RentPredict: A Locally Sourced Machine Learning Approach for House Rent Prediction in Mohammadpur, Dhaka

Abstract

In this study, we analyzed and predicted house rents in Mohammadpur town, Dhaka city. The dataset in the paper was physically collected. We prepared the dataset by preprocessing and encoding it. Different machine learning models, such as decision trees, random forest regression, etc., were applied to the final dataset and achieved a **R2_SCORE** of **0.78** and a **MAE** of **0.04** in the Ridge Regression model. Despite the availability of house-rent prediction research,limitations still remain, including the limited scope of proposed solutions, inefficient feature selections, and neglect of important factors that influence the prediction. We have tried to improve in these sectors to get a more accurate prediction.Therefore, this approach will facilitate accurate rent predictions, removing the information gap for rural people migrating to towns, and also benefiting real estate stakeholders.

Keywords: House rent prediction, Machine learning, Random forest,Predictive modeling, Real estate forecasting.

Introduction

Dhaka is Bangladesh's core, and it is regarded as one of the world's most densely populated cities.Every year, a large number of individuals migrate from rural areas of Bangladesh to Dhaka.There is still an information gap among the general population.Also, rapid population growth leads to an increase in dwelling rents. A criterion for determining house rent should be established to prevent landlords from engaging in aggressive increase. The availability of a house rent prediction model fills an important information gap and improves the general public's situation, as well as helping to standardize rent increases.It is also significant for prospective house buyers, developers, investors, appraisers, and other real estate market participants.

Existing works related to the study include a model can approximately match the volatility of the price-rent ratio in the data [1], spatiotemporal autoregressive model

which demonstrates its superiority for one-day-ahead forecasts [2].Another study by tom(2022) shows that the predictive patterns are highly dependent on whether housing returns and rents are measured in nominal or real terms across countries and time periods [3].

In comparison to previous study, the dataset was collected physically in order to achieve a prediction that is close to the real-time scenario.We included essential columns such as 'Floor', 'Economic class', and others, which broke the limitations of previous efforts and had a significant impact on the prediction.We also analyzed the data to make efficient predictions so that individual agents might enhance their forecasting accuracy by switching from the rational expectations model to the fundamentals-based forecast rule.Finally We achieved a greater accuracy compared to the existing models.

The next part of this paper contains related works of others, which contains works done by other people related to this topic, a Methodology which includes data collection, data preprocessing, data visualization, a Results & Discussion, which shows the performance metrics and finally the conclusion.

Related Works

Research on house rent prediction has garnered significant attention in recent years, leveraging various machine learning techniques & methodologies to enhance forecasting accuracy.This section reviews significant contributions in the field, highlighting key findings, advancements and areas for improvement.

Ahmed et al. explored the use of artificial neural networks to estimate house rents in various neighborhoods within Dhaka City. The study considered forty demographic attributes, including house size, age, type, bedrooms, bathrooms, garages, amenities, and geographical location.Multi-Layer Perceptions (MLP) Neural Network was used to process the dataset.Their model performed with Error size 1*506 and Class as Double.However the dataset contained only thirteen attributes, and results can be different because of variation of NN parameters. [4]

Zhang et al. proposed a joint model that can effectively improve the accuracy and stability of the rent prediction compared to other prediction models.Their methodology involved investigating various machine learning approaches, analyzing different

rent-related features, utilizing XGBoost, LightGBM, and CatBoost algorithms, and proposing a joint model using linear weighting.However narrow scope of investigation, lack of originality in the proposed model, reliance on competition performance for validation are constraints that should have been considered.[5]

Najib et al. addressed the challenge of understanding housing rent dynamics by proposing an Explainable AI framework. Utilizing regression algorithms, particularly tree-based models like Decision Tree, Random Forest, XGBoost, Gradient Boost, and Light Gradient Boost, the framework achieved accurate predictions with minimal error margins. A voting ensemble of these algorithms further enhanced performance, resulting in an MAPE of 11% and an impressive R2 Score of 86%. [6]

Zong and Song used KNN algorithm to perform weighted averaging based on the contribution of neighboring points to the prediction results, and combined the advantages of SVR in processing high-dimensional data and small samples, and proposed SVR and KNN_GBRT fusion models. The improved fusion model had been validated in a housing rental datasets and had better prediction results compared to SVR model and GBRT model. Limitations include taking a long time to optimize parameters using grid search during the model construction process which is not efficient.[7]

Heidari et al. proposed rent prediction models based on lazy learning algorithms lead to higher accuracy and lower prediction error compared to eager learning methods.The methodology involved analyzing seven machine learning algorithms for rent prediction, training models for different house types in the US, selecting features using filter methods, and employing hierarchical clustering based on house type and average rent estimate of zip codes.Limitations of the study include a focus on a specific industry (real estate), a limited selection of machine learning algorithms.[8]

Li et al. studied the pain points of the rental market, based on real renting market data after desensitization. Using the historical data of monthly rent tags to establish a LightGBM (Light Gradient Boosting) model based on machine learning, the accurate forecast of housing monthly rent based on basic housing information was provided having value 96%, which provided an objective measure for the city's rental market.However, some missing features in the dataset such as age of the house, economical class etc. can affect the results.[9]

Rupesh & Kumar.R proposed a model that exhibited the SVM and CNN, in which the Novel Convolutional Neural Network had the highest values. The accuracy rate of Novel convolutional Neural Network is 95.46 was higher compared with Support Vector Machine (SVM) that had an accuracy rate of 91.50 in analysis of prediction of house or flat rent in metro cities with improved accuracy rate.[10]

Kokaish et al. used a method to create a software for predicting property rental prices using the Extreme Gradient Boosting algorithm. Data preprocessing, feature selection, data cleaning, data standardization, and data aggregation are conducted to prepare the data for the machine learning process. The dataset is divided into independent and dependent variables, then split into train and test data. Hyper-parameter tuning is performed to optimize the model. Cross-validation is used to generate a score from the model EG algorithm is able to create property rental price prediction with the average of RMSE of 10.86 or 13.30%.However, the study does not discuss the generalizability of the prediction model to other locations or platforms beyond Airbnb Singapore, which is a big concern.[11]

In conclusion, the existing literature underscores the potential of machine learning models in improving house rent prediction accuracy. While various approaches have been proposed, the continuous refinement and integration of these models remain crucial for addressing the evolving challenges in the real estate market.

