Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/6/20230933

The transaction price prediction of second-hand cars based on model fusion

Zhiqiu Huang

School of Computer Engineer and Science, ShanghaUniversity, Shanghai, 200444, China.

1621471688@shu.edu.cn

Abstract. With the gradual improvement of the second-hand car trading market, buyers and sellers are increasingly demanding a more orderly market, and price forecasting plays an important role in regulating the market order. On one hand, buyers hope to buy second-hand cars with better quality at a lower price. On the other hand, the seller wants to know what the buyer's demand is, so as to better sell second-hand cars in the market. Therefore, the reasonable prediction of price plays an important role in regulating the buying and selling market. At present, some machine learning models, such as neural networks, tree models are very mature in regression prediction technology. By using the above algorithm for modeling, the model can play the function of reasonably predicting prices, so as to meet the needs of both buyers and sellers. Therefore, in this paper, the author predicts the second-hand car transaction price through modeling around some machine learning algorithms, and analyses the performance of the model.

Keywords: machine learning, price prediction, model fusion, second-hand car.

1. Introduction

According to online open source data, since the implementation of the administrative measures for the circulation of second-hand cars in 2005, China's second-hand car market has stepped into the growth period. Since 2015, the scale of second-hand car transactions has risen rapidly, and industry integration has accelerated the elimination [1, 2]. From 2004 to 2015, the historical trading volume of second-hand cars also increased from 250000 to 14.34 million [3]. In 2021, 17.59 million second-hand cars were traded in China. And in 2022, The trading volume of second-hand cars is still further increased. This phenomenon that the sales volume is increasing in the second-hand car market year by year can be explained from both the buyer and the seller. For buyers, on one hand, with the standard of living getting higher and higher, people not only shorten their commuting time by buying cars, but also take buying famous brand cars as a symbol of status. On the other hand, when the overall performance of a second-hand car is almost the same as that of a new car, second-hand cars have great advantages in terms of price compared with new cars. For sellers, due to the policy support and the gradual increase of market demand, they gradually expand their business scale, and the competition in the seller's market is becoming increasingly fierce, which also lead to the increase of the size and quality of the seller's market year by year [4]. Therefore, second-hand cars become increasingly inseparable in people's lives. If sellers can integrate the transaction records in the market and screen out valuable information, they will grasp consumers' preferences for second-hand cars and improve sales. The prediction of the transaction

DOI: 10.54254/2755-2721/6/20230933

price of used cars can not only guides the prices of buyers, but also can let the market be aware which aspect consumers value more of second-hand cars. Currently, some used car trading websites, such as "guazi second-hand car" and "renrenche", have indicated the guide prices of different second-hand cars on the official website. These guide prices are the predicted prices obtained by the platform through extracting useful information from massive data and modelling with this information [5]. From the successful experience of different second-hand car trading platforms, it can be concluded that a perfect used car price prediction model is crucial to the sales of used cars and will bring more credibility to the platform. Therefore, this experiment will build a model around the prediction of second-hand car transaction price, and make the deviation between the true price and predicted one as small as possible, so as to provide buyers with a reasonable and reliable guide price. The second-hand car transaction price model in this experiment will be constructed through the tree model and neural network model that have good performance in solving regression problems in current machine learning [6, 7].

2. Method

The dataset comes from the "Alibaba Cloud" website [8], where 150000 pieces of data were collected from the trading platform. It contains 31 characteristics, 15 of which are anonymous variables and desensitize some variables. Based on this, since the data has been desensitized, and many prices have been lost for each feature, it is difficult to complete a more specific analysis from the feature set. Therefore, the focus in this experiment is to select different models for multiple modelling through different data pre-processing methods, calculate and compare the evaluation indicators of each model, get the best model and analyse the relevant reasons.

2.1. Dataset

Features used in the experiment have 31 dimensions, including the brand, fuel type, sellers and so on. Some representative examples and features are shown in Table 1 and some representative feature's statistical information are shown in Table 2. In order to further explore the characteristics of the data, it is also necessary to view the statistical information of each feature, including the mean value, standard deviation, maximum value, etc., so as to facilitate subsequent data normalization or filling in missing values. It can be seen that the data is very dispersed, including integer, floating point, positive number, negative number, and date. At the same time, for the target variable price，the data presents a skewed distribution. In addition, if the data are converted into numerical values, the gap between the data is particularly large, some tens of thousands, some fractions of a few centimetres, which will inevitably ignore the role of some values in the prediction, so it is necessary to normalize them.

