Airbnb Listing Bangkok Data Analysis

Capstone Project 2

Brillian Adhiyaksa Kuswandi

WWW.AIRBNB.CO.ID

Purwadhika
Digital Technology School

Background

Airbnb is a prominent online marketplace that enables **users to rent out properties**, typically for short or long-term stays. The platform provides a convenient way **for hosts to offer spaces** and for travelers to find unique accommodations. With a range of services that includes property listings, booking management, and communication, Airbnb has become a significant player in the travel industry. As Airbnb listings grow in cities worldwide, analyzing these listings becomes essential for understanding trends, and pricing, which can help improve customer satisfaction and business operations.

Problem Statement

With the rise in Airbnb listings, especially in urban areas like Bangkok, there is a **need to analyze data to uncover patterns that influence pricing, and customer preferences**. Understanding these **factors can assist hosts in optimizing their listings** and **aid Airbnb in enhancing platform recommendations and offerings**.

Purwadhika
Digital Technology School

Objective

The objective of this analysis is to explore Airbnb listings in Bangkok, **examining aspects such as pricing, room types, and location**. This analysis **aims to provide insights that help hosts set competitive prices**, **identify popular locations**, and **understand customer preferences** based on listing data.

Purwadhika
Digital Technology School

Data Source

The data used in this analysis is from the Bangkok Airbnb dataset.
Bangkok Listing
Bangkok Geospatial Data

Data Preparation

Data preparation is the process of cleaning and organizing raw data so it’s ready for analysis.

1. Importing Library that needed
2. Read Data
3. Formatting Data
4. Missing Value Handling
5. Checking Anomalies Data
6. Extracting Features
7. Testing

Purwadhika
Digital Technology School

Library and Data overview

```
# Essential libraries for data manipulation, geospatial, and numerical operations
import pandas as pd
import numpy as np
import geopandas as gpd
import fiona
from shapely.geometry import Point, LineString, MultiLineString, Polygon, GeometryCollection, mapping
from scipy.stats import f_oneway, spearmanr, mannwhitneyu, ttest_ind, normaltest, chi2_contingency
from geojson_length import calculate_distance, Unit
from collections import Counter
import datetime as dt
import re
import string

# Visualization libraries
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import folium
from folium.plugins import MarkerCluster, MousePosition, FloatImage, TagFilterButton, HeatMap
from branca.element import Figure
from wordcloud import WordCloud
%matplotlib inline

# Text processing
import nltk
from nltk.corpus import stopwords

# Language models (LLM)
from langchain_ollama.llms import OllamaLLM

# Suppress warnings
import warnings
warnings.filterwarnings("ignore")

import requests
from PIL import Image
from io import BytesIO
import base64
```

Libraries that used are:

Data manipulation: pandas, numpy
Geospatial analysis: geopandas, fiona, shapely, geojson_length
Statistical testing: scipy.stats
Visualization: matplotlib, seaborn, plotly, and folium
Text Processing: nltk
Image Handling: PIL, requests, and io
cleaner outputs: warnings

```
df = pd.read_csv('Data/Airbnb Listings Bangkok.csv', index_col=0)
df.head(5)
```

	id	name	host_id	host_name	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_reviews	last_review	reviews_per_month	calculated_host_listings_count	availability_365	number_of_reviews_ltm
0	27934	Nice room with superb city view	120437	Nuttee	Ratchathewi	13.75983	100.54134	Entire home/apt	1905	3	65	2020-01-06	0.50	2	353	0
1	27979	Easy going landlord,easy place	120541	Emy	Bang Na	13.66818	100.61674	Private room	1316	1	0	NaN	NaN	2	358	0
2	28745	modern-style apartment in Bangkok	123784	Familyroom	Bang Kapi	13.75232	100.62402	Private room	800	60	0	NaN	NaN	1	365	0
3	35780	Spacious one bedroom at The Kris Condo Bldg. 3	153730	Sirilak	Din Daeng	13.78823	100.57256	Private room	1286	7	2	2022-04-01	0.03	1	323	1
4	941865	Suite Room 3 at MetroPoint	610315	Kasem	Bang Kapi	13.76872	100.63338	Private room	1905	1	0	NaN	NaN	3	365	0

