AIRBNB Use Case

Predicting The Price of AirBnB Rental

Outline

- Background
- Methodology
- Result
- Recommendation

Background

Introduction

I want to discuss about implementation of **Data Science and Statistics in Hospitality**. I am interested to explore use case in AirBnB. AirBnB is an online service for booking lodging is trending around the world. During COVID-19, hospitality business is down but soon after this pandemic over The Business will continue to grow again. AirBnB need to prepare because the amount of people who are interested in staycation will rapidly increase. One of the preparation that will be needed is new pricing strategy.

Problem Statement

"How might we **determine** the **optimal price** so we can predict **maximize current potential income** ?"

Pricing :

- Too high : Customer will **run into** the **competitor**
- Too low : Company **suffer a loss**

Goal :

We can get **better pricing strategy** so the **revenue will increase**

Approach

Utilizing **Artificial Intelligence (AI)**

There is an AI which is called regression, Using regression we can predict each of the price of Airbnb rental based on these following historical information (data scrapped June 25th 2019 and contained 8293 listing):

Features Group :

1. Host that taking care of the rental
2. Available facilities
3. Location and Accessibility
4. Reviews
5. Availability
6. Length Of Stay

Target Variable : Price

Success Criteria

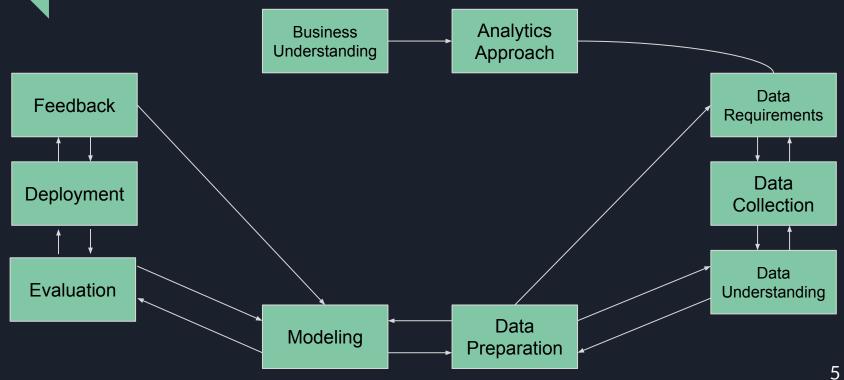Maximize Current Potential Income

Threshold

Minimum Price of Each Listing
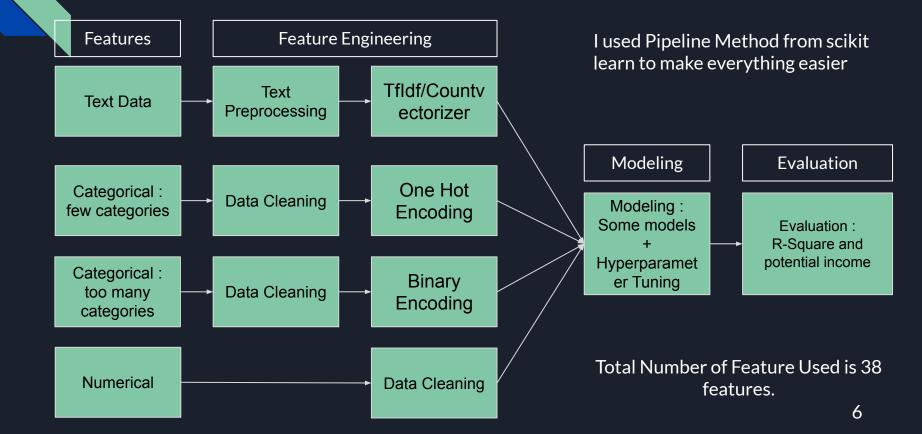
Why Should We Use This AI ?

- Utilizing advanced technology so we can get the prediction result automatically
- doesn't need to be specifically programmed
- Able to adapt easily
- There are several algorithm we can use. Such as Linear Regression, Extreme Gradient Boosting, Random Forest etc. Each algo has its own characteristics. We can do an experimentation before real implementation
- Able to utilize both structured and unstructured data like text
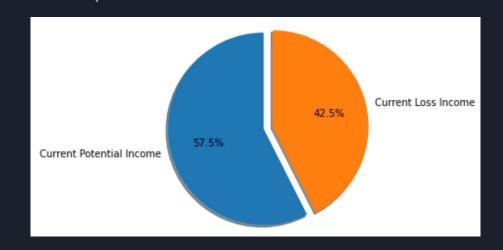
Methodology

Methodology

Pipeline : Data Preparation and Modeling

Features	Feature Engineering

Text Data	Text Preprocessing	TfIdf/Countvectorizer

Categorical : few categories	Data Cleaning	One Hot Encoding

Categorical : too many categories	Data Cleaning	Binary Encoding

Numerical		Data Cleaning

I used Pipeline Method from scikit learn to make everything easier

Modeling	Evaluation

Modeling : Some models + Hyperparameter Tuning	Evaluation : R-Square and potential income

Total Number of Feature Used is 38 features.
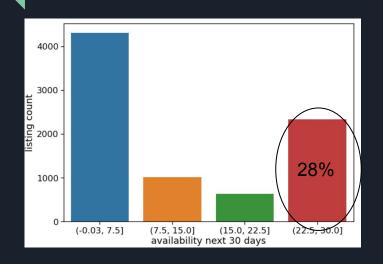
6

Result

What Happened ?

