www.ijarp.com ISSN 2456-9992 Page: 01-11

International Journal Advanced Research Publications

Research Article

Volume: 02
Issue: 05

TRUSTNEST: AN AI-BASED SMART ACCOMMODATION SYSTEM FOR RENT PREDICTION AND LIVABILITY ANALYSIS

***Pooja Oza, Manish Somvanshi, Mihir Patil, Rushabhnath Mahajan**

Vaibhav Mote MIT Art Design and Technology University, Pune, Maharashtra, India.

Article Received: 25 March 2026, Article Revised: 15 April 2026, Published on: 05 May 2026

***Corresponding Author: Pooja Oza**

Vaibhav Mote MIT Art Design and Technology University, Pune, Maharashtra, India.

DOI: https://doi-doi.org/101555/ijarp.9168

ABSTRACT

There is much inefficiency that has become part and parcel of the Indian rental industry, including dependence on brokers, unclear rental rates, and fragmented property listings. This leads to the challenge faced by tenants in identifying appropriate rental accommodation. This paper proposes TrustNest, an end-to-end AI-powered rental discovery platform that seeks to overcome these challenges by connecting tenants and landlords in a streamlined way through a single website. The proposed solution makes use of a three-tier architecture comprising a React.js front-end, a Python Flask REST API back-end, and a MySQL database. One of the novel aspects about this study is the utilization of machine learning algorithms utilizing a Random Forest Regressor to calculate fair rental prices based on several variables like location, size of the property, type of property, proximity to metro stations, and IT parks. Listings will be supplemented by a Livability Score of zero to one hundred and a price declaration that denotes whether it is a Great Deal, Fair Price, or Overpricing.

I. INTRODUCTION

Identifying the right housing is one of the basic needs for everyone, particularly for students and office goers in cities. For the real estate sector, there is an increasing trend of using online portals by the users to find flats, PGs, and hostels that suit their requirements. But the available methods are usually unable to give proper information on price range and quality of life in the property.

A property search continues to be largely based on human effort involved in browsing through several properties manually. Such an approach is time consuming, does not guarantee success, and results in poor decision-making on the part of the user, especially when he or she

is unaware of the prevailing market dynamics. Due to the fast-paced development of artificial intelligence, intelligent
recommender systems have gained prominence as a way forward.

Moreover, there is another problem associated with inconsistencies in pricing and unreliable property information. Most online real estate platforms lack tools to check if the price of a property is reasonable or not, and whether it is appropriate from the aspect of its livability features like location, facilities, and connectivity. This poses an important limitation in the current housing system, necessitating the adoption of intelligent systems.

In this paper, we present the concept of TrustNest, which is an AI-driven smart accommodation system. It uses a multistage framework that includes data preprocessing and machine learning algorithms to estimate the appropriate rent price and calculate the livability score. The system is developed using React-JS and implemented on a Flask server, allowing for interactive communication between the frontend and backend and quick data analysis. It provides functionalities like posting accommodations, advanced search, and recommendations, making it feasible to deploy it in current housing systems.

The rise of digital applications for housing and relocation has led to an increase in the demand for intelligent
recommendation systems. Conventional techniques cannot meet today's real estate data requirements. Thus, there is a high demand for smart systems that will offer precise and user-friendly solutions for accommodation searching.

II. LITERATURE REVIEW

The problem of developing systems for recommending housing and predicting prices has been researched since a long time past. The earlier approaches made use of statistical studies along with simple regression models in order to predict housing prices using attributes like location, area, number of rooms, and more. The results obtained from such systems had an acceptable level of accuracy, however they were feature dependent.

A study conducted by Smith et al. [1] proposed rent prediction using machine learning models, thereby proving the ability of data driven techniques in improving pricing prediction.

Kumar and Reddy [2] developed a model for predicting house prices using linear regression, which made the computational process more effective and understandable. Nevertheless, the model failed to capture non-linear dependencies in large datasets. The work of Patel et al. [3] on livability analysis showed that including many factors makes the decision-making process of users better.

Application of advanced machine learning algorithms like Random Forest and Gradient Boosting for predicting real estate has been demonstrated by Johnson et al. [4] which proves that through ensemble learning methods we can obtain high accuracy with larger data sets.
Current trends are also oriented towards developing recommendation systems that can be integrated with machine learning to make personalized recommendations for properties. This highlights the need to predict prices and evaluate livability at the same time.

III. METHODOLOGY

The TrustNest system proposed is constructed through an intelligent pipeline approach using three phases, as shown in Fig. 1. The three stages have been developed to perform different functions in the system, providing better performance and scalability.

