Journal of
Risk and Financial Management

MDPI

Article

Maximizing Profitability and Occupancy: An Optimal Pricing Strategy for Airbnb Hosts Using Regression Techniques and Natural Language Processing

Luca Di Persio [1,†] and Enis Lalmi [2,*,†]

[1] Department of Computer Science, College of Mathematics, University of Verona, 37134 Verona, Italy; luca.dipersio@univr.it
[2] Department of Computer Science, Faculty of Computer Science, University of Verona, 37121 Verona, Italy
[*] Correspondence: enis.lalmi@studenti.univr.it
[†] These authors contributed equally to this work.

Abstract: In the competitive landscape of Airbnb hosting, optimizing pricing strategies for properties is a complex challenge that requires revenue maximization with high occupancy rates. This research aimed to introduce a solution that leverages big data and machine learning techniques to help hosts improve their property's market performance. Our primary goal was to introduce a solution that can augment property owners' understanding of their property's market value within their urban context, thereby optimizing both the utilization and profitability of their listings. We employed a multi-faceted approach with diverse models, including support vector regression, XGBoost, and neural networks, to analyze the influence of factors such as location, host attributes, and guest reviews on a listing's financial performance. To further refine our predictive models, we integrated natural language processing techniques for in-depth listing review analysis, focusing on term frequency-inverse document frequency (TF-IDF), bag-of-words, and aspect-based sentiment analysis. Integrating such techniques allowed for in-depth listing review analysis, providing nuanced insights into guest preferences and satisfaction. Our findings demonstrated that AirBnB hosts can effectively utilize both state-of-the-art and traditional machine learning algorithms to better understand customer needs and preferences, more accurately assess their listings' market value, and focus on the importance of dynamic pricing strategies. By adopting this data-driven approach, hosts can achieve a balance between maintaining competitive pricing and ensuring high occupancy rates. This method not only enhances revenue potential but also contributes to improved guest satisfaction and the growing field of data-driven decisions in the sharing economy, specially tailored to the challenges of short-term rentals.

Keywords: AirBnB; price prediction; machine-learning; natural language processing; regression; neural-networks

check for updates

Citation: Di Persio, Luca, and Enis Lalmi. 2024. Maximizing Profitability and Occupancy: An Optimal Pricing Strategy for Airbnb Hosts Using Regression Techniques and Natural Language Processing. *Journal of Risk and Financial Management* 17: 414. https://doi.org/10.3390/jrfm17090414

Academic Editor: Thanasis Stengos

Received: 12 June 2024
Revised: 23 August 2024
Accepted: 28 August 2024
Published: 18 September 2024

1. Introduction

AirBnB, a widely utilized digital platform, facilitates the peer-to-peer leasing of residential spaces, such as homes and apartments, for transient accommodation purposes. Its operational scope encompasses over 220 countries and regions globally, exerting a notable influence on the hospitality sector and local economic structures. This platform presents an alternative to conventional hotel services by enabling private individuals to offer their residences as temporary lodgings for travelers. Recent empirical investigations have elucidated that AirBnB's market presence extends beyond merely supplementing the hospitality industry; it actively modulates hotel revenue dynamics. This interaction is quantifiable through the statistical examination of the correlation coefficient between the volume of AirBnB listings and the revenue generated from hotel room bookings. For example, a specific study Gerdeman (2018) highlighted that, in American urban areas with high AirBnB penetration, hotels experienced a decrement of 1.3% in booking rates and a 1.5% reduction in revenue, attributable to the competitive presence of AirBnB.

Regarding profitability, AirBnB reported its highest Airbnb (2022) revenues and profits in the third quarter of 2022, making it an attractive opportunity for real estate investors and entrepreneurs. In this study, we focused on analyzing AirBnB listing prices in Italy, with a specific emphasis **on Rome's listings**. Our goal was to identify the factors that contribute to the attractiveness of an Airbnb listing and to determine the variables that influence its price. We proposed using a range of machine-learning algorithms, such as support vector regression, XGBoost, neural networks, and natural language processing techniques, including TF-IDF, bag-of-words, and aspect-based sentiment analysis. By leveraging these techniques, we aimed to provide a better and more reliable way to create an optimal pricing strategy for AirBnB hosts, ultimately maximizing their utilization and profitability.

The paper is then structured as follows: We commence with a literature review that exposes our research's objectives and principal findings. Subsequently, we critically examine extant scholarly works pertinent to price indicators, methodologies, and natural language processing (NLP) applications within the hospitality sector. This is followed by a detailed account of our data acquisition and preprocessing procedures, alongside a description of the machine learning (ML) algorithms and NLP techniques implemented for the extraction and comparative analysis of price indicators from our dataset. In the Results section, we present and scrutinize the empirical outcomes of our investigative endeavors, assessing the extent to which our initial objectives were fulfilled. The Discussion section is devoted to a contemplative analysis of the methodological strengths and constraints encountered and the various challenges that emerged throughout the research process. We conclude by encapsulating our key contributions and proposing potential avenues for future scholarly inquiry.

2. Literature Review

The primary objective of the AirBnB service is to facilitate the connection between individuals possessing unutilized lodging assets, such as spare rooms or apartments, and individuals seeking temporary accommodation Mach (2020). The proliferation of AirBnB can be ascribed to its transformative impact on the market and its provision of diversified customer experiences Casamatta et al. (2022). The hospitality industry has long grappled with identifying the distinctive attributes and characteristics that set AirBnB apart from traditional hotel accommodation. Numerous studies have demonstrated that pricing is pivotal in the decision-making process for short-term rentals Xin and Xue (2022). Nonetheless, there remains a shortage of research exploring pricing determinants within sharing-economy-based lodging.

In econometrics and predictive analytics, traditional hotel pricing metrics such as star ratings and corporate affiliation, as delineated in Chattopadhyay and Mitra (2020), are not directly translatable to the pricing structure of AirBnB listings. Prior research has endeavored to trace the primary variables influencing AirBnB pricing strategies. The study by Wang and Nicolau (2017) categorized these variables into five distinct groups: host characteristics, geographical location, property-specific attributes, available amenities and services, rental regulations, and the volume and sentiment of online reviews.

An ordinary least squares (OLS) regression was employed in statistical analysis to establish a foundational understanding of these variables' impact on pricing. The study used three advanced machine learning algorithms, gradient boosting, support vector regression (SVR), and neural networks, to enhance predictive accuracy. The performance of these models was quantitatively assessed using metrics such as mean absolute error (MAE) and the coefficient of determination (R^2). The gradient boosting model exhibited an MAE of 0.24 and an R^2 of approximately 0.71, the SVR model an MAE of 0.21 and an R^2 of roughly 0.77, and the neural networks model an MAE of 0.26 and an R^2 of approximately 0.72. Furthermore, the study "AirBnB Price Prediction Using Machine Learning and Sentiment Analysis" by Rezazadeh Kalehbasti et al. (2019) also contributed to this field by applying diverse machine learning methodologies to predict AirBnB listing prices, underscoring the multifaceted approach required in this area of research. Recent research on optimizing AirBnB has yielded significant insights, as evidenced by several publications on the topic. In "AirBnB dynamic pricing using Machine Learning" Wang (2024), various algorithms, including linear regression, random forests, K-nearest neighbors, AdaBoost classifier, and naive Bayes were applied to New York's open dataset. The study revealed that the random forest models were most effective, achieving an R^2 of 0.997 and a root mean square error (RMSE) of 0.038. Naive Bayes classifiers showed promise, albeit with a different RMSE. The potential of boosting algorithms was further explored in "Forecasting Airbnb through Machine Learning" Tang et al. (2023), where a CatBoostRegressor algorithm outperformed other models by incorporating factors such as maximum guest capacity, number of bedrooms, and room privacy status. A more comprehensive approach was taken in "Predicting AirBnB pricing: a comparative analysis of artificial intelligence and traditional approaches" Camatti et al. (2024). This research expanded beyond individual regions, incorporating the financial histories of multiple rental offerings. The findings underscored the continued value of traditional methods in identifying significant predictors, while highlighting artificial intelligence techniques' robust predictive capabilities.

