Comparative Evaluation of Machine Learning and Deep Learning Models for Real Estate Price Prediction

Asad Ullah Khan[1], Muhammad Khateeb Khan[1*], Usama Khan[1], Faheem Shaukat[1]
[1]Department of Computing & Technology, Iqra University Islamabad Campus (IUIC).
*** Correspondence:** m.khateeb231@gmail.com

Citation | Khan. A. U, Khan. M. K, Khan. U, Shaukat. F, "Comparative Evaluation of Machine Learning and Deep Learning Models for Real Estate Price Prediction", IJIST, Vol. 07 Issue. 01 pp 83-97, January 2025
Received | Dec 21, 2024, **Revised** | Jan 13, 2025, **Accepted** | June 15, 2025, **Published** | Jan 16, 2025.

Abstract.
Accurate real estate price prediction is vital in informed decision-making for investors, policymakers, and stakeholders. This study evaluates various machine learning and deep learning models for predicting real estate prices using the House Prices 2023 dataset which contains 168,000 entries of Pakistani property data. In our proposed methodology we performed data preprocessing and features engineering to standardize the data. We performed extensive experiments by using Machine Learning (ML) and Deep Learning (DL) models on our preprocessed data. The model's performance was evaluated based on the R-squared (R^2) score and Mean Squared Error (MSE) metrics. Based on the provided metrics, the Decision Tree achieved the highest performance with an R^2 of 0.9968 and an MSE of 0.0021, followed by Random Forest with an R^2 of 0.990 and MSE of 0.0007. Similarly, other ML models like Gradient Boosting and XG Boost also outperformed by achieving (R^2 0.9959, MSE 0.0028 R^2 0.9747, and MSE 0.0170) respectively. In contrast, models like AdaBoost, Neural Network, and Convolutional Neural Network (CNN) showed comparatively lower performance due to the nature of the data. The study emphasizes that ensemble-based models like Decision Trees and Random Forests are highly effective at identifying patterns in real estate prices. Additionally, applying optimization techniques improves the model's ability to generalize and perform well on unseen data.

Abbreviations.
Machine Learning (ML)
Deep Learning (DL)
Mean Squared Error (MSE)
Convolutional Neural Network (CNN)
Support Vector Machines (SVM)
K-nearest neighbors (KNN)
Root Mean Squared Error (RMSE)
Interquartile Range (IQR)
Recursive Feature Elimination (RFE)
Multi-layer Perceptron (MLP)

Keywords: Real Estate, Price Prediction, Machine Learning, Deep Learning, Ensemble Methods

Introduction:

Real estate is a very useful business segment that has a huge impact on economic development and globalization. The features given above are crucial because determining the price of a property in question is of great importance to many involved parties such as investors, buyers, sellers, financial institutions, insurance companies, and policymakers [1]. Achieving timely, accurate forecasts is the key to ensuring that different stakeholders avoid the pitfalls that come with wrong investment decisions [2]. There are certain shifts in the market type in question real estate, especially after the COVID-19 period. For example, in 2019, real estate contributed approximately 7.62% to global GDP. Additionally, transactions increased by 6% in the Asia-Pacific region, and an 8% growth was recorded in Europe [3]. Forecasting these trends accurately is vital for developing effective investment strategies with minimized risks. Moreover, governments can leverage precise forecasts for urban planning, while financial institutions can set suitable mortgage rates with reduced risks [4][5].

As we know, in recent years, ML and deep learning methods have transformed real estate price prediction, enabling significant advancements in accuracy and reliability across diverse datasets [6]. This study presents advanced models with modern techniques to improve the accuracy of the models, including boosting and optimization, ensemble methods, feature engineering, and feature selection. We also aim to apply Neural Network architectures to predict real estate prices with higher precision [7]. Using the "House Prices 2023" dataset from Kaggle, which provides detailed property data for Pakistan, we implemented key preprocessing steps such as filling missing values, removing outliers, scaling, encoding, and feature engineering to prepare the data for high-performance modeling.

While several studies have aimed to enhance house price prediction accuracy, few have explored comprehensive ensemble methods and advanced transformation techniques. [8] applied ML and DL models using the Clean_data_for_model dataset, which is already preprocessed by the user from Kaggle, identifying KNN as the most effective model with a promising R^2 score of 0.85 and MSE of 1.9e+4. Another research [9] explores a range of ML and DL models, including Decision Tree, Random Forest, AdaBoost, Gradient Boosting, KNN, XG Boost, Neural Network (NN), and CNN. We used different techniques like Boosting, Optimization, Ensemble methods, feature engineering, and feature selection to enhance model performance.

The continuous evolution of real estate markets underscores the need for accurate and robust price-prediction models. Consequently, this paper aims to do its part in filling the current gap within the utilization and deployment of machine learning and deep learning, enhancing decision-making, and thus strengthening the forecasting for multiple markets and datasets. From utilitarian action, the outcomes demonstrate that, in addition to the straightforward ensemble methods, beginning with the Decision Tree, Gradient Boosting, as well as XG Boost, can be even more significant in boosting efficiency for the provision of prediction. However, Adam optimization and some of the feature engineering are two procedures, that help fine-tune such models.

Research Objectives:

More specifically, the main objective of this research is to expand existing models of price forecasts for real estate about various markets and combined data. It is therefore aimed at the improvement of the prediction accuracy through the implementation of ML and DL coupled with great consideration on the data preprocessing as well as very appropriate feature selection strategies. The research objectives are as follows.

- Design and implement advanced ML and DL models capable of accurately predicting real estate prices across diverse markets and datasets.

- Identify and prioritize the most impactful features influencing real estate prices. Optimize feature selection and engineering to enhance model performance and accuracy.
- Assess the performance of different Machine Learning and Deep Learning algorithms, which are boosting methods, ensemble models, and neural networks, in forecasting property values using the "House Prices 2023" dataset.

Literature Review:

Real estate price prediction has been a subject of research focus mainly because its predictability affects decision-making and business stability. Several conventional and ML approaches have been developed for estimating property values in the past several years, and many studies have addressed location, economic variables, and market trends for this purpose. Real estate price prediction was explored using the 'House Prices 2023 Dataset' available on Kaggle. Some studies also indicated that Support Vector Machines (SVM) and Lasso Regression are less suited for house price prediction due to their inability to effectively manage non-linear relationships and multicollinearity. Studies have emphasized that ensemble methods, particularly Gradient Boosting, provide superior performance in capturing these complexities [10].