Methodology

The methodology section of our work proposes a stepwise approach to maintain an organized workflow and avoid mistakes.Initially, raw data was collected physically and then preprocessed.The preprocessing involved several steps including dropping unnecessary columns that either have no effect or negative effect on the research, filling up missing values,encoding and visualizing the data for better analysis.Finally, to obtain the results, we selected features and used a variety of machine learning techniques.

Below is a workflow diagram that illustrates the procedure.

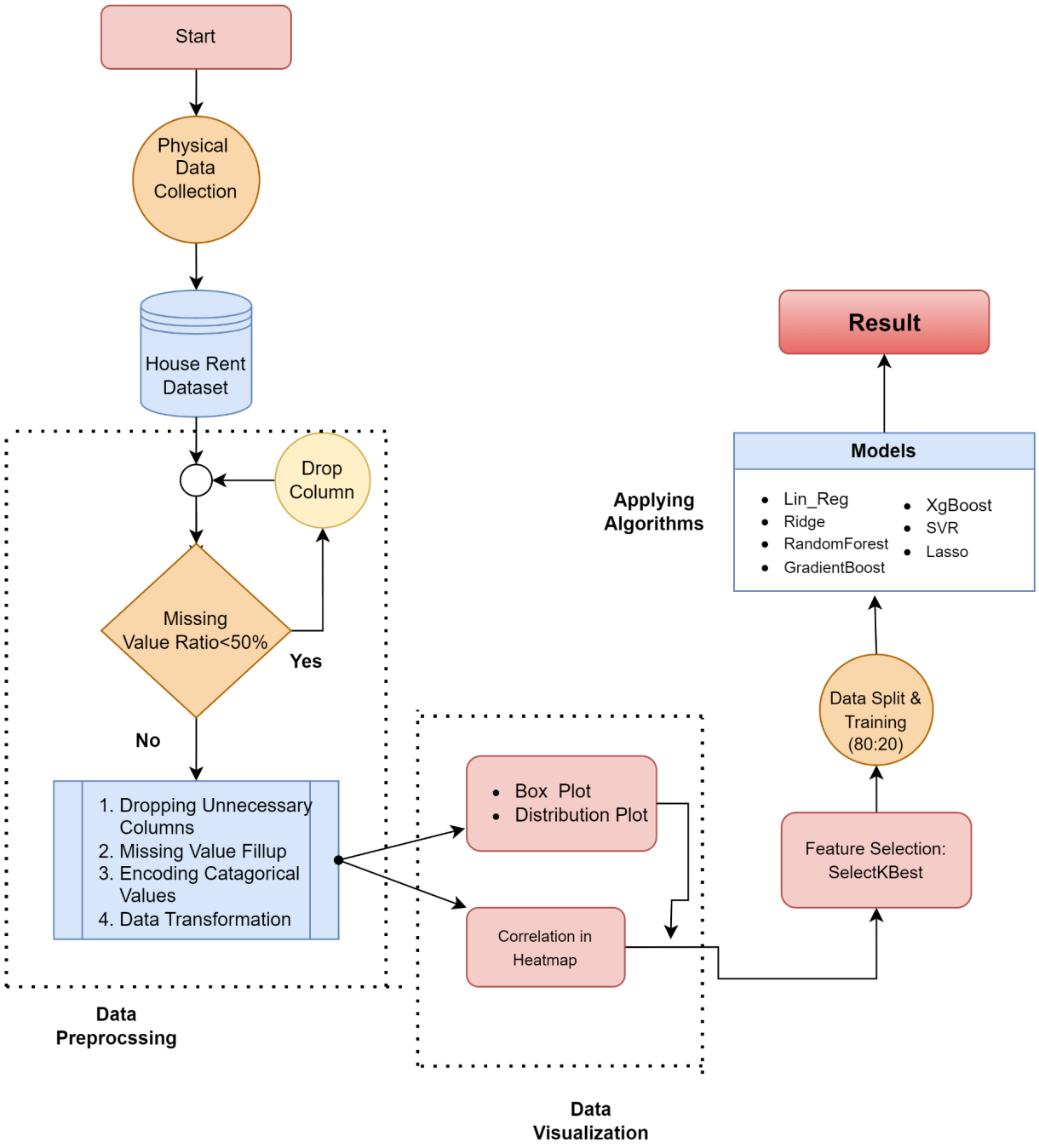

Fig-01: Methodology Diagram

I. Dataset Collection: The dataset used in the paper has been collected from Mohammadpur area in Dhaka city. It is entirely made up of unstructured raw data. It consists of 14 columns and 103 rows.We have included basic features such as 'Beds' ,'Baths','Area' etc. and some unique features such as 'Neighborhood' which shows the economic class of the neighborhood.

	Area	Beds	Baths	Drawing	Dining	Bachelor	Floor	Rent	Service Charge	Total Cost
count	85.000000	103.000000	103.000000	103.000000	103.000000	103.000000	99.00000	103.000000	47.000000	103.000000
mean	1062.764706	2.495146	2.339806	0.572816	0.941748	0.145631	5.00000	18636.407767	3756.595745	20350.582524
std	349.010085	0.669706	0.693858	0.497088	0.235365	0.354461	2.07512	6868.360231	1783.790879	8272.761450
min	350.000000	1.000000	1.000000	0.000000	0.000000	0.000000	1.00000	550.000000	800.000000	1550.000000
25%	800.000000	2.000000	2.000000	0.000000	1.000000	0.000000	3.50000	13500.000000	2375.000000	14000.000000
50%	1050.000000	2.000000	2.000000	1.000000	1.000000	0.000000	5.00000	18000.000000	4000.000000	19000.000000
75%	1350.000000	3.000000	3.000000	1.000000	1.000000	0.000000	6.50000	22000.000000	5000.000000	24500.000000
max	1800.000000	5.000000	4.000000	1.000000	1.000000	1.000000	10.00000	37500.000000	8770.000000	42500.000000

Fig-02: Initial Dataset Description

The integer columns described above shows the standard deviation, minimum and maximum values for each column, and other information.The maximum rent is 37500, as the "Rent" column shows.The 'count' function also reveals missing values in some columns; for example, it indicates a significant number of missing values in the 'Service Charge' column.