Moreover, the pursuit of discovering the optimal pricing strategy Toader et al. (2022) has captivated the attention of numerous online services, exemplified by platforms like alltherooms.com, accessible at https://www.alltherooms.com/ in 24 March 2023, and airbtics.com accessible at https://airbtics.com/AirBnB-income-calculator/ in 21 June 2023. However, these platforms predominantly focus on a limited subset of factors, while pricing additional services.

Regrettably, these websites fail to offer users a comprehensive alternative for refining their pricing tactics or enhancing occupancy rates. To illustrate this, airbtics.com exclusively accounts for location and the number of bedrooms and bathrooms. In contrast, alltherooms.com incorporates a property score, a metric analyzing bookings, earnings, reviews, and competitive intelligence, as a paid service. Although these platforms can provide insights into the short-term rental market, they impose supplementary charges, which may deter new hosts. Moreover, many similar websites primarily concentrate on the United States (US) market, as evident in the case of airbtics.com, which exclusively serves the US region.

It is worth noting that a property's pricing is influenced not only by its location, size, and bathroom count, but also by various other critical factors, including amenities, minimum and maximum stay durations, and even the host's interactions with guests Bobrovskaya and Polbin (2022). We must delve deeper and gain insights into these additional influential elements to advance in this field.

This study aimed to build upon prior research, offering a comprehensive and data-driven approach to assist Airbnb hosts in optimizing their pricing strategies. Our objectives were multifaceted: to develop and evaluate machine learning models for accurate price prediction, incorporating factors such as location, property attributes, and market conditions; to implement NLP techniques for extracting valuable insights from guest reviews; to identify and rank significant price indicators for Airbnb listings; to understand the balance between revenue maximization and high occupancy rates; and to evaluate the effectiveness of our approach against traditional pricing methods. Through these objectives, we sought to provide Airbnb hosts with a powerful, data-driven tool that enhances their understanding of market dynamics, enables informed decision-making, and ultimately improves their competitiveness in the short-term rental market.

3. Methodology

3.1. Dataset

This paper analyzed the public Airbnb dataset for Rome, Italy, obtained from Inside Airbnb. The dataset used in this study was last scraped in May 2023, providing a recent snapshot of the short-term rental market in the city. While several Airbnb datasets for other cities are available, we specifically chose Rome due to the significant variance between datasets from different locations. These variations include differences in the periods covered, sampling frequency, and the unique characteristics of each city. For instance, some cities are more tourism-focused, while others cater more to business travelers or have seasonal fluctuations. This level of heterogeneity could complicate a comparative analysis, so focusing on Rome allowed for a more controlled and consistent study. Initially, this dataset comprised approximately 75 columns. Then, it underwent a meticulous evaluation and preprocessing to enhance its predictive capability. Our data preparation process involved several key steps: (i) removing listings with improper or incomplete information, (ii) eliminating records with missing values, (iii) converting categorical features into one-hot vectors for machine learning compatibility, and (iv) removing irrelevant and uninformative features. Specifically, columns such as hostName, listingUrl, and scrapeId were eliminated, while the remaining features underwent rigorous assessment using various feature-selection techniques. Further, we normalized the dataset and removed anomalies to ensure data quality. The refined dataset was split into training and test sets to facilitate model development and evaluation. This comprehensive data preparation process ensured the reliability and relevance of our analysis, contributing to the reproducibility and credibility of our research findings in the context of Airbnb pricing optimization for Rome's short-term rental market. Further, the dataset was normalized, and anomalies were removed. Then, the data were split into two sets, namely the train set and the test set. Let us underline that the latter method of exploiting data is not *written in stone*, meaning that different approaches are possible. Indeed, one could opt for, e.g., cross-validation, stratified sampling, time-series splitting, etc. The best method for dividing data depends on the specifics of the particular problem we have to deal with. We experimented with different approaches, concluding that a train/test split offered better performance. Moreover, having conducted other attempts with various percentages of train/test, we concluded that the best option was the one realized with 80% of the dataset as train and the remaining 20% as a test.

3.2. Feature Selection

Forward feature selection (FFS) significantly outperformed principal component analysis (PCA), leading to a 27 percent improvement in the R^2 metric.

Forward feature selection works by incrementally adding features that improve model performance, using the ANOVA F-statistic to identify features with the strongest relationship to the target variable. In our case, this method selected the top 10 features directly relevant to the prediction task. Principal component analysis, by contrast, is a dimensionality reduction technique that transforms the original features into a set of linearly uncorrelated principal components. We reduced the data to four principal components designed to capture the maximum variance in the dataset.

Despite PCA's strength in reducing dimensionality and minimizing the risk of overfitting, FFS proved more effective for our purposes. When applied to models such as XGBoost and SVR, the features selected by FFS resulted in a lower MAE than those derived from PCA, underscoring FFS's superiority in preserving features with direct predictive relevance.

In addition to curating the most productive features (Figure 1), we introduced attributes with significant predictive power. These included activeDaysHost, which measures the duration of host activity; amenitiesTotal, which counts the total number of amenities in a listing; and verificationsTotal, representing the number of verifications completed by the host. Given the high dimensionality of our dataset, we were careful to avoid the "curse of dimensionality" and the risk of overfitting, ensuring that our model remained robust and generalizable. The curse of dimensionality is a pressing concern when dealing with high-dimensional data, as it renders the data more sparse and considerably more challenging to manipulate. As previously expounded, the optimal approach is comprehending and prioritizing the most salient dimensions. Furthermore, overfitting looms when the model becomes excessively attuned to idiosyncratic patterns in the training data, compromising its generalization capacity on new, unseen data. We must, however, caution that both PCA and forward selection necessitate careful management, as they are inclined to assign undue importance to features with high variance, even when these may have limited relevance to the predictive target.

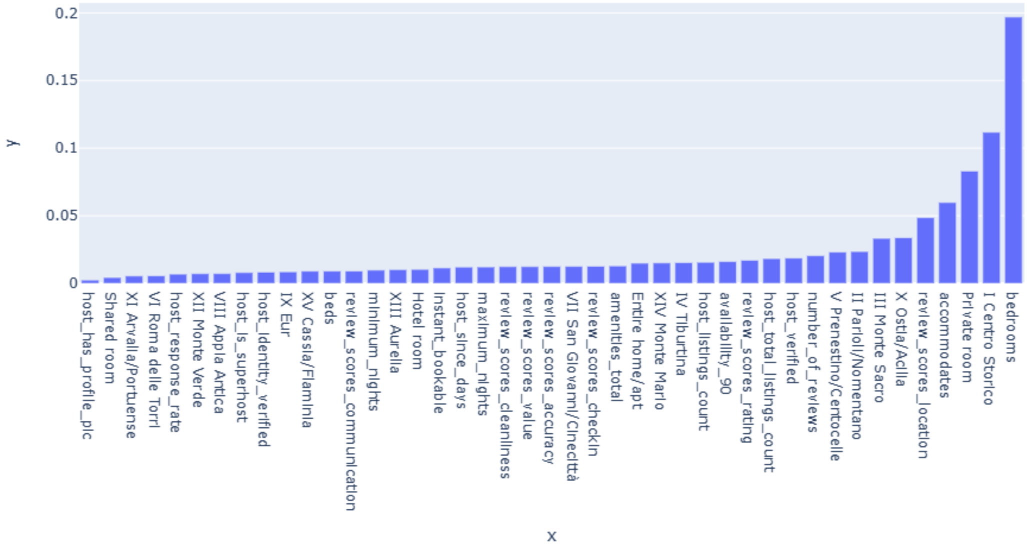

Figure 1. Graph derived from the most important features in the used models.

