Stay More Project

Problem and proposed solution overview

The aim of this project is to provide StayMore company a way of maximizing the efficiency of their business decisions. The problem faced by StayMore is that although a significant volume of data is collected, their current structure is difficult to make use of in order to gain a useful insight about the company's state and challenges. The solution that our team proposes is divided in three parts. Firstly, the creation of a data warehouse, for a multidimensional representation of the data, which will allow the fast and easy aggregations and calculations of underlying data sets. Following, the creation of dynamic reports and interactive dashboards, for the convenient extraction of meaningful insight from the data, which will enable more effective management decisions. Finally, the development of a model that describes the relationship between a property and its value, which will facilitate relevant cost decisions.

Analysis of the problems and needs that will be covered

The proposed solution is aspiring to cover certain problems and needs that StayMore is facing. On the first hand, the suggestion of a data warehouse creation will make feasible the easy manipulation of available data with minimum time resources, as it gathers data from all sources, stores it in uniform format and allows the quick sharing of insigths. At the same time, the creation of user-friendly dashboards and reports will make StayMore capable of having an updated (and historical) overview, at any time, about questions on the current company state. Such questions might concern for example the availability time lime of a certain property or the lower rated properties among the top priced ones. Subsequently, the problem of having available the underlying intelligence of the data, at any time requested, for the maximization of decisions' effectiveness is addressed. On the other hand, the suggestion of an attributes-value model development concerning the properties, will allow a better understanding of the attritibes-value relationship and the identification of relevant patterns. The company will be able to suggest renting values to new hosts and adjustments in value to old hosts. This satisfies the need of effective regulations over renting values in order to maximize the profits.

Team organization, tasks identified and project timeline

In this project our goal was to analyze and extract useful information about the Airbnb renting system in Netherlands. Our team consisted of 4 members who are presented below with alphabetical order:

- Dimitris Gkaimanis (Junior Data Scientist)
- Elena Pappa (Junior Data Scientist)
- Iosif Petrakis (Junior Data Scientist)
- Lampros Mitsiogiannis (Junior Data Scientist)

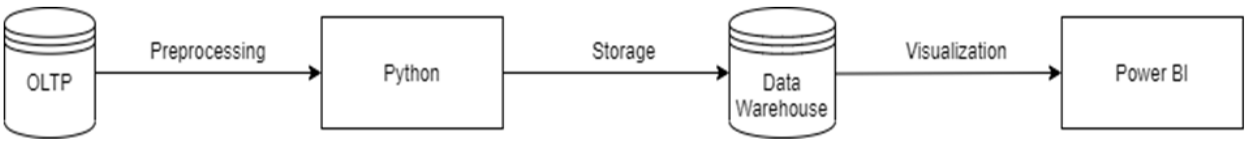
After carefully discussed about the project and created a first overview of how we are going to work, we set the task that each one of us was going to develop. In the beginning Dimitris, Elena and Lampros worked in the preprocessing of the data in order to clean them and be ready for the machine learning process and Iosif worked with SQL in order to built our OLTP database and data warehouse tables and diagrams. Those tasks have been accomplished during the first week of our project (May 25- June 3). After we finished our first tasks, we assigned new tasks where Dimitris and Elena worked the visualization process of our project and Iosif and Lampros continued to develop the Data warehouse (June 3-June 8). During the last 2 days we worked all together toy cerate the final charts and plots in order to be ready for our presentation.

