AirBnB Price Suggestion Model

Varsha Meghana Kanmuri
UNI:vk2497
Department of Computer Science
Columbia University
vk2497@columbia.edu

Yatharth Bansal
UNI:yb2540
Department of Electrical Engineering
Columbia University
yb2540@columbia.edu

Allison Reiling
UNI:ar4513
Department of Applied Physics and Applied Mathematics
Columbia University
ar4513@columbia.edu

Lylybell Teran
UNI:klt2162
Data Science Institute
Columbia University
klt2162@columbia.edu

Index Terms

Price suggestion model, machine learning, regression, random forest, neural networks, SHAP.

I. Abstract

In this project, we aim to analyze the 2020 Bay Area Airbnb dataset to gain insights into the factors that affect the pricing and perceived value of Airbnb listings. For this purpose, we propose to incorporate regression models to predict listing prices based on listing features. The goal is to help hosts optimize their prices and improve their listings to increase visitor satisfaction and maximize earnings, and to provide insights into the Bay Area Airbnb market to inform the development of new products and services for Airbnb hosts and travelers.

II. Background

Airbnb is a platform where people can rent lodging accommodations. It has become a popular alternative to traditional hotels, and the number of Airbnb listings has grown significantly in recent years. The Bay Area, which includes cities like San Francisco, Oakland, and San Jose, is one of the most popular destinations for Airbnb rentals in the United States. Understanding the factors that affect Airbnb pricing in the Bay Area can provide insights into how hosts can maximize their earnings and how travelers can make informed decisions about where to stay.

III. Dataset Description

For this project, we plan to use the Bay Area Airbnb dataset updated in 2020, which is available on Kaggle. The dataset contains information on Airbnb listings in the Bay Area, with 106 columns including listing price, the location of the listing, and features of the listing such as the number of bedrooms, bathrooms, and amenities. There are also reviews for each listing, including the date of the review, the reviewer's ID, and the comments they left.

IV. Exploratory Data Analysis and Preprocessing

Our target variable is the Airbnb listing price and our aim is to build a model that predicts price based on features and amenities. As displayed in Fig. 1, the distribution of listing prices is highly skewed, which could eventually lead to a poor-performing model. As a solution, we performed a logarithmic transformation of the data and split the data using stratified sampling based on quantiles.

In addition, we visualized the variation in average listing price by categorical variables such as room type and bed type (Fig. 2). For example, we can see that listings that are an entire home or apartment-style are associated with higher listing prices. Additionally, listings with a real bed, airbed, or couch tend to be priced higher than those with a pull-out sofa or futon-type bed.

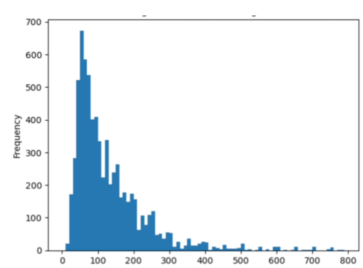

Fig. 1. Average Price by Host Neighborhood.

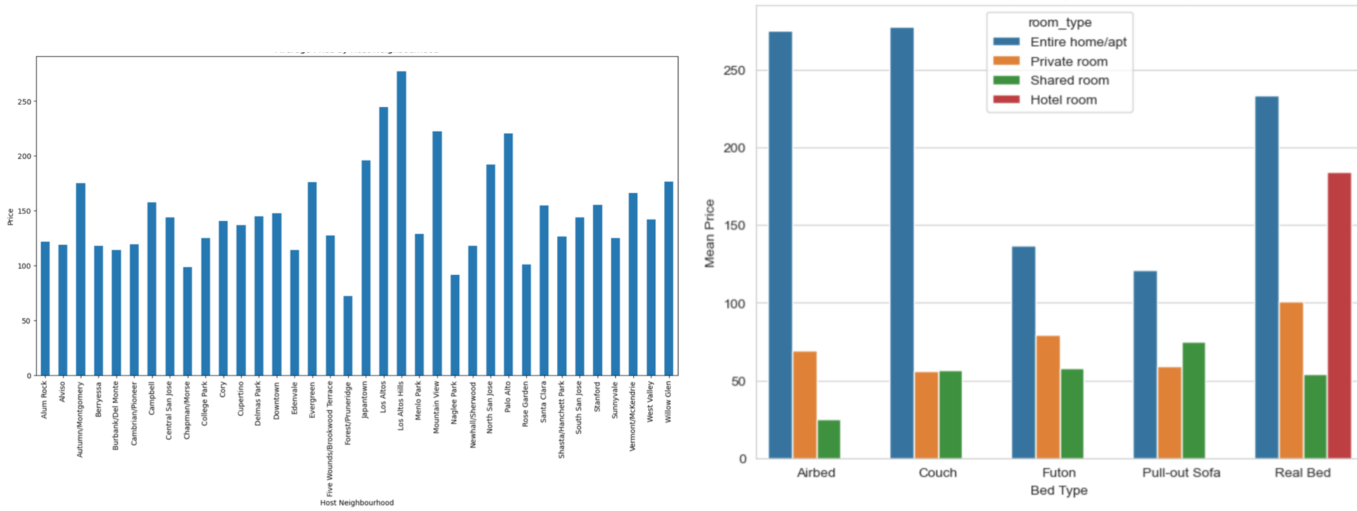

Fig. 2. Average Price by Host Neighborhood (left) and Average Price by Bed Type and Room Type (right).

