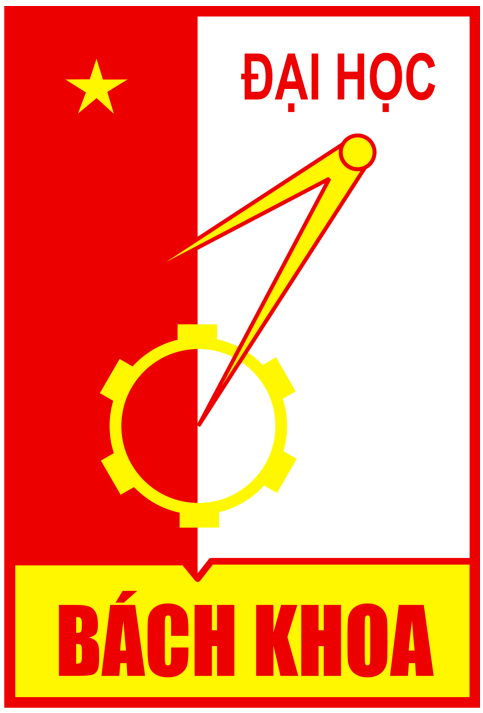
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY



INTRODUCTION TO DATA SCIENCE
HOTEL PRICE PREDICTION

Do Thanh Lam	20194785

Hanoi, 12/23

I. Introduction to our project

Our venture into machine learning aims to revolutionize the prediction of car prices by leveraging data sourced from MakeMyTrip.com. Our objective is to construct a robust predictive model capable of accurately estimating forthcoming hotel room prices. Through the implementation of sophisticated algorithms, our ultimate goal is to furnish users with a tool that facilitates informed decision-making throughout the booking journey. This project embodies a tangible application of machine learning within the realms of the real world, showcasing its immense potential in augmenting capabilities and refining decision-making processes within the booking sector.

Process

In our hotel price prediction project, we embark on a comprehensive four-step process to handle data.

Data Crawling: We begin by sourcing relevant data from MakeMyTrip.com, meticulously extracting pertinent information regarding hotel prices, features, and other crucial details. This step involves accessing and collecting data from the website's resources.

Data Cleaning: Following data acquisition, our focus shifts to data refinement. We meticulously clean the gathered information, addressing missing values, eliminating duplicates, and rectifying inconsistencies or errors within the dataset. This stage ensures the data's integrity and quality for subsequent analysis.

Data Visualization: With the refined dataset in hand, we delve into data visualization techniques. Utilizing graphs, charts, and other visual tools, we explore the relationships between various hotel attributes and prices. Visual representation aids in uncovering patterns, trends, and correlations, offering valuable insights into factors influencing hotel pricing.

Machine Learning: Leveraging advanced machine learning algorithms, we construct predictive models. These models are trained on the cleaned dataset, incorporating features such as hotel location, amenities, ratings, and more to predict future hotel room prices accurately. Through model evaluation and fine-tuning, we strive to create a reliable tool that empowers users to make informed decisions during their booking process.

II. Data Crawling

Data Crawling:

In the context of our hotel price prediction project, the data crawling phase involves scientifically collecting relevant information from MakeMyTrip.com, a popular travel booking website. This process typically entails using web scraping techniques to extract data from the website's pages related to hotel listings, prices, amenities, locations, ratings, and other essential details.

Identifying Relevant Data: We pinpoint and define the specific data attributes needed for our predictive model, included : rating, total of reviews, star rating, location, price and tax

Web Scraping Implementation: Using web scraping libraries selenium in Python, we craft scripts that navigate through the website's pages, simulate user interactions (like clicking through search results or hotel profiles), and extract structured data from HTML content.

Handling Data Structure: Data is retrieved to CSV format file for further processing

Data Preparation:

Following data crawling, the next step involves preparing the collected data for analysis and model building:

Data Cleaning: This phase focuses on cleaning the acquired data. It includes handling missing values, removing duplicates, addressing inconsistencies, and performing necessary transformations to ensure data integrity and quality.