3.3. Methods

This study employed a multi-faceted approach to predict Airbnb listing prices, leveraging three machine learning methodologies: XGBoost, neural networks, and support vector regression (Figure 2). These methods were chosen for their complementary strengths in handling the complexities of Airbnb pricing data, including structured and unstructured elements. To enhance the predictive power of our models, we integrated natural language processing (NLP) techniques to extract meaningful features from textual reviews. This approach allowed us to capture both quantitative and qualitative factors influencing price determination. The following table (Table 1) concisely compares these methodologies, highlighting their strengths, applications, and roles within our study. By employing this diverse set of techniques, we aimed to comprehensively analyze Airbnb pricing dynamics, while balancing model diversity with computational efficiency.

Table 1. Comparison of methodologies for Airbnb pricing prediction.

Aspect	**XGBoost (XGB)**	**Neural Networks (NN)**	**Support Vector Regression (SVR)**
Strengths	• Excels with structured data • Captures non-linear relationships • Effective for complex feature interactions	• Captures complex, non-linear relationships • Models intricate patterns in pricing	• Handles high-dimensional spaces • Robust to overfitting • Useful for non-linearly separable data
Application in Study	Modeling complex interactions among features (location, amenities, reviews)	Modeling intricate patterns in Airbnb pricing	Baseline model for comparison
Performance	Better than SVR	Outperformed other methods	Used as a baseline
Role in Comparison	Represents ensemble learning approach	Represents deep learning approach	Represents traditional machine learning approach
NLP Integration			• Used TF-IDF and bag-of-words for feature extraction from reviews • Incorporated ABSA sentiment scores as additional features

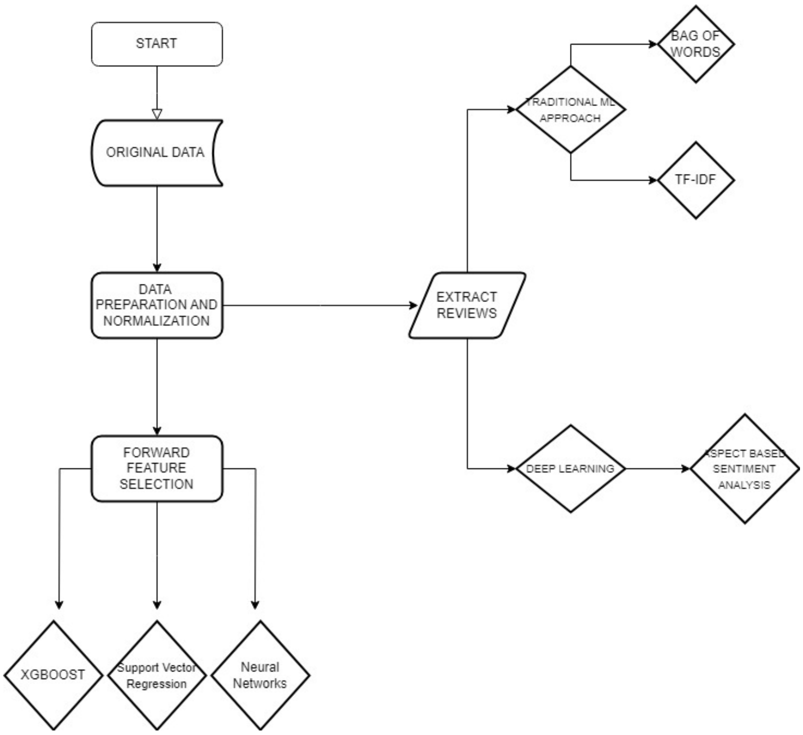
START
ORIGINAL DATA
DATA PREPARATION AND NORMALIZATION
FORWARD FEATURE SELECTION
XGBOOST
Support Vector Regression
Neural Networks
EXTRACT REVIEWS
TRADITIONAL ML APPROACH
BAG OF WORDS
TF-IDF
DEEP LEARNING
ASPECT BASED SENTIMENT ANALYSIS

Figure 2. Workflow chart of development of the research.

3.3.1. Ensemble Algorithms: XGBoost Regressor

Extreme gradient boosting, commonly called XGBoost, constitutes an open-source framework that endows us with a proficient and productive implementation of the gradient boosting technique. For a comprehensive understanding, let us reiterate that XGBoost is grounded in a tree-based ensemble approach to anticipating the target variable. This process entails amalgamating forecasts from numerous weak models, systematically incorporating those models that are especially adept at addressing challenging prediction scenarios. This incremental refinement bolsters the overall predictive accuracy of the model. The final prognostication of the model emerges from the summation of the individual model forecasts encompassed within the assembled ensemble. An array of hyperparameters, encompassing considerations such as the number of trees, the learning rate, and the maximum depth of each tree, oversee the learning dynamics. These hyperparameters are meticulously fine-tuned through the crucible of cross-validation to discern the values that yield peak performance. To forestall the perils of overfitting and augment the model's generalization capability, XGBoost incorporates a regularization term within the objective function. This objective function is articulated as follows:

$$Obj(\Theta) = L(\Theta) + \Omega(\Theta) \tag{1}$$

where Θ represents the model parameters, $L(\Theta)$ is the loss function that measures the difference between the predicted and true values, and $\Omega(\Theta)$ is the regularization term that penalizes complex models. The regularization term is defined as

$$\Omega(\Theta) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{2}$$

where T is the number of trees, w_j is the weight assigned to the j-th tree, γ controls the balance between the complexity and accuracy of the model, and λ holds the degree of regularization. In particular, XGBoost works according to the following steps:

1. Initialize the model with a constant value.
2. For each tree in the ensemble perform the following:

   (a) Compute the gradient and Hessian of the loss function concerning the current predictions.
   (b) Train a new tree to fit the negative gradient of the loss function using the gradient and Hessian as weights.
   (c) Compute the optimal weight for the new tree using a line search algorithm.
   (d) Update the model by adding the new tree with the optimal weight.

3. Repeat steps 2–3 until the desired number of trees is reached.

XGBoost has gained popularity across various machine learning competitions, due to its efficacy and direct applicability to regression predictive modeling. We decided to assess its performance in the context of the AirBnB Scenario. Our analysis focused on a particular instantiation of the XGBoost methodology, specifically the XGBoost regressor (XGBR), which is primarily employed for regression tasks. It is worth emphasizing that XGBR constitutes a machine learning model harnessing the XGBoost algorithm to forecast continuous numerical values based on input features. More precisely, XGBR can be viewed as an ensemble-based algorithm that amalgamates multiple decision trees to construct a more precise and resilient predictive model. It can be mathematically expressed concerning the input features represented by X, the target variable denoted as y, and the prediction model encapsulated within F.