Purwadhika
Digital Technology School

Data Formatting

```
df['id'] = df['id'].astype('str')
df['host_id'] = df['host_id'].astype('str')
df['last_review'] = pd.to_datetime(df['last_review'])

def convert_to_category(col):
    if col.dtype == 'object':
        if col.nunique() / df.shape[0] < 0.05:
            return col.astype('category')
        else:
            return col
    else:
        return col

df = df.apply(lambda col: convert_to_category(col))
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 15854 entries, 0 to 15853
Data columns (total 16 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   id                              15854 non-null  object
 1   name                            15846 non-null  object
 2   host_id                         15854 non-null  object
 3   host_name                       15853 non-null  object
 4   neighbourhood                   15854 non-null  category
 5   latitude                        15854 non-null  float64
 6   longitude                       15854 non-null  float64
 7   room_type                       15854 non-null  category
 8   price                           15854 non-null  int64
 9   minimum_nights                  15854 non-null  int64
 10  number_of_reviews               15854 non-null  int64
 11  last_review                     10064 non-null  datetime64[ns]
 12  reviews_per_month               10064 non-null  float64
 13  calculated_host_listings_count  15854 non-null  int64
 14  availability_365                15854 non-null  int64
 15  number_of_reviews_ltm           15854 non-null  int64
dtypes: category(2), datetime64[ns](1), float64(3), int64(6), object(4)
memory usage: 1.8+ MB
```

convert the **id and host_id** columns **to strings** and **the last_review** column **to datetime**.
Define a function to convert columns with **low cardinality (fewer than 5% unique values) to the 'category' datatype**
This Dataset contains: 2 category, 1 datetime, 3 float, 6 int, and 4 object data

Purwadhika
Digital Technology School

Descriptive Statistic

	count	unique	top	freq
id	15854	15854	27934	1
name	15846	14794	New! La Chada Night Market studio 2PPL near MRT	45
host_id	15854	6659	201677068	228
host_name	15853	5312	Curry	228

	count	mean	min	25%	50%	75%	max	std	Range	Variansi
latitude	15854.0	13.75	13.53	13.72	13.74	13.76	13.95	0.04	0.42	0.000000e+00
longitude	15854.0	100.56	100.33	100.53	100.56	100.59	100.92	0.05	0.59	0.000000e+00
price	15854.0	3217.7	0.0	900.0	1429.0	2429.0	1100000.0	24972.12	1100000.0	6.236069e+08
minimum_nights	15854.0	15.29	1.0	1.0	1.0	7.0	1125.0	50.82	1124.0	2.582170e+03
number_of_reviews	15854.0	16.65	0.0	0.0	2.0	13.0	1224.0	40.61	1224.0	1.649440e+03
last_review	10064	2021-08-30 08:37:49.316375296	2012-12-15 00:00:00	2020-02-20 00:00:00	2022-10-24 00:00:00	2022-12-08 00:00:00	2022-12-28 00:00:00	NaN	3665 days 00:00:00	NaN
reviews_per_month	10064.0	0.81	0.01	0.12	0.44	1.06	19.13	1.09	19.12	1.190000e+00
calculated_host_listings_count	15854.0	13.89	1.0	1.0	4.0	13.0	228.0	30.27	227.0	9.162600e+02
availability_365	15854.0	244.38	0.0	138.0	309.0	360.0	365.0	125.84	365.0	1.583652e+04
number_of_reviews_ltm	15854.0	3.48	0.0	0.0	0.0	3.0	325.0	8.92	325.0	7.951000e+01

From this, we can see descriptive statistics such as **unique values**, **counts**, **means**, **medians**, **minimums**, **maximums**, **standard deviations**, **ranges**, and **variances** for **both categorical and numerical data** in the Airbnb listings dataset.

