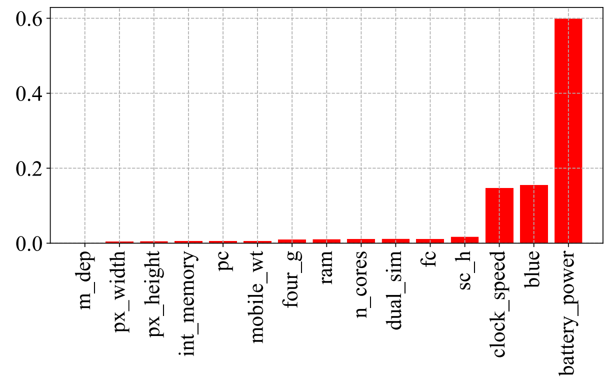

Fig. 2. The results of feature importance.

The dataset after deleting the above features is used to construct new features by principal component analysis, and the new features eliminate the redundant information of the original features to make the data more distinguishable. Fig. 3 plots the interpretable variance images of the features after PCA processing, and the data are dimensionalized according to the size of the interpretable variance. In order to find the appropriate data dimension, the data were downscaled to 5, 10, and 15 with PCA and trained on standard XGBoost, respectively. The experimental results are shown in Table I.

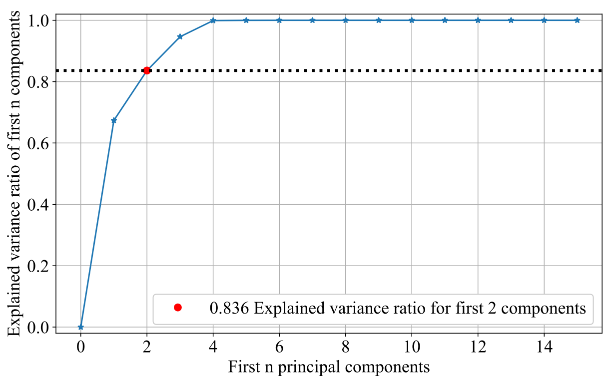

Fig. 3. PCA explained variance ratio.

TABLE I. COMPARISON OF PCA DOWNSCALING IN SEVERAL DIMENSIONS

Model	Accuracy		
	Dimension=5	***Dimension=10***	***Dimension=15***
LR	0.543	0.552	0.552
Decision Tree	0.853	0.853	0.853
Random Forest	0.913	0.903	0.847
AdaBoost	0.715	0.715	0.715
XGBoost	0.927	0.915	0.912

According to the results in Table I, the data reduced to 5 dimensions by PCA performs better on XGBoost, Random Forest algorithm, so the filtered data will be reduced to 5 dimensions by PCA.

C. Application of DBO-XGBoost model

XGBoost uses a gradient boosting tree as the base learner, and the main parameters of the XGBoost model include: number of base learners (n_estimators), maximum depth of base learners (max_deepth), learning rate, downsampling rate (subsample), L1 regularization weight (reg_alpha), L2 regularization weight (reg_lambda), random sampling ratio of base learner features (colsample_bytree), and the minimum sample weight sum of child nodes (min_child_weight).

Define the adjustment range and default values of each parameter in XGBoost as shown in Table II. Experiments are conducted on the number of base learners, maximum depth of base learners, min_child_weight, and subsample under the condition of using default parameters to discuss the effects of the parameters on the model effects.

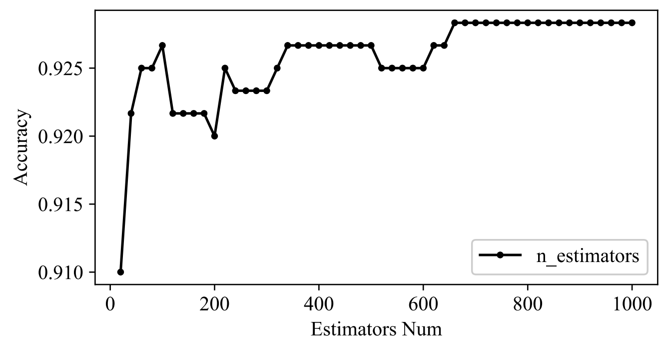

Fig. 4. Effect of the number of base learners on XGBoost

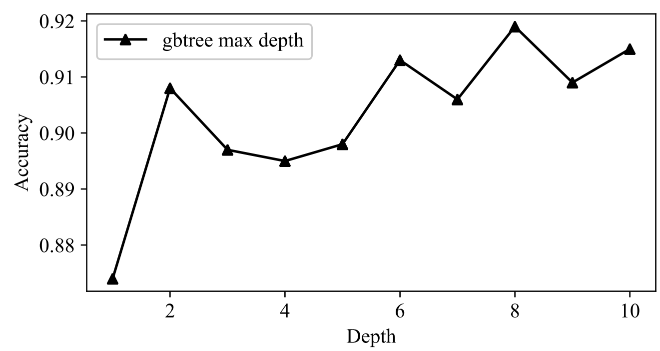

Fig. 5. Impact of gradient boosting tree maximum depth on XGBoost.