Then, XGBR aims to find a prediction model F that minimizes the mean squared error (MSE) of the predictions, defined as

$$\text{MSE}(y, F(\text{X})) = \frac{1}{n}\sum_{i=1}^{n}(y_i - F(\text{X}_i))^2 \tag{3}$$

where n is the number of training samples.

The prediction model F is defined as the weighted sum of K decision trees $f_k(x)$:

$$F(x) = \sum_{k=1}^{K} f_k(x) \tag{4}$$

where x is an input feature vector.

Each decision tree $f_k(x)$ is a function of the input features X. It is learned sequentially by minimizing a loss function L that measures the difference between predicted and true values. The loss function L is defined as

$$L(y, \hat{y}) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{5}$$

where l is the individual loss function that measures the difference between the predicted and actual values for a single data point, $\hat{y}_i$ is the expected value for data point i, $\Omega(f_k)$ is a regularization term that penalizes complex models, and K is the number of trees in the ensemble.

The individual loss function l can be chosen based on the type of regression problem being solved. For example, for a mean squared error (MSE) regression problem, the individual loss function is defined as

$$l(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2 \tag{6}$$

The regularization term $\Omega(f_k)$ is typically defined as the sum of the leaf weights in the decision tree f_k:

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{7}$$

where T is the number of leaves in the tree, w_j is the weight of the jth leaf, γ and λ are regularization parameters that control the complexity of the model. Accordingly, XGBR optimizes the loss function L by iteratively adding decision trees to the ensemble. In particular, at each iteration, the algorithm calculates the gradients and the Hessians of the loss function concerning the predictions and uses them to train a new decision tree. Then, XGBR updates the weights of the existing trees in the ensemble to improve the accuracy of the predictions.

To get the most out of XGBRegressor, we used the following parameters: *colsamplebytree = 0.75, gamma = 1, learningrate = 0.05, maxdepth = 8, nestimators = 200, subsample = 0.75*. These values were chosen as the optimal result obtained by performing a grid search cross-validation. In particular, we tried a range of different hyperparameters and to give us the best-performing ones:

- colsamplebytree [0.5, 0.75, 1] expresses the subsample ratio of columns when constructing each tree;
- gamma [0, 0.1, 0.5, 1] gives the minimum loss reduction required to make another portion on a leaf node of the tree. As gamma increases, the algorithm becomes more conservative.
- learning rate [0.01, 0.05, 0.1] helps prevent overfitting as the step size shrinks. Maximum depth shows the maximum depth of a tree. If this value is increased, the model becomes more complex and prone to overfitting.
- sub-sample [0.5, 0.75, 1] ratio gives information about the ratio of the training instances. Having a sub-sample ratio of 0.75 means that XGBoost would randomly sample 75 percent of the training data before growing trees.

Lastly, the grid search cross-validation for this task took the following values:

```
grid_search = GridSearchCV(
    xgb_model,
    param_grid=param_grid,
    cv=10,
    n_jobs=-1,
    verbose=1
)
```

3.3.2. Support Vector Regression

Support vector regression (SVR) is a regression algorithm essentially based on the support vector machines (SVM) approach to solving regression problems, where the goal is to find a hyperplane:

$$y = \omega^T x + b \tag{8}$$

where x is the input vector, y is the output value, ω is the weight vector, and b is the bias term, maximizing the margin between the training data and the hyperplane, while minimizing the regression error.

SVR achieves this by taking the mean of a prescribed loss function that penalizes errors in the regression, e.g., with reference to the epsilon-insensitive type:

$$L_\epsilon(y, \hat{y}) = \begin{cases} 0, & \text{if } |y - \hat{y}| \leq \epsilon \\ |y - \hat{y}| - \epsilon, & \text{otherwise} \end{cases} \tag{9}$$

y is the true output value, $\hat{y}$ is the predicted output value, and ϵ is a hyperparameter determining the hyperplane's margin size. Then, SVR minimizes the sum of the loss function and uses a regularization term to prevent overfitting. Accordingly, the optimization problem for SVR is given by

$$\min_{\omega, b, \xi, \xi^*} \frac{1}{2}\omega^T \omega + C \sum_{i=1}^{n} (\xi_i + \xi_i) \tag{10}$$

subject to

$$y_i - \omega^T x_i - b \leq \epsilon + \xi_i \tag{11}$$

$$\omega^T x_i + b - y_i \leq \epsilon + \xi_i^* \tag{12}$$

$$\xi_i, \xi_i^* \geq 0 \tag{13}$$

where C is a hyperparameter that controls the trade-off between maximizing the margin and minimizing the regression error, ξ_i and ξ_i^* are slack variables that allow for some training data points to violate the margin, and n is the number of training data points.

The optimization problem can be solved using various techniques, such as quadratic programming or gradient descent. Once the optimal values of ω and b have been found, they can be used to predict new input data by applying the equation for the hyperplane.

In contrast to the general mindset towards SVR, it did not perform better in our scenario, even after hyper-tuning with the following parameters that were chosen as best from this range by performing a grid search cross-validation:

- 'kernel': ('linear', 'rbf','poly')
- 'C': [1.5, 10]
- 'gamma': [1e-7, 1e-4]
- 'epsilon': [0.1,0.2,0.5,0.3]

The parameters *'C': 10, 'epsilon': 0.1, 'gamma': 1e-07, 'kernel': 'linear*, where C shows the L2-regularization parameter, *epsilon* is the epsilon-SVR model, which specifies the epsilon-tube, and *kernel* indicates the type of kernel implemented, performed best, but still not well enough to beat the other methods.

3.3.3. Neural Networks

Machine learning (ML) has demonstrated its supremacy over classical statistical methodologies by utilizing computationally intensive models, as demonstrated in the preceding sections. Nevertheless, recent years have witnessed a burgeoning trend towards integrating deep learning (DL) as a substitute for conventional ML techniques. Statistical methodologies have exhibited commendable performance across diverse scenarios in numerous instances, particularly within straightforward regression analyses. However, the convenience of implementing DL methodologies involving neural networks (NN) via widely adopted frameworks such as TensorFlow or PyTorch has yielded substantial performance enhancements, yielding noteworthy managerial utility.

In business analytics, marketing, and economics, there has been a growing interest in understanding the potential impact of data science and ML. Our hypothesis suggests that substituting conventional ML techniques with deep learning (DL) methods can significantly influence decision-making for Airbnb hosts, enabling enhanced data-driven predictions that surpass alternative strategies. Our study's results illustrate that DL maintains superiority over basic ML models even for tasks previously deemed straightforward. Notably, the deployment of NNs, even in concise architectures, yields substantial performance improvements. It is acknowledged that a trade-off exists between algorithmic complexity and computational expense. Nevertheless, given the accessibility of open-source libraries and GPUs on platforms like Google Colab and Kaggle, we contend that NNs represent the optimal choice for a wide array of tasks.