In addition, it is also necessary to analyse other information, such as missing values and duplicate value. By analysing the distribution of price, plotting it, and comparing it with common distribution curves, it has been found that price is more consistent with the unbounded Johnson distribution. At the same time, most of the price data are below 20000 as shown in Figure 1. Based on this, a log transformation should be implemented on the price.

Table 1. Interception of partial characteristics of dataset.

SaleID	regDate	brand	power	kilometre	v_9	v_10	v_14
0	20040402	6	60	12.5	0.097462	-2.881803	0.914762
1	20030301	1	0	15.0	0.020582	-4.900482	0.245522
2	20040403	15	163	12.5	0.027075	-4.846749	-0.229963
3	20120103	10	193	15.0	0.000000	-4.509599	-0.478699

DOI: 10.54254/2755-2721/6/20230933

Table 2. Statistical information of representative features.

	bodyType	gearbox	regionCode	seller	v_1
Count	145494.0	144019.0	150000.0	150000.0	150000.0
Mean	1.792369	0.224943	2583.077267	0.000007	-0.044809
Std	1.760640	0.417546	1885.363218	0.002582	3.641893
Min	0.000000	0.000000	0.000000	0.000000	-4.295589
25%	0.000000	0.000000	1018.000000	0.000000	-3.192349
50%	1.000000	0.000000	2196.000000	0.000000	-3.052671
75%	3.000000	0.000000	3843.000000	0.000000	4.000670
Max	7.000000	1.000000	8120.000000	1.000000	4.000670

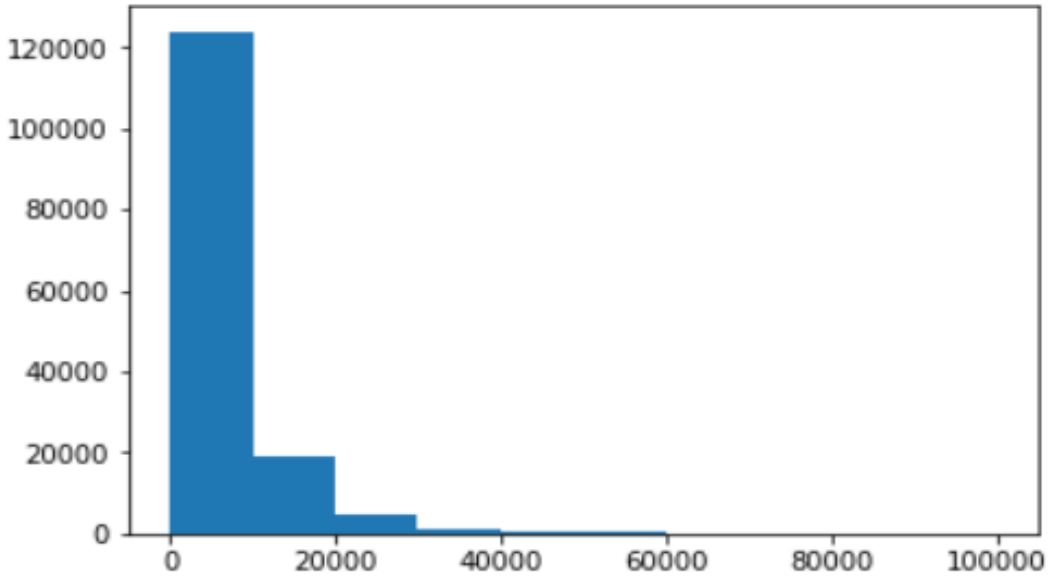

Figure 1. Price distribution.

2.2. Data pre-processing

Based on the above dataset overview, a general understanding of the characteristics of each feature of this dataset is obtained, and the data pre-processing will be divided into four parts: data cleaning, data dimensionality reduction, data normalization and feature construction.