Feature Engineering: This step involves creating new features or modifying existing ones to enhance the predictive power of the model. For instance, deriving new attributes like the rating per hotel,

extracting average geographical features from addresses, or categorizing amenities can enrich the dataset.

Encoding: Numerical features might need scaling to bring them within a similar range, while categorical variables might require encoding (such as one-hot encoding) for model compatibility.

By meticulously executing these data crawling and preparation steps, we create a well-structured and refined dataset, laying the foundation for accurate model training and prediction in the subsequent phases of the project.

III. Data Cleaning

Read raw Dataset

```
data1 = pd.read_csv('hoteldata.csv', on_bad_lines='skip')
```

```
data1.head()
```

	Unnamed: 0	hotelname	rating	reviews	star rating	location	nearest	distance nearest	price	tax	tax.1
0	0	Grande Collection Hotel & Spa	5.0	322.0	4.0	Old Quarter	NaN	NaN	D 41	10	NaN
1	1	Diamond Legend Hotel	4.5	1142.0	3.0	Old Quarter	Old Quarter	840 m	D 20	5	NaN
2	2	Grand Mercure Hanoi	5.0	197.0	NaN	Cat Linh	Old Quarter	2.6 km	D 137		NaN
3	3	Lotte Hotel Hanoi	5.0	2469.0	5.0	Ba Dinh	NaN	NaN	D 93	28	NaN
4	4	Silk Path Hotel Hanoi	4.5	4361.0	4.0	Old Quarter	NaN	NaN	D 59	13	NaN