However, it is imperative to acknowledge that NNs may not always be the most suitable choice, mainly when the problem is relatively elementary. This limitation arises due to the inherent complexity of the model, which can result in overfitting. Overfitting emerges when the model becomes excessively tailored to the training data, leading to poor generalization to new data and consequently causing suboptimal performance on the test dataset.

$$\text{Overfitting} = \text{Bias}^2 + \text{Variance}$$

The prevailing mathematical literature exhibits notable skepticism concerning the utilization of deep learning (DL) for analogous computational tasks. Numerous research investigations have emphasized apprehensions regarding the problem of overfitting due to constraints posed by limited dataset availability and the inherent complexity of the model. Nevertheless, our research endeavors reaped substantial benefits from an expansive dataset encompassing diverse variables. This comprehensive dataset led to the NN delivering performance far exceeding initial expectations. The overfitting challenges were effectively mitigated through the refinement of regularization rates, adjustment of epoch counts, and optimizing learning rates. Consequently, the NN model demonstrated remarkable efficacy, surpassing alternative methodologies and emerging as a potent mathematical instrument for modeling intricate and non-linear relationships. Owing to its feedforward capabilities, the NN proficiently captured the complex interconnections among various features characterizing an AirBnB listing and its associated pricing. In this light, our analysis illuminated the most influential factors impacting pricing, thereby furnishing invaluable mathematical insights for AirBnB proprietors to maximize their revenue. The implementation phase commenced with a keen consideration of the pervasive vanishing gradient and exploding issues. These phenomena can severely impede the training process, rendering weight adjustments either excessively minuscule or excessively substantial to yield effectiveness. Specifically, for elaborately nested NN structures, the loss function about the model parameters could approach vanishingly small values during the back-propagation process, resulting in sluggish or non-existent convergence.

Specifically, one has to consider that we are dealing with a general NN with weights θ, input data x, and output y, and we define a loss function $L(\theta, x, y)$ that measures the difference between the predicted output and the actual output.

The goal of training the neural network is to find the values of the weights that minimize the loss function, i.e., $\theta^* = \text{argmin}_\theta L(\theta, x, y)$.

During back-propagation, the gradients of the loss function concerning the weights are computed using the chain rule:

$$\frac{\partial L}{\partial \theta} = \frac{\partial L}{\partial a_n} \frac{\partial a_n}{\partial a_{n-1}} \cdots \frac{\partial a_2}{\partial a_1} \frac{\partial a_1}{\partial \theta} \tag{14}$$

where a_i denotes the activation of the i-th layer.

The problem arises when the gradients $\frac{\partial L}{\partial a_i}$ become very small as they are propagated backwards through the layers, leading to small values of $\frac{\partial L}{\partial \theta}$. This can happen when the network weights are initialized in a way that causes the activation to saturate, meaning that they are pushed towards the extremes of their range and are no longer sensitive to small changes in the input. Consequently, our NN model becomes challenging to train, since the gradients become too small to cause significant weight changes. It is worth noting that it is not always possible to overcome this problem using different weight initialization strategies or different activation functions to avoid saturation. At the same time, it could be reasonable to use gradient clipping to limit the size of the gradients during the training phase itself.

In our quest to mitigate overfitting, we deployed many strategies within our implementation. We harnessed the power of the dropout rate, regularization parameters, epoch reduction, and a judiciously tuned regularized learning rate. These strategic interventions acted as mathematical constraints to foster a tendency for generalization. The dropout rate, thoughtfully set at 0.4, engendered stochastic activations among neurons during training, enhancing the model's resilience and preventing the undue dominance of specific neurons. Moreover, our judicious application of regularization introduced a penalty term into the loss function during training, discouraging the model from ascribing disproportionate importance to particular features. This twofold mathematical tuning facilitated a snug fit to the training data and mitigated the perilous over-emphasis on capricious features, enhancing the generalization prowess when confronted with unseen data. Leveraging the TensorFlow callback functionality, we embarked on an exhaustive exploration, meticulously manipulating the epochs and learning rates to discern the optimal configuration for our model. By systematically traversing the parameter space, we quantified their mathematical influence on the model's performance, facilitating a data-driven approach to fine-tuning.

Through a series of iterative adjustments to the number of epochs, our objective was to find an equilibrium point between achieving model convergence and mitigating the peril of overfitting. We conducted a diligent examination of the training and validation metrics, with a focus on ensuring the model exhibited continuous enhancement, without reaching a plateau or manifesting indications of overfitting.

Concurrently, we engaged in fine-tuning the learning rate, a pivotal hyperparameter governing the gradient descent optimization's step size. We aimed to pinpoint the optimal learning rate to facilitate ideal convergence and model stability. We rigorously monitored both the training progress and validation performance, allowing us to discern the learning rate that produced superior results in terms of accuracy, convergence speed, and generalization capacity. Consequently, our efforts led to promising outcomes in neural networks (NNs).

In our implementation, we employed a neural network architecture composed of two fully connected dense layers, each containing 64 neurons, followed by a final layer with a single neuron tailored to our regression task (Figure 3). This architecture was chosen after extensive experimentation with various network configurations. Notably, this relatively simple structure outperformed more complex architectures, including those with additional layers or more neurons per layer. The superior performance of this leaner network can be attributed to the moderate complexity of our dataset, which did not require an overly sophisticated model to capture its underlying patterns. Our chosen architecture strikes an optimal balance: it is capable of modeling the non-linear relationships within the data, while avoiding overfitting, a common issue with larger networks on datasets of this scale. We determined an optimal learning rate of 0.078 and applied a dropout rate of 0.4 to enhance generalization further. The model was trained for 50 epochs, which we found provided sufficient iterations for learning without risking overfitting. This configuration allowed the model to effectively capture the nuances of Airbnb pricing patterns, while maintaining robust performance on unseen data. By carefully monitoring the training progress and validation performance, we confirmed that this architecture achieved strong predictive capabilities while preserving excellent generalization, outperforming simpler linear models and more complex neural networks.

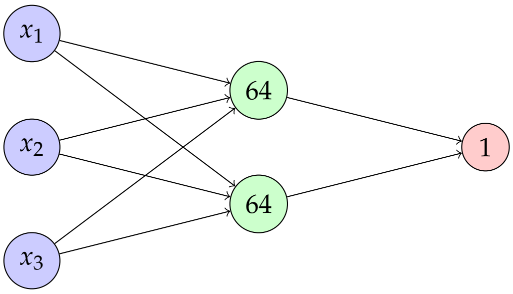

Figure 3. Neural network architecture.

3.3.4. Natural Language Processing

With significant digitized text, customer experience has changed thoroughly among different markets. The digital transformation of diverse industries has increased the quality and volume of data. With the vast amount of big data available for studies, the research community has used different methodologies to explore this field. This has been an excellent opportunity to analyze the importance of a product in the market, analyze its growth, and further develop strategic plans for improving said products and services Kumar et al. (2021). With the Internet's tremendous growth, along with the help of social media networks, where users can freely express their opinions, the need to find an effective way to extract the essentials of texts is mandatory.

The first approaches in the field started with qualitative approaches Mayring (2015) or even hard-coding. However, these processes tend to be time-consuming, especially in today's world of big data. Further, with the increase in user numbers, the reliability of the Internet's sources, and more and more data to be analyzed, better approaches should be pursued.