II. Data Preprocessing:

A. Dropping Unnecessary columns:

Removing columns that don't contribute to the analysis is a good practice to streamline the dataset and focus only on relevant features.We started by removing any unnecessary columns from my dataset that had no bearing on the data. For example, the column named "location'' which contains the road number ,was dropped. Next, we checked the missing value ratio and dropped the "Service charge" column, which had a missing value ratio of 54%. Finally, we decided to drop the "Total cost" column because it was no longer useful after dropping 'Service Charge'.

Column	Non-Null Count	Data Type
Region	103	object
Neighborhood	103	object
Area	85	float64
Beds	103	int64
Baths	103	int64
Drawing	103	int64

Column	Non-null count	Data Type
Floor	99 non-null	int64
Dining	103 non-null	int64
Bachelor	103 non-null	int64
Features	103 non-null	object
Rent	103 non-null	int64

Fig-03: Aftermath of column elimination

As we can see, after elimination, we have 10 columns left in the dataset information, 4 are categorical, 5 are integer, and 2 have floating-point values.It is evident that there is missing data in the columns labeled "Area" and "Floor".

B. Missing Values:

We then looked at the columns that had values missing.We attempted using each column's mean, median, and modes to fill in the missing values.The dataset differs and varies from the original dataset due to these three processes.Finally, we filled in the missing numbers using the sampling method, which involves taking a sample from the data.

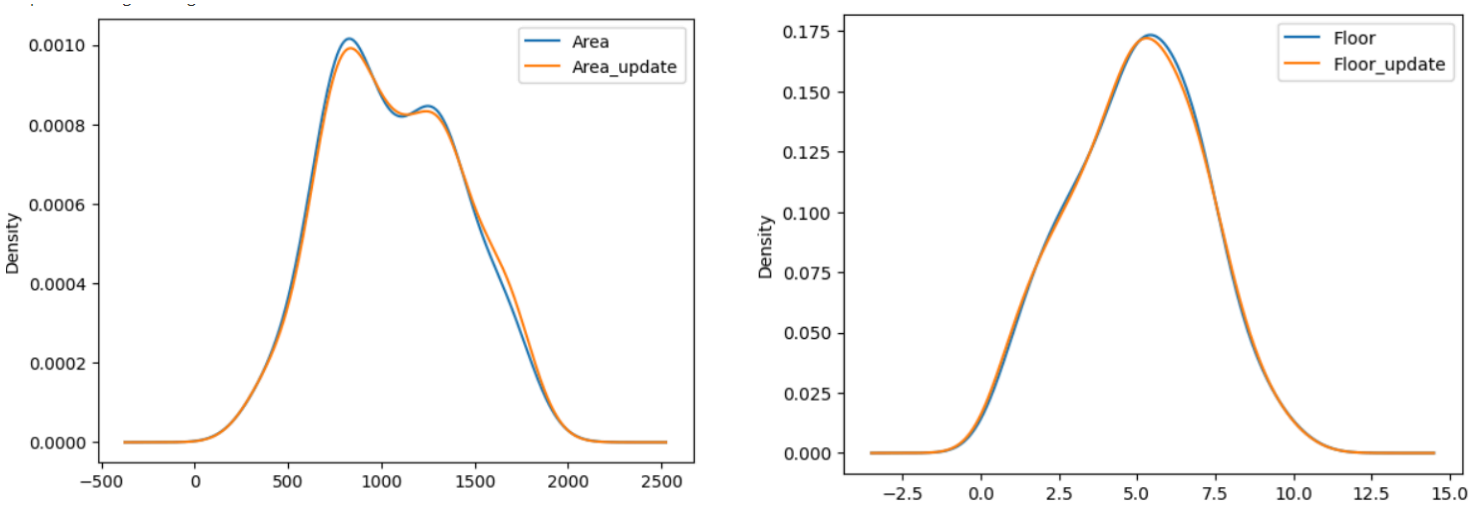

Fig-04: Dataset Variation after missing value fillup

The plots show two lines,the blue one representing the original column and yellow one representing the filled up column.We can see there is almost no variation between the lines.So, we were able to fill up the missing values in the columns with no variation.which is a positive progress on research.

C. Encoding Alphanumeric values:

Categorical values cannot be processed by machine learning models.In order to convert categorical or alphabetic values into numeric values, we encoded them.Three category columns were present in our set: "Neighborhood," "Region," and "Features."The values were encoded using the sklearn preprocessing tool ‘Label Encoder’.