2.2.1 Data cleaning. In terms of data cleaning, it can be divided into missing value handling, outlier handling and duplicate value handling.

First of all, for missing value processing, it can be obtained from the dataset overview described above that except for the three features of "bodyType", "fuelType" and "gearbox", the rest of the features have no missing values. On one hand, if there are too many missing values, it can be considered being deleted. On the other hand, if there are relatively few missing values, it can be filled through the tree model such as LGB, and let the tree optimize itself. The missing value scale is shown in the Table 3. So, in this experiment, the tree mode is made to complete the missing values itself, and the neural network is made to fill in the missing values by using mean.

DOI: 10.54254/2755-2721/6/20230933

Table 3. The missing values.

Feature	Missing value
bodyType	4506
fuelType	8680
gearbox	5981

Second, for outlier handling, the feature "notRepaireDamage" should be an integer variable, but there are many character variables, so they are actually missing values. Therefore, the outliers in this experiment are replaced with the mode. In addition, such as the "power" feature, the range given in the dataset is [0,600], but some data values exceed 600, so these values are replaced by 600, and those values less than 1 are replaced by 1.

Finally, for duplicate value processing, since the SaleID given by the dataset is unique, checking for duplicate values only needs to filter the SaleID, and judges whether there are duplicate SaleID. After filtering, there are no duplicate values in this dataset.

2.2.2 Data dimensionality reduction. It can be learned from the dataset that there must be a linear relationship between some feature, such as the sale time of cars, the kilometres travelled by cars and the date of car registration. According to common sense, under the condition that factors such as automobile brand remain unchanged, and only the sale time of cars, the kilometres travelled by cars and the date of car registration are changed, the earlier the sale time and registration data are, the longer the car may travel. As one of the important indicators of automobile performance, the travel kilometres are directly proportional to the final transaction price. At the same time, as the data has been desensitized, it is uncertain that all variables are linear with the price. Therefore, two more classical methods of linear and nonlinear dimensionality reduction are applied in this experiment to reduce data dimensions: PCA and ISOMAP.

PCA is a classic method applied to dimension reduction, and it' s purpose is to find a linear mapping that makes the original input vector to be mapped to a lower dimensional vector space, while maximizing the variance between vectors. Moreover, the linear mapping is not an ordinary linear mapping, but orthogonal. The solution steps of PCA [9] can be divided into the following six steps: 1. Removing average, that is, each feature minus its respective average. 2. Calculating the covariance matrix by the formula. 3. Calculating the eigenvectors and eigenvalues of the covariance matrix by the formula. 4. Make features sorted from largest to smallest. 5.Select the k largest values. Then the eigenvector matrix could be constructed by corresponding k eigenvectors. 6. Make the original features to be converted to the new space which is made up of k feature vectors obtained above.

It can be learned that the last two steps realize feature compression. Since the covariance matrix possesses the character of symmetry, the k feature vectors have orthogonality to each other. Since the orthogonality is linear and uncorrelated, PCA can eliminate the correlation between original features

The basic idea of ISOMAP [10] is to find the "geodesic distance" between any two high-dimensional data points, and to achieve the "geodesic distance" approximately unchanged in low dimensional space through mapping. The basic algorithm steps are as follows: 1. For each data point x, calculate its k-nearest neighbour. 2. Set Euclidean distance as the distance between x and its corresponding k-nearest neighbour. 3. Set infinity as the distance between x and other points. 4. Calculate the distance between any two points through the shortest path algorithm, and thus the distance matrix is obtained. 5. Obtain the matrix Z of the sample set through the multidimensional scaling MDS algorithm in the lower dimensional space.

In general, most of the data in high-dimensional space will be embedded in a manifold model, rather than randomly distributed in high-dimensional space. The number of high-dimensional data with the same characteristics is the dimension of the manifold model, that is, the dimension of the target space which is needed to reduce dimensionality. At the same time, if the data is not embedded in a manifold

DOI: 10.54254/2755-2721/6/20230933

model in the high-dimensional space, the ISOMAP algorithm may not be suitable. If the data is classified, and the data will not be embedded in a manifold model, the effect of the ISOMAP dimension reduction algorithm is poor, but the data is continuous, and the data is likely to be embedded in a manifold model. At this time, the effect of the ISOMAP algorithm will be better.