Purwadhika
Digital Technology School

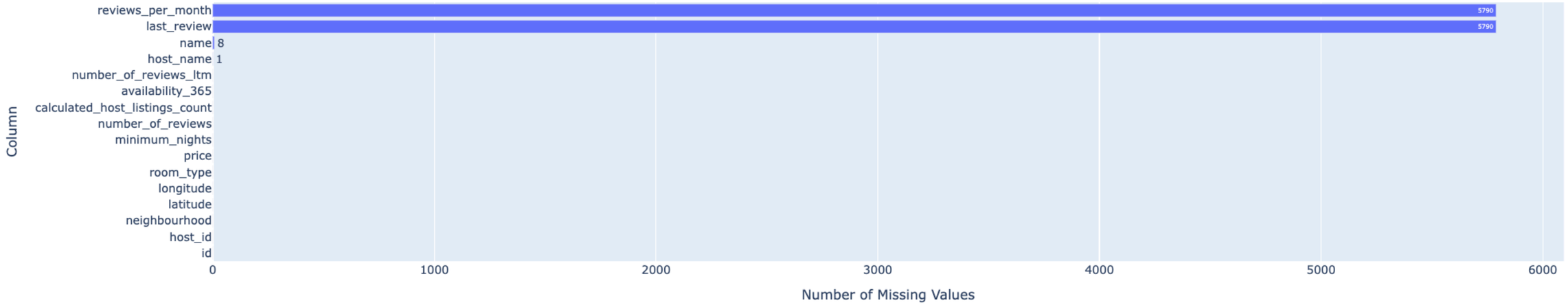
Dataset Summary

	Column Name	Datatype	Missing Values (%)	Number of Unique	Percentage of Unique	Unique Value
0	id	object	0.0%	15854	100.0%	[27934, 27979, 28745, 35780, 941865, 1704776, ...
1	name	object	0.05%	14794	93.31%	[Nice room with superb city view, Easy going l...
2	host_id	object	0.0%	6659	42.0%	[120437, 120541, 123784, 153730, 610315, 21296...
3	host_name	object	0.01%	5312	33.51%	[Nuttee, Emy, Familyroom, Sirilak, Kasem, Wimo...
4	neighbourhood	category	0.0%	50	0.32%	['Ratchathewi', 'Bang Na', 'Bang Kapi', 'Din D...
5	latitude	float64	0.0%	9606	60.59%	[13.75983, 13.66818, 13.75232, 13.78823, 13.76...
6	longitude	float64	0.0%	10224	64.49%	[100.54134, 100.61674, 100.62402, 100.57256, 1...
7	room_type	category	0.0%	4	0.03%	['Entire home/apt', 'Private room', 'Hotel roo...
8	price	int64	0.0%	3040	19.17%	[1905, 1316, 800, 1286, 1000, 1558, 1461, 700,...
9	minimum_nights	int64	0.0%	86	0.54%	[3, 1, 60, 7, 250, 2, 15, 30, 28, 21, 27, 4, 1...
10	number_of_reviews	int64	0.0%	298	1.88%	[65, 0, 2, 19, 1, 10, 4, 27, 129, 208, 3, 78, ...
11	last_review	datetime64[ns]	36.52%	1669	10.53%	[2020-01-06 00:00:00, NaT, 2022-04-01 00:00:00...
12	reviews_per_month	float64	36.52%	513	3.24%	[0.5, nan, 0.03, 0.17, 0.01, 0.09, 0.19, 1.17,...
13	calculated_host_listings_count	int64	0.0%	50	0.32%	[2, 1, 3, 41, 10, 7, 6, 4, 37, 8, 19, 5, 53, 4...
14	availability_365	int64	0.0%	366	2.31%	[353, 358, 365, 323, 87, 320, 356, 361, 330, 1...
15	number_of_reviews_ltm	int64	0.0%	85	0.54%	[0, 1, 3, 13, 2, 7, 5, 10, 9, 12, 29, 4, 19, 5...

	Value
Number of columns	16
Number of rows	15854
Duplicate rows	0
Text type	4
Numeric type	9
Categorical type	2
Datetime type	1

The tables summarize the dataset, showing each column's datatype, **missing values**, **unique values**, and their **counts**, along with the **total number of columns**, **rows**, **duplicates**, and the **distribution of text, numeric, categorical, and datetime** columns. This provides an overview of the dataset's structure and quality.

Missing Value



It visualizes the count of missing values per column in the dataset. In this case, **only 4 columns have missing values**: the "**review_per_month**" and "**last_review**" columns each **have 5,790 missing values**, while the "**name**" column **has 8 missing values**, and the "**host_name**" column **has 1 missing** value.