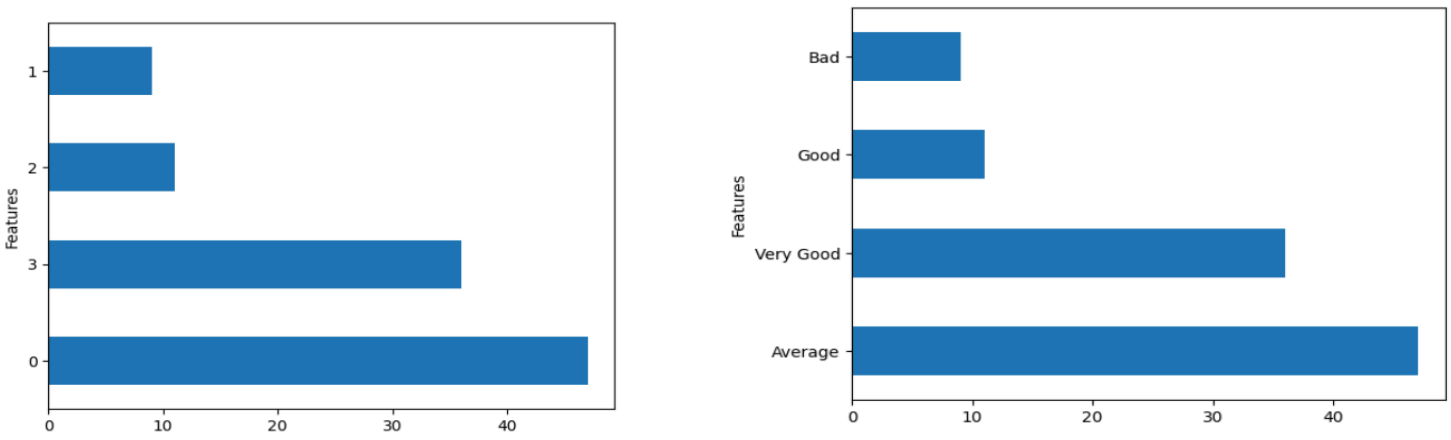

Fig-05:Encoding Aftermath (Feature)

The bar plot shows the before and after encoding of dataset's 'Feature' column.Here the values of average were set to 0, Bad as 1, Good as 2 and Very good as 3.The other columns were encoded using the same format.

D.Data transformation:

Data transformation is a preprocessing approach that ensures fair representation and optimal performance of machine learning algorithms by transforming a dataset's numerical features to a consistent scale.We scaled the encoded dataset within a range in this section.'MinMaxScaler' was selected to scale.The minimum was set to 0, the maximum to 3.

The **MinMax Scaler** uses the following equation:

$$Z_{scaled} = \frac{Z_i - Z_{min}}{Z_{max} - Z_{min}}$$

Where Z is the value to be scaled. Z_max and Z_min denote maximum and minimum values in the individual feature column, respectively.

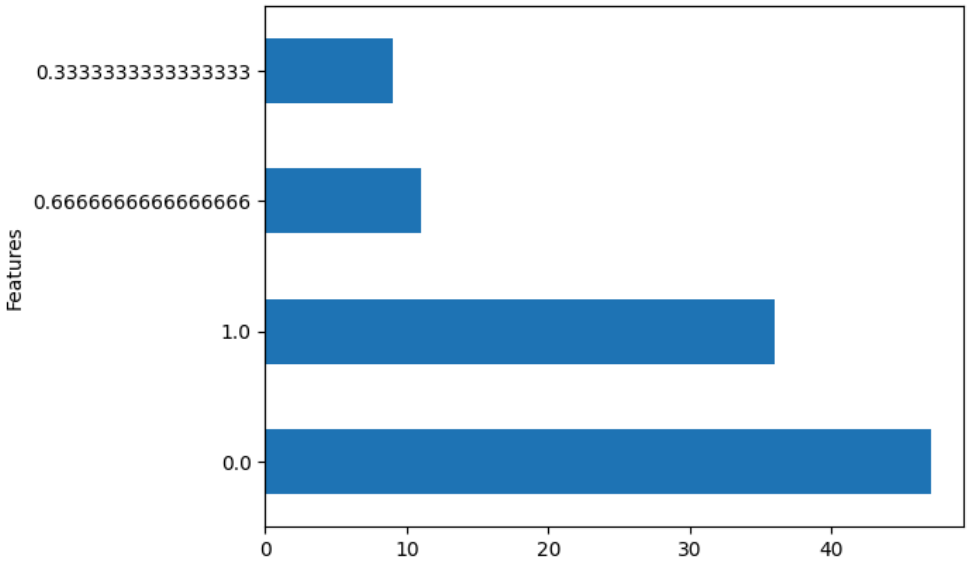

Fig-06:Scaling Aftermath (Feature)

The bar plot illustrates the distribution of values in the 'Features' column after data scaling, showing the frequency of different scaled values ranging from 0 to 1.0. This visualization helps in understanding how the scaling process has normalized the feature values within the specified range.

IV. Data Visualization

The dataset was visualized at various stages of the procedure to help with the anlaysis.To visualize the data, we employed boxplots, histograms, etc..Additionally, we plot feature correlations using a heatmap, which has an impact on feature selection.

HeatMap:

Correlation among different columns of the dataset.