To improve the forecast accuracy the study used machine learning techniques as well as deep learning such palm; Linear Regression, Gradient Boosting, Random Forest, CNN, and KNN. The authors discussed a considerable amount of data preprocessing and feature engineering and the assessment of the models based on MSE, RMSE, Coefficient of determination R-squared and Accuracy. In the present study, KNN turned out to be the most accurate model yielding an R-squared value of 0.85 (86% accuracy) and an RMSE of 13.79 more accurately predicting the ratio of order price to the total price. This study provides a solid foundation for improving the accuracy of real estate price prediction models and emphasizes the efficacy of machine learning approaches [11]. Advancements in data science and machine learning have introduced a sophisticated tool that can analyze the various variables that will have an impact on property prices [8].

However, factors such as property type, location, and market trends add complexity, making price prediction challenging. Traditional models often struggle to capture these complexities, particularly non-linear relationships, thus limiting their effectiveness in intricate real estate price forecasting [12]. Recent studies, however, have employed advanced techniques, such as Convolutional Neural Networks and ensemble learning, which show improved accuracy in real estate price predictions [12].[13] compared various methods, including polynomial regression, multivariate regression, and linear regression models. By using features such as square feet (SqFt), bedrooms, and bathrooms, they demonstrated that accounting for multiple influential factors improved prediction accuracy, especially when considering various aspects of property value. Machine learning models, especially ensemble methods, have emerged as efficient alternatives due to their ability to recognize complex correlations and patterns [14].

Ensemble methods combine the strengths of multiple models, thereby improving prediction accuracy and robustness [15]. In feature engineering, we created new features to ensure consistency and improve model accuracy. For example, we derived a 'sqft' feature by converting units of area from Marla and Kanal to square feet, standardizing the measurement across all records. Additionally, we encoded location data using target encoding to capture regional pricing trends, which provided a richer context for model training. Feature scaling was also applied to numerical attributes like area size, number of rooms, and price, ensuring a standardized input range for optimal model performance.

There is another research incorporating advanced feature engineering techniques, such as calculating 'Area Size' and 'price per square foot' and trimming extreme values at the 1st and 99th percentiles or using the IQR [16]. Traditional machine learning models, including Random Forest, Gradient Boosting, KNN, Decision Tree, MLP, AdaBoost, XG Boost, and deep learning

models like NN, were evaluated using metrics such as MSE, RMSE, R-squared, and Mean Absolute Error (MAE) [17]. Furthermore, the study also introduced the use of a Stacking Regressor, an ensemble model, to enhance the prediction accuracy of the model. While prior studies focused on a limited set of models, this research illustrates the effectiveness of integrating several models and techniques, including Support Vector Regression and advanced ensemble methods, for enhanced prediction performance [18].

Table 1. Comparison Table

Ref.	Algorithms Used	Performance Metric	Best Algorithm Declared
[11]	XG Boost, Light GBM, RF	RMSE	Light GBM
[19]	LSTM, GRU, CNN	R^2	LSTM
[20]	Decision Trees, SVM, KNN, XG Boost	MAE	XG Boost
[21]	Random Forest, Neural Networks, AdaBoost	RMSE	Neural Networks
[22]	LSTM, RNN, CNN	ME	LSTM
[23]	Decision Trees, Ridge Regression, Neural Networks	MSE, R^2	Neural Networks

This work is based on earlier research by analyzing a wide variety of models and implementing hyperparameters tuning tools like grid and random search with early stopping. What is important is to underline the accuracy of the last concepts of modeling, such as neural networks and ensemble methods, in real estate price prediction. Further, subsequent studies will be more dedicated to enhancing feature extraction, exploration of transfer learning, and the implementation of stronger models to enhance the predictive capability of the model.

Research Methodology:

In this particular investigation, therefore, an attempt will be made to examine how the phases may be sequenced more effectively towards the prediction of house prices. Data acquisition follows where data on housing is obtained from various data sources. Data cleaning then occurs as a part of the Data preprocessing activity which involves cleaning of data. So, feature engineering is applied to design meaningful variables that represent important characteristics of the housing markets. Feature selection enables the choice of the most important features of the model and as such enhances model performance. Next is model selection where the best model that will predict the results best is selected. Last but not least, price prediction is conducted to predict future house prices with the help of such findings described in Figure 1.

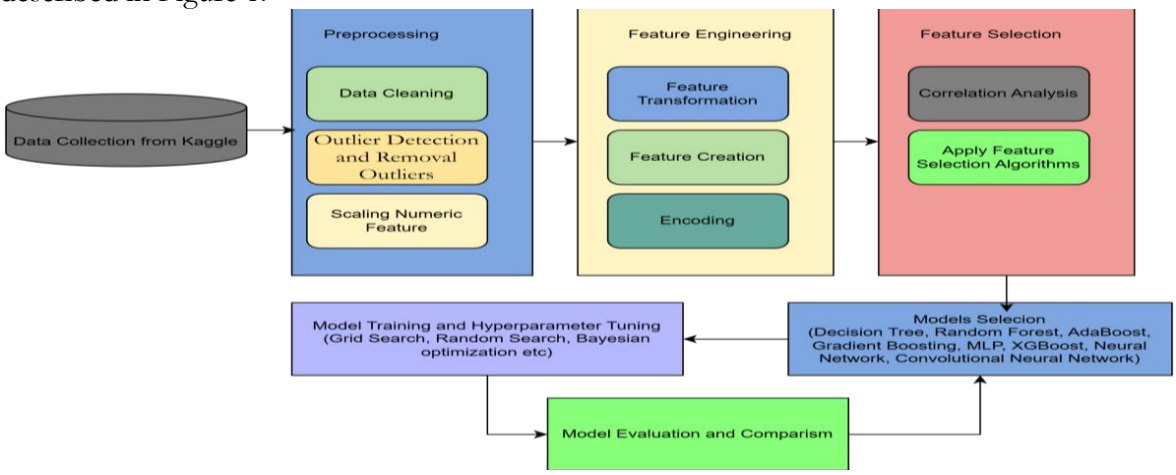

Figure 1. Research Methodology

Data Collection:

The current dataset to be used for this was obtained from Kaggle and is called the House Prices 2023 Dataset. It includes 168000 rows reflecting different characteristics of real estate properties in several cities of Pakistan. Key variables observable from the dataset are property type, price, number of bedrooms, bathrooms, location, and area, which make the dataset suitable and exhaustive for the analysis of the real estate market as depicted below in Table (2). These extensive attributes provide strong feasibility for model building and assessment of predictive models.

Table 2. Dataset Attributes

S.no	Variables	Data Types
1	propertied	Integer
2	location	Integer
3	Page URL	String
4	property type	String
5	price	Integer
6	location	String
7	City	String
8	province name	String
9	latitude	Real
10	longitude	Real
11	baths	Integer
12	purpose	String
13	bedrooms	Integer
14	date added	String
15	agency	String
16	agent	String
17	Area Type	String
18	Area Size	String
19	Area Category	String
20	area	String

Data Preprocessing:

Data preprocessing is crucial for ensuring the quality and consistency of the dataset before feeding it into machine learning models. The following steps were implemented. Removal Features like propertied, location, page URL, longitude, and latitude were removed as they did not contribute to price prediction and could introduce noise. Missing values, particularly in the 'Area Size' and bathrooms attributes, were handled by imputing them with the median or mean value to maintain data integrity without removing entire rows [24]. Therefore, in this study, we conducted experiments to evaluate different techniques for combining the models, to improve. Initially, we addressed data preprocessing issues, which include a multiple approach how to detect and eliminate outliers. Among these methods, the IQR method proved to be the most suitable for this dataset since it followed the most accurate precision in filtering out extreme outliers.