In recent years, sentiment analysis has garnered significant attention within the deep learning community, revolutionizing the approach to handling complex textual data. Unlike traditional rule-based or machine learning approaches, deep learning methods have demonstrated remarkable capabilities in automatically learning features from large-scale datasets, substantially enhancing the accuracy and efficiency of sentiment classification and analysis Wu et al. (2024). Adopting deep learning for sentiment analysis offers considerable advantages, particularly in terms of precision and performance. Neural network models excel at identifying intricate patterns and relationships in data, often surpassing the accuracy of traditional machine learning methods. DL utilizes a layered approach in its neural network structure, particularly in the hidden layers. Unlike conventional ML methods, where feature definition and extractions are performed manually or rely on feature selection techniques, DL automatically learns and extracts features Dang et al. (2020). Furthermore, these models exhibit versatility across various domains and languages, effectively adapting to different data types and scales. This is particularly valuable for analyzing sentiment on platforms like social media Islam et al. (2024). The layered approach in deep learning neural network structures, especially in the hidden layers, sets them apart from conventional machine learning methods. While traditional approaches rely on manual feature definition and extraction or feature selection techniques, deep learning algorithms automatically learn and extract relevant features Dang et al. (2020). This capability has led to a surge in research exploring various deep learning architectures for sentiment analysis. The growing body of literature in this field underscores the potential of deep learning in advancing sentiment analysis techniques and their applications across diverse domains.

3.3.5. NLP in AirBnB's Context

Understanding human sentiment has been one of the most important fields of DL, as it can significantly impact the analysis of reviews and facilitate closer customer engagement. In today's era of abundant data and vast collections of user opinions, gathering comprehensive information has become essential to adapt businesses to meet the evolving needs of their customers. By gathering valuable insights, decision-makers can improve their efficiency and productivity, allowing them to provide a personalized experience to their customers, further optimizing revenues and sales process. In the context of AirBnB, hosts can leverage the reviews of their guests, enabling them to make informed changes and at the same time drive innovation. One of the advantages that AirBnB offers to the parties it serves is the ability to express their opinion freely. Such a feature then allows the hosts to learn from their previous customers, but at the same time, new customers have a way of evaluating how to spend their money.

Recognizing the sentiments expressed by guests towards an AirBnB property holds immense value in driving further enhancements, both in terms of the listing itself and the host's characteristics. Practical understanding and analysis of user attitudes and opinions have proven instrumental in boosting profits for businesses across various industries.

To achieve this, a range of natural language processing (NLP) techniques are available, encompassing simple statistical methods, traditional machine learning (ML) techniques, and advanced DL approaches. We thoroughly analyzed these techniques to identify the most effective strategy for extracting valuable insights from user reviews.

By delving into NLP, we aimed to decipher user reviews' sentiments and underlying meanings. This deeper understanding will enable us to gain actionable insights and make informed decisions that align with customer preferences and requirements. Ultimately, utilizing appropriate NLP techniques will empower businesses to refine their offerings, enhance customer satisfaction, and drive profitability in an increasingly competitive market landscape.

3.3.6. Bag-of-Words Approach

As mentioned before, the techniques we proposed for the SA problem revolve around traditional ML and DL models. Classic ML methods include basic techniques that emphasize word frequencies and usage in a document.

Text modeling is a messy problem requiring flexible approaches to extract meaning from the input. We initially started with a “bag-of-words” approach to understand the frequency of words. This is a way to extract features from text with a straightforward approach. Here, we used the occurrence of words within the document, the dataset of reviews involving two aspects: (1) a vocabulary of known words, and (2) a measure of the presence of known words. Even though bag-of-words is an easy model to implement and understand, it is still very effective for information retrieval. By representing each review as a set of word counts, we had a powerful tool for text analysis. Then, we could easily visualize these words using a word cloud (Figure 4) , where the more significant the size of the word, the greater its frequency. This approach showed that, for users, it is essential for an apartment to be in a *great location* and for the host to be *always available*.

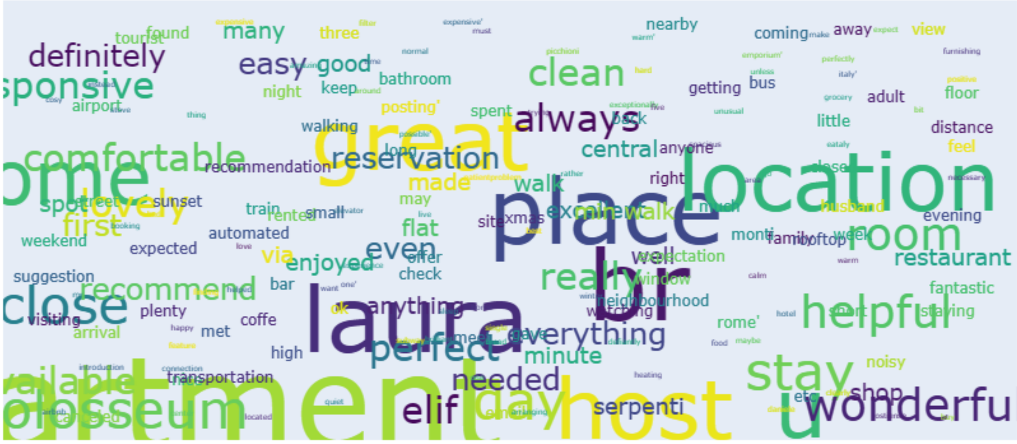

Figure 4. Word cloud for most common words in the reviews. Here, we can see what guests talk about most in their reviews and what is worth mentioning.

In the case of Rome, for instance, it was important for guests to have their property close to the Colosseum.

3.3.7. Term-Frequency/Inverse Document Frequency

One of the problems with bag-of-words is that the highly frequent words very soon start to dominate. However, they may not be words that contain the actual “informational content”. One approach to solve this is to rescale the frequency of the phrases by measuring their frequency in all documents. Further, we also used a stop-word vocabulary, where words such as “the”, “a”, “an”, etc, that may appear frequently but do not add extra meaning to the final review were penalized.

This approach is called term-frequency/inverse document frequency (TF-IDF), where TF stands for the scoring of the frequency of the word in the current document (review), and IDF stands for the scoring on the rarity of the word in all documents (all reviews). Using this approach, we saw that AirBnB guests are usually very generous with their reviews, with the top words being *top*, *perfect*, *excellent*, *good*, *brilliant* (Figure 5), and so on.

These two simple methods showed that they work well to determine what makes an Airbnb property successful. Understanding what guests like most and ensuring that property owners can meet their needs in the future is essential for running a good business.

Looking at guest reviews is also helpful in planning what to invest in next, as it helps us see what parts of the guest experience we should focus on improving.

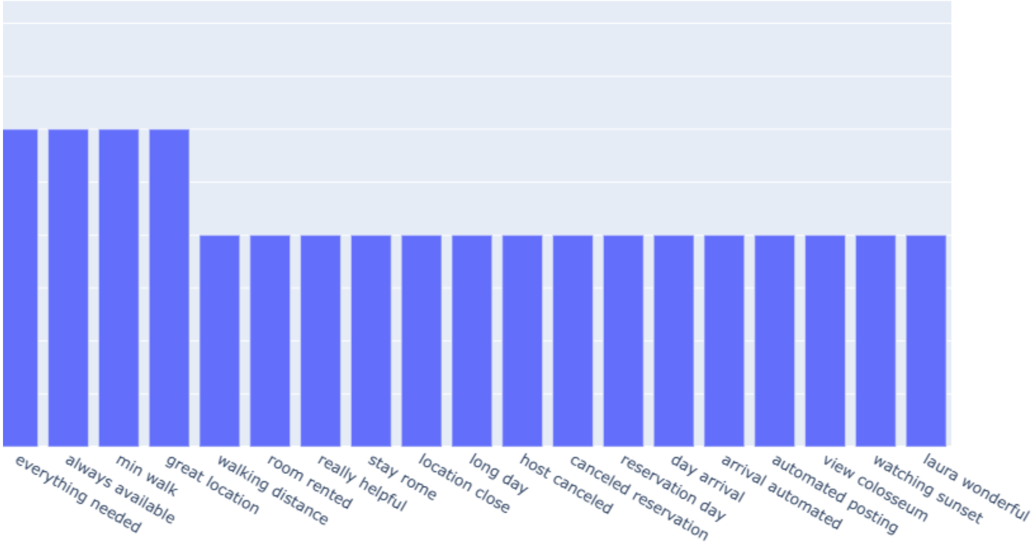

Figure 5. The TF-IDF showed the most common phrases found in the review, from which we can understand how important are the location, the host behavior, and other features like *being close to the Colosseum* or having *an automated check-in*.