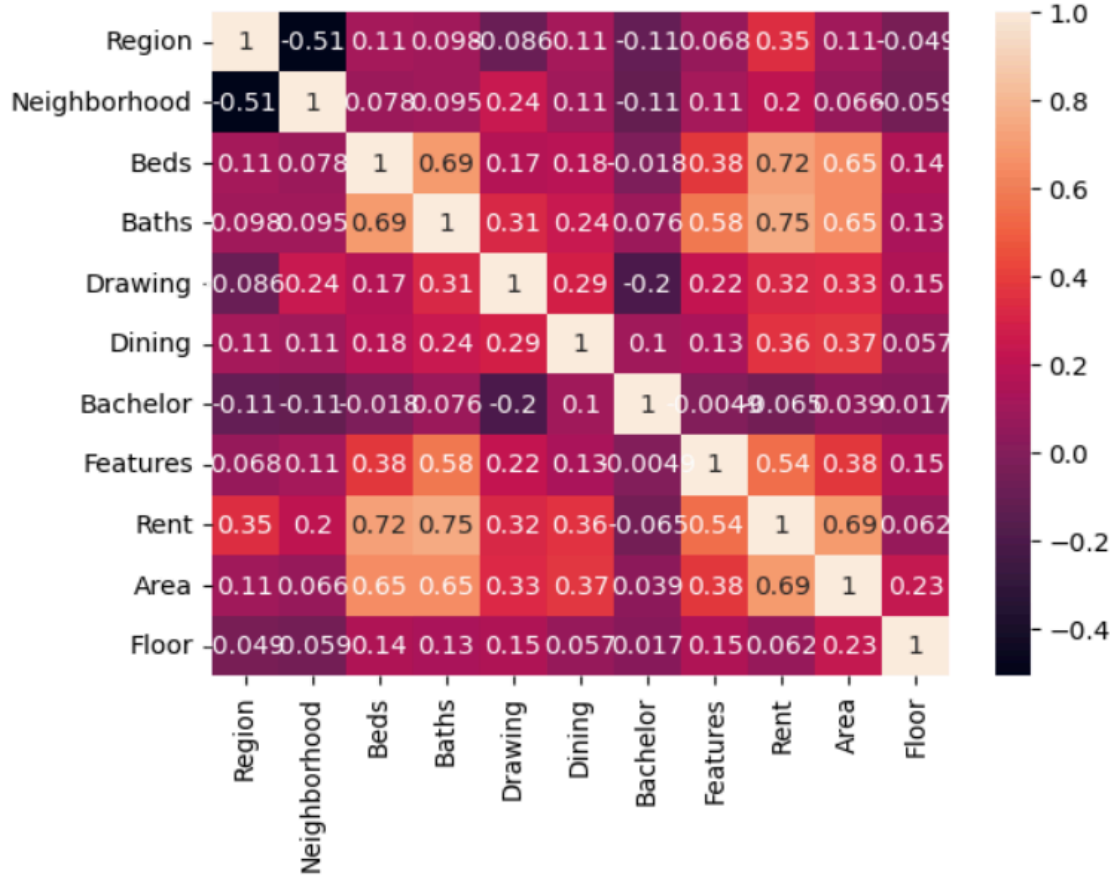

Fig-07: Correlation matrix

The heatmap above illustrates the correlation between the dataset's features. When looking at the overall points, the features "Area" and "Rent" showed strong correlations with all other features, however the features ``Neighborhood" and "Region" showed weak correlations. The correlation between the remaining features was moderate.

Distribution of Rent

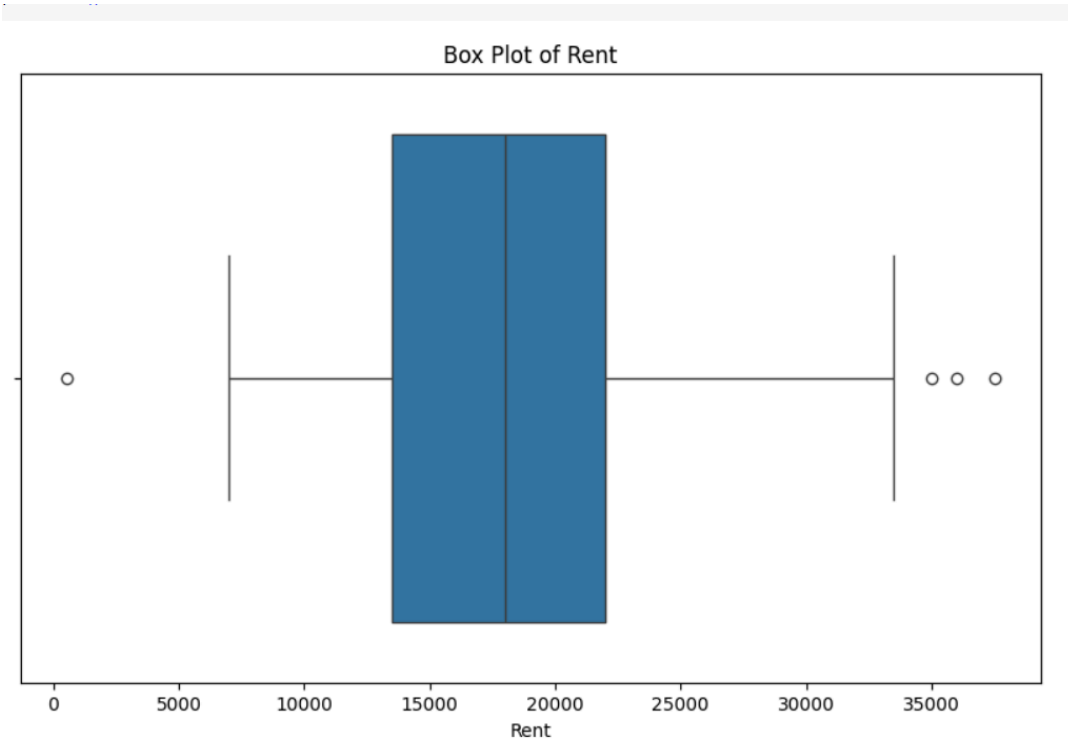

Fig-08: Boxplot

The rent distribution, the interquartile range, and the outliers are displayed in this boxplot.As we can see, the maximum values are concentrated around 13000–23000, with very few values appearing outside of this range. Furthermore, we identify outliers in the dataset that fall outside of the interquartile range of values greater than 35000 and less than 5500.