Development Tools

The tool that we used an helped us with our project was:

- SQL Server Management Studio (SSMS) for our OLTP and Data Warehouse development
- Python Programming Languages with all the useful libraries such as pandas, NumPy etc.
- Power BI for our Visualization

BI System overview, architecture and system workflow



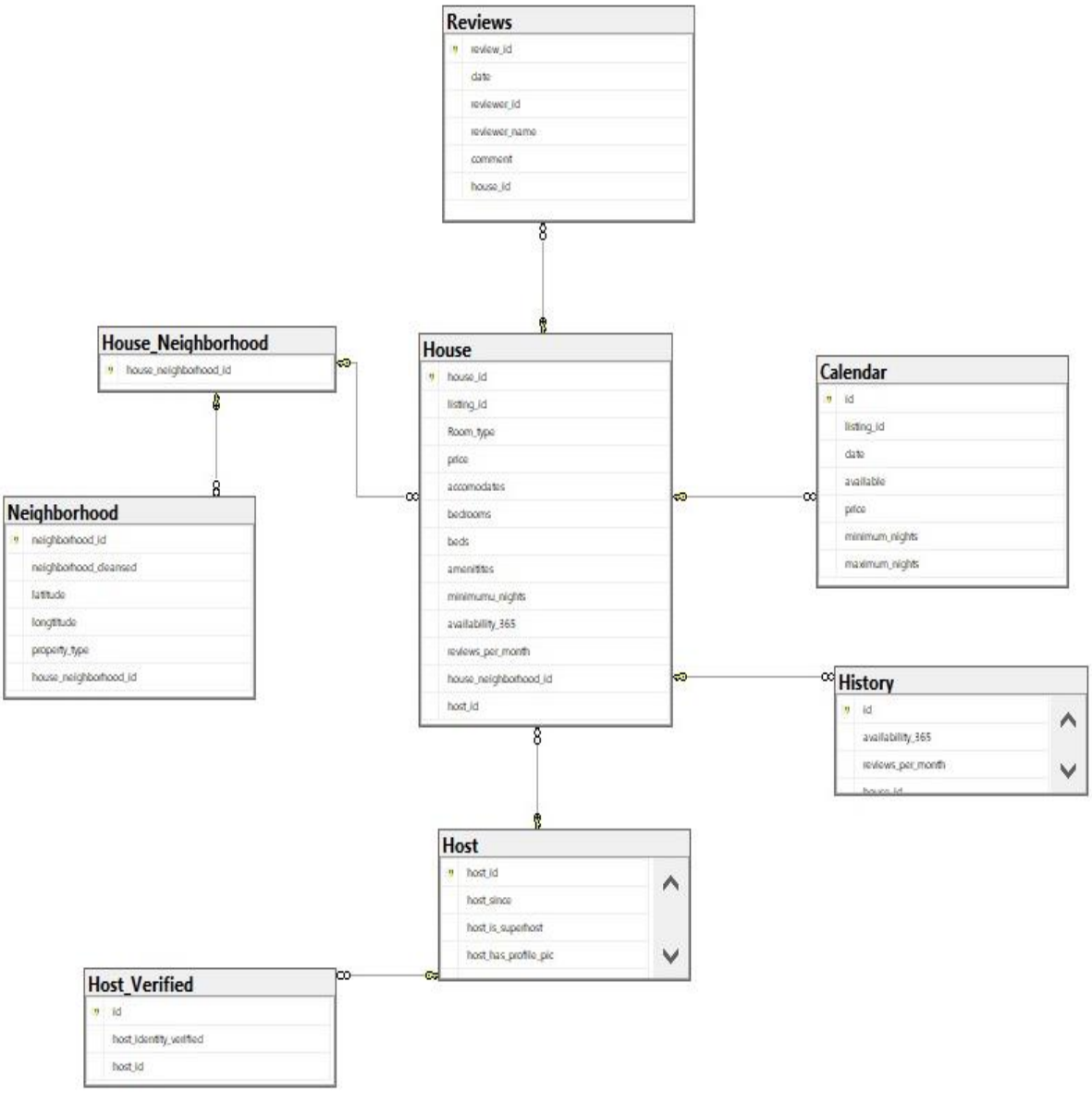
A first look on the workflow process of the project. Initially, we had our data stored in an OLTP database, followed by the preprocessing of the data via Python, in order to end up creating the final tables for our data warehouse. The final step, is the visualization process of the data using the Power BI.