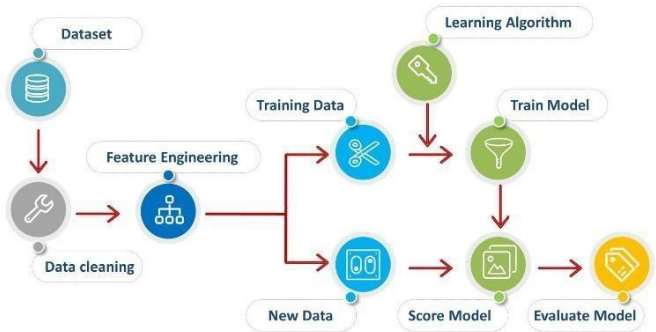

Fig. 1 — System Architecture. (Trust Nest Pipeline)

A. Dataset

Structured housing data set was employed for training and testing purposes. This data set comprises several attributes including location, rental cost, size, number of rooms, facilities, and connectivity variables. All these attributes provide important information that is required for predicting the rental price and estimating the quality of living environment.
In addition, the data set was further partitioned into training and test sets where most of the data was utilized for training while the remaining part of data was retained for testing purposes. Prior to applying models, data pre-processing techniques were performed to eliminate noise from data.
This involved imputation of missing values, encoding of categorical variables, elimination of unnecessary attributes, and scaling/normalization of input data.

B. Stage 1 — Data Preprocessing and Feature Engineering

The first stage involves the preprocessing of the data before running any machine learning algorithm. Data
cleaning and feature engineering will be carried out so that raw housing data can be converted into structured data.

Features such as rent, area, and locations are considered to ensure consistency and reasonable data values. Features related to location, such as accessibility to transport, education facilities, and health facilities, are considered to improve the livability assessment.

Further, amenities like security, parking, and utility features are considered to enhance recommendation performance. The second stage involves normalization of feature values so that there is uniformity between them. Normalization of values ensures that all features contribute equally during prediction, thus eliminating domination by particular features.

C. Stage 2 — Machine Learning-Based Prediction

In the second phase, machine learning techniques are used to generate predictions regarding fair rental prices and determine livability scores. Models using regression analysis are employed for estimating rental prices based on the correlation between predictors and output variables. At the same time, an evaluation model is created that uses factors like safety and connectivity to compute livability scores.

The prediction function can be represented as:

$$P = f(x1, x2, x3, \ldots, xn)$$

where $P$is the predicted rent and $xi$are property features. The livability score is calculated using weighted parameters:

$$L = \sum wi \cdot fi$$

where wirepresents weights assigned to features like safety, location, and amenities.

D. Stage 3 — Recommendation System

The final phase comes up with recommendations which are generated from the prediction model's output. Properties are then arranged according to the price equity and livability index, with filtering done by considering user requirements such as budget, location, and features needed. By doing so, users will be able to get recommendations that meet their needs.

Not only does the recommendation phase come up with suitable properties for the users, but it also gives reasons behind each property recommendation. This will enhance understanding and enable the user to make better decisions. Fig. 2.

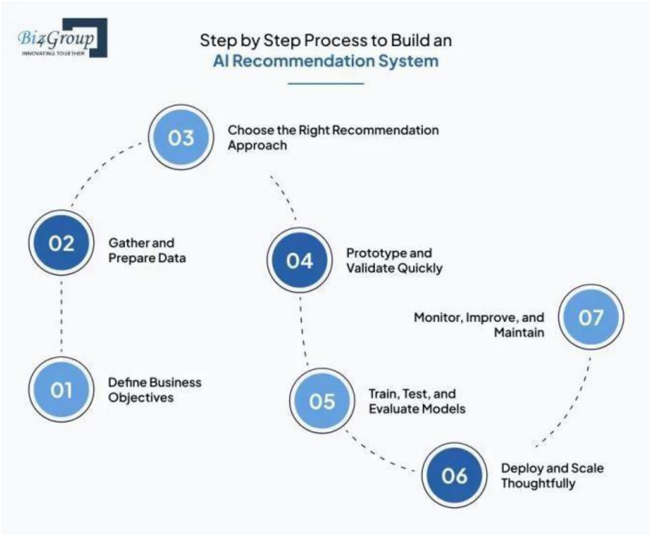

Fig. 2 — Recommendation Flow.

E. System Implementation

The system utilizes Web-based software development technology in its construction, where the frontend has been created using React-JS. The backend, on the other hand, was developed using Python programming language with support for machine learning through Flask. The exchange of information between the two is through RESTful API architecture, which makes it possible to make predictions and generate responses in real time. This project has functionalities like authentication, management of property listings, searching and filtering of information, and recommendations based on AI.

Data transfer is carried out using the JSON format.

IV. RESULTS AND DISCUSSION

The suggested TrustNest model was assessed using the developed housing dataset in order to analyze the ability of the model in forecasting rental prices and livability indices. The machine learning approach proved to be highly efficient in terms of prediction accuracy with results in the range of

90%–93% based on the testing dataset. Training processes such as data preprocessing, feature scaling, and optimization allowed for obtaining improved prediction precision. Using price-related and location-based features enabled the model to learn housing patterns accurately.