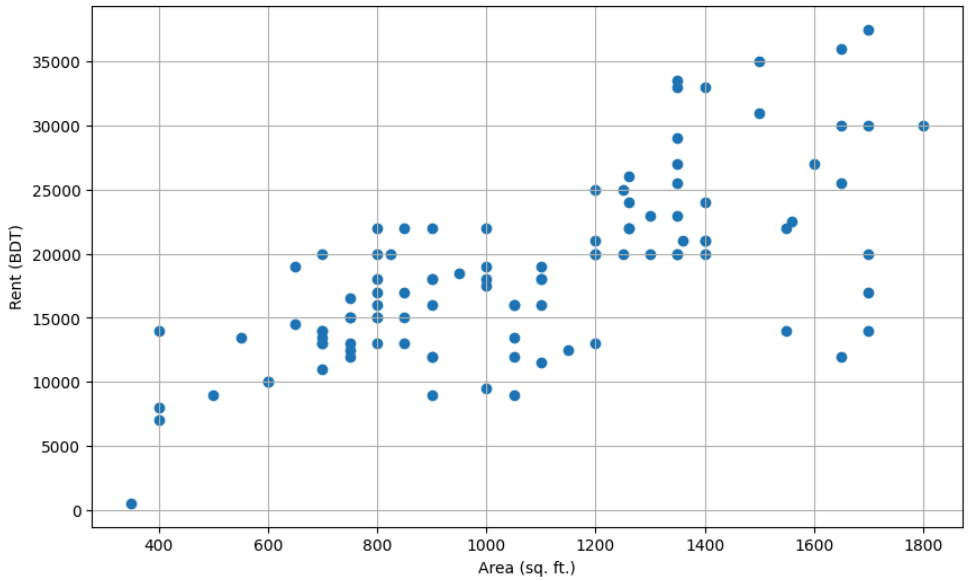

Fig-06: Scatter plot of Rent vs. Area

The scatter plot illustrates the relationship between 'Rent'(BDT) and 'Area'(SQ.ft), indicating a positive correlation where larger areas generally correspond to higher rent values. This trend suggests that as the area of a property increases, the rent tends to increase as well, with some variability.

V. Feature Selection

We choose the features for the dependent and independent variables after scaling.We employed the "SelectKbest" algorithm, which ranks the best features according to a given score by using the "chi2" algorithm. “Area’ feature received the highest score.

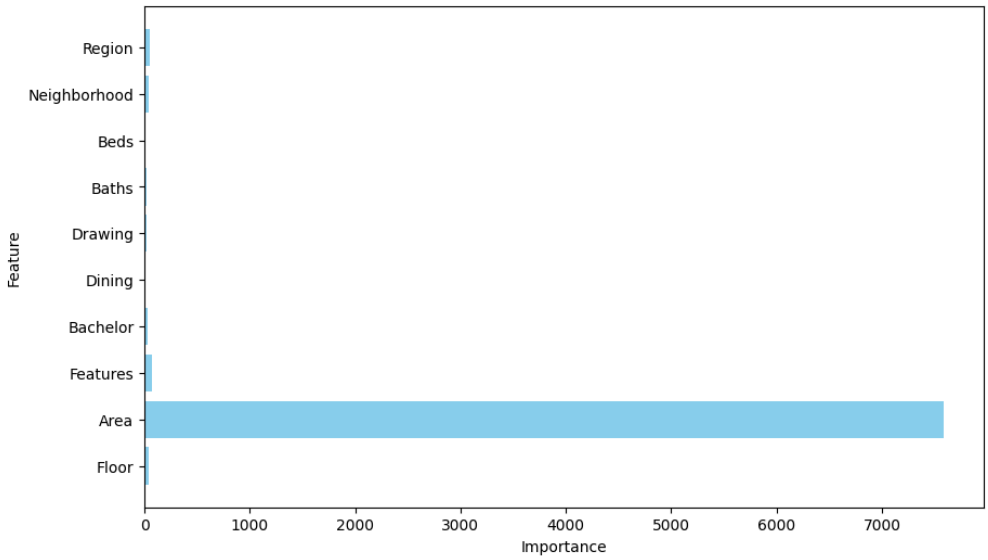

Fig-07: Feature Importance by Score

The above bar plot shows the importance of among all the features or dependent variables.''Area'' performed highly with a score of almost (7000+) while others performed low.

VI. Data Split & Model Evaluation

There were training and testing sets inside the dataset.For testing, 20% of the data was used, and for training, 80%.Various regression algorithms were put into practice.Notable models include **Random Forest**, **LinearRegression**, **Ridge Regression, XgBoost** etc.The use of different models enhances us to improve and gives more choice of choosing the best fit model.

Ridge Regression: It is a regularization technique used in linear regression to prevent overfitting by adding a penalty term to the cost function. This penalty term, controlled by the hyperparameter \(\alpha \), constrains the size of the coefficients, thus reducing model complexity and mitigating multicollinearity issues.

$$\min_w ||Xw - y||_2^2 + \alpha||w||_2^2$$

Here, X represents the feature matrix, y is the target variable, w are the coefficients, and Alpha is the regularization parameter.

Linear Regression: It is a simple and widely-used method for modeling the relationship between a dependent variable and one or more independent variables, assuming a linear relationship.

$$y = w_0 + w_1x_1 + w_2x_2 + ... + w_nx_n$$

In this equation, y is the dependent variable, w_0, w_1, ..., w_n are the coefficients, and x_1, x_2, ..., x_n are the independent variables.

Gradient Boosting: It is an ensemble learning method that builds models sequentially, each new model correcting errors made by the previous ones, by fitting to the residuals. There is no specific equation for Gradient Boosting as it's an ensemble method combining multiple weak learners.

$$\gamma = \frac{Sum\ of\ residuals}{Sum\ of\ each\ p(1-p)\ for\ each\ sample\ in\ the\ leaf}$$

Where Gamma is the coefficient.

Random Forest: It is another ensemble learning method that constructs a multitude of decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) of the individual trees. There is no specific equation for Random Forest as it's an ensemble method combining multiple decision trees.

$$Gini_{\text{RF}} = \sum_{t=1}^{T} \frac{n_t}{N} Gini_t$$

T represents the total number of trees, n_t is the number of samples in a tree's leaf node, N denotes the total number of samples, and Gini_t is the Gini index for the leaf node of the t^{th} tree.

Support Vector Regression (SVR): It is a variation of Support Vector Machines used for regression tasks. It identifies a hyperplane that best fits the data while limiting deviations from it within a threshold determined by a parameter \(\epsilon \). The equation for SVR is:

$$\min_{w,b,\zeta,\zeta^*} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n}(\zeta_i + \zeta_i^*)$$

Where w is the weight vector, b is the bias, C is a penalty parameter, and zeta and zeta* are slack variables.