Outlier Detection and Removal Outliers:

Especially in the price attribute, were filtered out using The Interquartile Range method (IQR).

Filtering:

Only properties listed as 'For Sale' were retained to maintain consistency in the target prediction task. Additional filtering was applied based on property types relevant to the analysis.

Scaling: We also standardized the numerical features of the datasets by using the 'Standard Scaler' to ensure all the features have the same scales. The target variable or label variable 'price' transforms using the np.log1p to improve the model performance.

Feature Engineering:

During the feature engineering stage, we were formulating useful and relevant features to improve the models' interpretability and performance. For the area size, we converted the measurement to an area in square feet. We also took the logarithm of the price variable– our metric of interest– to address skewed distribution. Additionally, we added new attributes were also designed including price per square footage and bedroom-to-bathroom ratio that gave a forecast regarding property pricing as shown in Figure (2).

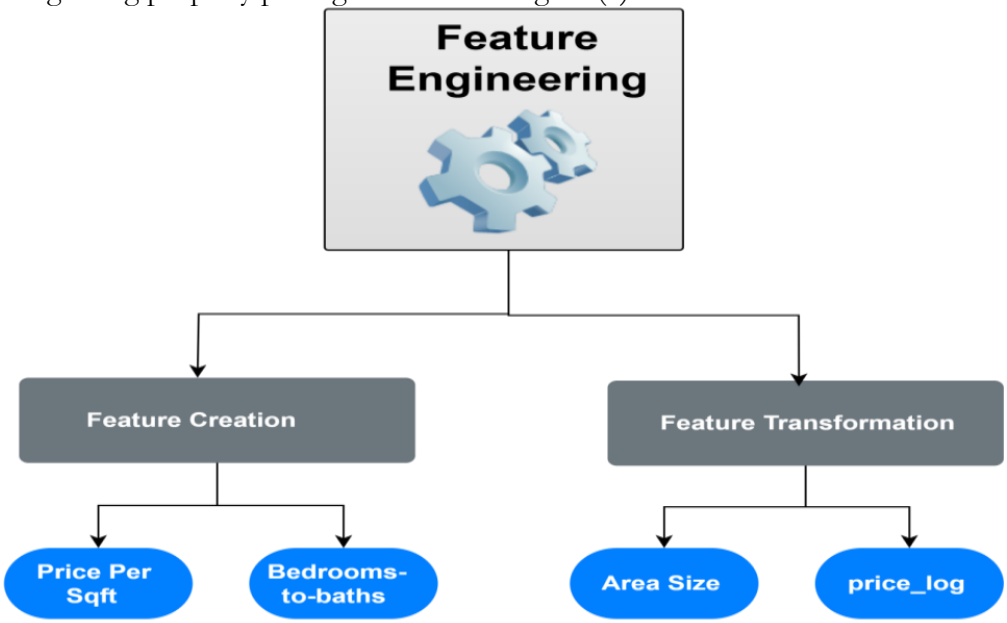

Figure 2. Features Engineering

Feature Transformation:

In the feature transformation, the conversion of the unit's land area Marla's and Kanal's into a standard unit. Area Size is transformed to square feet. Which is the common feature transformation technique in real estate. We also transformed the price features by applying the natural logarithm.

Feature Creation:

A new feature, 'Price Per Sqft. The Area Size is computed by multiplying the Kanal's with 5445.0 and Marla's with 272.25. The Price Per Sqft is computed by dividing the property price by its area in square feet, providing a normalized perspective on pricing. Also, create a new feature which is the Bedroom-to-Bath Ratio.

Encoding Categorical Variables:

We examined several encoding methods to use when dealing with categorical data. One hot encoding could have been used, but when this method is used, it increases the dimensionality of the dataset, which will also reduce the computational efficiency. Models have to process data with many new dimensions which greatly slows down computation. Nevertheless, the label encoding turned out to be the best way as it provides a less complex and shorter format for the characteristics of the dataset.

Feature Selection:

Correlation Analysis:

In the correlation, we created the heat map of all features to observe the correlation between the variables which features are strongly correlated, and which will have the most effect

on the target variable.