Database Diagram and source code description

Our goal in this step was to take the OLTP given system and construct a Data Warehouse system which would be much more useful for business and operational purposes. We decided not to work with Data Marks, as we did not have a clear vision for the business departments and the specific use of the data for business operations. Given that we wanted to combine this Data Warehouse with the machine learning algorithm, we kept only the necessary features for our analysis.

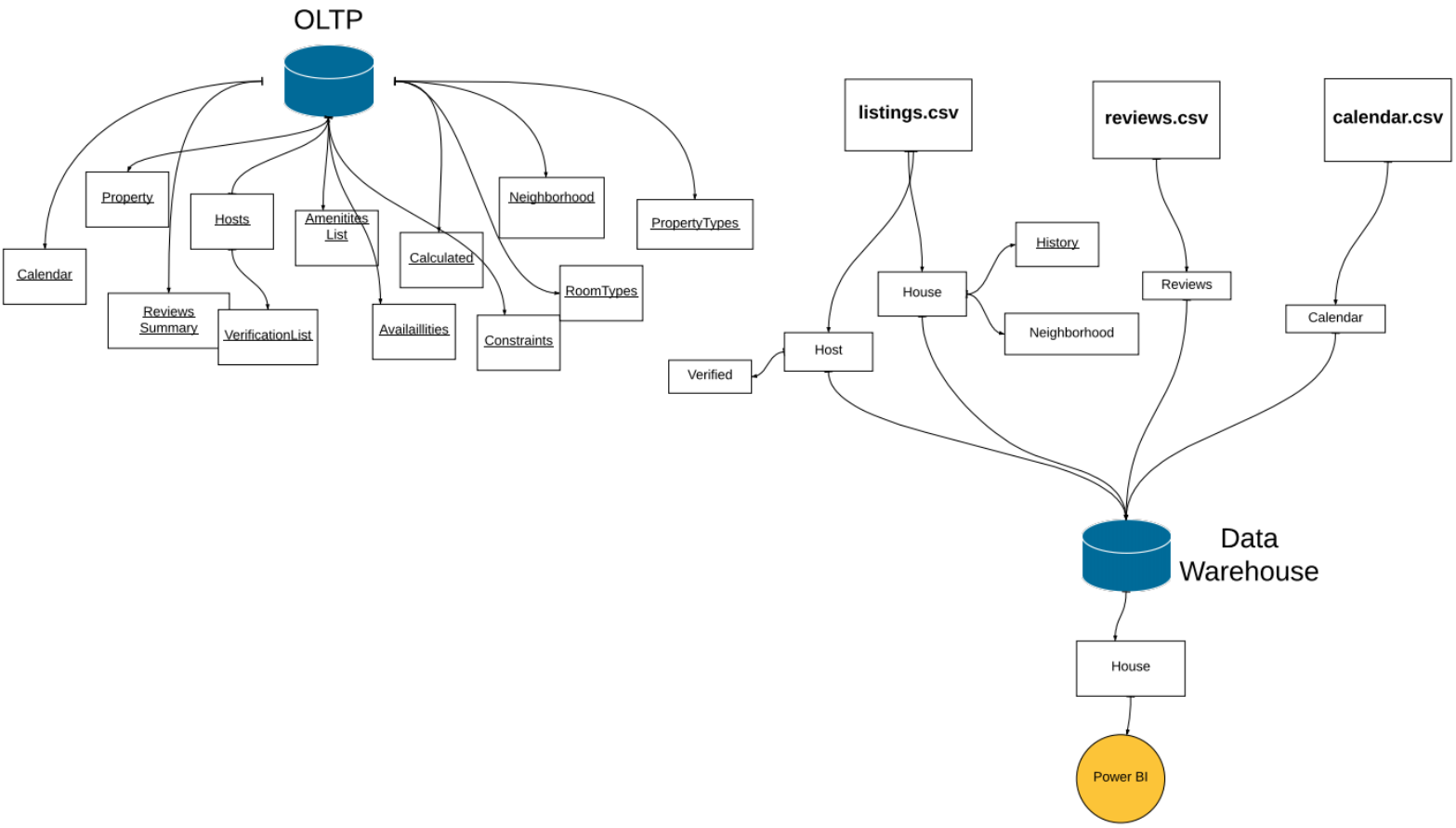
In the diagram below, it is observed a snowflake schema with two fact tables, ‘House’ and ‘Reviews’ and there are 6 more tables that coplete the diagram.