Purwadhika
Digital Technology School

Cleaning Name Column

	id	name	host_id	host_name
9489	39762269	Executive Deluxe Room, I Residence Hotel Sathorn	301148796	Sawassakorn
14351	730403485467994514	Easy going, 2 mins from BTS Ratchathewi, Block740	40563982	Nattarat
6264	30086067	One Bedroom Asoke area close to MRT/ARL /Nice unit	154634369	Mycondo
1662	11425056	1BR condo in Center of Bangkok!	60017041	Panuwat
5674	28089290	อพาร์ตเมนต์ 1 ห้องนอนที่ทันสมัย	212187760	Charlotte
7881	34920489	近onnut BTS站/高速WIFI/7-11便利店/15分钟到暹罗广场四面佛&泳池健身房&每客消毒	145673155	Cora
903	6949824	studio, 900 m to BTS Phrom Phong BTS	23888912	Nalinee
13484	665087191522315628	THE SPRING BLOSSOM SUITE:1BR/WIFI/ASOKE/SHOPPING	133082557	Jean
6621	30974375	PLATINUM MALL | FAMILY ROOM | upto 4 ppl	192217276	Pratunam Design
12264	53873868	Escape Condo	436384240	บวร
2069	12687676	The Hyde "6th floor" Sukumvith 13	37225700	Chani
5725	27618932	Suanchitlada dusit Maneewan Place	129158607	Sirawut
9638	40660343	One Condo Ratchada- Latprao/ 5m walk MRT	105304529	My
9322	39656036	BTS rachatiwi 3min Airportlink phayathai 5min	163260650	Seitaro
5880	28616305	TanTanGuestHouse(ตาลตาลเกสต์เฮาส์)	215970644	♂Yutthakarn

New! La Chada Night Market studio 2PPL near MRT	45
New! La Chada Night Market 1BR 2PPL near MRT	35
30days! AirportLink Sukhumvit NANA MaxValu 2BR(4P)	35
New! Gateway/ Bangkok University 1BR 2PPL near BTS	29
30days! Sukhumvit NANA spacious 1BR 2PPL near BTS	25
1BR Twin Suit 2ppl/Surasak BTS Sathorn/Pool /WIFI	24
1BR Twin Suit 2ppl/Surasak BTS Sathorn/Pool /Wifi	21
Near The Grand Palace/ASOK Station/Sukhumvit	18
2BR! New! Near The Grand Palace/ASOK Station/4PPL	15
New!Near The Grand Palace/ASOK Station/4PPL	14

1 br twin suit 2 ppl surasak bts sathorn pool wifi	51
new la chada night market studio 2 ppl near mrt	48
days airportlink sukhumvit nana maxvalu 2 br 4 p	35
new la chada night market 1 br 2 ppl near mrt	35
new gateway bangkok university 1 br 2 ppl near bts	29
near the grand palace asok station sukhumvit	25
days sukhumvit nana spacious 1 br 2 ppl near bts	25
studio at luxx xl langsuan of	21
2 br new near the grand palace asok station 4 ppl	15

The "**name**" column contains **non-alphabetic characters** and text in **non-English languages**. This indicates the need to **remove all non-alphabetic characters** and **translate the non-English text** to English using a language model (LLM).