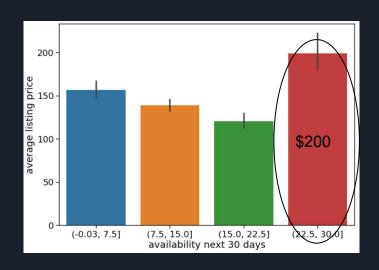12.15 % listing still has maximum availability

In the Dataset, we analyze potential income for all the rental for next 30 days. Unfortunately, there is still 12.15% listing that has maximum availability. We need to put more attention to those 12.15% listing.

Now, we look closer into the income as a whole. Maximum potential income for next 30 days that we can get is SG$ 40,794,120 but the current potential income is only SG$ 23,428,767. It's equal to 57.5% of maximum potential income.

We are interested to develop the AI based on this data so we can keep this good performance in the future or even improve it.

Why Did It Happen ?

Besides listing with maximum availability. There are **28% listing** that have very **high availability** (23 - 30 days) and yet the price are still set high. I think we can make price adjustment for these listing so we can decrease the availability and eventually increase the income.

Solution

Modeling Result

Model	R-square Test(%)
Extreme Gradient Boosting	72.41
Tuned Extreme Gradient Boosting	75.29

Final model we used is Tuned extreme gradient boosting regression with R-square 75.29%. This model is already tuned optimized by R-Square. This model also utilize both structured and unstructured data. We will illustrate the changes in income in the next slides.

Modeling Process

Model	R-square CV(%)
Decision Tree	50.93
Random Forest	53.11
Adaptive Boosting	45.28
Gradient Boosting	70.08
Extreme Gradient Boosting	70.18

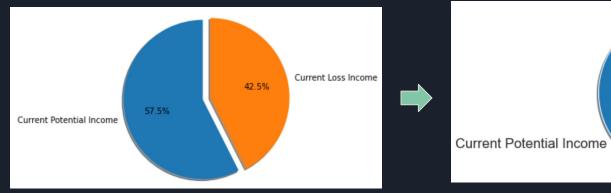| Model	R-square CV(%)
Extreme Gradient Boosting	70.18
Tuned Extreme Gradient Boosting	72.88

How To Make It Happen ?
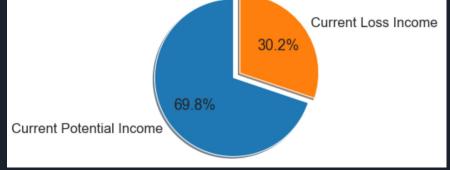
Price = Cost + Location + Reviews + Length of Stay + Availability

Recommendation :

- Cost, Location , Reviews, length of stay and availability are some groups of feature that used in the modeling and Price as target
- Cost, Location and Reviews are fixed term
- We can make Price adjustment based on length of stay and availability using model that we already develop, Refined Extreme Gradient Boosting then predict the price
- So, which listing that we need to adjust. We need to **adjust** the **price** for **listing** that still has **high availability**
- This **adjustment** will be given as **price recommendation** for listing
- **Adjustment threshold, minimum price of each listing**
- **Example Adjustment,** It's predicted that current potential income will increase from SG$ 23,428,767 to SG$27,670,076
  - *note : now i don't know the minimum price for each listing if it is available i would have.used it. So, i only adjust it by length of stay and expected availability. Then I set the minimum price 80% of current price

10

Adjustment Result

Current potential income is SG$ 23,428,767 of SG$ 40,794,120

Predicted Current potential income after adjustment is SG$27,670,076 of SG$39,620,547

Deployment Recommendation

Recommendation for Deployment

Doing further analysis for another potential strategy like competitor based, forecasting based etc but this need another data source

Doing some testing:

1. Let's say method that we discuss is a new features
2. We need to do some testing and get any feedback before it go live
3. How to compare → we can use KPI in form revenue:
    a. Compute revenue before implementation in certain border of time
    b. Compute revenue after trial implementation in certain border of time
    c. Compare the revenue, using A/B testing will be better
    d. If success continue the process
    e. If fail, go back to modeling and evaluation or even further

After GO Live, We need to monitor the deployment time to time :

1. Data Versioning
2. Dictionary Versioning
3. Feature Versioning
4. Algorithm Versioning
5. Interpretation/Insight Versioning

Thank You

References

https://www.hotelogix.com/blog/2016/11/17/5-intuitive-ways-boost-hotels-revenue-smarter-pricing/
https://www.ezeeabsolute.com/blog/hotel-pricing-strategies/
https://www.revfine.com/pricing-strategies-hotel-industry/