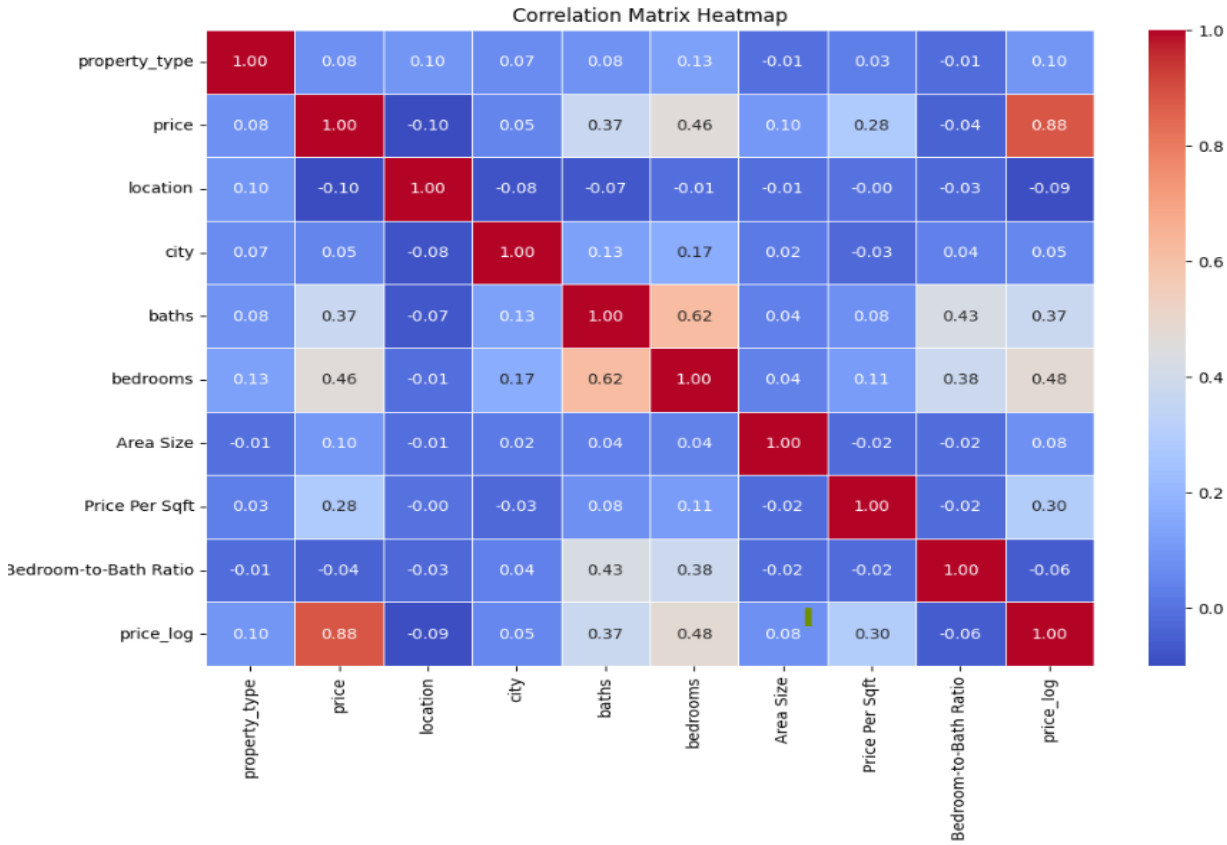

Figure 3. Correlation Heat map Matrix

As shown in Figure (3) the target variable, 'Price log', demonstrates a strong positive correlation with several features, including price (0.88)**,** bedrooms (0.48)**,** baths (0.37)**,** and price per sqft (0.30). The correlation with price is especially strong, indicating a significant relationship. On the other hand, location (-0.09) and Bedroom-to-Bath Ratio (-0.06) show weak negative correlations with Price log, suggesting a minor inverse relationship. Additionally, city and Area Size exhibit very low or near-zero correlations with Price log**,** implying that these factors have little to no effect on the target variable in this dataset.

Feature Selection Algorithms:

Select Best from the sklearn. feature selection package is applied with the f regression scoring model to consecutively select the best k features to create a correlation with the target variable. And we also tried RFE and Select Best methods were used to identify the most impactful features, reducing dimensionality and improving model efficiency by retaining only the most relevant features.

Model Selections:

We selected a total of eight models for training on this dataset which are the Decision Tree, Random Forest, AdaBoost, Gradient Boosting, MLP, XG Boost, Neural Network, and Convolution Neural Network.

Model Training and Hyperparameter:

The study developed and evaluated various ML and DL models to predict real estate prices, aiming to identify the best-performing approaches. Hyperparameter tuning played a pivotal role in optimizing the models' performance by systematically searching as shown in Table (3) for the most effective parameter configurations. To ensure an unbiased and thorough exploration of hyperparameter space, methods such as Randomized Search CV and Grid Search

CV were employed. For neural networks, tuning involved parameters like learning rate, dropout rate, activation functions, number of layers and nodes, batch size, and epochs. Similarly, for tree-based models such as Random Forest, key parameters like the number of estimators, maximum tree depth, and minimum samples required for splitting or constructing nodes were optimized. These adjustments led to significant improvements in key performance metrics, including MSE and R2R^2R2, showcasing the critical role of hyperparameter tuning in enhancing model effectiveness.

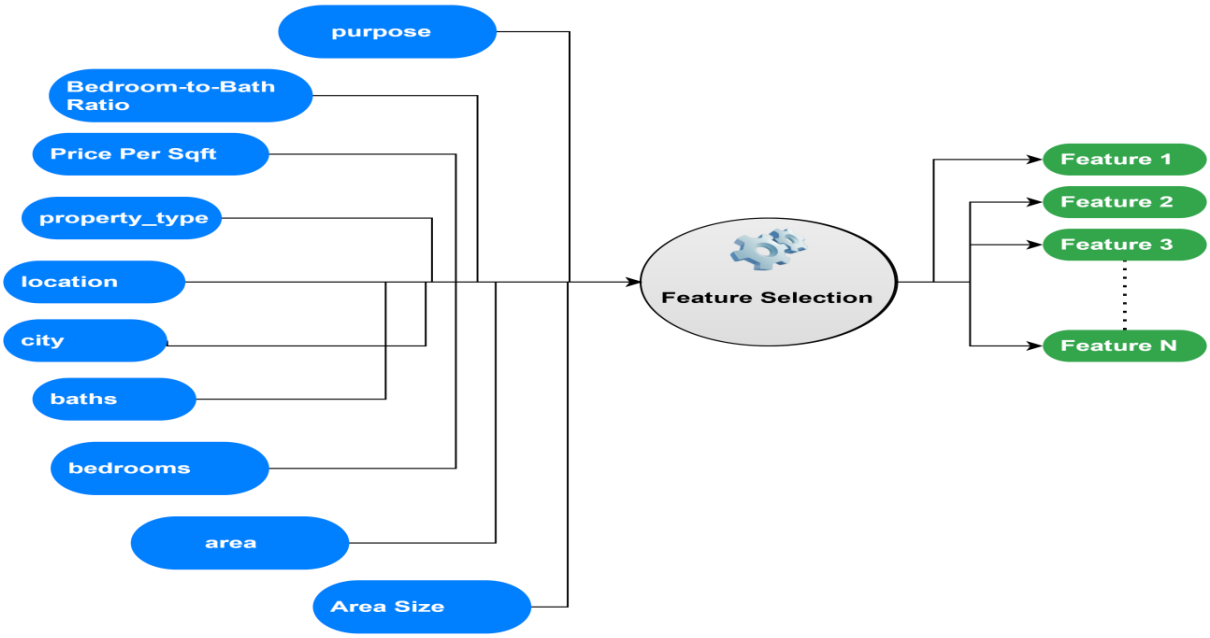

Figure 3. Diagram of Feature Selection of Algorithms

Table 3. Hyperparameter Search Space for each Algorithm

Model Name	With Hyperparameters tuning	Values Explored
Decision Tree	max_depth	[3, 5, 7]
Random Forest	estimators, max_depth	[100, 200, 500], [3, 5, None]
AdaBoost	n_estimators, learning_rate	[50, 100, 200], [0.01, 0.05, 0.1]
Gradient Boosting	n_estimators, learning_rate, max_depth	[100, 300, 500], [0.01, 0.05, 0.1], [3, 5, 7]
MLP	hidden_layer_sizes, alpha, learning_rate_init	[(50,), (100,)], [0.0001, 0.001], [0.001, 0.01]
XG Boost	n_estimators, learning_rate, max_depth	[100, 300, 500], [0.01, 0.05, 0.1], [3, 5, 7]
Neural Network	learning_rate, batch size, epochs	0.0005, 64, 50
CNN	learning_rate, batch size, epochs	0.0005, 64, 50

Model Evaluation and Comparison:

After training all the models of machine learning and deep learning, we all evaluate the models using the evaluation metrics of R-squared (R^2) and MSE. After finding the evaluation metrics, we will compare which model is performing well.