Translating non english using name LLM

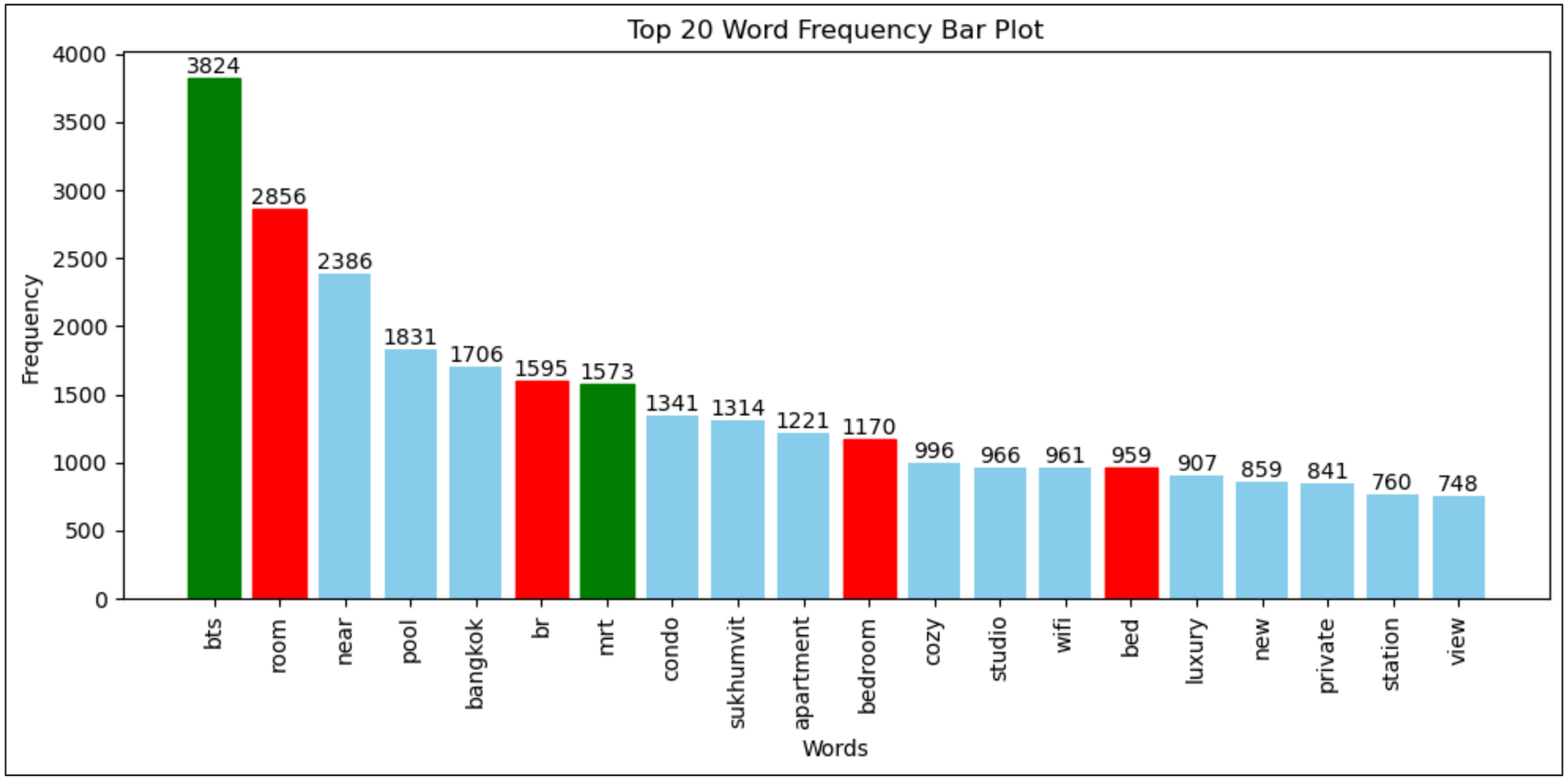
```python
model = OllamaLLM(model="llama3.2")
ulangan = 1

def translate_to_english(text):
    global ulangan
    prompt = f"Translate '{text}' to English. Respond in one line only, in this format: 'The translation is:
[translated text]' without any extra details."
    response = model(prompt)
    print(f"{ulangan}: {response.strip()}")
    ulangan += 1
    return response.strip()

df_nonlatin['name'] = df_nonlatin['name'].apply(translate_to_english)
df_nonlatin['name'] = df_nonlatin['name'].apply(lambda x: x.replace('The translation is: ', ''))
df_nonlatin['name'] = df_nonlatin['name'].apply(clean_text)
```

```
1: The translation is: Granby Mansions Ladprao.
2: The translation is: House Mode
3: The translation is: Varaphada Noy Pong.
4: The translation is: House near the Suvarnabhumi Airport across from Saphan Chinair Airport.
5: The translation is: Home with clubhouse
6: The translation is: French Vintage Around the World B&B with Chinese Service and Language Support
7: The translation is: Siam City Around The World B.B., Chinese Service with Thai Language Communication.
8: The translation is: Wild West Room Around the World - B.B. Service: Chinese Translation and Language Assistance
9: The translation is: Little India Around the World B B Chinese Service Mandarin Communication
10: The translation is: Sumo Village Around The World B B - Chinese Service with Mandarin Language Communication.
11: The translation is: Room Bali Hut Around the World B B Chinese Service Language Communication
12: The translation is: Crystal View Apartment.
13: The translation is: A charming Casablanca-style cozy retreat with Chinese service and French communication.
14: The translation is: China Town Around the World B.B. Chinese Service Mandarin Communication
15: The translation is: Chime in.
16: The translation is: Park View Vibhawdee.
17: The translation is: Concord Condo Saphalai Kazaa Riva Vista 2 Tower.
18: The translation is: "One bedroom (1br), 6m (bathroom) Ekmai BTS Thonglor Japanese Style Wood (Teekee)"
19: The translation is: A perfect condo in the CBD serves up to 6.
20: The translation is: Apartment near BTS
21: The translation is: Family room with safari-themed decor and seating for up to 5 people.
22: The translation is: Perfect Park Suvarnabhumi Village.
23: The translation is: "Coconut Shrimp BBQ at a cozy apartment 5 minutes from Ekamai Train Station Eastern Bus Terminal Shopping Area Dragon's Den Massage"
24: The translation is: "Big 1 B.R Step to BTS Comfy Bed Welcome everyone."
25: The translation is: condominium Chatchotevan Intawon Rachada Building 2.
...
1952: The translation is: High space garden gym pool sauna luxury apartment Central Bangkok BTS Asok Nanamowai cowboy
1953: The translation is: Loft apartment near BTS, located in the heart of Bangkok, with restaurants and night market nearby, free WiFi, swimming pool, etc.
1954: The translation is: "Bkk bts surasak warmth loft apartment 5 free pool gym wifi free parking".
1955: The translation is: The Sukhothai Resort & Spa Core Two Bedroom Apartment near BTS On Nut Close to No boundary pool overlooking Bangkok city skyline with local street food available outside.
```