The features of the ‘Reviews’ table are derived from the ‘reviews.csv’ file and the features of the ‘Calendar’ table are coming from the ‘calendar.csv’ file. The features in all the other tables are coming from the ‘listings.csv’ file and are distributed on the ‘House’, ‘Neighbourhood’, ‘Host’, ‘History’, and ‘Host_Verified’ tables. The table ‘House_Neighbourhood’ has been created to help on the communication between ‘House’ and ‘Neighbourhood’.



Dataflow diagram and transformations explanations

The image below represents the data flow diagram. On the left side is presented the system overview of the OLTP system. On the right side we can see the architecture and the workflow of the model that we create, keeping only the necessary for the analysis features, extracted from the 3 csv files. These features represent the tables of the Data Warehouse, while only some features from the 'House' table have been kept for building the predicting models and for visualization purposes.



Code used for data processing description

A machine learning model was developed for the evaluation of the suitable value of a property based on its attributes. The available data were preprocessed and then used on experimentations in training several machine learning models so as to find the best one fitting the data. Specifically, the steps followed were:

- Thorough view of the database. Firstly, it was necessary to remove the features that were not helpful for the problem faced. In this step, most of the variables measured in the database were discarded as they were thought to be not so relevant in predicting the price of a property. The **25** features used were: host_acceptance_rate, host_is_superhost, host_neighbourhood, host_listings_count, host_total_listings_count, host_has_profile_pic, host_identity_verified, latitude, longitude, neighbourhood_cleansed, room_type, accommodates, bathrooms_text, bedrooms, beds, amenities, minimum_nights, maximum_nights, minimum_nights_avg_ntm, maximum_nights_avg_ntm, number_of_reviews, review_scores_rating, instant_bookable, calculated_host_listings_count, reviews_per_month.

However, the experimentation on this 25-features dataset (which followed all the below mentioned steps) did not give adequate performance. As a result, the dataset was reexamined and a further extraction of features lead to the final dataset used which consists of **15** features: host_is_superhost, host_has_profile_pic, host_identity_verified, latitude, longitude, room_type, accommodates, bedrooms, beds, minimum_nights, minimum_minimum_nights, instant_bookable, reviews_per_month, amenities_number, price, neighborhood_cleansed.

- Preprocessing of the data. This step included handling of missing values, transformation of certain variables, encoding categorical features into numerical, removing outliers and scaling.
    - All rows containing a missing value in the features 'host_is_superhost', 'host_has_profile', 'host_identity_verified', 'bedrooms' and 'beds' were dropped. In the features 'bedrooms' and 'reviews_per_month', missing values where filled with the median value of the corresponding feature. In the feature 'beds' the missing values were filled with the value of the variable 'accommdates' in the same row.
    - The feature 'amenities' which included a list of the available amenities of the property was replaced by 'amenities_number' which represents the number of amenities listed (offered by the property).
    - One hot encoding was used for the categorical features 'host_has_profile_pic', 'host_identity_verified', 'room_type', 'instant_bookable' and 'neighborhood_cleansed'.
    - Values that were extreme and rare were considered outliers and so removed from the final dataset. The features with outliers were 'beds', 'bedrooms' and the target variable 'price'. Outliers in the target variable seemed to significantly decrease the model's performance.
    - Features were scaled using the standardization method so that all variables are measured in the same unit and contribute equally to the analysis.
- Model training. Several machine learning models were used to experiment on data fitting. The metric used for the model's performance evaluation was the mean absolute error (MAE) in dollars.
    - K-Nearest Neighbors Regressor: For the hyperparameter of the number k of neighbors to inspect several values of k were examined and k = 15 gave the minimum error MAE = 35.76. The weights were chosen to be either uniform (giving the same weight in each neighbor) or by 'distance' (giving more importance to closest neighbors).
    - Decision Tree: This model was used with the default parameters giving MAE = 35.39.
    - Random Forest: The best performance given by the Random Forest model was MAE = 35.23.