Results:

The performance of several machine learning models was evaluated for house price prediction based on two key metrics: R-squared (R^2) and MSE. These metrics were used to assess the accuracy and effectiveness of each model in predicting house prices. The models tested include Decision Tree, Random Forest, AdaBoost, Gradient Boosting, MLP, XG Boost, Neural Network, and CNN. The results, as shown in Table (5), indicate varying levels of performance

across the models, with some models demonstrating high accuracy and low error rates, while others performed less effectively.

Table 4. Model Performance Metrics without Hyperparameters

No.	Model Name	R-Squared Score (R^2)	MSE
1	Decision Tree	0.9958	1.82e+12
2	Random Forest	0.99	9.15e+11
3	AdaBoost	0.6171	5.38e+14
4	Gradient Boosting	0.9921	1.11e+12
5	MLP	-7.13e+35	1.00e+isth68
6	XG Boost	0.9743	8.04e+11

We performed extensive experiments in our dataset with and without hyperparameter tuning. As results displayed in Table (4) were obtained without parameter tuning. When hyperparameter tuning was not applied, Decision Tree and Random Forest models exhibited high R^2 values but also high MSE, indicating overfitting and suboptimal generalization. Poorly performing models such as AdaBoost and MLP demonstrated substantial improvement post-tuning, with the tuned versions achieving an R^2 of 0.862 and a significantly reduced MSE. Additionally, techniques like Gradient Boosting and XG Boost benefited from tuning, achieving enhanced MSE reduction and more stable R^2 values. Other neural network architectures, including CNNs, also showed marked performance improvements with precise parameter adjustments. Overall, hyperparameter tuning proves critical in optimizing weaker models, and refining strong models and is an indispensable component of the machine learning pipeline.

As reported in Table (5) herein above, Random Forest, Gradient Boosting, and Decision Tree-based ensemble created a breath-taking impression achieving a nearly perfect model fit as indicated by the nearly converging R^2 scores. According to the table, Random Forest had the least MSE, equal to 0.0007, meaning the tested models it was the most accurate one. The using of AdaBoost was less satisfactory relative to other ensemble techniques and demonstrated reasonable performance, but lower R^2, and higher MSE. The deep learning models (MLP, CNN, and basic Neural Networks) in general performed lesser when compared with the tree-based models. The results obtained from XG Boost were good as it obtained high R^2 and relatively low MSE.

Table 5 Model Performance Metrics with Hyperparameters

No.	Model Name	R-Squared Score (R^2)	MSE
1	Decision Tree	0.9968	0.0021
2	Random Forest	0.990	0.0007
3	AdaBoost	0.862	0.0924
4	Gradient Boosting	0.9959	0.0028
5	MLP	0.6590	0.2289
6	XG Boost	0.9747	0.0170
7	Neural Network	0.9433	0.0381
8	CNN	0.6418	0.2404

The results presented as shown in Table (5) are in line with the effectiveness of ensemble methods, particularly Random Forests and Gradient Boosting for real estate price prediction. Perhaps due to their capacity to generate an ensemble of decision trees and map non-linear data patterns, both methods afford high accuracy. The suboptimal performance of AdaBoost could be due to the rich-relevance factor of adversarial outliers to the weighting decision of weak learners. The below-par performance of MLP and CNN may have been due to issues such as restricted data quantity to model capacity, the tabular data structure that is suitable for tree-based algorithms, or larger hyperparameter search space for these deep learning architectures. The neural network model was found to be better than MLP and CNN but not better than tree-

based models. The performance of the models based on R^2 and MSE highlights the strengths of tree-based classifiers. The Decision Tree achieved the highest R^2 value of 0.9968, indicating excellent explanatory power, along with a minimal MSE of 0.0021, reflecting very low prediction errors.

Similarly, Gradient Boosting demonstrated strong predictive capabilities with an R^2 of 0.9959 and a slightly higher MSE of 0.0028, making it a close competitor to the Decision Tree. The Random Forest also performed exceptionally well, with an R^2 of 0.990 and the lowest MSE of 0.0007, showcasing its ability to generalize effectively across the data. In comparison, AdaBoost achieved a decent R^2 of 0.862 but struggled with a higher MSE of 0.0924, suggesting greater variability and less accuracy in its predictions compared to other tree-based models. Among the neural network-based models, the Neural Network performed moderately well with an R^2 of 0.9433 and an MSE of 0.0381, indicating reasonable predictive accuracy with some room for improvement. Nevertheless, MLP and CNN had a poor fit to the data as confirmed by the low R^2 values of (0.6590) and (0.6418) and high MSE values of (0.2289) and (0.2404) respectively. When assessing the performance of the ML algorithms, it was found that tree-based models including Decision Tree, Gradient Boosting, and Random Forest are the least erroneous and offer high explain-ability scores for this task.

The R^2 squared score as depicted in the Figure (4) shows the following performance of the classifiers in mapping the variance in the data set. The models in comparison produced the best R^2 of 0.9968 for the Decision Tree, then Gradient Boosting of (0.9959), and finally the Random Forest model of 0.990, showing that all the models are excellent models of the variability of the data. However, AdaBoost tended to give a lower R^2 of 0.862 hence signifying less capability of making future predictions than the tree-based models. Of those based on stacked neural networks, the Neural Network has a fair degree of fit, having an R^2 of 0.9433, while the MLP model gave a very low fit (0.6590) and the CNN (0.6418), suggesting they give a poor fit to the data. These findings support tree-based models being more appropriate for this work although neural networks need enhancement to explain this work better.