The next step is to **separate the non-Latin and Latin dataframes**. The **non-Latin dataframe will be translated** using an LLM, and the result is shown below the code. After translation, the **two dataframes will be merged** into one for further analysis.

Purwadhika
Digital Technology School

Showing Word Count



This visualization describes the word count, and from it, we can identify two patterns that can be extracted as new features, such as "**bedroom**" and "**station type**."

Checking Anomaly Data

host_id & host_name

```
if (df.groupby('host_id')['host_name'].nunique() > 1).any():
    print("There are multiple hosts with the same host_id")
else:
    print("There are no multiple hosts with the same host_id")

# "There are no multiple hosts with the same host_id"
```

In the "host_id" and "host_name" columns, there are no anomaly data, such as a single "host_id" associated with multiple "host_name" values. This indicates that there is **no anomaly** data in these columns.

Geometry (longitude & latitiude)

```
df.groupby(
  ['longitude', 'latitude']
  )['host_id'].nunique().reset_index(
  name='count').sort_values(by='count',
                            ascending=False).head(10)
```

	longitude	latitude	count
7888	100.563269	13.754188	7
2865	100.519891	13.710777	7
6451	100.555503	13.750300	5
11304	100.584781	13.716847	4
6157	100.553311	13.737687	4
7026	100.557960	13.734856	4
7513	100.561119	13.723233	3
12219	100.593528	13.714623	3
8961	100.568486	13.757097	3
8920	100.568269	13.756184	3

In the same geometry, multiple hosts can list the accommodation. This is **not considered anomaly data** because it aligns with Airbnb's policy.