At the same time, feature selection can also be used to reduce data dimensions. There are many features in the data set, such as SaleID, and automobile transaction name, are only the codes assigned by the vehicle exchange, which have little impact on the project, so they can be deleted directly. This part is mainly carried out in the following feature construction.

2.2.3 Data normalization. In this experiment, the expected transaction price prediction model is built based on the weighted fusion by combining a tree model and a neural network. In the tree model, data normalization is not required. It is mainly because the scaling of values has nothing to do with the position of the split point, scaling pre-processing the data does not have any effect on the tree model, and the construction of the tree model is a process of constantly looking for the best split point, instead of using gradient descent to continuously update. However, in neural network model, when multiple feature attributes of the data have different dimensions, but they need to use gradient descent to continuously update to build the model. If there are different ranges of values between features, so when the gradient is updated, it will oscillate back and forth. Furthermore, this process that the gradient reaches the local optimal value or the global optimal value will take a long time. The addition of normalization can increase the model learning ability to a certain extent. In the dataset, some feature variables are distributed 0-1, and some feature variables can have values up to 10^7, however, the size of the amount of data does not necessarily judge the importance of a feature. Since the variance of the features of the dataset is small, each feature is linearized to between [0,1] by max-min normalization, the formula is shown below:

$$x = \frac{x - \min}{\max - \min} \tag{1}$$

What's more, as is mentioned above, price is more consistent with the unbounded Johnson distribution, but some algorithms require that the data conform to a normal distribution, if the target variable is seriously skewed, it will affect the prediction effect of the model, so the logarithmic normalization will be carried out. So, in the experiment, the logarithmic transformation is applied to the target variable price in order to make it follow a normal distribution. Besides, it will not change the nature and relevance of the data to use the log transform, but to reduce the scale of the variable, which can make the final prediction more reliable.

2.2.3 Feature construction. In the dataset, features include time features, category features and anonymous features.

For time characteristics, including vehicle registration date, vehicle launch date and sales start time, firstly the vehicle registration date is subtracted from the vehicle launch date to reflect the vehicle use time. It is known from life experience that price is inversely proportional to the duration of use.. Among them, there are some sample time formats that are problematic and cannot be added or subtracted directly. Considering that the number of outliers is relatively small, they are deleted directly.

For category features, they are first applied bucket splitting algorithm. And then features are crossed to get their statistical data, peaks and skews, constructing more features. Through bucket splitting, the following models can be optimized: 1. The discrete sparse vector inner product multiplication is faster. It makes the calculation results easy expand and store; 2. Discrete features are less affected by outliers. For example, if the mileage is greater than the average value, it will be 1; if the mileage is less than the average value, it will be 0. In this way, if the mileage is infinite, it will not cause great interference to the model. 3. LR is a kind of linear model, which is generalized but has limited expression ability. After discretization, each variable is assigned a separate weight, which introduce non-linear, which can increase fitting; 4. After discrete, feature crossover can enhance the expression ability, which further

DOI: 10.54254/2755-2721/6/20230933

brings the introduction of non-linear; 5. The model is more stable after the characteristics are dispersed. For example, the mileage interval will not change because of the extra 1 or 2 kilometres.

For anonymous features, from the initial overview of the dataset, it can see that 14 anonymous features are evenly distributed, and some features with the same distribution can be retained for feature combination to construct more features, especially those strongly related to the predicted goals.

2.3. Model selection

2.3.1 XGBoost. XGBoost is an optimized distributed gradient enhancement library, which aims to achieve efficiency, flexibility and portability. Its basic element is decision tree, which are turned into "weak learners". In traditional GBDT model, CART is used as the base classifier, while in XGBoost model, linear classifier is also supported. Its objective function formula is as follows:

$$\sum_{i}^{n} \frac{1}{2}\left(f_t(x_i) - \frac{g_i}{h_i}\right)^2 + \Omega(h_t) + constan \tag{2}$$