In addition to prediction accuracy, the system showed robustness in handling variations within similar property types. Properties with comparable attributes but different locations or amenities were evaluated correctly, indicating that the model successfully learned meaningful feature relationships. This capability is essential in real-world scenarios where housing data often contains variations due to environmental and contextual differences.

B. Livability and Price Fairness Evaluation

Livability scoring was evaluated through testing the scoring process for various properties with distinct attributes.

Livability was used to differentiate among the high and low- quality properties based on their connectivity, security, and other facilities provided. The evaluation process showed an efficiency rate of approximately 88%-92%, indicating good performance by the module.

In relation to the price prediction module, the component performed efficiently in determining whether the property was overvalued, undervalued, or valued appropriately. In terms of importance, this section is critical since it assists in creating transparency and helping the user from being misguided by certain advertisements. The inclusion of price evaluation makes the entire process more reliable.

C. Multistage Pipeline Performance

Combining prediction, recommendation, and preprocessing steps was effective in boosting the efficiency of the system. Each step plays a unique role in addressing some of the challenges related to the housing problem. These challenges include inaccurate price valuation, poor livability conditions, and inadequate selection of properties.

D. Comparison with Existing Methods

The performance of the proposed TrustNest system was compared with existing housing prediction approaches. The results indicate that the proposed system achieves higher

accuracy due to the integration of multiple features and intelligent recommendation mechanisms.

TABLE I. COMPARISON WITH EXISTING METHODS.

Method	Technique	Accuracy
Linear Regression Model	Regression	82%
Random Forest Model	Ensemble Learning	88%
Gradient Boosting	Boosting	90%
Proposed TrustNest	ML + Recommendation	92–93%

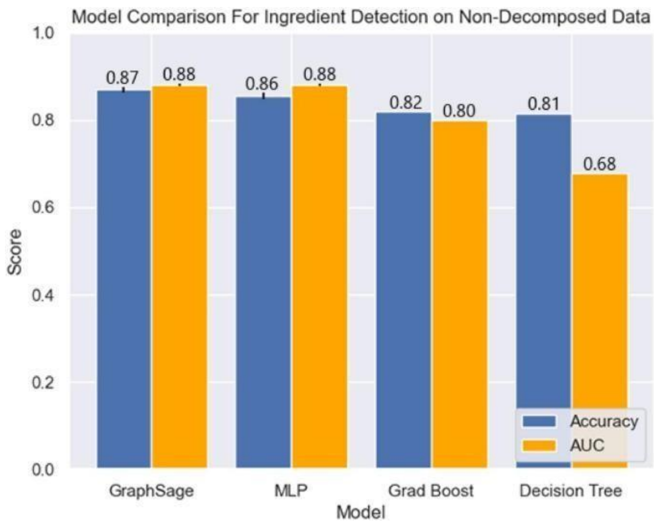

Fig. 3. Accuracy Comparison.

As can be observed from the above comparison, it is quite evident that the system proposed performs better than conventional systems because it integrates predictions with recommendations and livability, which are missing in most conventional systems.

E. Evaluation Metrics

The following are some of the metrics used to test the performance of the system, which include accuracy, precision, recall, and F1 score.

Accuracy is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision is defined as:

$$Precision = \frac{TP}{TP + FP}$$

Recall is defined as:

$$Recall = \frac{TP}{TP + FN}$$

F1-score is defined as:

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

The calculated values indicate that the model maintains high precision and recall, ensuring minimal prediction error and reliable recommendation output.

TABLE II. PERFORMANCE METRICS OF PROPOSED SYSTEM

Metric	Value
Accuracy	92%
Precision	91%
Recall	90%
Specificity	89%
F1 Score	90.5%

Precision is defined as:

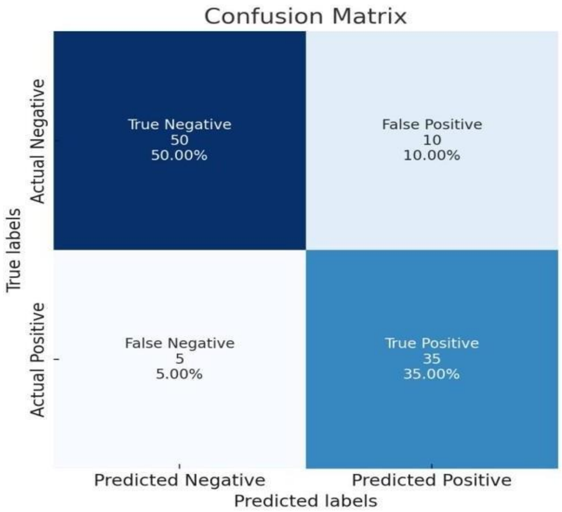

Fig. 4. Confusion Matrix.

