QBS
Vol. 40, No. 2, pp. 63-69 (2021)
https://doi.org/10.22283/qbs.2021.40.2.63
pISSN 2288-1344 eISSN 2508-7185 **Original Article**

Quantitative Bio-Science
Official Journal of the Institute of Natural Science at Keimyung University
Available Online at https://qbs.kmu.ac.kr:442/

Prediction of Bidding Rate of Residential Auction Items using Model Stacking

Tae Hyun Park[1], Jae Joon Ahn[2], Kyong Joo Oh[3,*]

[1]Department of Industrial Engineering, Yonsei University, Seoul 03722, Korea
[2]Department of Information and Statistics, Yonsei University, Wonju 26493, Korea
[3]Department of Information & Industrial Engineering, Yonsei University, Seoul 03722, Korea
(Received September 6, 2021; Revised October 28, 2021; Accepted November 8, 2021)

ABSTRACT

Recently, as interest in real estate auctions has grown, research on predicting the bidding price has been actively underway. The hedonic price model, one that is traditionally used to evaluate the value of real estate, uses simple linear regression and has the disadvantage of delivering a poor predictive performance. To compensate for this, recent research using machine learning or deep learning has been actively conducted in the real estate sector. It was found that the predictive performance of machine learning surpassed that of the traditional hedonic price model in predicting the bidding rate. In this paper, we try to find a method that performs better than the current machine learning model while predicting the bidding rate. In previous studies, Bagging and Boosting methodologies, among ensemble methodologies, have been used. In this paper, we use the model stacking method to predict the bidding price of a real estate auction.

Key words : Real estate, Auction, Forecasting, Machine learning, Deep learning

1. Introduction

Recently, the interest in investment has increased as the price of various assets has risen sharply. As a result, demand for real estate investment, a traditional investment asset, is also pouring in. There are many ways to invest in real estate assets, and among them, the method through an auction is in the spotlight. Since the auction is a method in which investors take a certain amount of risk, it has the advantage of being able to receive a bidding at a price relatively lower than the market price. In Korea, the auction method for real estate is usually based on a deadline or a period auction. In the write-in bidding method, bids are submitted on the day of the auction, so the bids of other bidders cannot be known. In such a situation, in order to receive a bidding for an auction item most efficiently, it is necessary to accurately predict the bidding rate of the item and receive a bid.

The hedonic price model, which is a linear regression model, has been mainly used to predict the value of real estate. The hedonic price model is a method used for estimating the value of non-market goods. Since residential real estate has both characteristics of market goods and characteristics of non-market goods, this model was mainly used for price prediction. As it is a method to measure the intrinsic value of real estate, the price of real estate is used as the dependent variable, and the intrinsic characteristics of real estate, such as the number of floors, exclusive area, etc., are used as independent variables. However, since it is basically a linear regression model, it assumes equal variance, linearity, normality, and independence [1,2].

Machine learning is used in many ways, such as supervised

* Correspondence should be addressed to Dr. Kyoung Joo Oh, Department of Information & Industrial Engineering, Yonsei University, Seoul 03722, Korea. Tel: +82-2-2123-5720, Fax: +82-2-364-7807, E-mail: johanoh@yonsei.ac.kr
Tae Hyun Park is a Master Candidate, Department of Industrial Engineering, Yonsei University, Seoul 03722, Korea. Jae Joon Ahn is a Professor, Division of Data Science, Yonsei University, Wonju 26493, Korea.

learning, unsupervised learning, and reinforcement learning, and has been in the spotlight in many fields because it shows better performance than the existing statistical-based regression model. The machine learning method basically consists of gradient descent. If the error function is a concave or convex function, there must be a minimum point, and the direction in which the gradient vector has the maximum change is the direction of the gradient. Therefore, if you go in the opposite direction of the slope, you can find the minimum point in principle. In the past, when the number of layers increased, a vanishing gradient problem occurred or the learning time was too long, making it difficult to use. These days, algorithms and optimization functions have been developed to solve the above problems, and computing power has increased exponentially compared to the past, reducing the difficulty of using machine learning [3,4].

Real estate data is unstructured data, and there is no fixed arrangement method and it does not have linearity. In predicting nonlinear data, machine learning methodologies can be more useful than traditional linear regression methodologies. For this reason, recent studies to apply machine learning or deep learning algorithms to evaluating the value of real estate are actively underway.

Recently, machine learning models showing good performance in regression problems are ensemble-based models such as Random Forest, XGBoost, and CatBoost, and deep learning still mainly uses DNN. In previous studies, machine learning techniques showed better performance than simple linear regression. In this paper, we try to show that Model Stacking, an ensemble technique not used in previous studies, can derive better predictions than other existing machine learning methods.