Furthermore, we plotted a scatterplot matrix (Fig. 3) of top amenities versus listing price. We can see a positive correlation between listing price and amenities — as the value of the amenity in question increases, the price of the Airbnb also increases. We can also see left-sided clusters, or subgroups in the data, which indicate that there is a concentration of data points with low values for one or both of the variables being plotted.

We also examined the property ratings versus listing price and could see that higher-priced listings are not necessarily a better value, meaning customers are not necessarily getting more for their money. Our Machine Learning model will aim to help Airbnb owners list their properties at a price that will set them up to receive optimal reviews from their customers.

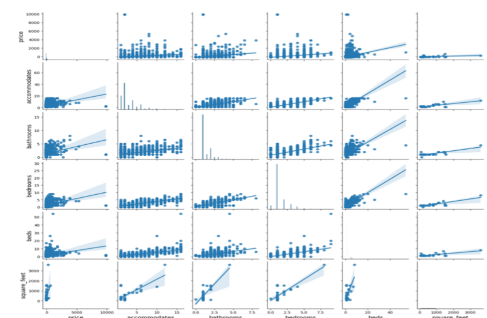

Fig. 3. Amenities of Airbnb Listings versus Price.

As part of our data preprocessing, we decided to drop few columns that were not insightful, like host id. We imputed columns consisting of more than 75% null values with the median (for numerical features) and mode (for categorical features). We used ordinal encoding for continuous columns like bathrooms, bedrooms, beds, and host response time. We used one-hot encoding for categorical columns such as location, bed type, room type, and property type. Using Standard Scaler, we scaled the data to easily compare our machine learning models later on.

V. Feature Engineering

In this study, we present our feature engineering process on the Bay Area Airbnb dataset obtained from Kaggle. Our goal was to prepare the data for a regression task that would predict the price of a rental property based on its features.

We began by conducting an exploratory data analysis to identify the data types and unique values of the data distribution. Based on our EDA insights, we split the dataset into development and test sets using quantile distribution to balance the target variable in both sets.

Next, we identified categorical and numerical variables based on their data type and unique values, with few exceptions due to misinterpretation by the pandas library. We handled missing values by imputing the null values less than 75% using the mean and mode for numerical and categorical columns, respectively, on the development set.

Outlier analysis was performed to identify and replace outliers. For numerical values, we replaced values above the 99th percentile with the 99th percentile value. For categorical columns, we replaced outliers with the most common category.

To handle differences in the scales of different features, we standardized the data by scaling the numerical data. We then performed one-hot encoding based on the cardinality of categorical data.

We also performed feature selection by analyzing the correlation matrix and dropping highly correlated features to deal with multicollinearity in the regression task.

Finally, we used the prepared features to train our model. Our feature engineering process resulted in a final feature set with 443 features, which we used to train our model for the regression task.

In summary, our feature engineering process involved handling missing values, outlier analysis, scaling, one-hot encoding, and feature selection. The resulting feature set allowed us to effectively train our model to predict the price of a rental property based on its features.

VI. Machine Learning Models

We first split the entire dataset into development data (80%) and test data (20%) using the stratified sampling technique to account for the listing price skewness in the dataset. Next, we use 80% of the development data for training the models with the randomized search 5-fold cross-validation technique (20% of development data for validation in each cross-validation fold and for hyper-parameter tuning in each model). We then assess each model's test performance on the remaining 20% of test data. For our experiments, we used the Decision Tree, Random Forest, CatBoost, and Multi-layer Perceptron regression models to predict the listing price based on the features of the listing, such as the number of bedrooms, bathrooms, and amenities. The goal is to create models and techniques to help hosts optimize their prices and improve their listings to increase visitors' satisfaction and maximize their earnings.

VII. Results

Table 1. Best Hyperparameters obtained after RandomizedSearchCV

Machine Learning/Deep Learning model	Best Hyperparameters
RandomForest Regressor	{'n_estimators': 200, 'min_samples_split': 5, 'min_samples_leaf': 1, 'max_features': None, 'max_depth': 15}
CatBoost Regressor	{'task_type': 'CPU', 'learning_rate': 0.1, 'l2_leaf_reg': 1, 'iterations': 200, 'grow_policy': 'Lossguide', 'depth': 4, 'border_count': 32, 'bagging_temperature': 5}
Multi-layer Perceptron regressor	(activation='logistic', alpha=0.01, hidden_layer_sizes=(512, 512), learning_rate='adaptive', random_state=42, solver='sgd')

Table 2. Model performance metrics

Machine Learning/Deep Learning model	Mean Squared Error (Dev set)	Mean Absolute Error (Dev set)	R-squared (Dev set)	Mean Squared Error (Test set)	Mean Absolute Error (Test set)	R-squared (Test set)
Decision Trees Regressor	8.023e-33	1.099e-17	1.0	0.2427	0.3327	0.5492
Random Forest Regressor	0.0519	0.1619	0.9133	0.1112	0.2368	0.7934
CatBoost Regressor	0.1092	0.2359	0.8177	0.1074	0.240	0.8005
Multi-layer Perceptron regressor	0.1504	0.2706	0.7489	0.1274	0.2661	0.7634

CatBoost has the best performance, followed by the RandomForest regressor model. While the MLPRegressor(Neural Net) performs decently well, future work could use other architectures like increasing the number of layers/units and testing on other activation functions to gain higher performance.