Property owners can make better choices that make guests happy and return by listening to what guests say. This approach makes the property more attractive in the short term and helps build a solid and positive reputation over time. Paying attention to feedback and acting on it means the property can keep improving, helping it stand out in the crowded Airbnb market.

3.3.8. Deep Learning in NLP

We know that extracting information accurately remains challenging, as we deal with massive amounts of data. NLP can be divided into different levels: documents, sentences, and aspects. In the previous statistical methods, we used the entire set of reviews to find the most insightful keywords and review-per-review to analyze each review, specifically using bag-of-words. Another helpful tool that we decided to experiment with is aspect-based sentiment analysis. We believed this approach could be beneficial, especially in understanding the target aspect in various reviews. For example, if the AirBnB listing has everything needed, it is spacious, comfortable, and well-located. Nevertheless, the host is arrogant and indifferent, which complicates the check-in process; it can be hard to identify whether this listing is worth staying in. Therefore, aspect-based sentiment analysis (ABSA) was proposed to solve this problem. ABSA is a complex task within the field of sentiment analysis that goes beyond determining the overall sentiment of a text; it aims to identify the sentiment towards specific aspects or features mentioned in the text. With the rise of DL and its breakthroughs in industry and academia, we believed that applying ABSA to our reviews would be pretty efficient.

In recent years, ABSA has captured the research community's attention as a much finer level of analysis, where understanding different aspects and their polarities can give good insights into the most critical targets to focus on. Even though ABSA with deep learning is still in the beginning stages of its evolution, we believed it would be enlightening to see how it performed on the reviews of AirBnB properties. Different research papers Ruder et al. (2016), Mohan and Sunitha (2018), Yadav (2021) saw convolutional neural networks (CNNs) as a good fit that led to good results for different benchmarks of sentiment analysis and ABSA.

To be precise, CNNs are a category of deep learning algorithms primarily used for processing structured grid data such as images. They are highly effective in areas like image recognition, video analysis, natural language processing, and even in recommender systems, where the ability to detect patterns and features is crucial.

CNNs apply a series of filters (convolutions) to the input data to create feature maps highlighting essential characteristics. These filters automatically adjust during training to capture the most relevant patterns. After convolutional layers, pooling layers reduce the dimensionality of these feature maps, summarizing their most important information and making the network more efficient and less sensitive to the exact location of features within the input.

The architecture of a CNN typically starts with several convolutional and pooling layers to process and simplify the input data. This is followed by one or more fully connected layers that analyze the features extracted by the earlier layers to make a final classification or prediction.

The strength of CNNs lies in their ability to learn feature representations directly from the input data, without the need for manual feature extraction, making them highly efficient for tasks that rely on pattern recognition within large datasets. This has led to their widespread use in various applications, including facial recognition systems, autonomous vehicles, and medical image analysis, where they can identify patterns and anomalies within complex visual inputs.

CNNs are particularly suited for ABSA because they can efficiently process and analyze textual data to capture the context and semantic nuances of different aspects of sentences or documents. By applying convolutional layers to text data, CNNs can extract meaningful feature representations for words and phrases associated with various elements. These features can then determine the sentiment (positive, negative, or neutral) expressed towards each aspect.

For example, in AirBnB reviews, a CNN can distinguish sentiments directed towards different attributes of a listing, such as its host, home design, or amenities, by analyzing the text through convolutional and pooling layers. This granular analysis can allow hosts to gain deeper insights into customer opinions and preferences regarding specific features of a product or service.

Using CNNs in ABSA showcases the versatility and power of convolutional neural networks in handling visual data and complex textual analysis tasks, offering precise and detailed sentiment insights that can significantly benefit customer feedback analysis, market research, and personalized recommendation systems.

In this study, we also used a CNN for the task at hand, using a one-dimensional CNN with a kernel size of 3 and a ReLU activation function, followed by a 1D max pooling layer and a dropout layer of 0.2 to reduce overfitting. The model was finalized using one dense layer, a dropout layer, and a final dense layer with different activation functions. Our primary focus was on host characteristics, as we aimed to understand how customers value a host's behavior and communication skills. We decided to focus on HOST-GENERAL to understand the sentiment towards hosts. Each review would be classified into polarities:

- Positive Sentiment: The aspect has a positive sentiment.
- Negative Sentiment: The aspect has a negative sentiment.
- Neutral Sentiment: The aspect is neither positive nor negative.

Our review dataset had 1,048,576 rows, featuring 249,765 reviews where the host was mentioned. We focused on 30,000 reviews, as per the computational resources. Out of these, we extracted only the reviews written in English, approximately 22,649 reviews, of which 85 were automated posting *"The host cancelled this reservation x days before arrivals. This is an automated posting."*, leaving us with 22,564 reviews to analyze, where the host was mentioned 7865 times. Along with the host being mentioned, we found words such as *wonderful, exceptional, brilliant, or friendly*, proven from our ABSA analysis, where around 93 percent of the reviews had a positive sentiment. However, the other part was driven by hosts being *terrible, the worst, or even a nightmare*. We checked all these reviews to understand what would make a property owner receive such bad reviews. From our data exploration and analysis, some hosts failed to explain everything adequately, leaving room for misunderstanding in communications, invading the guests' privacy by entering the rooms without notice, being late for the check-in, falsely advertising their properties, etc. All such behaviors tended to leave a bad impression with guests, which was then generally represented in the reviews.

We believe that by taking into account the previous price prediction analysis, understanding the key features which affect the pricing and occupancy of a property as well as the NLP analysis, followed by ABSA, the proposed strategy would help hosts to improve the AirBnB experience for their guests, as well as to better utilize their properties.

4. Results

To evaluate the trained models and understand which gave the most accurate pricing strategy, we used two methods of measuring error: mean absolute error (MAE) and R^2 (Table 2). We aimed to increase the R^2 value in the test set, while lowering the MAE. In this way, we could create a reliable way for the user to input the features of their property and obtain a good price estimation for it. Further, we aimed to understand each feature's weight on the product price, to give a prospective host the right setting on where and what to invest in. Without a doubt, the best place to buy a property in the case of Rome was *I Centro Storico*, surprisingly followed by *X Ostia/Acilia* and *II Pariolu/Nomentano*. In addition, the number of bedrooms and how many people it can accommodate was significant to users, along with the **reviews**, specifically *the number of reviews* that a property has had, as well as the review score for location. This gave us the insight that when people go to Rome, they mostly care about the location of the property rather than any other factor. Lastly, to ensure that the property is fully booked, guests are more likely to go for *Entire properties* rather than *Shared rooms or homes*.

The following results were a product of different experiments conducted on the AirBnB Dataset for Rome. The outcomes showed that neural networks performed best by learning complex relationships between the dependent and independent variables, and overall, this was a well-suited model for the task at hand. Nevertheless, the ensemble models also tended to improve the performance, as they reduced the variance in predictions, giving us a stable performance. Surprisingly, SVR tended not to perform as well as thought. However, this model still fit well with the observed data.