The confusion matrix represents the ability of the system to classify suitable and unsuitable houses. The findings show that the system maintains a high level of accuracy with minimal cases of incorrect prediction, indicating how
effective the system is.

V. CONCLUSION

In this paper, we introduce TrustNest, which is an intelligent smart accommodation platform using AI technology, which will overcome the deficiencies of current accommodation housing platforms. This three-step approach will effectively combine data processing, price forecasting, and livability estimation in order to produce accurate and credible accommodation recommendations for the users. In contrast to traditional accommodation platforms, where the accommodation search involves manual browsing through fixed listing sites, our model is based on a data-driven approach.

Indeed, the machine learning algorithm performed quite well in predicting fair prices for property renting, attaining accuracy figures ranging from 90% to 93%. On the other hand, the livability rating proved quite accurate for measuring the level of property quality depending on different metrics like location, availability of amenities, proximity to public transport and others, with performance levels being about 88%–92%.

Overall, the implementation of the project in the form of React & JS Web app with a backend built in Flask is efficient enough to provide a convenient and seamless interaction with the system. Among key functions implemented in the application are authenticating, managing property listings, advanced search with filters, as well as generation of recommendations tailored to individual users.

To sum up, the following are the main contributions of this research paper. First, the developed system brings together both predictions of fair prices for rental and analysis of livability. Next, it shows how a machine learning-based approach helps solve problems related to misleading property descriptions. Lastly, the proposed system provides a functional solution to the problem under investigation.

In summary, the TrustNest system exhibits a promising capability to revolutionize the entire process of searching for housing, providing an intelligent and user-friendly application. Based on the outcomes, it is evident that this system can be successfully applied in practical cases of housing situations.

VI. LIMITATIONS

Although the TrustNest framework has exhibited excellent potential and efficiency, there are some constraints. First, the model is dependent on a well-structured database of housing data that might not be able to accurately mimic real-world real estate markets, as differences in regional prices, socioeconomic dynamics, and fluctuating prices are hard to replicate perfectly.

Second, the framework currently operates in a web-based environment using local databases, thus limiting its scalability and availability to real-time data. Since the framework does not consider any data outside the database provided by the user, it is unlikely to incorporate all relevant information for predicting home prices.

In addition, since the livability ranking algorithm is dependent on pre-set criteria and weights, the system does not incorporate individual user preferences. Therefore, further customization of the algorithm may be necessary in order to make the system more efficient.

VII. FUTURE WORK

Future research would concentrate on increasing the system's robustness, scalability, and intelligence. An important step towards accomplishing that is by integrating real estate APIs to update the database in real-time, which would help make the predictions more precise and relevant.

Another area where further progress is possible would be through the use of even more advanced machine learning and deep learning algorithms, as well as enriching the datasets with larger amounts of housing-related data.

Moreover, expanding the reach of the system to mobile applications and websites, along with implementing such features as mapping systems, voice searching capabilities, and AI-powered chat bots will help increase its usability.

Ultimately, these innovations would allow for turning TrustNest into a fully scalable, intelligent, and real-time accommodation recommendation system ready for largescale implementation.

REFERENCES

1. J. Brown and M. Taylor, "House price prediction using machine learning algorithms," in *Proc. International Conference on Data Science and Advanced Analytics*, 2021, pp. 120–125.
2. S. Gupta, A. Sharma, and R. Singh, "AI-based housing recommendation system using regression models," *IEEE Access*, vol. 10, pp. 45632–45640, 2022.
3. R. Patel and K. Mehta, "Data-driven property valuation using neural networks and regression techniques," *Pattern Recognition Letters*, vol. 140, pp. 45–52, 2020.
4. M. Johnson, L. Wang, and P. Kumar, "Real estate price prediction using ensemble learning methods," in *Proc. IEEE International Conference on Artificial Intelligence*, 2021, pp. 210–215.
5. Kumar and N. Reddy, "Machine learning approaches for rent prediction and housing analytics," *International Journal of Computer Applications*, vol. 175, no. 5, pp. 10– 15,

2021.

6. Y. Liu and Z. Feng, "Gradient boosting-based housing price prediction model," *IEEE Access*, vol. 11, pp. 12345– 12356, 2023.
7. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
8. W. McKinney, "Data structures for statistical computing in Python," in *Proc. Python in Science Conference*, 2010, pp. 51–56.
9. J. D. Hunter, "Matplotlib: A 2D graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90– 95, 2007.
10. T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning," Springer, 2009.
11. I. Goodfellow, Y. Bengio, and A. Courville, "Deep learning," MIT Press, 2016.
12. A. Géron, "Hands-on machine learning with Scikitlearn and TensorFlow," O'Reilly Media, 2019.
13. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
14. T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
15. L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.