XGBoost 's optimization over GBDT is mainly demonstrated in the following four points: 1. In XGBoost model, a regularizing term which has the number of leaf nodes of the tree and the sum of squares of the score of output's L2 modulus on each leaf node is added to the cost function. In this way, model' s complexity can be controlled because the variance of the model is reduced by the regularizing term, which makes the model simpler, and prevents overfitting; 2. Shrinkage. After XGBoost finishes one round of iteration, it will multiply the weight of the leaf nodes by this coefficient, aiming to weaken the influence of each tree and vacate more room for the later learning. There is a sequence between the decision trees that make up XGBoost: the generation of the latter decision tree takes into account the predictions of the previous one. Therefore, the generation of each decision tree can be regarded as a complete decision tree generation process; 3. Column subsampling. XGBoost uses the strategy of random forests for reference column sampling. This mechanism reduces both overfitting and calculations; 4. Handling of missing values. XGBoost could fill in missing values automatically.

Based on the above characteristics of XGBoost, the reasons for selecting XGBoost in this experiment can be mainly attributed to the following points: 1. a regularizing term is added to punish the training if the model is too complex, making the learned model simpler and avoiding overfitting. 2. Considering that there are too many features in this experiment, the approximation algorithm is chosen to divide the value of each feature into different buckets according to the quantile. The boundary value of the bucket is used as the candidate set of split nodes. Each time traversing, it is not necessary to traverse all feature values, but only several buckets of the feature. This can reduce the number of times traversing feature values. 3. some features of this dataset have missing values, and XGBoost model can directly fill in missing values.

2.3.2 LightGBM. The differences between LightGBM and XGBoost are mainly in the following two aspects: 1. Segmentation algorithm. LightGBM uses the histogram algorithm. It first makes the continuous floating-point eigenvalues discretized and to be turned into k integers. Afterwards, a histogram with width k based on the k integers. When the data is being traversed, the discretized value is used as the index to accumulate statistics in the histogram. When data is traversed once, the optimal separation points could be found according to the histogram accumulation. It is helpful to reduce the calculation. The pre-sorting algorithm requires to calculate the gain of the split once for each eigenvalue, while the histogram algorithm only needs to calculate k times; 2. Decision tree growth strategy. XGBoost adopts a level-wise growth strategy together with depth limitation. Leaves of the same layer can be split by each iteration of the data at the same time, which avoids overfitting. While LightGBM adopts the leaf-wise growth strategy. Each time the leaf with the largest split gain is found from all the current leaves, and then splits, and so on, but may lead to overfitting. To solve this problem, a limit of

DOI: 10.54254/2755-2721/6/20230933

depth on the leaf-wise growth strategy is added to avoid over fitting to the greatest extent while ensuring high efficiency.

The reason why this experiment uses LightGBM is that XGBoost has too much time and space overhead. The histogram algorithm makes the LightGBM model have smaller memory footprint and computing cost. At the same time, LightGBM's leaf-wise growth strategy may produce overfitting. But when the maximum depth of the tree is limited, it can avoid over fitting to the greatest extent while ensuring high efficiency.

2.3.3 Neural network. The back propagation neural network, learns to optimized the weights in the network, targeting to the direction of reducing the total objective function. It is a widely used for forecasting problems.

In this experiment, by analysing the expected results and properties of the model, it is believed that CNN and RNN are suitable for some problems such as image recognition and time series, but not for regression problems, so neural network is the most suitable model. Neural network is highly nonlinear and has strong generalization ability. With the number of layers of the network increasing, the feature representation becomes more abstract and more semantic meaningful. By extracting these features to distinguish things, superior ability to distinguish and classify can be obtained.

2.4. Model building

At the beginning of the article, it is mentioned that since the characteristics of the dataset have been desensitized, it is difficult to analyse many valuable things from the model prediction results, so the focus of the experiment is to evaluate the model performance by establishing different models. In the last section, four different models are mentioned, but the training effect of a single model may not be the best. For example, LightGBM can make up for XGBoost's disadvantage in time and space spending, and XGBoost can also increase the accuracy of the final model. Therefore, the strategy for this experiment is to integrate some models on the basis of a single model in order to obtain a better modelling effect.