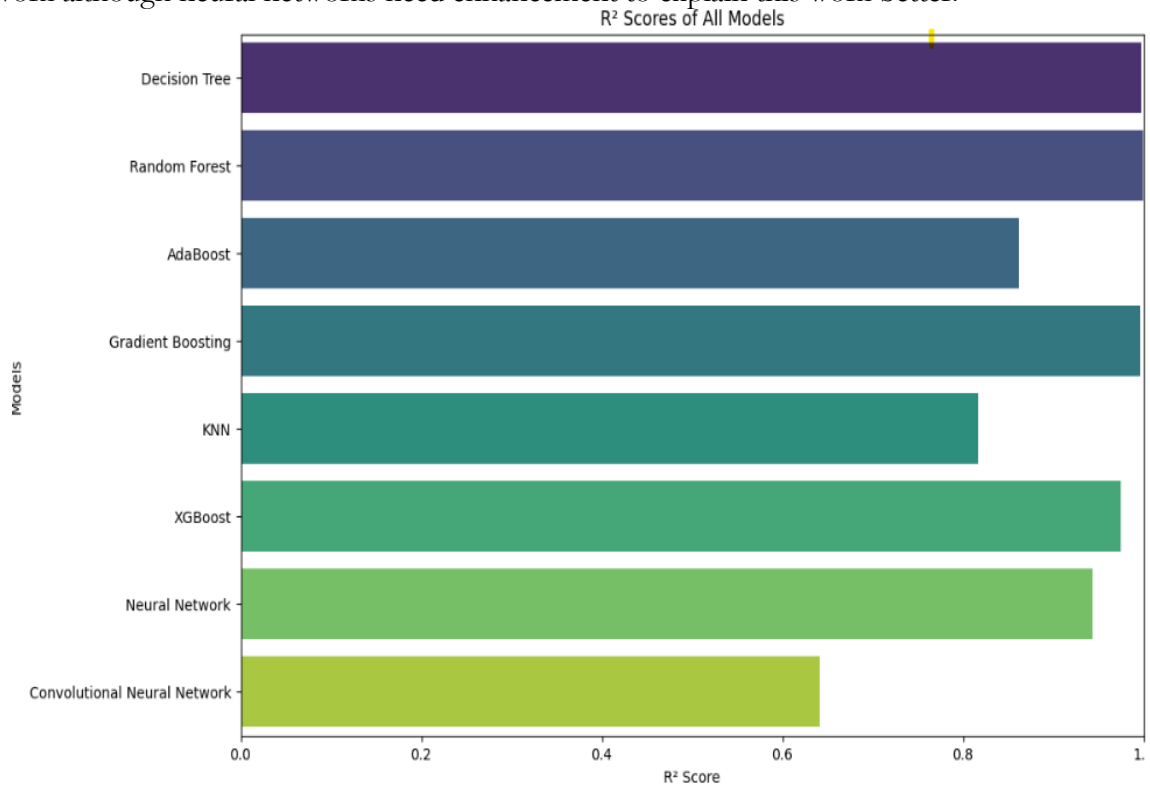

Figure 4. R Squared Scores of Models

The MSE (Mean Squared Error) values, as shown in Figure (5), highlight the variation in prediction error across the classifiers. The Random Forest achieved the lowest MSE of 0.0007, reflecting its remarkable accuracy and minimal error in predictions. Similarly, the Decision Tree and Gradient Boosting demonstrated very low MSE values of 0.0021 and 0.0028, respectively, indicating their strong predictive performance. In contrast, AdaBoost showed a significantly higher MSE of 0.0924, suggesting greater variability and error in its predictions compared to the top-performing tree-based models.

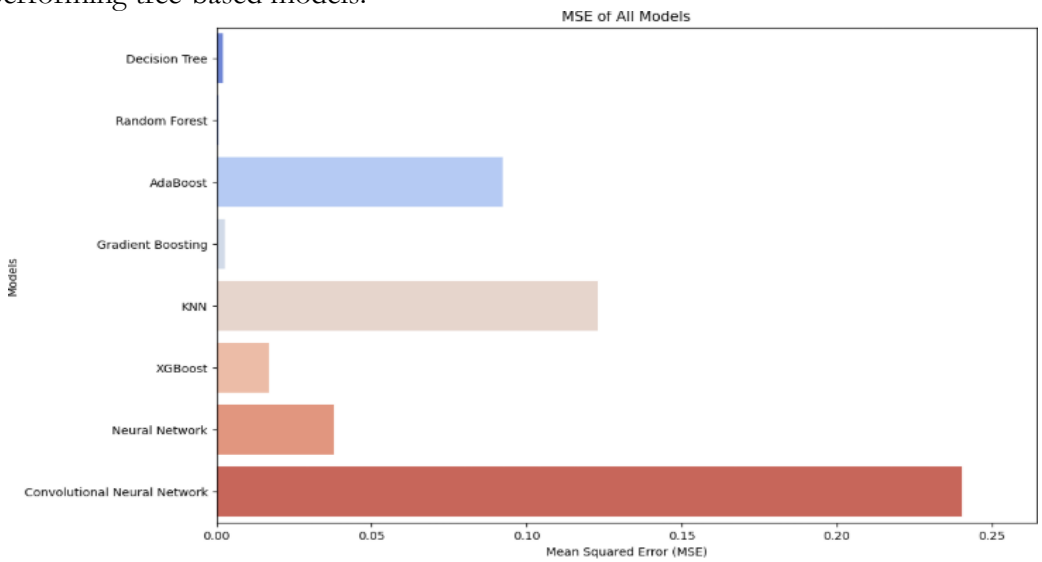

Figure 5. Mean Squared Error of all models

Among neural network-based classifiers, the Neural Network exhibited a moderate MSE of 0.0381, while the MLP (0.2289) and CNN (0.2404) displayed much higher MSE values, reflecting their inability to produce accurate predictions. These results emphasize the minimal errors of tree-based models, while neural network-based models require further refinement to reduce their prediction errors.

The learning curve for the Decision Tree model is shown in Figure (6), highlighting the relationship between the training set size and the MSE for both the training and cross-validation datasets. The graph shows that the training error is almost zero across all training set sizes, indicating a good fit on the training data. However, the cross-validation error is initially high when the training set is small and gradually decreases as the training samples increase, reaching a certain point This behavior suggests overfitting on the training set, particularly evident in the dramatic divergence between training and cross-validation errors when the training set is relatively small. As the training set grows, the cross-validation error decreases, indicating improved generalization due to the increased diversity in the data. Despite this, the Decision Tree model does not appear to suffer from severe overfitting. As a result, the gap between training and cross-validation errors reduces, and the cross-validation error stabilizes with a larger dataset. Therefore, additional regularization or ensemble methods may not be necessary unless further testing reveals instability or poor performance on new datasets.

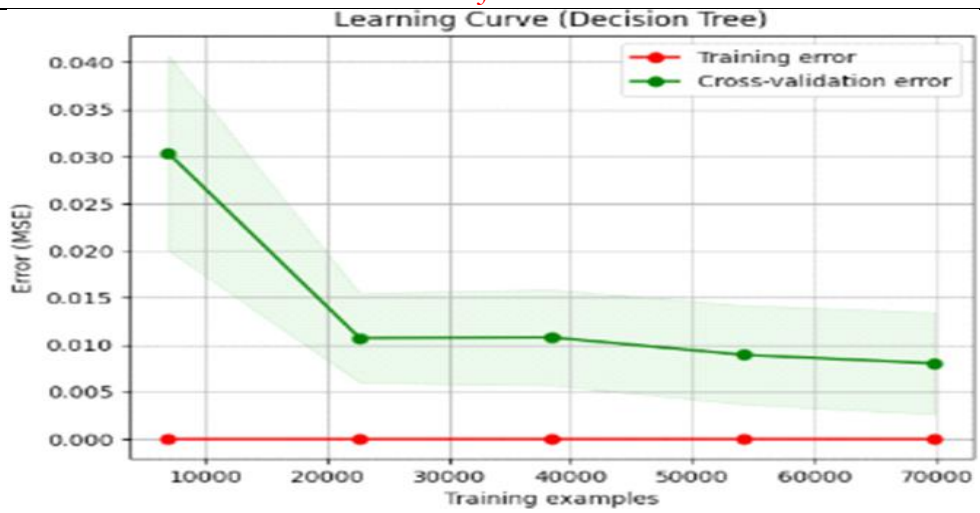

Figure 6. Decision Tree Learning Curve

The learning curve for the Random Forest model as shown in Figure (7), displays the progression of the MSE as the training set size increases and compares it to other models. The red curve for the training error consistently remains below the other diagrams, indicating that the model fits the training data well. The green curve, representing the cross-validation error, starts higher with fewer training examples and decreases as more data is added, though at a slower rate, before stabilizing at a low value.