Decision Tree: It is a non-parametric supervised learning method used for classification and regression. It partitions the data into subsets based on features at each node, aiming to minimize impurity. There is no specific equation for Decision Trees as they are hierarchical structures formed by recursive binary splits based on feature thresholds.

$$Gini(D) = 1 - \sum_{i=1}^{c}(p_i)^2$$
$$Gini(D, A) = \frac{|D_1|}{|D|}Gini(D_1) + \frac{|D_2|}{|D|}Gini(D_2)$$

Here, variable D denotes the dataset or subset, c represents the number of classes, pi signifies the probability of selecting an element from class iii, and A represents the attribute used for splitting.

Lasso Regression: It is a regularization technique for linear regression. It adds a penalty term to the linear regression cost function, but it uses the absolute value of the coefficients instead of their squares. The equation for Lasso Regression is:

$$\min_w \frac{1}{2n_{samples}}||Xw - y||_2^2 + \alpha||w||_1$$

Where the variables are the same as Ridge Regression, with the penalty term using ||w||_1 instead of ||w||^2_2

Experimental Results and Comparison

In this phase, we will assess the models' effectiveness, examine how the models produced results, and conduct an analysis.We measured our models performance on different type of metrics including **MAE**, **MSE** and **R2 Score** to ensure the accuracy of our predictions.

Table I : Models Ranked by R2_Score(R-Squared Score)

Rank	Model	R^2 Score
1	Ridge Regression	0.7811
2	Linear Regression	0.7798
3	Gradient Boost	0.7596
4	Random Forest	0.7291
5	SVR	0.7091
6	Decision Tree	0.6221
7	Lasso Regression	-0.0690

From Table I, we see the results of all models ranked on the basis of their R2 Score.The R-squared (R2) score assesses the proportion of variance in the dependent variable that is explained by the independent variables in a regression model. A high R2 score indicates a strong relationship between the predictors and the target variable, while a low R2 score suggests a weaker relationship. In our table, Ridge Regression performed as the top with a score of 0.7811 and Linear Regression performed second with a slightly less score of 0.7798.Additionally, we see Lasso regression having a negative score, indicating very low performance.

Table II : Models Ranked by MAE (Mean Absolute Error)

Rank	Model	MAE
1	Linear Regression	0.0471
2	Ridge Regression	0.0475
3	Random Forest	0.0573
4	Gradient Boost	0.0555
5	SVR	0.0590
6	Decision Tree	0.0657
7	Lasso Regression	0.1165

Table II was ranked in terms of **MAE**.MAE measures the average absolute difference between predicted and actual values in a regression model, with lower values indicating higher accuracy.In the table, we see Linear regression performed at the top with the lowest value of 0.047 and Ridge in 2nd position with 0.0475.Other models performed moderately, except Lasso.

Table III : Performance in all metrics

Metric	Random Forest	Decision Tree	Linear Regression	Lasso Regression	Ridge Regression	Gradient Boost	SVR
Mean Absolute Percentage Error (MAPE)	15.48	15.73	12.88	34.51	12.90	14.98	16.43
Mean Squared Error (MSE)	0.0054	0.0075	0.0044	0.0212	0.0043	0.0048	0.0058
Mean Absolute Error (MAE)	0.0573	0.0657	0.0471	0.1165	0.0475	0.0555	0.0590
R^2 Score	0.7291	0.6221	0.7798	-0.0690	0.7811	0.7596	0.7091
Root Mean Squared Error (RMSE)	0.0733	0.0866	0.0661	0.1456	0.0659	0.0691	0.0760

From table III, we see that Ridge Regression and Linear Regression models exhibit superior performance.

In R2_score, Ridge Regression (0.7811) and Linear Regression (0.7798) have the highest R^2 scores, indicating a strong fit to the data. These models effectively capture the variance in house rents.In terms of MAE, Linear Regression (0.0471) and Ridge Regression (0.0475) have the lowest MAE values, suggesting high prediction accuracy.And in terms of MSE, Both Ridge Regression (0.0659) and Linear Regression (0.0661) show the lowest RMSE, further supporting their accuracy.

To get a better understanding a visualization of MAE,MSE & RMSE of all models are given below.