2.4.1 Tree Model (XGBoost + LightGBM). The first model is the model fusion using XGBoost and LightGBM. XGBoost is more accurate than LightGBM, but its training speed is not as fast as LightGBM. By adopting simple weighted fusion, the model's accuracy can greatly be improved. Theoretically, the effect of stack fusion should be better than that of weighted fusion, but it may be limited to only two fusion models, which leads to the fact that the effect of stack fusion is not as good as that of weighted fusion. Because tree models have the characteristic of self-selecting features, if the data dimension reduction is added, the model may lose some key features, so in this model, data dimension reduction module, data normalization and feature selection are not added, leaving only data cleaning and feature construction.

First, dataset is split the dataset. Second XGBoost and LightGBM models are defined and kept. Third viewing the results separately, and then weighted fusion of the results is performed to view the fusion effect of the models.

2.4.2 Neural Network Model. Because ordinary fully connected networks are difficult to train due to gradient attenuation when the number of layers is relatively deep, function is defined in the network to adjust the learning rate of the training process. When the epoch reaches 1400, 1700 and 1900, the learning rate will be reduced to 1/10 of the original. During the training, five-fold cross-validation is used to generate five models, and then the prediction results of the five models on the test set were averaged to get the final prediction results. The optimizer uses Adam, the initial learning rate is set to 1e-1, and the learning rate attenuation uses ReduceLROnPlateau, with 2000 iterations and a batch size of 512. In addition, five hidden layers are used in the middle of the neural network model, which will neither over fit nor under fit.

DOI: 10.54254/2755-2721/6/20230933

2.4.3 Final Model (Tree Model + Neural Network Model). In section 2.4.2 and 2.4.3, the tree and the neural networks have been obtained respectively. However, according to the experimental results, the training effect of the weighted ensemble tree model is not good, so in the final model, the tree model uses stacking fusion. The general idea of constructing the final model is to mix the output of the tree model and that of the neural network. Since the tree model and the neural network are completely different architectures, the score output they get is similar, the predicted value is different, often the difference in MAE is about 200, so they can get a better result by mixing, the weighted average selection coefficient is selected 0.5, although the score of the neural network will be a little higher than the tree model, but the highest score is a combination of the optimal output of multiple sets of lines, so it can make up for each other's advantages.

2.5. Evaluation index

2.5.1 Mean Absolute Error (MAE). MAE is short for mean absolute error. The value measures the error between the predicted and the observed values. It is a linear score, and all differences are assigned equal weight on the average value. Small MAE means the prediction is accurate. The following is the calculation formula of MAE, where y_i represents the true value of the i-th sample, $\hat{y}_i$ represents the predicted value of the i-th sample.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{3}$$

2.5.2 R-square. Since MAE can only reflect a relative degree of model quality: the lower the value of MAE is, the lower the model error is, but it is impossible to quantitatively describe the quality of the model. Therefore, the R square of the model is also calculated. R-Square refers to the quality of regression prediction. The maximum value of 1, the closer it is to 1, the better the regression line fits the observed values. The formula is as follows, where y_i represents the true value of the i-th sample, $\hat{y}_i$ represents the predicted value of the i-th sample, and $\overline{y_i}$ Is the mean of the original data.

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{i=1}^{n} w_i |y_i - \hat{y}_i|^2}{\sum_{i=1}^{n} w_i |y_i - \overline{y_i}|^2} \tag{4}$$

3. Result

3.1. Experiment setting

When using training set to train parameters, the entire training set is usually separated into training, evaluation and test set. Because during training, the model usually fits the training set very well, but its generalization ability is usually less satisfactory, then five-fold cross-validation can be used to increase the generalization ability of the model. Five-fold cross-validation is to divide the data into 5 equal parts on average, take one copy for each experiment for testing, and use the rest for training. The average value was calculated for 5 experiments. In this experiment, in order to increase the generalization ability of the model and reduce the error of the model in theory, five-fold cross-validation is applied. It is expected that the model will also have a good performance in the training set rather than just being applied to the training set.

3.2. Tree model training results

For the tree model, XGBoost, LightGBM, and the fused MAE values are respectively calculated and demonstrated in Table 4 and Table 5.

DOI: 10.54254/2755-2721/6/20230933

Table 4. Statistics of price after LightGBM model training.

