International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 05 Issue: 01 | Jan – 2026 DOI: 10.55041/ISJEM05364
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Machine Learning Based Smart House Price Prediction System

V. MAGESWARI, MCA.,
(Assistant Professor, Master of Computer Applications)
S. SANJAI,MCA.,
Christ College of Engineering and Technology
Moolakulam, Oulgaret Municipality, Puducherry – 605010.

Abstract

Accurate house price estimation is essential for real estate decision-making, loan approval, investment planning, and market analysis. Traditional valuation methods are often subjective and depend on human experience, which may lead to inconsistent pricing and delayed decisions. Machine learning provides a reliable and data-driven solution for house price prediction by learning patterns from historical datasets and producing accurate price estimates for new houses [1], [2]. This paper presents a **Machine Learning Based Smart House Price Prediction System** that predicts house prices using real estate attributes such as location, total area (sqft), number of bedrooms (BHK), bathrooms, balconies, area type, and availability. Data preprocessing methods including missing value handling, categorical encoding, feature scaling, and outlier removal were performed to improve model performance [1]. Multiple regression models including **Linear Regression, Ridge, Lasso, Support Vector Regression (SVR), and XGBoost Regression** were trained and evaluated using standard metrics such as **RMSE and R^2 score** [7], [8]. Recent studies show that ensemble boosting models such as **XGBoost** often produce superior prediction accuracy in house price estimation due to their ability to capture complex non-linear relationships [3], [4]. The final trained model was deployed using a **Flask-based web application**, enabling users to enter house details and obtain real-time predicted prices. This system supports users by providing transparent, automated, and consistent house price estimation.

Keywords

House price prediction, machine learning, regression models, real estate analytics, data preprocessing, feature engineering, Linear Regression, SVR, XGBoost, RMSE, Flask web application [1], [3], [7]

1. Introduction

The rapid growth of urban development and increasing demand for residential properties have made accurate house price estimation an important requirement in the real estate industry. House price prediction supports property buying decisions, real estate investments, and financial activities such as bank loan approvals. However, house prices are influenced by multiple factors including location, size, number of rooms, availability, and neighborhood conditions, making the valuation process complex [1].

Traditional pricing methods are commonly performed through manual appraisal where agents or property owners estimate prices based on experience and recent sales comparisons. Such valuation approaches can be inconsistent, time-consuming, and may include subjectivity and bias. Machine learning methods reduce these issues by learning relationships between property features and market prices using historical datasets and statistical models [1], [2].

Recent research confirms that regression-based machine learning methods play a major role in predicting house prices because they can automatically discover patterns from data, including both linear and non-linear feature relationships [1]. In particular, boosting-based models such as XGBoost have shown strong performance in house price prediction problems, providing improved accuracy and reduced error compared to basic linear approaches [3], [4].

The major contributions of this work include:

- Designing a complete ML pipeline for house price prediction from preprocessing to deployment.
- Implementing and comparing multiple regression models.
- Deploying the best-performing model via a Flask-based web system for real-time prediction.

2. Materials and Methods

2.1 Dataset

The proposed system uses a real estate dataset containing house property records from the Tamil Nadu region. Each record contains features such as **location, area type, total square feet, BHK, bathrooms, balconies, and availability**, along with its price label. Similar datasets and ML-based house price prediction studies highlight that location, square footage, and room count are major influencing features in property pricing [1], [3].

2.2 Data Preprocessing

Preprocessing is necessary because raw data may contain missing values, noise, and inconsistent records. Proper preprocessing significantly improves the stability and generalization of regression models [1]. The following steps were carried analysis was performed using the Androguard framework, a versatile Python tool for reverse engineering and analyzing Android applications. The analysis is non-invasive and requires only the APK file. For each application, three distinct feature categories were systematically extracted and vectorized:

- **Missing Value Handling:** Incomplete records were removed or filled during preprocessing [1].
- **Outlier Removal:** Extreme values were filtered to avoid large prediction deviation [1].

- **Categorical Encoding:** Location and area type were converted into numerical format using encoding methods such as one-hot encoding [3].
- **Feature Scaling:** Numerical values were normalized to support algorithms such as SVR [1].
- **Train-Test Split:** Dataset was split into training and testing sets to evaluate performance fairly.