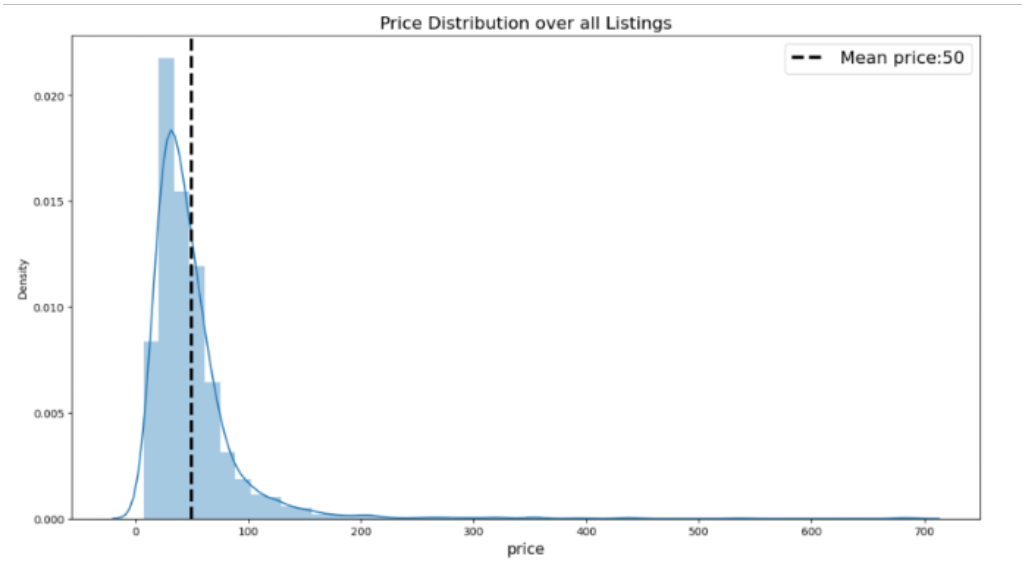
- Raw data set has 11 columns : index, hotel name, rating, review , star rating, location, nearest , distance nearest, price, tax and tax1.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3601 entries, 0 to 3600
Data columns (total 11 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Unnamed: 0         3601 non-null   int64  
 1   hotelname          3601 non-null   object 
 2   rating             2733 non-null   float64
 3   reviews            2733 non-null   float64
 4   star rating        3490 non-null   float64
 5    location          3601 non-null   object 
 6    nearest           1153 non-null   object 
 7    distance nearest  1153 non-null   object 
 8   price              3601 non-null   object 
 9   tax                740 non-null    object 
 10  tax.1              2085 non-null   object 
dtypes: float64(3), int64(1), object(7)
memory usage: 309.6+ KB
```

- From 3601 raw records,we removed duplicate row, and we have 2447 entries.
- After that is handle missing data :
  - •rating : mean of all rating
  - •reviews : mean of all rating
  - •star rating : no row has NaN
  - •location : no row has NaN
  - •price : no row has NaN
  - •tax : fill 0 if Nan or empty

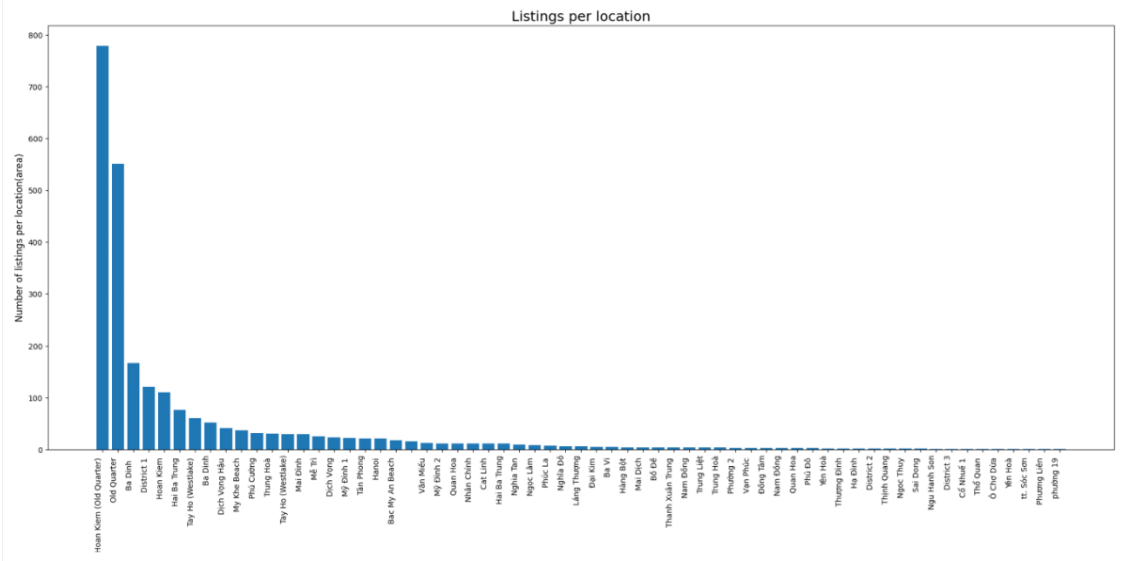
III. Data Visualization

1. Price distribution



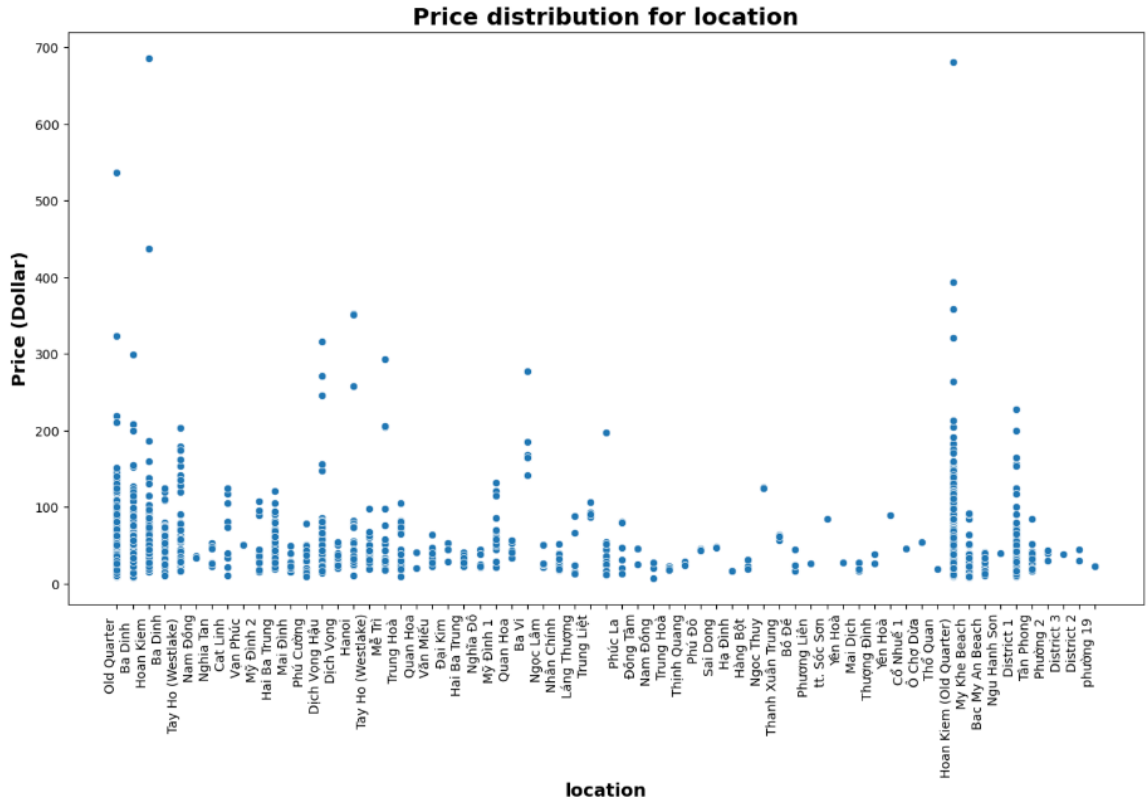
Our data visualization for the hotel price prediction machine learning project showcases the overall distribution of hotel prices. The mean price stands at 50 dollars, representing the average price across the various accommodations. Interestingly, the majority of prices fall within the range of 10 to 100 dollars, indicating a wide and popular price range within the hotel market.