The effects of the amount of gradient boosting trees and the maximum depth on the model are shown in Fig. 4, 5. From Fig. 4, it can be seen that: as the number of base learners increases, the model classification accuracy reaches the maximum when the number of trees is 700, and the accuracy does not change significantly as the number of regression trees rises. As can be seen from Fig. 5, for the maximum depth of the regression tree, the error is minimized when it is equal to 8. On the parameter optimization of XGBoost, two approaches are taken to conduct the experiments separately. The first approach is noted as DBO-XGBoost(1), which is using the DBO algorithm to optimize all the parameters of Table II. The second approach, denoted as DBO-Xgboost(2), is to fix the number of base learners and the maximum depth and optimize the other parameters.

During our experiments, we leverage the validation set metrics to fine-tune these parameters, enabling us to mitigate overfitting issues that may arise in the training set. This approach allows us to rectify overfitting based on the insights gained from the validation set, ensuring better generalization performance of the XGBoost model. The iterative process of the DBO-XGBoost model is shown in Fig. 6.

TABLE II. MODEL PARAMETERS AFTER OPTIMIZATION WITH DBO

Parameter	**Range of value**	**Default value**	**Value**	
			DBO-XGBoost(1)	***DBO-XGBoost(2)***
n_estimators	[10-1000]	50	344	700
max_deepth	[2-10]	6	4	8
learning rate	(0-1]	0.3	0.8604	0.1052
min_child_weight	(0-1]	1	0.3573	0.2637
subsample	[0.2-1]	1	0.7049	0.3661
colsample_bytree	[0.2-1]	1	1.0	1.0
L1	[0-1]	1	0.3281	0.1145
L2	[0-1]	1	0.9279	0.8397

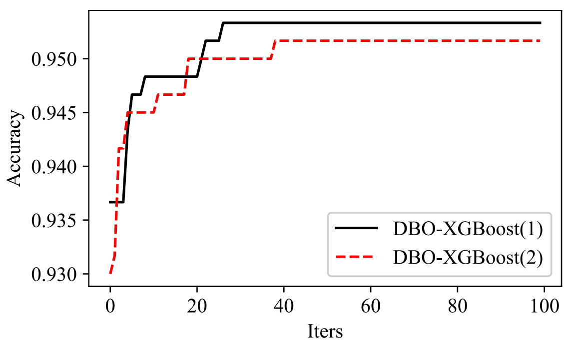

Fig. 6. Model iteration process.

After the experiments as in Fig. 6, we choose the parameters trained by the DBO-XGBoost(1) method with the highest accuracy as the optimal parameters for the model.After several experiments, the accuracy of the test set reaches 95.5% when the number of population iterations reaches 100. To validate the DBO-XGBoost effect, we trained many other machine learning algorithms for comparison, and the results are shown in Table III.

TABLE III. COMPARISON OF THE EFFECT FROM DIFFERENT MODELS TRAINED IN THIS PAPER

Model	**Recall**	**Precision**	**Accuracy**
LR	0.573	0.682	0.551
KNN	0.878	0.873	0.873
MLP	0.841	0.848	0.845
DecisionTree	0.861	0.856	0.853
RandomForest	0.907	0.905	0.905
AdaBoost	0.732	0.725	0.715
XGBoost	0.921	0.919	0.918
DBO-XGBoost	**0.955**	**0.956**	**0.955**

As shown in Fig.7, we plot the test set classification confusion matrix for several algorithms.