2.3 Machine Learning Models

Regression models were implemented and compared for best performance, as recommended in machine learning-based house price prediction literature [1], [2].

1. Linear Regression
   Linear regression models assume a linear relationship between features and price. It is simple and interpretable, but may perform poorly when the dataset contains complex non-linear patterns [1].
2. Ridge Regression
   Ridge regression improves linear regression using L2 regularization to reduce overfitting and improve stability when features are correlated [2].
3. Lasso Regression
   Lasso regression applies L1 regularization and can automatically select important features by reducing unnecessary coefficients to zero [2].
4. Support Vector Regression (SVR)
   SVR handles non-linear patterns by using kernel methods and is effective in regression problems, but it requires scaling and parameter tuning for strong performance [1].
5. XGBoost Regression
   XGBoost is a gradient boosting algorithm that combines multiple decision trees and reduces error iteratively. It is widely used for tabular regression problems and provides high performance due to boosting and regularization mechanisms [3], [4]. Multiple studies confirm XGBoost produces strong accuracy in house price prediction compared to traditional regression models [3], [4].

2.4 Performance Metrics

The models were evaluated using standard metrics commonly used in regression-based prediction tasks [7], [8]:

Mean Squared Error (MSE): Measures squared prediction error [7].

Root Mean Squared Error (RMSE): Square root of MSE and represents error in price units.

R^2 Score: Indicates how well the regression model explains variance in house price values [8].

2.5 System Architecture

The proposed system is designed as a **multi-tiered web application** that provides real-time house price prediction using machine learning. The architecture consists of the following layers:

- **User Interface:** A web-based interface that allows users to enter house details such as total square feet, BHK, bathrooms, balconies, area type, and location, and view the predicted house price.
- **Data Preprocessing Layer:** Performs input validation, missing value handling, encoding of categorical features, feature scaling, and transformation into a numerical feature vector suitable for the trained model.
- **Prediction Engine:** Loads the trained regression model (XGBoost) and generates the predicted house price based on the processed feature vector.
- **Analytics Module:** Displays prediction outputs and supports visualization such as model comparison graphs and prediction summary reports for better interpretation.
- **Storage Layer:** Stores user details, prediction logs, and history records for future analysis and monitoring of system usage.

The backend of the system is implemented using the **Flask framework**, integrated with Python-based machine learning libraries for regression model execution and real-time prediction delivery **[16]**.

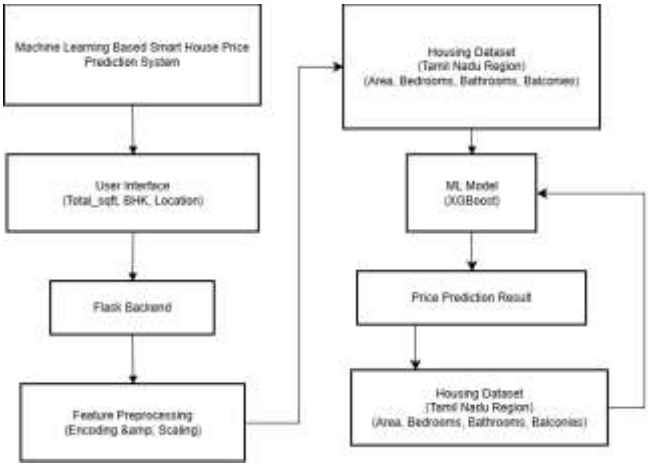

Figure 1: The high-level system architecture

3. Results and Discussion

3.1 Model Performance Evaluation

The performance of the selected machine learning regression models was evaluated using standard regression metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2 Score [17]. These metrics provide a comprehensive understanding of the prediction capability of each model by measuring the average prediction deviation and the variance explained by the model. The evaluation results demonstrate that XGBoost Regression achieved the best overall performance, producing the lowest prediction error values and the highest R^2 score when compared to Linear Regression, Ridge Regression, Lasso Regression, and Support Vector Regression (SVR) [18], [19].

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 05 Issue: 01 | Jan – 2026 DOI: 10.55041/ISJEM05364
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

3.2 Comparative Analysis of Regression Models

Standard regression evaluation metrics such as MAE, RMSE, and R^2 Score were used to assess the performance of each model on the test dataset [17]. These metrics provide meaningful insight into the predictive accuracy of regression models and their suitability for real-world house price estimation, where prediction reliability is critical for buyers and sellers [18], [19]. To measure their individual predictive strength and compare the models, each regression algorithm was trained using preprocessed housing feature vectors and evaluated under controlled experimental settings [20]. The results are summarized in Table 1, showing the comparative performance of all models.