Feature importance plots : Based on the results from the feature importance graphs for the RandomForest and CatBoost models, we see that the features like number of bedrooms, cleaning fee , accommodates (number of people) and private room type and are the most important factors influencing the price of the listing.

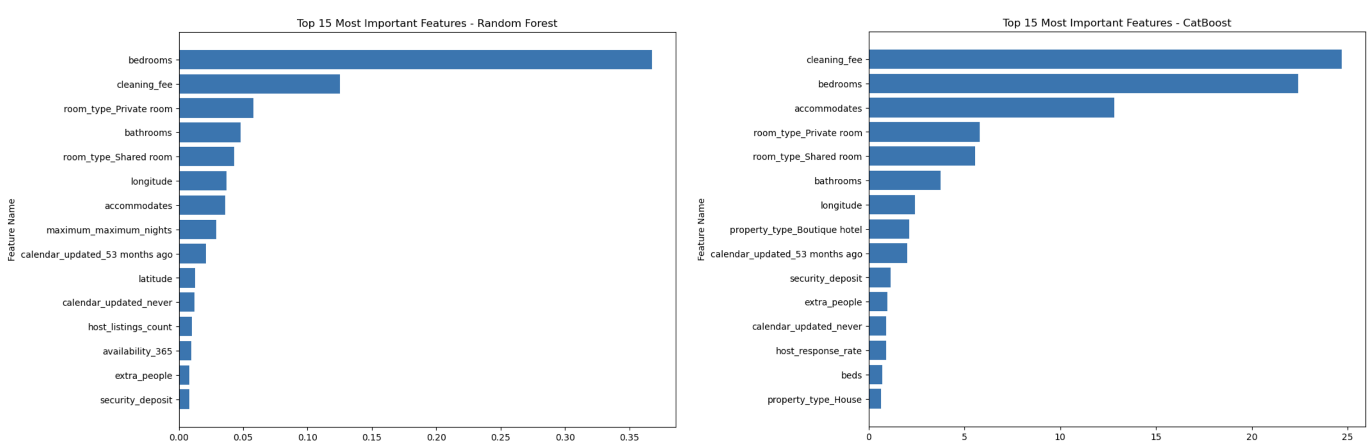

Fig. 4. Feature Importance Plots

SHAP summary plot: We plot the SHAP summary plot for 300 data samples for the Neural Networks (MultiLayer Perceptron) regressor. We observe that features like private room time and accommodates are the most important to here as well.

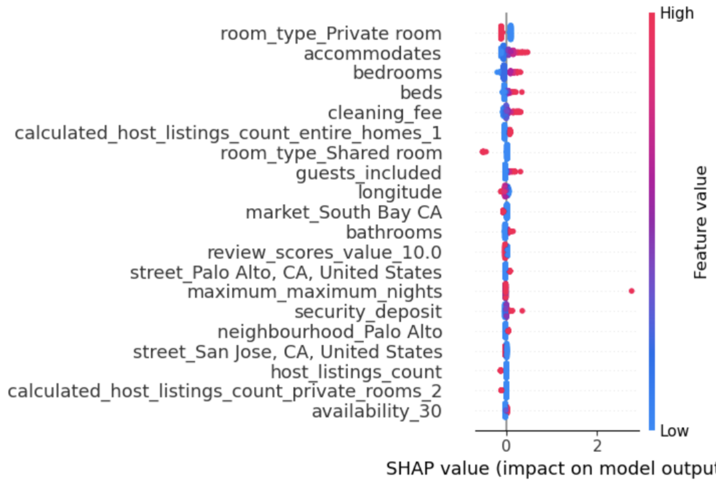

Fig. 5. SHAP plot for MLP Regressor

VIII. Conclusion

According to the feature importance graphs of the Machine/Deep Learning models, we could identify the features of the listing (like bedrooms) that have the most significant impact on the perceived value of the listing. This project successfully provides insights into what features hosts should focus on improving to increase the perceived value of their listings. Additionally, the exploratory data analysis can provide insights into trends and patterns in the Bay Area Airbnb market that can inform the development of new products and services for Airbnb hosts and travelers.

IX. Future Work

The project can be extended to utilize textual features like name, description, and neighborhood overview to create embeddings using a language model/TF-IDF/word2vec and then do the regression to identify how it influences the price of the listings. We can also plan to identify critical topics/themes of top-rated listings.