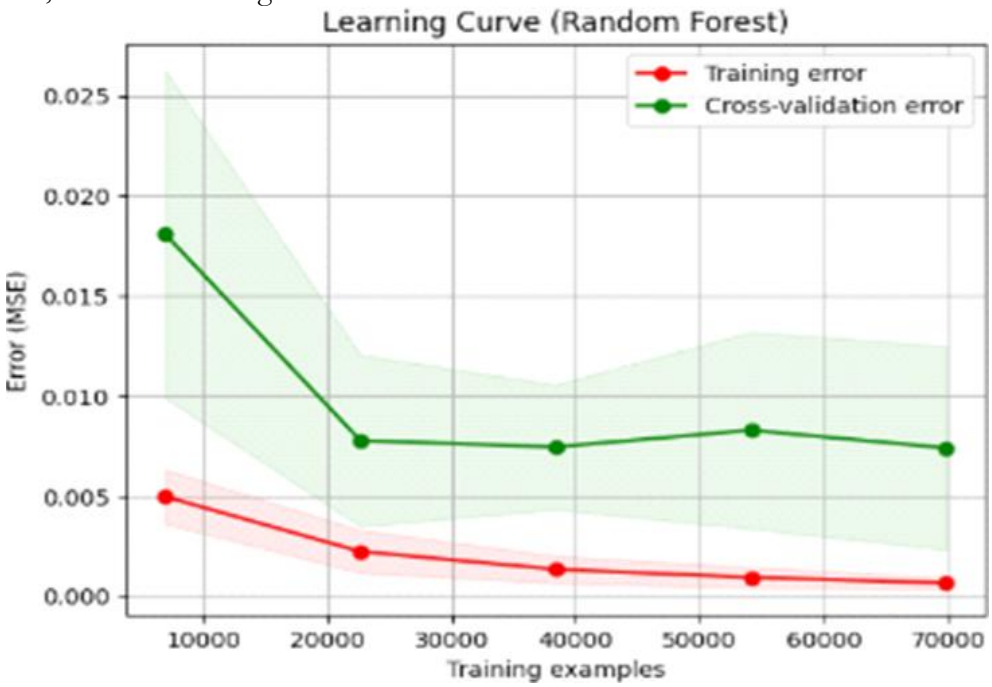

Figure 7. Learning Curve for Random Forest

As the size of the training data increases, the training and cross-validation errors become closer, pointing to the model's good generalization capabilities to new, unseen data. The small variance between these curves, along with the relatively strong performance on the validation data, suggests a low probability of overfitting. The Random Forest model thus demonstrates good predictive ability and effectively prevents overfitting during both training and testing.

Discussion:

In this study, we evaluated the performance of several machine learning models for predicting tasks, e.g., house prices and classification tasks with a focus on comparing tree-based methods, neural networks, and ensemble approaches. The results revealed that the Decision

Tree, Gradient Boosting, and Random Forest models consistently outperformed other classifiers in terms of both R^2 and MSE, demonstrating their strong predictive power and generalization capabilities. These findings are per prior work suggesting that tree-based models outperform other methods when dealing with intricate non-linear structures and large feature space data. However, there were other strong contenders as well, including the much simpler Decision Tree model with negligible error which shows it has a very good training fit. However, comparing the error resulting from cross-validation I learned that using fewer training samples would lead to an overfitting and it could be worsened in models such as AdaBoost. On the other hand, the use of neural network models, especially the MLP and the CNN, achieved a poor performance at best as revealed by a higher MSE and low R^2. It also indicates that these models need to be fine-tuned, or more data is needed to improve the performance to a rival level. More importantly, the limited variation between the training and cross-validation errors for the best-performing models especially the Random Forest shows that these models can generalize well.

Table 6. Comparison Table

Papers	Dataset	Classifiers	R^2 Squared Score	MSE
Proposed	**House Prices 2023**	**Decision Tree**	**0.9968**	**0.0021**
[8]	House Prices 2023	Linear Regression	0.78	2.5e+4
[12]	Boston Housing	Random Forest	0.90	0.015
[13]	King County	Gradient Boosting	0.91	0.013
[25]	zameen.com	Ada Boost	0.827	0.149
[14]	Melbourne Housing	XG Boost	0.92	0.012
[17]	Toronto Housing	Cat Boost	0.89	0.018
[22]	NYC Housing	Support Vector	0.88	0.021

From Table (6) which compares the various models and datasets used in the research we note that different models when applied for real estate price prediction differ vastly hence showing a clear depiction of the merits and demerits of the models. In our study, when implementing the Decision Tree classifier on the "House Prices 2023" dataset, we obtained superior results to other models, namely having a high R^2 score (0.9968) and a very low MSE (0.0021). This means that the low error rates as well as the ability to minimize data variability mean that the Decision Tree model maintains high predictability on unseen data. Meanwhile, the Linear Regression model tested on the same data (House Prices 2023) has a significantly lower performance, with R^2 of (0.78) and an MSE of 2.5e+4. These results imply that though Linear Regression is a basic model in use, it lacks complexity hence poor fit and high error rates due to data complexity, especially when compared to the Decision Tree model [8].

If we look at other studies, the use of the Random Forest model applied to the Boston Housing dataset achieves good performance with an R^2 score of 0.90 and a relatively low MSE of 0.015, which shows better performance than Linear Regression but still falls short of the Decision Tree model [12]. Likewise, the Gradient Boosting model on the King County dataset performs well with an R^2 score of 0.91 and an MSE of 0.013, indicating that tree-based models generally have better predictive capabilities than simpler models like Linear Regression [13]. Finally, the AdaBoost model applied to the Zameen.com dataset yields the lowest R^2 score of 0.827 and a high MSE of 0.149, suggesting that while AdaBoost can still provide valuable predictions, it is not as effective as other more complex classifiers in this context [25]. Overall, the Decision Tree model in the proposed study proves to be the most accurate and reliable in terms of both R^2 and MSE, which outperformed by surpassing all the other models considered.