Checking Anomaly Data

Geometry inside bangkok polygon

```
gdf = gpd.GeoDataFrame(df, geometry=gpd.points_from_xy(df.longitude, df.latitude))

# Set the same CRS for both GeoDataFrames (using a suitable projection)
gdf_district.set_crs(epsg=4326, inplace=True)
gdf_sub_district.set_crs(epsg=4326, inplace=True)
gdf.set_crs(epsg=4326, inplace=True)

gdf = gpd.sjoin(gdf, gdf_district[['dname_e', 'centroid_district', 'geometry']], predicate='within')
gdf.drop(columns=['index_right'], inplace=True)

gdf = gpd.sjoin(gdf, gdf_sub_district[['SNAME', 'centroid_sub_district', 'geometry']], predicate='within')
gdf.drop(columns=['index_right'], inplace=True)
```

```
if gdf['dname_e'].isna().sum() > 0 or gdf['SNAME'].isna().sum() > 0:
    print("There are some geometry outsite bangkok")
else:
    print("The Geometry is inside Bangkok")

# The Geometry is inside Bangkok
```

```
gdf[gdf['neighbourhood'].str.lower() != gdf['dname_e'].str.lower()].shape[0]

# 0
```

The next step is to check for anomalies in the geometry column, using external data for validation. The result shows that there are **no geometries outside of Bangkok**, and there are **no misspellings in the district names**.

Checking Anomaly Data

Price before drop anomaly

	name	price
9887	somerset maison asoke bangkok	0
13650	artist private airroom min 2 skytrainpetfriendly	278
13579	1 lower bunk bed w shared bath	280
13531	mixed dorm bunkbed at amazing khaosan hostel 2	280
13578	1 upper bunk bed w shared bath	280

Price after drop anomaly

	name	price
13650	artist private airroom min 2 skytrainpetfriendly	278
13579	1 lower bunk bed w shared bath	280
13531	mixed dorm bunkbed at amazing khaosan hostel 2	280
13578	1 upper bunk bed w shared bath	280
13592	flourish capsule hostel	295

The "price" column contains **values of 0**, which is **considered an anomaly** since there are **no free listings** on Airbnb. **A price of 0 is unrealistic**. Therefore, rows with a price **value of 0 will be removed** from the dataset to ensure data consistency.

Checking Anomaly Data

calculated_host_listing_count

```
check = gdf.groupby(['host_id',
'calculated_host_listings_count']).size().reset_index(name='count').sort_values(by='count', ascending=False)
check[check['count'] != check['calculated_host_listings_count']]

print(f'result: {check.shape[0]}')
```

This checks if the **count of listings** for each "**host_id**" **matches the values** in the "**calculated_host_listing_count**" column. The result shows that **all data is consistent** with no mismatches.

Purwadhika
Digital Technology School

Extracting the length from a point to the district center

```
def LongLine(x, y):
    line = LineString([x, y])
    geojson_line = {
        'type': 'Feature',
        'properties': {},
        'geometry': mapping(line)
    }
    return calculate_distance(geojson_line, Unit.meters)

gdf['distance_from_district_center'] = gdf.apply(lambda x: LongLine(x['geometry'], x['centroid_district']), axis=1)
gdf['distance_from_sub_district_center'] = gdf.apply(lambda x: LongLine(x['geometry'], x['centroid_sub_district']),
axis=1)
```

This code **calculates the distance** from each **listing to the district and sub-district centers**, creating two new features: distance_from_district_center and distance_from_sub_district_center. These distances **can help explore correlations** with other variables, like price.

Missing Value Handling

last_review

	name	last_review
0	nice room with superb city view	2020-01-06
1	easy going landlord easy place	NaN
2	modern style apartment in bangkok	NaN
3	spacious 1 bedroom at the kris condo bldg 3	2022-04-01
4	suite room 3 at metropoint	NaN

```
gdf['status'] = gdf['last_review'].apply(lambda x: 'unreviewed' if pd.isna(x) else 'reviewed')
```

For the "last_review" column, **missing values are transformed into a new feature**. If the value is missing, it is labeled as "unreviewed," and if it is not missing, it is labeled as "reviewed."