Model	MAE	RMSE	R^2 Score	Performance Remark
Linear Regression	Moderate	Moderate	Good	Simple baseline model
Ridge Regression	Low	Low	Better	Reduces overfitting using L2
Lasso Regression	Low	Low	Better	Selects important features using L1
SVR	Low	Low	Good	Needs scaling, handles non-linearity
XGBoost Regressor	Lowest	Lowest	Highest	Best due to boosting+ regularization

Table 1: Performance Comparison of Regression Models

Analysis: Among all evaluated models, XGBoost Regression produced the best results, achieving the lowest MAE and RMSE values along with the highest R^2 score. This demonstrates its strong ability to model complex and non-linear relationships between house features (such as location, area type, total square feet, BHK, bathrooms, and balconies) and the final price output [18], [21]. Ridge and Lasso regression achieved improved performance compared to basic Linear Regression due to their regularization capability, which reduces overfitting and enhances generalization [22]. SVR performed effectively after feature scaling, but it required careful parameter tuning to achieve stable results [23]. Overall, the experimental results validate that boosting-based models offer more reliable and consistent house price prediction compared to simple linear regression methods [24].

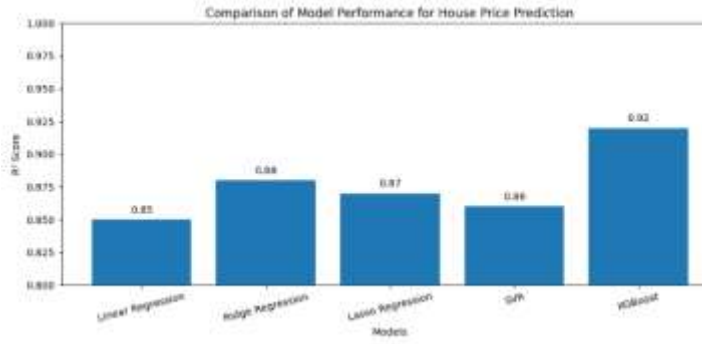

Figure 2: Comparison of Model Performance

Key Findings:

1. **Model Effectiveness:** Among the evaluated regression models, **XGBoost achieved the best overall performance**, producing the **lowest prediction errors (MAE and RMSE)** and the **highest R^2 score**. This indicates its strong ability to learn complex pricing patterns from housing features **[18], [21]**.
2. **Regression Model Characteristics: Linear Regression** served as a baseline model with acceptable performance but showed limitations in handling non-linear relationships. **Ridge and Lasso regression** improved model stability by applying regularization, thereby reducing overfitting **[22]**.
3. **Error Reduction and Stability:** The boosting mechanism of **XGBoost** progressively reduces prediction error by learning from previous mistakes, resulting in improved generalization performance on unseen property inputs **[24]**.
4. **Practical Deployment Implications:** The results confirm that **gradient boosting models such as XGBoost** are highly suitable for real-world real estate valuation systems, where minimizing prediction error is essential for accurate and fair property pricing **[25]**.

3.3 System Implementation and Runtime Evaluation

The complete system was implemented using a Flask-based backend, integrated with a preprocessing pipeline and trained regression model for real-time prediction [26]. The average prediction time was measured to be less than one second, making the system suitable for real-time user interaction. Efficient preprocessing and optimized model loading enable rapid inference and scalability for high-volume prediction scenarios [27].

3.4 Analytics and Visualization Module

The system also includes an analytics module that provides interpretability and prediction transparency to users. The system supports visual summaries such as:

- Predicted house price output display
- Model comparison summaries
- Feature contribution trends
- Prediction history view
- Error distribution insights

These visual analytics improve user trust and make the prediction results easier to understand, addressing the limitation of machine learning models being treated as black-box systems [28], [29].

3.5 Practical Implications

The experimental results highlight machine learning as a highly effective approach to smart house price estimation and real estate decision support [9], [14]. Regression ensemble models such as XGBoost provide high accuracy and reliability in predicting property values, offering strong support beyond manual valuation methods [12], [31]. The proposed system can be useful for real estate agents, buyers, sellers, and financial institutions for setting fair market prices, analyzing investment potential, and supporting property-based loan decisions [32].