The bidding rate prediction problem has a correct answer, so supervised learning must be used, and since the bidding rate, an independent variable, is a real number, a regression model must be used, so the above-mentioned model is suitable. Machine learning is data-based learning, and the amount of data is large and the better the quality, the better the performance. The quantity can be said to be sufficient.

In real estate auctions, there are several types of objects, and the reason for limiting the analysis to residential objects is that it is not good to analyze all uses with the same model because the distribution or nature of data is different for each use. Residential use accounts for the largest portion of real estate auctions, and as mentioned earlier, about 400,000 cases of data have been accumulated over 10 years, so the number of data is sufficient. This is because a large percentage of those who participate in the auction want to buy a house or invest in a house, so the demand for predicting the bidding rate is also the highest.

2. Methodology

2.1 Literature review

The hedonic price model is a model invented by Rosen to calculate the intrinsic value of goods by calculating how many characteristics affect the price when Rosen determines the price of goods with various characteristics [5]. Since real estate prices are determined by numerous heterogeneous factors of real estate, it is good to apply the hedonic price model. In previous studies that tried to predict real estate auction prices or real estate prices, the hedonic price model was mainly used.

Lim [6] classified the uses in real estate auctions into 5 types and 21 types in detail, and examined the regional and time-series characteristics of the bidding price. He analyzed the time-series characteristics of bidding rates through the stationary test and the Granger-causality test. It was found that changes in the bidding price of apartments in Seoul have a significant effect on the changes in the bidding price in the metropolitan area between 2 and 6 months.

Rhee [7] predicted the bidding rate for an apartment item during a real estate auction. He showed that machine learning methodologies performed better than linear regression, which has been mainly used for predicting bidding rates. Kim [8] analyzed the housing auction market through a time series forecasting method. This study also showed that RNN, a machine learning method, yields better results than GARCH, a traditional statistical method.

Eom [9] showed that model stacking showed better predictive performance than a single machine learning model in the corporate default risk prediction model.

2.2 Ensemble learning

Ensemble learning is a technique for deriving more accurate predictions by creating several classifiers and combining the predictions. Instead of using one strong model, it is a way to help make more accurate predictions by combining several weaker models. There are two main types of ensemble learning: bagging, boosting, and stacking. In previous studies, Bagging and Boosting-based models were used, but in this study,

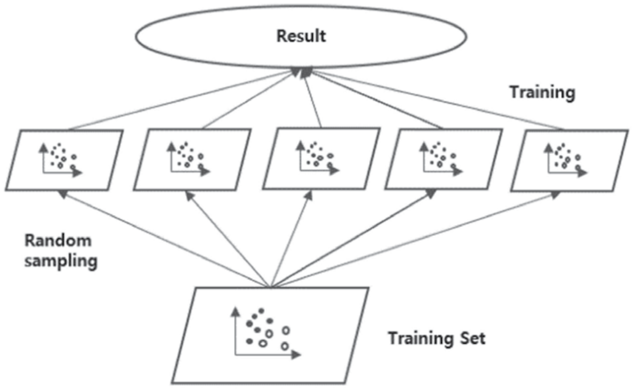

Fig. 1. Bagging.

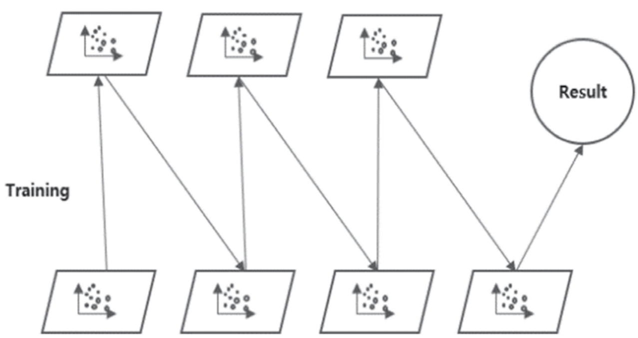

Fig. 2. Boosting.

we try to show that Stacking has better performance than Bagging and Boosting-based single models.

2.2.1 Random forest

Among the ensemble models, it is a model using the bagging methodology. The bagging methodology is a method of synthesizing the learning results after learning several decision trees in parallel to prevent overfitting of the decision trees. Random Forest uses soft voting and derives the final result from the average of the results of numerous predictors, so it has the advantage of low model variance [10]. In addition, since the number of hyperparameters is small, it is convenient for researchers to handle the model. Because it is a CART-based decision tree-based model, it can be used for both regression and classification.