Table 2. Performance metrics of models.

Model	R^2	MAE
XGB	0.77	0.21
Neural Networks	0.81	0.18
SVR	0.62	0.25

5. Discussion

The obtained results provided insights into the factors influencing a property's price and availability, enabling us to develop a strategy for investing in and maintaining an AirBnB listing. We had to be cautious during our experiments to avoid issues such as overfitting and the curse of dimensionality. Although the available data produced satisfactory results, we believe that incorporating more data would further enhance the accuracy of our findings. Throughout the process, we faced challenges such as missing values, erroneous information, and the need to engineer new features. We overcame these by hyper-tuning parameters and performing feature selection to reduce overfitting.

6. Conclusions

In this paper, we faced the challenge of finding the *best actions* an AirBnB host can perform aiming at maximizing his investment's profits. We carefully considered both sides of the problem, namely, how the property characteristics influence the profits and how hosts' behavior influences the profits. With a diversity of models tested, we provided robust and reliable results in the test set that worked as the baseline for the optimal strategy. We listed the most critical districts guests would like to vacation in; and we saw how the type of property affects its availability, how the number of amenities affects the price, and the influence of hosts' characteristics versus the property characteristics in a review. By conducting such experiments, we understood how important it is for the guests to have a good relationship with the hosts and how important it is that the guests feel heard and respected by their hosts.

In the future, we will examine how the derived optimal strategy compares across other platforms, e.g., Booking.com and Vrbo.com, while also factoring in variables such as social media presence and marketing budget.

Author Contributions: Conceptualization, E.L.; Methodology, E.L.; Software, E.L.; Formal analysis, L.D.P. and E.L.; Data curation, E.L.; Writing—original draft, E.L.; Writing—review; L.D.P.; editing, L.D.P.; Supervision, L.D.P.; Project administration, L.D.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets used are publicly available in https://insideairbnb.com/get-the-data/ (accessed on 19 April 2023). The data has been extracted for the periods specified in Section 3.1.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

NLP	Natural Language Processing
ML	Machine Learning
SVR	Support Vector Regression
XGBoost	extreme gradient boosting
AI	Artificial Intelligence
TF-IDF	Term Frequency—Inverse Document Frequency
PCA	Principal Component Analysis
NN	Neural Networks
MAE	Mean Absolute Error
ABSA	Aspect Based Sentiment Analysis
DL	Deep Learning

References

Airbnb. 2022. Airbnb Q4 2022 and Full Year Financial Results. Available online: https://news.airbnb.com/airbnb-q4-2022-and-full-year-financial-results/ (accessed on 19 April 2023).

Bobrovskaya, Ekaterina, and Andrey Polbin. 2022. Determinants of short-term rental prices in the sharing economy: The case of Airbnb in Moscow. *Applied Econometrics* 65: 5–28. [CrossRef]

Camatti, Nicola, Giacomo di Tollo, Gianni Filograsso, and Sara Ghilardi. 2024. Predicting Airbnb pricing: A comparative analysis of artificial intelligence and traditional approaches. *Computational Management Science* 21: 30. [CrossRef]

Casamatta, Georges, Sauveur Giannoni, Daniel Brunstein, and Johan Jouve. 2022. Host type and pricing on Airbnb: Seasonality and perceived market power. *Tourism Management* 88: 104433. [CrossRef]

Chattopadhyay, Manas, and Subrata Kumar Mitra. 2020. What Airbnb host listings influence peer-to-peer tourist accommodation price? *Journal of Hospitality and Tourism Research* 44: 597–623. [CrossRef]

Dang, Nhan Cach, María N. Moreno-García, and Fernando De la Prieta. 2020. Sentiment analysis based on deep learning: A comparative study. *Electronics* 9: 483. [CrossRef]

Gerdeman, Dina. 2018. The Airbnb Effect: Cheaper Rooms for Travelers, Less Revenue for Hotels. Available online: https://hbswk.hbs.edu/item/the-airbnb-effect-cheaper-rooms-for-travelers-less-revenue-for-hotels (accessed on 19 April 2023).

Islam, Md. Shahinur, Md. Nurul Kabir, Nor Azlina Ghani, Kamal Zuhairi Zamli, Nor Saradatul Akmar Zulkifli, Md. Mustafizur Rahman, and Mohammad Ali Moni. 2024. Challenges and future in deep learning for sentiment analysis: A comprehensive review and a proposed novel hybrid approach. *Artificial Intelligence Review* 57: 62. [CrossRef]

Kumar, Sunil, Arpan Kumar Kar, and P. Vigneswara Ilavarasan. 2021. Applications of text mining in services management: A systematic literature review. *International Journal of Information Management Data Insights* 1: 100008. [CrossRef]

Mach, Łukasz. 2020. Prices of accommodation rental as functioning on the basis of a sharing economy in the capitals of CEE states. *Argumenta Oeconomica* 2020: 141–62. [CrossRef]

Mayring, Philipp. 2015. Qualitative content analysis: Theoretical background and procedures. In *Approaches to Qualitative Research in Mathematics Education*. Dordrecht: Springer, pp. 365–80.

Mohan, Syam, and R. Sunitha. 2018. Survey on aspect based sentiment analysis using machine learning techniques. *International Journal of Computer Sciences and Engineering* 6: 174–81.

Rezazadeh Kalehbasti, Pouya, Liubov Nikolenko, and Hoormazd Rezaei. 2019. Airbnb price prediction using machine learning and sentiment analysis. *arXiv* arXiv:1907.12665.

Ruder, Sebastian, Parsa Ghaffari, and John G. Breslin. 2016. INSIGHT-1 at SemEval-2016 Task 5: Deep learning for multilingual aspect-based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego: Association for Computational Linguistics.

Tang, Jinwen, Jinlin Cheng, and Min Zhang. 2023. Forecasting Airbnb prices through machine learning. *Managerial and Decision Economics* 45: 148–60. [CrossRef]

Toader, Vlad, Ana-Larisa Negrușa, Oana-Ramona Bode, and Roxana-Valentina Rus. 2022. Analysis of price determinants in the case of Airbnb listings. *Economic Research-Ekonomska Istrazivanja* 35: 2493–509. [CrossRef]

Wang, Yuhan. 2024. Airbnb dynamic pricing using machine learning. In *New Perspectives and Paradigms in Applied Economics and Business*. Edited by William Gartner. Cham: Springer Nature Switzerland, pp. 37–51.

Wang, Dan, and Juan Luis Nicolau. 2017. Price determinants of sharing economy based accommodation rental: A study of listings from 33 cities on Airbnb.com. *International Journal of Hospitality Management* 62: 120–31. [CrossRef]

Wu, Yichao, Zhengyu Jin, Chenxi Shi, Penghao Liang, and Tong Zhan. 2024. Research on the application of deep learning-based BERT model in sentiment analysis. *arXiv* arXiv:2403.08217.

Xin, Jin, and Lei Xue. 2022. The Hedonic Price Model of Online Short-Term Rental Market Based on Machine Learning. In ACM International Conference Proceeding Series, pp. 972–6. Available online: https://dl.acm.org/doi/10.1145/3558819.3565227 (accessed on 24 March 2023).

Yadav, Kaustubh. 2021. A comprehensive survey on aspect based sentiment analysis. *International Journal of Advanced Research in Computer Science* 12: 1–9. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.