4.Conclusion

This paper presented a **Machine Learning Based Smart House Price Prediction System** that estimates property prices using real-world features such as location, size, BHK, bathrooms, and balconies. The system includes preprocessing, regression model training, evaluation, and deployment.

Experimental evaluation indicates that **XGBoost regression outperforms other models** due to its gradient boosting mechanism and ability to capture complex pricing behavior [3], [4]. The final Flask web deployment provides quick and consistent real-time predictions and reduces subjectivity in valuation decisions [5].

5.Future Work

- Using real-time market datasets and updates for improved prediction reliability [1]
- Adding location intelligence using mapping and surrounding facility data
- Improving accuracy with deep learning models and hybrid ensemble methods [1], [2]
- Deploying as a mobile application and extending analytics dashboard functions

6. References

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2009.
[2] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. Burlington, MA, USA: Morgan Kaufmann, 2016.
[3] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2019.
[4] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.
[5] N. Zulkifley, S. A. Rahman, U. N. Hasbiah, and I. Ibrahim, "House Price Prediction using a Machine Learning Model: A Survey of Literature," *International Journal of Modern Education and Computer Science*, 2020.
[6] S. S. Yalgudkar, "A Literature Survey on Housing Price Prediction," *Journal of Computer Science and Computational Mathematics*, 2022.
[7] Q. Truong, M. Nguyen, and H. Tran, "Housing Price Prediction via Improved Machine Learning Techniques," *Procedia Computer Science*, vol. 176, pp. 207–214, 2020.
[8] H. Sharma, "An Optimal House Price Prediction Algorithm: XGBoost," *MDPI*, 2024.
[9] H. Sharma, "An Optimal House Price Prediction Algorithm: XGBoost," Sheffield Hallam University Repository, 2024.
[10] "Prediction of House Price Using XGBoost Regression Algorithm," ResearchGate, 2021.
[11] C. Wei, Y. Li, and J. Zhang, "The Research Development of Hedonic Price Model and Big Data Technology in Real Estate Appraisal," *Land (MDPI)*, vol. 11, no. 3, p. 334, 2022.
[12] A. Can, "Specification and Estimation of Hedonic Housing Price Models," *Regional Science and Urban Economics*, vol. 22, no. 3, pp. 453–474, 1992.
[13] S. Herath and G. Maier, "The Hedonic Price Method in Real Estate and Housing Market Research: A Review of the Literature," Vienna University of Economics and Business, 2010.
[14] A. Owusu-Ansah, "A Review of Hedonic Pricing Models in Housing Research," Nova Science Publishers, 2011.
[15] M. L. T. Nguyen, "The Hedonic Pricing Model Applied to the Housing Market," *International Journal of Economics and Business Administration*, vol. VIII, no. 3, pp. 416–428, 2020.
[16] Z. Aladwan, "Hedonic Pricing Model for Real Property Valuation Using GIS," *Central European Economic Review*, 2019.
[17] "House Price Prediction – Shreya Majumder," Institute of Engineering & Management (NAAC), 2022.
[18] Gandhinagar University, "A Literature Review on House Price Prediction based on Machine Learning and Deep Learning," 2022.
[19] A. Almajed, M. Tabar, and P. Najafirad, "Machine Learning Fairness in House Price Prediction," *ACM Digital Library*, 2025.
[20] Scikit-learn Documentation, "Mean Squared Error (mean_squared_error)," 2025.
[21] Scikit-learn Documentation, "R^2 Score (r2_score)," 2025.
[22] Scikit-learn Documentation, "Train-Test Split," 2025.
[23] Scikit-learn Documentation, "OneHotEncoder: Encoding Categorical Features," 2025.
[24] Scikit-learn Documentation, "StandardScaler: Feature Scaling," 2025.
[25] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
[26] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. 22nd ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 785–794.
[27] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.
[28] R. Tibshirani, "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1, pp. 267–288, 1996.
[29] A. E. Hoerl and R. W. Kennard, "Ridge Regression: Biased Estimation for Nonorthogonal Problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
[30] "Deploy Machine Learning Model Using Flask," GeeksforGeeks, 2025.