2. Listing by location



Our listings by location chart provides a clear overview of different regions and their listing distribution. Hoan Kiem and the Old Quarter stand out, surpassing other areas in terms of number of listings.

3. Price distribution for location



The location of the hotel greatly affects the price. Only a few areas have high-priced hotels, while most others do not.

IV. Training the model

We used 2 model RandomForest and GradientBoosting

Let comparison between 2 model

RandomForest

- Combine multiple decision tree to make prediction, each tree is trained on a random subset.

- Uses a voting technique from multiple trees to make the final prediction. Each tree has equal weight.

- Typically faster to train because it can train trees independently and in parallel.

GradientBoosting

- Builds decision trees sequentially, with each tree trying to correct the errors. Each new tree is trained to focus on data points

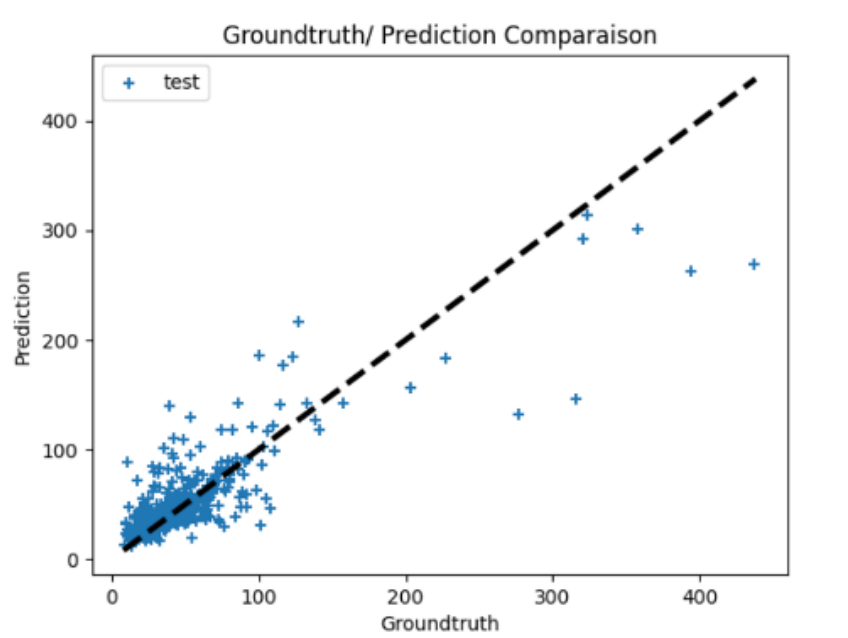
- Combines predictions from multiple trees sequentially, with each tree adjusted to minimize the errors made by the previous ones.

- Training time can be longer due to the sequential tree-building process and individual tree adjustments.

The result when using 2 model :

1. RandomForest:

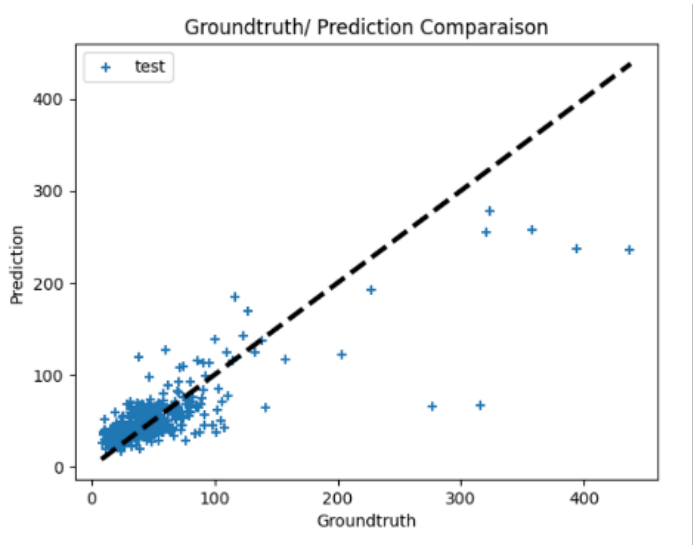
```
Mean absolute Error: 13.357448979591837
Mean Squared Error: 563.5928291836734
Root Mean Squared Error: 26.603592841997273
r squared 0.7139279466607497
```