- Linear Regression: The performance Linear Regression gave was MAE = 35.76.
- Multilinear Perceptron: The parameters handled here were the number of the hidden layers (1, 2 or 3), the number neurons in each layer, the activation function (identity, logistic, tanh, and ReLU) and the learning rate (constant and adaptive). The technique of early stopping was also used to avoid the data overfitting. The best model had 3 hidden layers of 9 neurons each, used the ReLU activation function and constant learning rate. The performance of this model was MAE = 34.71.

Summarizing, the goal of estimating a suitable price for a given property, based on its attributes, was achieved effectively by many different machine learning models. The best one was a Multilinear Perceptron which produced mean absolute error MAE = 34.83.

Data Visualizations and Advanced Analytics

Using Power BI, interactive dashboards were created, which are useful in answering intrinsic questions fast and easily. Specifically:

- For the easy overview of the top rated and top priced properties, a dynamic diagram was made along with the mapping of the property's coordinates.
- To answer the question of which hosts are the busiest, two variables of the dataset were used. There is no direct information in the dataset for the specific request, so the variables 'reviews_per_month' and 'numbers_of_reviews' can be combined by the user interested. Both can be translated as a measure of renting frequency, however, if both variables have a high value then the inference that the property is busy (and subsequently its host) is more reliable.

The previously mentioned attributes-price models regarding the properties are also very effective in answering other type of questions.

- The most direct application of the model is the evaluation of overrated and underrated properties, by comparing their present price with the one predicted by the model.
- For the evaluation of the most critical features, the standardized coefficients calculated by Linear Regression are insightful. The results showed that the first most important feature, for predicting the price, is the property's neighborhood, the second is the number of bedrooms and the third the number of people that accommodates.
- Another interesting issue that can be solved by using the model, is the selection of the hosts that can be notified to increase their price. This can be done by taking into consideration the rating of the property and the prediction

of the price that the attributes-price model gives. Our suggestion is that it would be meaningful to use the model for predictions among the top (or adequately) rated properties and notify the ones found underrated.

Project evaluation, advantages, limitations and future work suggestions

The project has successfully completed the goals set. Specifically, the creation of a data warehouse has been completed, user-friendly reports were created, and properties attributes were modeled properly to be able to predict the property price. The results show that the problems faced by StayMore company are solved effectively.

However, there is always room for improvement. One limitation that the present solution is encountering, is the poor price prediction for properties whose value is “unexpected”. The reasons justifying such deviations are possibly not fully grasped in the current dataset or they are present in a form not not recognizable for the model. This means that present features should be transformed or new variables should be measured so that the model can perform better.

Some suggestions that could improve the performance of the attributes-price model are the further analysis the feature ‘amenities’ and the insertion of a new variable that represents the size of the house. Specifically, the feature representing the list of the amenities that the property is offering, could be replaced by a feature with fewer and more meaningful categories. For example, the amenities could be grouped in “luxurious”, “semi- luxurious” and “basic” which will probably be more insightful on the expected property price, comparing to a simple number of available amenities used in this experimentation. Another suggestion is the insertion of a new variable regarding the square footage of the property which is probably greatly helpful for predicting the property price.