2.2.2 XGBoost

It is a Boosting-based model among Ensemble models. Boosting methodology is a model that gradually reduces bias by putting the results of each decision tree into the next decision tree. It is a model that supplements the learning speed

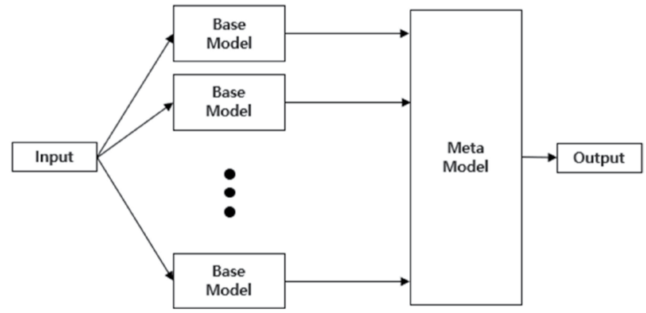

Fig. 3. Model stacking ensemble.

and overfitting problems, which are the disadvantages of the existing boosting-based model [11]. There is a feature that a regularization term exists to prevent overfitting. Like Random Forest, both classification and regression are possible with a CART-based model.

2.2.3 Light GBM

Light GBM is also a Boosting-based model among Ensemble models. It has leaf-wise growth characteristics unlike other existing decision tree-based algorithms. Due to this, the learning speed of the model is very fast, and only a small amount of memory is used when dealing with large-sized data [12]. However, it shows similar performance to other decision tree based algorithms such as XGBoost.

2.2.4 CatBoost

It is a Boosting model among Ensemble models. Like XGBoost, it creates a tree in a level-wise growth method. Unlike the existing boosting model, after calculating the residual with some data, a model is created with it, and then the residual of the data uses the value predicted by the created model. This method is called Ordered Boosting, and at this time, the order is randomly changed to prevent overfitting. It has the advantage of not having to worry too much about parameter tuning because basic parameter optimization is well done. However, there is a problem in processing sparse data, and if most of the data are numeric variables, it is slower than Light GBM and the learning performance is low [13].

2.2.5 Model stacking

Model Stacking is a type of ensemble because it uses several algorithms [14]. It is a model that learns by using the data predicted by the individual model again as a training set. The train set and test set of the original data must exist, and three or more machine learning models learn the original train set and accumulate the results. It is to predict the result by using the generated data as a new train set. Although model stacking has better performance than a single model, it is always a model with a high risk of overfitting. So, to supplement this problem, the overfitting problem was solved by using Cross Validation.

2.3 DNN

It is a deep learning model that increases the number of hidden layers to 3 or more layers in the Artificial Neural Network and puts an activation function between each layer. Activation functions can be used to model non-linear relationships. DNN learns data based on the principle of backpropagation. There are adaptive weights and biases in the neural network, and these weights and biases are adjusted to adapt to the train data.

2.4 Randomized grid search

One of the important factors in machine learning and deep learning learning is hyperparameter tuning. Even with the same model, performance may differ depending on how hyperparameters are tuned. Randomized Grid Search is a search method to find the hyperparameter with the highest performance after sequentially entering the values that can be put into the inputable hyperparameters. In this paper, Randomized Grid Search was used as a hyperparameter tuning method for models. Because the number of models used is large and the number of train data is large, the training time is long, so instead of Grid Search that searches all grids, Randomized Grid Search that searches several grids was used. Randomized Grid Search has a very short learning time compared to Grid Search, but its performance is similar.

3. Results

3.1 Data

This paper used real estate auction data from January 1, 2010 to February 2021, provided by GiGi Auction. Among them, objects classified as residential use were used according to the classification of Gigi Auction, such as apartments, multi-family houses, multi-family houses, and neighborhood houses. Most of the previous studies have very limited areas or uses, but in this paper, the entire country and residential use were used. The bidding rate was used as the dependent variable. As independent variables, variables such as exclusive area and number of floors, which are factors that reflect the characteristics of real estate itself, variables representing regional characteristics, and various variables representing characteristics of auctions were used.

Among the data provided, there were many cases in which the price was not paid even if the item was winningly bid. In this case, in most cases, an unreasonably large price was simply written incorrectly or deliberately used an unreasonably large price to prevent anyone from winning the bid, rather than for rational reasons such as a change in market price after winning the bid. Therefore, after winning the bid, only the data for which the amount was paid was used, as it was judged that using only the data for which the amount was paid would generate less noise.

There was also a variable in the data indicating the status of the auction, whether the auction was closed or bid. However, the data in the case where the auction was closed normally existed, but the data in the case of the bidding was wrongly recorded or there was a lot of missing data. So, only the finalized data was used, but when conducting the actual test, it is not known whether the current auction will be bid or winning, so bias occurs. It is judged that better results can be obtained if the related data are supplemented in future studies.