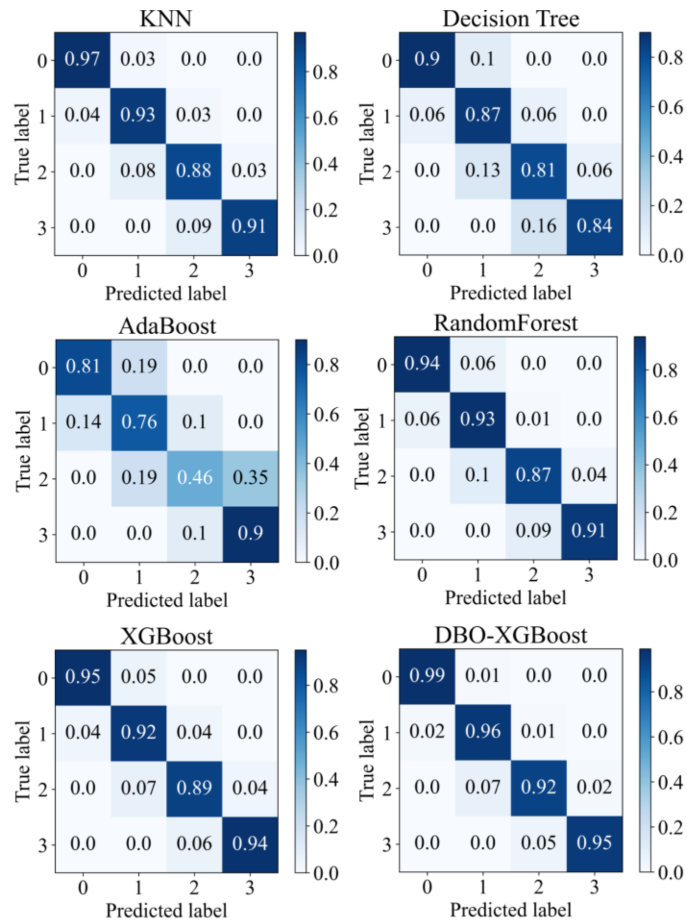

Fig. 7. Confusion matrix for classification of several algorithms.

From the confusion matrix in Fig. 7, it can be seen that DBO-XGBoost outperforms the other algorithms in terms of classification accuracy on all categories, and the data with label 2 has the lowest classification accuracy. The results in Table III show that DBO-XGBoost achieves a 3.7% improvement over the standard XGBoost algorithm and significantly outperforms other machine learning algorithms. We also compared the improved model with the effectiveness of other studies on mobile phone price classification tasks, and the results are shown below.

TABLE IV. COMPARISON OF MODELS IN DIFFERENT PAPERS

Author	**Algorithms**	**Accuracy**
Kumuda S et al. [15]	K-Means	0.89
Asim & Khan [9]	Naive Bayes	0.75
	Decision Tree	0.78
Pipalia & Bhadja [2]	KNN	0.55
	Decision Tree	0.82
	SVM	0.84
	Linear Regression	0.91
	Gradient Boosting	0.90
Sakib et al. [4]	Decision Tree	0.87
	Random Forest	0.92
Our Study	**DBO-XGBoost**	**0.96**

The results in Table IV show that DBO-XGBoost achieves the best classification performance. In addition to the task of classifying mobile prices, we conducted experiments on several Kaggle open-source datasets to compare the performance of DBO-XGBoost and standard XGBoost. The results of these experiments are shown in Table V.

TABLE V. COMPARISON OF CLASSIFICATION ACCURACY RESULTS BETWEEN DBO-XGBOOST AND XGBOOST

Data set	DBO-XGBoost	XGBoost
Heart	0.912	0.846
Cancer	0.977	0.947
Drug	0.983	0.917
Wine Quality	0.680	0.654
Credit Customers	0.793	0.743

From Table V, it can be observed that DBO-XGBoost outperforms XGBoost on all five commonly used Kaggle community classification datasets. This indicates that utilizing DBO optimization with XGBoost is effective and exhibits a certain degree of generalization capability.

V. CONCLUSION

In this paper, we propose an improved model for optimizing XGBoost parameters using DBO: DBO-XGBoost. To accomplish the mobile phone price classification task, DBO-XGBoost is combined with feature filtering and PCA dimensionality reduction to form a classifier. The experiments first compare KNN algorithm, Logistic Regression algorithm, Multilayer Perceptron algorithm, Decision Tree algorithm, Random Forest algorithm, AdaBoost algorithm, and XGBoost model using grid method to search parameters on the mobile phone price dataset. The results show that the classifier using DBO-XGBoost improves the classification accuracy while reducing the dimensionality of the data. We also conducted experiments on several classification datasets, and the classification accuracy was effectively improved, which verified that the model has certain generalization property. Therefore, the model not only provides a new approach for mobile phone classification tasks, but also provides new ideas for classification tasks in other fields. However, the DBO algorithm for parameter optimization also suffers from the issue of high time cost. In our future work, we will strive to further enhance the algorithm to reduce the training time of the model.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 62073300).

REFERENCES

[1] J. Liu and Z. Mo, “The effects of review’s mobile phone price on consumers’ purchase intention: An event-related potential study.,” *Journal of Neuroscience, Psychology, and Economics*, vol. 14, no. 4, pp. 197–206, Dec. 2021, doi: 10.1037/npe0000152.

[2] K. Pipalia and R. Bhadja, *Performance Evaluation of Different Supervised Learning Algorithms for Mobile Price Classification*. 2020.