MAE	685.094950635
MIN	-858.247525369
MAX	91332.8090592
MEAN	5905.80827188
PTP	92191.0565846
STD	7341.16204407
VAR	53892660.1572

Table 5. Statistics of price after XGBoost model training.

MAE	688.745509976
MIN	-896.798
MAX	90702.6
MEAN	5905.57
PTP	91599.4
STD	7352.81
VAR	54063871.65

After that, two models are combined according to the formula below:

$$val_{weight} = \left(1 - \frac{MAE_{lgb}}{MAE_{xgb} + MAE_{lgb}}\right) * val_{lgb} + \left(1 - \frac{MAE_{xgb}}{MAE_{xgb} + MAE_{lgb}}\right) * val_{xg} \quad (5)$$

After fusion, the MAE of tree model result is 666.939086478.

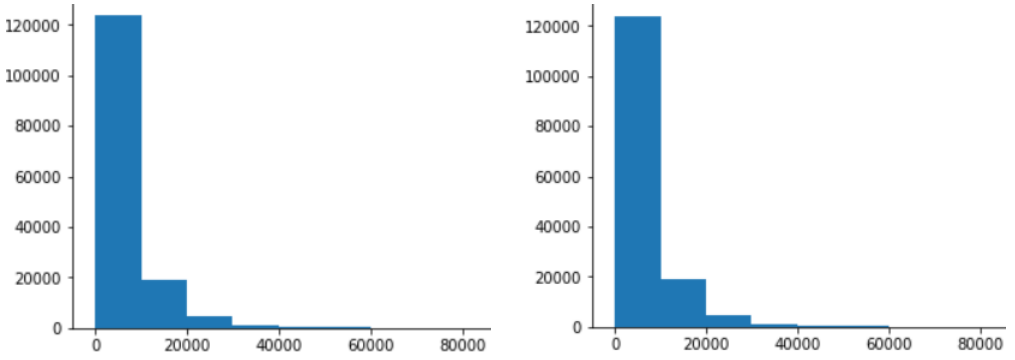

Figure 2. Actual (left) and predicted (right) price distribution.

Table 6 demonstrates the predicted prices' MAE and R-square after Neural Network model training.

DOI: 10.54254/2755-2721/6/20230933

Table 6. Result of the neural network.

MAE	523
R-square	57.9224%

The MAE value of the final model training is 423.

4. Discussion

4.1. Analysis of experimental problems

In this experiment, it can be seen that the training effect of the tree model and the neural network model is not very satisfying. So, some reflections have been made from the analysis: 1. The tree model uses weighted fusion to fuse XGBoost and LightGBM models, but the effect is not good. If stacking fusion is used, the effect is worse, but the effect is better if stacking fusion is used in the final model. This phenomenon may be due to the omission of information capture in the first layer of stacking, so it may be necessary to add raw data to the second layer. 2. The model of the neural network is too simple. Due to computer performance problems and to prevent overfitting, the neural network model has only five hidden layers, too few hidden layers may make the model less generalizable. 3. XGBoost is used in the tree model, but because it needs to use the second order derivative, the objective function does not have MAE, so some custom functions used for approximation are not ideal. 4. The advantages of the neural network are not fully used, and more features should be artificially created; 5. The reasons why the final model works well may be divided into two main points. Firstly, the tree model can automatically compensate for missing values and make feature selection. It is suitable for this dataset with a small number of missing values and large features. Second, the predicted value of this dataset, price, can be seen from the graph that the distribution is more concentrated, which also results in the test values of training data and test data being close, reducing the error.

4.2. Future prospect

Firstly, for the tree model, the parameter adjustments are optimized by Bayesian parameter adjustments, and the MAE values are selected. Later, some systematic methods, such as Bayesian parameter adjustments, could be used. Bayesian optimization finds a value that minimizes an objective function by establishing a surrogate function based on past evaluations of the objective function. However, the result of tree model is not satisfying, so grid tuning can be considered to replace Bayesian parameter adjustments, because when the effect of the algorithm model is not very good, grid tuning traverses through loops, tries each parameter combination, and returns the parameter combination with the best score value.