Real estate auctions are data that do not have a time-series characteristic because once the auction is closed, the auction is not conducted again. However, since the degree of activation of the real estate market has a time-series characteristic, a process to reflect this is necessary. In this paper, we do not use temporal data directly to learn data, but use a sliding window method that divides the entire section for learning and evaluation. Rather than learning and evaluating data for a total of 11 years at a time, we tried to reflect the trend at the time of the auction by proceeding to the end of the first quarter evaluation method of 5-year learning.

3.2 Empirical study

The nationwide residential auction data from 2010 to 2021 was tested for a total of 25 sections with a train period of 5 years and a test period of 1 quarter. Randomized Grid Search was used to find the optimal parameter settings for each basic machine learning model. Cross validation was set to 5 times to maintain robustness of hyperparameter setting. With the hyperparameter thus determined, the entire sliding window section is re-learned and evaluated. The hyperparameters of the base models of the ensemble stacking model to be compared with the basic machine learning model were used as they were previously, and in the case of the meta model, the default parameters of each model were used. As performance evaluation indicators, Mean absolute percentage error (“MAPE”), which is mainly used for financial data evaluation, and Median absolute percentage error (“MdAPE”), which is similar but robust to outliers, were used.

The hyperparameters put in the Randomized Grid Search to find the optimal hyperparameters of Random Forest, XGBoost, Light GBM, CatBoost, and DNN, which are models to be used as benchmarks, are in Tables 1 to 5 below.

Table 1. Random forest hyperparameters

n_estimators	200, 400, 800, 1200, 1600
max_depth	4, 7, 10, 15, 20, 25, -1
criterion	mse, mae

Table 2. XGBoost hyperparameters

n_estimators	200, 400, 800, 1200, 1600
learning_rate	0.001, 0.005, 0.01, 0.05, 0.1, 0.15
max_depth	4, 7, 10, 15, 20, -1
early_stopping_rounds	100, 200
eval_metric	mae, rmse

Table 3. Light GBM hyperparameters

n_estimators	200, 400, 800, 1200, 1600
learning_rate	0.001, 0.005, 0.01, 0.05, 0.1, 0.15
max_depth	4, 7, 10, 15, 20, 25, -1
eval_metric	L1, L2

Table 4. CatBoost hyperparameters

n_estimators	200, 400, 800, 1200, 1600
learning_rate	0.001, 0.005, 0.01, 0.05, 0.1, 0.15
max_depth	4, 7, 10, 15, 20, 25, -1
early_stopping_rounds	100, 200
Eval_metric	MAE, RMSE

Table 5. DNN hyperparameters

lr	0.001, 0.005, 0.01, 0.05, 0.1, 0.15
epoch	200, 400, 800, 1200, 1600
batch_size	256, 512, 1024, 2048, 4096
hidden	[16,16,16], [16,32,16], [32,16,32], [16,64,16], [64,16,32], [32,32,32], [64,64,32]

n_estimators and epoch are the number of iterations of training. max_depth means the max depth of the decision tree in the decision tree-based model, and as it gets larger, the explanatory power of the train set increases, but there is a risk of overfitting. When max_depth is -1, learning is performed without limiting max_dapth. In early_stopping_rounds, if the results of the validation set do not improve for a certain number of training times, training is stopped and the model with the best performance is used before that. criterion is to decide which indicator to use when evaluating with the validation set. learning_rate and lr are hyperparameters that determine the learning rate of the model. If the learning rate is large, the data may deviate randomly and may not converge to the lowest point. Conversely, if the learning rate is too small, it takes a long time to learn, and it may not be possible to find a global minimum by converging to a local minimum. In DNN, batch size is the size of data given to each batch. Batch means a divided data set. All activation functions used in DNN used ReLU, and Adam optimizer was used as the optimizer. The number of layers was set to 3, and the number of nodes in each layer was designated as 2 to the nth power.

The hyperparameters adopted with the best performance for each model are as follows.