[3] M. Cetin and Y. Koc, “Mobile Phone Price Class Prediction Using Different Classification Algorithms with Feature Selection and Parameter Optimization,” in *2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, Ankara, Turkey: IEEE, Oct. 2021, pp. 483–487. doi: 10.1109/ISMSIT52890.2021.9604550.

[4] A. H. Sakib, A. K. Shakir, S. Sutradhar, MD. A. Saleh, W. Akram, and K. B. MD. B. Biplop, “A hybrid model for predicting Mobile Price Range using machine learning techniques,” in *2022 The 8th International Conference on Computing and Data Engineering*, Bangkok Thailand: ACM, Jan. 2022, pp. 86–91. doi: 10.1145/3512850.3512860.

[5] Z. Huang, “Logistic Regression in Rental Price and Room Type Prediction Based on Airbnb Open Dataset,” in *2022 6th International Conference on E-Commerce, E-Business and E-Government*, Plymouth United Kingdom: ACM, Apr. 2022, pp. 260–265. doi: 10.1145/3537693.3537732.

[6] A. Gupta, S. K. Dargar, and A. Dargar, “House Prices Prediction Using Machine Learning Regression Models,” in *2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, Dec. 2022, pp. 1–5. doi: 10.1109/ICMNWC56175.2022.10031728.

[7] K. L. Chowdary, Ch. N. Krishna, K. S. Manaswini, and B. Jithendra, “Electricity Price Prediction using Machine Learning,” in *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, Feb. 2023, pp. 611–615. doi: 10.1109/ICAIS56108.2023.10073777.

[8] B. Rajesh, M. V. Sai Vardhan, and L. Sujihelen, “Leaf Disease Detection and Classification by Decision Tree,” in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, Jun. 2020, pp. 705–708. doi: 10.1109/ICOEI48184.2020.9142988.

[9] M. Asim and Z. Khan, “Mobile Price Class prediction using Machine Learning Techniques,” *IJCA*, vol. 179, no. 29, pp. 6–11, Mar. 2018, doi: 10.5120/ijca2018916555.

[10] Moch. Lutfi, S. P. Agustin, and I. Nurma Yulita, “LQ45 Stock Price Prediction Using Linear Regression Algorithm, Smo Regression, And Random Forest,” in *2021 International Conference on Artificial Intelligence and Big Data Analytics*, Oct. 2021, pp. 1–5. doi: 10.1109/ICAIBDA53487.2021.9689749.

[11] A. Parameswari, K. V. Kumar, and S. Gopinath, “Thermal analysis of Alzheimer’s disease prediction using random forest classification model,” *Materials Today: Proceedings*, vol. 66, pp. 815–821, 2022, doi: 10.1016/j.matpr.2022.04.357.

[12] P. Durganjali and M. V. Pujitha, “House Resale Price Prediction Using Classification Algorithms,” in *6th IEEE International Conference on “Smart Structures and Systems”, ICSSS 2019, March 14, 2019 - March 15, 2019*, in 6th IEEE International Conference on “Smart Structures and Systems”, ICSSS 2019. Chennai, India: Institute of Electrical and Electronics Engineers Inc., 2019. doi: 10.1109/ICSSS.2019.8882842.

[13] P. Thamilselvan, “Lung Cancer Prediction and Classification Using Adaboost Data Mining Algorithm,” *IJCTE*, vol. 14, no. 4, pp. 149–154, 2022, doi: 10.7763/IJCTE.2022.V14.1322.

[14] Department of Information Science and Engineering, Shibaura Institute of Technology, Tokyo, Japan, V.-D. Minh, and M. Kimura, “Subspace-Based Method to Improve Classification Accuracy of High-Dimensional Data,” *IJCTE*, vol. 10, no. 6, pp. 180–184, 2018, doi: 10.7763/IJCTE.2018.V10.1222.

[15] Department of Electronics and Communication in NIE Institute of Technology, Mandya (Karnataka), India., Kumuda, V. Karur, Department of Electronics and Communication in NIE Institute of Technology, Mandya (Karnataka), India., K. B. S E., and Department of Electronics and Communication in NIE Institute of Technology, Mandya (Karnataka), India., “Prediction of Mobile Model Price us ing Machine Learning Techniques,” *IJEAT*, vol. 11, no. 1, pp. 273–275, Oct. 2021, doi: 10.35940/ijeat.A3219.1011121.

[16] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[17] J. Xue and B. Shen, “Dung beetle optimizer: a new meta-heuristic algorithm for global optimization,” *J Supercomput*, vol. 79, no. 7, pp. 7305–7336, May 2023, doi: 10.1007/s11227-022-04959-6.

[18] “Mobile Price Classification.” https://www.kaggle.com/datasets/iabhishekofficial/mobile-price-classification (accessed May 02, 2023).