Comments about the Random Forest model :MAE and MSE: MAE evaluates the average difference between the predicted value and the actual value as about 13 units. The MSE is 563.59, which represents a larger difference between the prediction and the actual value, especially since it squares the errors.
RMSE: RMSE (26.60) is the square root of MSE, which indicates that the average difference from the original unit is about 26.6 units. It retains the units of the dependent variable.
R-squared (R^2): R^2 is 0.714,
Overall comments:
This model can predict hotel prices with a fairly good degree of accuracy. However, there remains a large percentage of variation that cannot be explained by the model.
MAE and RMSE also show that, on average, the model's prediction error is not too large, which can be considered a positive point.

2. Gradient Boosting

```
Mean absolute Error: 15.377394710986474
Mean Squared Error:707.7511521027685
Root Mean Squared Error : 26.603592841997273
r squared: 0.640755142984129
```



Comments about the model:
MAE and MSE:

The MAE is at 15.377, which is higher than the previous result, indicating a larger average difference between the predicted and actual values.
The MSE is 707.75, which is larger than the previous result, which is also a larger difference between the prediction and the actual value, especially because it squares the errors.
RMSE:

RMSE remained at 26.60, indicating an unchanged mean difference from the original unit compared to the previous result.
R-squared (R^2):

R^2 is 0.641, lower than the previous result.
Overall comments:
The model is capable of predicting hotel prices, but with greater variation than the previous model.
Decreasing R^2 and increasing MAE and MSE may indicate that the model is not performing as well as expected.