Secondly, for the neural network model, as many hidden layers and neurons as possible are hoped to be added to make the model more accurate without overfitting. If too few neurons are used in the hidden layer, it will lead to under fitting; If too few neurons are used in the hidden layer, it will lead to overfitting, so it needs plenty of experiments to determine the number of neurons.

Thirdly, in order to improve the fitting ability of the neural network, an attention module can be added.

Fourthly, the effect of the tree model based on stacking is worse than that of the weighted fusion tree model. This may be due to the omission of information capture in the first layer of stacking. Raw data can be added to the second layer for optimization.

Finally, the advantages of the neural network should be fully used, and multiplication processing should be used on 14 anonymous features to obtain 14*14 features.

5. Conclusion

In this paper, the author introduces three models: tree model, neural network model and final model to predict the transaction price of used cars. Among them, the tree model is composed of weighted fusion of XGBoost and LightGBM, and the final model is composed of weighted fusion of tree model and

DOI: 10.54254/2755-2721/6/20230933

neural network model. First, the author makes an overview of the data set, analyses the statistical information of each feature and the distribution of the target variable, and lays the foundation for the subsequent data pre-processing. After that, the author performs data pre-processing in four aspects: data cleaning, data dimensionality reduction, data normalization and feature construction. Among them, in terms of data cleaning, outliers, duplicate values, and missing values have been dealt with emphatically; in terms of data dimensionality reduction, PCA and ISOMAP two dimensionality reduction methods have been used; in terms of data normalization, in order to summarize and unify the statistics of samples distribution, so the values of each feature are mapped to the [0,1] interval; in terms of feature construction, the features mainly include time features, category features and anonymous features. When building the tree model, due to the characteristics of the tree model itself, the three aspects of data dimensionality reduction, data normalization and feature construction are discarded, and only data cleaning is retained. When building a neural network, regularization is used to prevent overfitting by adjusting the regularization coefficient. Finally, when constructing the final model, the final model is obtained by weighted fusion of the previously obtained tree model and the performance of the neural network. At this time, the tree model is obtained by combining XGBoost and LightGBM through stacking. By comparing the above three models, it could be concluded that the results of the final one is superior than that of the neural network model and the tree model. Effectiveness of the final model is very good, and it' s error is relatively small. Generally speaking, it can precisely predict the transaction price of used cars and play a certain guiding role in the market.

References

[1] Beuving, J. J. (2004). Cotonou's Klondike: African traders and second-hand car markets in Bénin. The Journal of Modern African Studies, 42(4), 511-537.

[2] Yakob, R. (2018). Augmenting local managerial capacity through knowledge collectivities: The case of Volvo Car China. Journal of International Management, 24(4), 386-403.

[3] Cheng, W. (2015, June). Research on the Decision Making Model of Purchasing Second-hand Car. In 2015 International Conference on Management, Education, Information and Control, 1288-1294.

[4] Chen, Z., Chen, D., Wang, T., & Hu, S. (2015). Policies on end-of-life passenger cars in China: dynamic modelling and cost-benefit analysis. Journal of Cleaner Production, 108, 1140-1148.

[5] Zhang, W., & Ma, L. (2021). Research and application of second-hand commodity price evaluation methods on B2C platform: take the used car platform as an example. Annals of Operations Research, 1-13.

[6] Çelik, Ö., & Osmanoğlu, U. Ö. (2019). Prediction of the prices of second-hand cars. Avrupa Bilim ve Teknoloji Dergisi, (16), 77-83.

[7] Yadav, A., Kumar, E., & Yadav, P. K. (2021). Object detection and used car price predicting analysis system (UCPAS) using machine learning technique. Linguistics and Culture Review, 5(S2), 1131-1147.

[8] Aliyun (2020) Second hand car price prediction. URL: https://tianchi.aliyun.com/competition/entrance/231784/information

[9] Daffertshofer, A., Lamoth, C. J., Meijer, O. G., & Beek, P. J. (2004). PCA in studying coordination and variability: a tutorial. Clinical biomechanics, 19(4), 415-428.

[10] Balasubramanian, M., & Schwartz, E. L. (2002). The isomap algorithm and topological stability. Science, 295(5552), 7-7.