In Model Stacking Ensemble, using 3 or more base models

Table 6. Selected hyperparameter

Models	Selected hyperparameter
Random forest	n_estimators: 800 max_depth: 15 criterion: mse
XGBoost	n_estimaors: 10 learning_rate: 0.15 max_depth: 4 early_stopping_rounds: not used eval_metric: mae
Light GBM	n_estimators: 400 learning_rate: 0.1 max_depth: 15
CatBoost	n_estimators: 1200 learning_rate: 0.005 max_depth: 10 early_stopping_rounds: 200 eval_metric: MAE
DNN	lr: 0.01 epoch: 1600 batch_size: 4096 hidden: [32,32,32] threshold: 0.95

Table 7. Selected model stacking structure

Base model	Meta model
Random forest, XGBoost, Light GBM	XGBoost

Table 8. The average value of the results of each window when the training period is 5 years and the test period is 1 quarter

Model	MAPE	MdAPE
Random forest	8.838	6.151
XGBoost	6.569	4.642
Light GBM	6.857	4.720
CatBoost	10.050	6.800
DNN	7.130	5.041
Model stacking ensemble	6.179	4.187

is good for performance, so at least 3 of Random Forest, XGBoost, Light GBM, CatBoost, and DNN were used. For base model selection, a total of 16 combinations were used, including 10 selected 3 out of 5 models, 5 selected 4, and 1 selected all, and 3 types of Random Forest, XGBoost, and Light GBM were used as meta models. A total of 48 combinations were used. The hyperparameter of each base model was used as it was previously set, and the default value was used for the meta model. As a result, the model using Random Forest, XGBoost, and Light GBM as the base model and XGBoost as the meta model showed the best performance.

The final result of this model is shown in Table 8.

When looking at the average value of the results of each window when the train period was 5 years and the test period was 1quarter, XGBoost showed the best performance among the base models, and the Model Stacking Ensemble showed the best performance among all models.

4. Discussion

In this paper, the bidding rate was studied using the data of Korean residential auctions from January 2010 to February 2021 provided by Gigi Auction. Among the base models, XGBoost showed the best performance, and CatBoost showed the lowest performance because there were not many nominal variables. In the case of DNN, it is worse than XGBoost and Light GBM among machine learning models, but it shows better performance than Random Forest and CatBoost. Model Stacking Ensemble shows better performance than base models. It can be said that the prediction is definitely better, given that both MAPE, which are relatively sensitive to outliers, and MdAPE, which are robust to outliers, perform well. Of course, there is only a 0.4 difference from XGBoost in percentage terms, but considering that the bidding price of each item is hundreds of millions of won, it can be said to be a significant difference in predictive performance.

In this paper, hyperparameter tuning for meta models was not performed when model stacking, but machine learning models have a large effect on the results of hyperparameters, so improving this point has room to produce better results. In addition, there is room for performance improvement by using other machine learning models as the base model or by using a different stacking structure.

Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No. 2020-0-02162, Development of a deep learning based real estate auction fair value forecasting system for non-face-to-face auction information provision service).

References

1. Lee HK, Bang SH, Lee YM. True auction price ratio for condominium: the case of gangnam area, Seoul, Korea. Housing Stud 2009;17:233-258.
2. Lee KT, Shin IJ, Kim HK, Shin JC. A study on the influencing factors on sale price of partitioned ownership retail store in seoul real estate auction market. KREAA 2017;23:87-104.
3. Yang JH, Kim YM, Oh KJ. Forecasting the KOSPI 200 stock index based on lstm autoencoder. QBS 2020;39:101-109.
4. Lee SG, Lee HS, Oh KJ. KOSPI200 index prediction using sequence-to-sequence based on denoising filter and attention mechanism. QBS 2020;39:127-135.
5. Rosen S. Hedonic prices and implicit markets: product differentiation in pure competition. J Polit Econ 1974;82:34-55.
6. Lim BK. A study on regional fluctuations in housing auction prices in real estate auctions. Domestic Master's Thesis Kyung Hee University Graduate School of Public Administration, 2006. Seoul.
7. Rhee JT, Ahn WB, Oh KJ. Using machine learning algorithms to forecast the optimal bidding rate in apartment auctions. QBS

2021;40:31-37.

8. Kim SA, Chun HJ. Prediction on the housing auction market using deep learning. KREMA 2020;21:7-26.
9. Eom HN, Kim JS, Choi SO. Machine learning-based corporate default risk prediction model verification and policy recommendation: Focusing on improvement through stacking ensemble model. JIIS 2020;26:105-129.
10. Breiman L. Random forests. Machine learn 2001;45:5-32.
11. Chen T, Guestrin C. Xgboost: a scalable tree boosting system. Lect Notes Artif Int: 22; 785-794. San Francisco; Aug. 2016.
12. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, et al. Lightgbm: a highly efficient gradient boosting decision tree. NIPS 2017;30:3146-3154.
13. Prokhorenkova L, Gusev G, Vorobev A, Veronika A, Gulin A. CatBoost: unbiased boosting with categorical features. arXiv preprint arXiv: 1706.09516, 2019.
14. Wolpert DH. Stacked generalization. Neural Networks 1992;5:241-259.