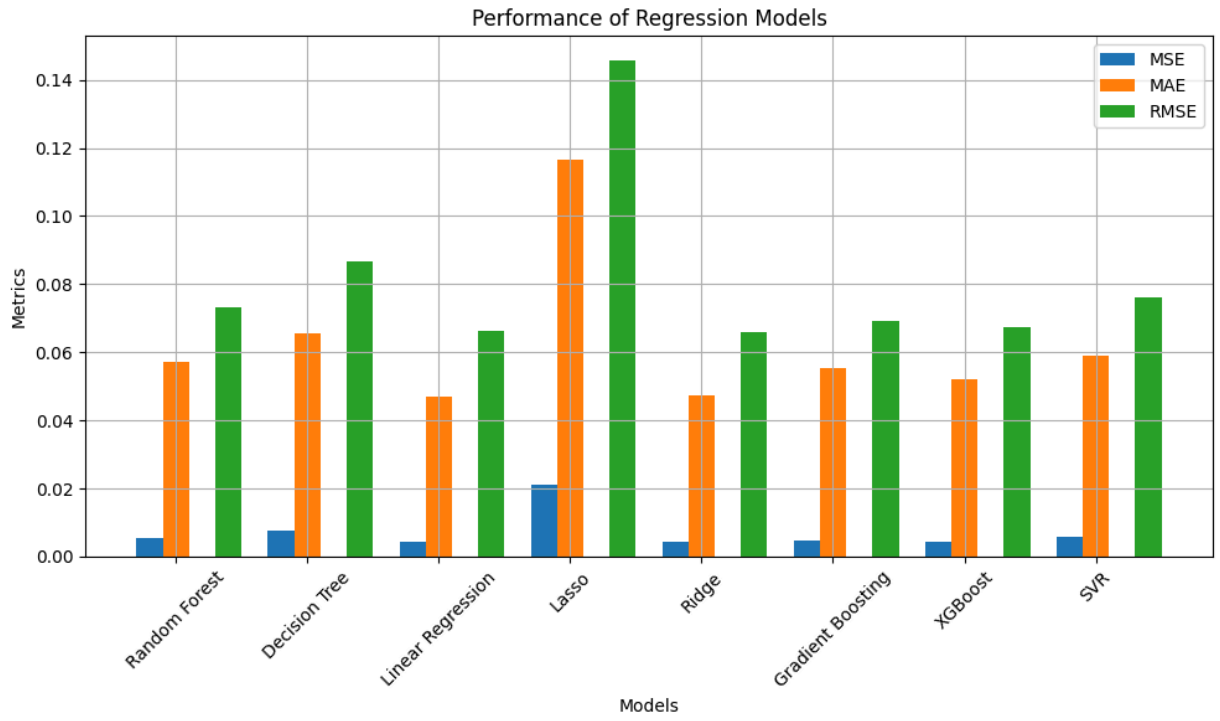

Fig-08: Model Performance

As may be seen from the plot and tables,

Random Forest, Gradient Boost Regression & XGBoost also perform well, though not as effectively as Ridge and Linear Regression, as indicated by slightly higher MAE and RMSE values.

Lasso Regression significantly underperforms, with a negative R^2 score (-0.0690), indicating it fails to model the data effectively. This can result in potential issues with over-penalization of coefficients,which can lead to underfitting.

Decision Tree and SVR models display moderate performance but do not match the accuracy of the top-performing models. The Decision Tree's lower R^2 score (0.6221) indicates a propensity to overfit, while SVR's results show it is less effective compared to ensemble methods.

In summary, Ridge Regression and Linear Regression are the most suitable models for house rent prediction in this context, demonstrating high accuracy and robustness. Potential improvements can be achieved in future research through ensemble methods or hybrid models to further enhance predictive performance.

Discussion

The discussion section of this academic research paper delves into the implications, and limitations arising from the methodology and experimental findings.A useful foundation for examining the datasets is provided by the methodology used in this study.It includes gathering data physically, choosing models and features methodically, and performing meticulous preprocessing, which includes cleaning ,encoding, and transforming the dataset.A flow chart diagram was also employed to enhance the explanation of the whole process.Finally, Ridge Regressor was found to be the best-performing model among the others after an examination and comparative analysis of the models using performance metrics.

However, limitations persist even after a thorough investigation.One is the study's scope, which is restricted to the Mohammadpur Area of Dhaka.Furthermore, the outcomes are impacted by the lack of ensemble strategies.

Conclusion

In this work, we were able to forecast housing rent in the Mohammadpur Area of Dhaka using machine learning techniques; the results showed two models with a significantly improved **MSE** of **0.04** and **R2_score** of **0.78**.Although *unique*, this strategy of predicting property rent by fusing machine learning algorithms with actual, physically acquired data is quite successful.It can help close the knowledge gap that exists between rural residents and metropolitan rents. It can also encourage investment in the real estate sector, which will boost the economy by fostering stability, growth, and the availability of cheap housing.

References:

1. Engsted, Tom, and Thomas Q. Pedersen. "Predicting Returns and Rent Growth in the Housing Market Using the Rent-Price Ratio: Evidence from the OECD Countries." *Journal of International Money and Finance* 53 (2015): 257-275. doi:10.1016/j.jimonfin.2015.02.006.

2. Füss, Roland, and Johannes Koller. "Residential Rent Predictions." *Real Estate Economics* (2015): 1-28.

3. Engsted, Tom, and Thomas Q. Pedersen. "Predicting Returns and Rent Growth in the Housing Market Using the Rent-Price Ratio: Evidence from the OECD Countries." *Journal of International Money and Finance* 53 (2015): 257-275. doi:10.1016/j.jimonfin.2015.02.006.

4. Ahmed, S., Md. M. Rahman, and S. Islam. "House Rent Estimation in Dhaka City by Multi-Layer Perceptions Neural Network." *International Journal of U-and e-Service, Science and Technology* 7, no. 4 (2014): 287-300. doi:10.14257/ijunesst.2014.7.4.26.

5. Zhang, Ke, Liang Shen, and Ning Liu. "House Rent Prediction Based on Joint Model." In *Proceedings of the 2019 8th International Conference on Computing and Pattern Recognition*, 2019.

6. Najib, Tanzina, Farhana Muntasir, and Wasif Al Wazed. "Transparency in House Rent of Dhaka: An Explainable AI Based Predictive Framework." In *Proceedings of the 6th Industrial Engineering and Operations Management Bangladesh Conference*, December 2023. doi:10.46254/BA06.20230146.

7. Zong, Hui, and Jianping Song. "Research on the Prediction Model of House Rent Based on Machine Learning." In *Proceedings of the 2nd International Conference on Bigdata Blockchain and Economy Management*, ICBBEM 2023, May 19-21, 2023, Hangzhou, China.

8. Heidari, Mahdi, Saeed Zad, and Siavash Rafatirad. "Ensemble of Supervised and Unsupervised Learning Models to Predict a Profitable Business Decision." In *2021 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS)*, 1-6, 2021.

9. Li, Jingwen. "Monthly Housing Rent Forecast Based on LightGBM (Light Gradient Boosting) Model." *International Journal of Intelligent Information and Management Science* 7, no. 6 (2018): 9-13.
10. Rupesh, K. S. "Better Accuracy for House or Apartment Rent Prediction in Metro Cities Using CNN Compared to Support Vector Machine." *Journal of Survey in Fisheries Sciences* 10, no. 1S (2023): 2631-2641.

11. Kokasih, Muhammad F., and Agnes S. Paramita. "Property Rental Price Prediction Using the Extreme Gradient Boosting Algorithm." *Journal of Artificial Intelligence Research* (2020): 1-15.