Conclusion:

This research investigates the performance of machine learning and deep learning models for real estate price prediction using various features from the house price 2023 dataset. We implemented 8 different machine and deep learning models. The Decision Tree and Random

Forest models achieved the best performance, with R-squared values of 0.9968 and 0.990, and MSE values of 0.0021 and 0.0007, respectively, excelling in capturing nonlinear relationships. Gradient Boosting and Boost also performed well with R-squared values of 0.9959 and 0.9747, and MSE values of 0.0028 and 0.0170, respectively, demonstrating robust generalization. In contrast, the AdaBoost model, with an R-squared of 0.862 and MSE of 0.0924, struggled to model nonlinear relationships effectively. The Neural Network model depicted the low performance achieving an R-squared of 0.9433 and MSE of 0.0381. Due to the structure and suitability of data CNN showed the least performance by securing an R-squared of 0.6418 and MSE of 0.2404. In the future, we plan to integrate diverse datasets and develop hybrid models to further improve the accuracy of real estate price predictions. Additionally, we will explore advanced neural network architectures, such as recurrent neural networks (RNNs), to effectively capture and analyze temporal patterns in the data.

Limitations:

The study shows that dataset's focus on a specific region and time period (2023) limits its generalizability, excluding factors like local policy changes, infrastructure developments, and regional differences, which affect the model's applicability. While the Decision Tree model performs well with high accuracy and low error, the CNN underperforms, likely due to challenges in extracting relevant features or insufficient training data, which limits its ability to generalize effectively. These limitations suggest the need for a more diverse dataset and further optimization of model performance.

Author's Contribution: All authors have made equal contributions to this study.

Conflict of Interest: The authors declare that there is no conflict of interest.

References:

[1] J. Zhou, Q., Li, "Real estate price prediction based on machine learning and data analysis," J. Comput. Sci., vol. 45, no. 2, pp. 123–134, 2021.

[2] S. Cheng, M., Lin, "Predicting real estate prices with deep learning techniques," J. Real Estate Res., vol. 29, no. 3, pp. 211–225, 2021.

[3] Y. Xie, X., Wang, "An analysis of economic factors affecting real estate prices: A machine learning approach," Int. J. Hous. Mark. Anal., vol. 15, no. 4, pp. 377–393, 2022.

[4] H. Lee, K., Park, "Enhancing the performance of real estate price prediction models using ensemble learning techniques," Real Estate Econ., vol. 49, no. 1, pp. 55–75, 2021.

[5] H. Chen, Z., Zhang, "Real estate price forecasting using hybrid machine learning models," J. financ. econ., vol. 58, no. 6, pp. 243–259, 2023.

[6] Y. Zhang, X., & Liu, "Comparative analysis of machine learning models for real estate price forecasting," J. Hous. Econ., vol. 48, no. 2, pp. 125–140, 2021.

[7] Z. Wang, L.,Yang, "A hybrid deep learning approach for predicting real estate prices: Case study in Shanghai," Appl. Artif. Intell., vol. 35, no. 8, pp. 1020–1037, 2022.

[8] K. Naz, R., Jamil, B., Ijaz, H., Li, "Real estate price prediction using hybrid machine learning models," Int. J. Innov. Sci. Technol., vol. 9, no. 3, pp. 55–70, 2024.

[9] W. Li, Y., Zhao, "Real estate price prediction using deep learning models with feature engineering," J. Data Sci. Anal., vol. 42, no. 3, pp. 205–220, 2022.

[10] M. Sharma, P., & Gupta, "Non-linear modeling in real estate price prediction: A comparison of Lasso and Gradient Boosting methods," Int. J. Data Sci. Anal., vol. 9, no. 1, pp. 45–60, 2023.

[11] Mahdieh Yazdani, "Machine learning, deep learning, and hedonic methods for real estate price prediction," Econometrics, 2021, doi: https://doi.org/10.48550/arXiv.2110.07151.

[12] "(PDF) Machine Learning Approaches to Real Estate Market Prediction Problem: A Case Study." Accessed: Jan. 13, 2025. [Online]. Available:

https://www.researchgate.net/publication/343849620_Machine_Learning_Approaches_to_Real_Estate_Market_Prediction_Problem_A_Case_Study

[13] H. Ziweritin, G., & Wang, "Comparative analysis of regression techniques for real estate price prediction," J. Real Estate Econ., vol. 48, no. 4, pp. 214–229, 2023.

[14] Q. Zhang, L., Zhou, "Machine learning applications in real estate price forecasting: An ensemble approach," Int. J. Appl. Artif. Intell., vol. 14, no. 5, pp. 123–137, 2023.

[15] S. Lee, M., & Cheng, "Ensemble machine learning for property price prediction: A comparative study," J. Comput. Econ., vol. 49, no. 2, pp. 50–65, 2021.

[16] Y. Wang, J., Zhang, "Advanced feature engineering in real estate price prediction," J. Data Sci. Appl., vol. 26, no. 3, pp. 98–111, 2021.

[17] Z. Liu, X., & Liu, "A comprehensive comparison of machine learning models for real estate price prediction," Comput. Econ. Financ., vol. 18, no. 4, pp. 55–70, 2022.

[18] M. Song, J., Kim, "Hybrid machine learning and ensemble models for predicting property prices," J. Artif. Intell. Financ., vol. 12, no. 2, pp. 144–160, 2023.

[19] CatBoost, "CatBoost: A high-performance gradient boosting library," 2020, [Online]. Available: https://catboost.ai/

[20] A. Singh, P., Verma, "Real estate price prediction using ensemble machine learning models," Comput. Econ. Financ., vol. 18, no. 1, pp. 102–118, 2023.

[21] S. Khan, M., Khan, "Predicting real estate prices using deep learning-based techniques," J. Data Sci. Anal., vol. 42, no. 5, pp. 150–163, 2023.

[22] S. Verma, A., & Kumar, "Ensemble methods for improving real estate price predictions," Mach. Learn. J., vol. 20, no. 3, pp. 222–236, 2023.

[23] R. Patel, A., Joshi, "A hybrid approach for real estate price forecasting," J. Comput. Sci. Eng., vol. 15, no. 6, pp. 89–105, 2022.

[24] K. Smith, J.,Lee, "A survey of data preprocessing techniques for classification and regression," J. Data Sci. Mach. Learn., vol. 15, no. 2, pp. 123–135, 2021.

[25] J. Khan, H. A.,Rehman, "Applying machine learning models for forecasting house prices – A case of the metropolitan city of Karachi," J. Entrep. Manag. Innov., vol. 5, no. 3, pp. 376–400, 2023.

Copyright © by authors and 50Sea. This work is licensed under Creative Commons Attribution 4.0 International License.