Purwadhika
Digital Technology School

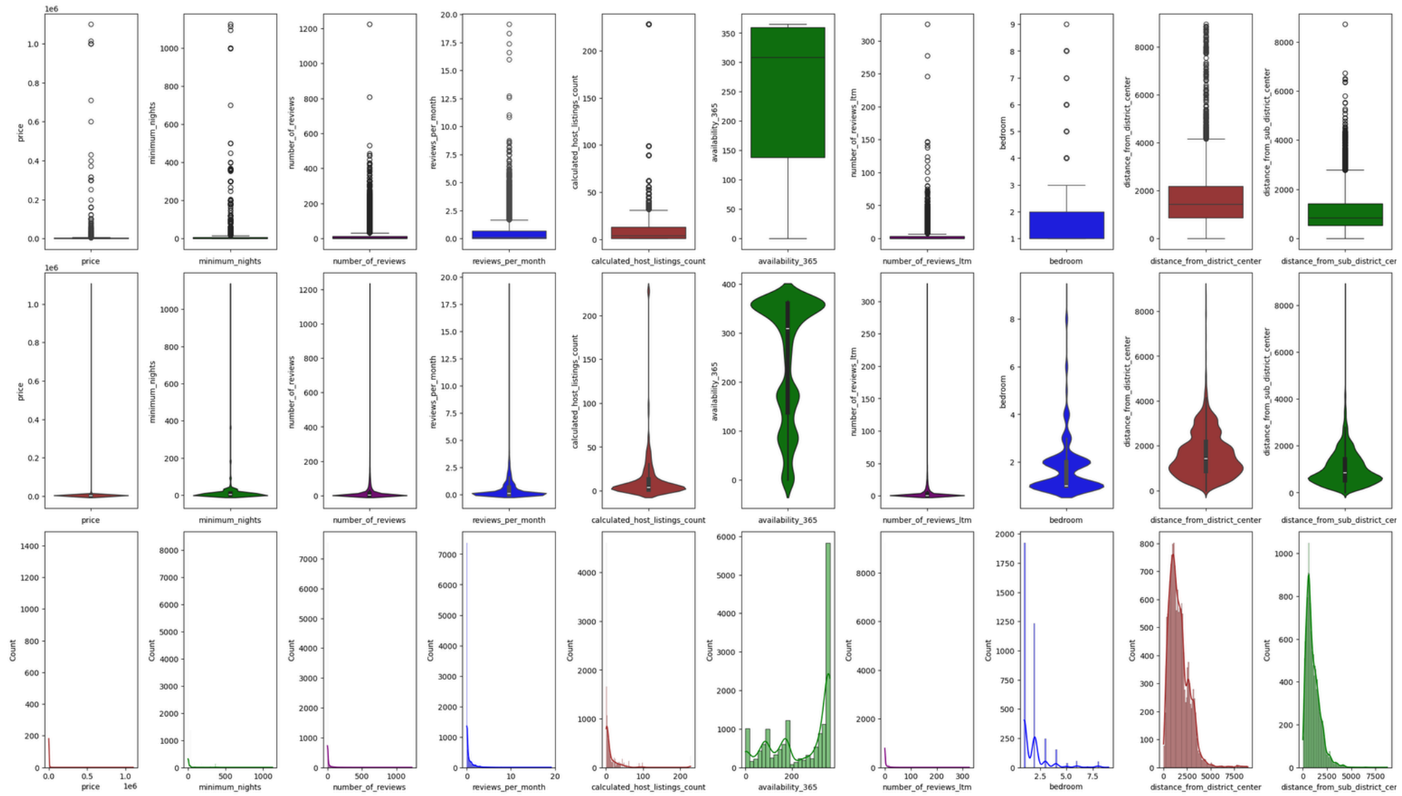
Missing Value Handling

review_per_month

```
print(f'before: {gdf.reviews_per_month.isna().sum()}')

condition = (gdf['number_of_reviews'] == 0) & (gdf['reviews_per_month'].isna())
gdf.loc[condition, 'reviews_per_month'] = gdf.loc[condition, 'reviews_per_month'].fillna(0)

print(f'after: {gdf.reviews_per_month.isna().sum()}')

# before: 5789
# after: 0
```

For the "**review_per_month**" column, missing values are **filled with 0**, but **only if the "number_of_reviews" is also 0**.

Normality Testing



	column	Test Statistic	p_value	result
0	id	2897.118116	0.000000e+00	not normally distributed
1	host_id	1416.490985	2.587562e-308	not normally distributed
2	latitude	4356.959559	0.000000e+00	not normally distributed
3	longitude	2077.433462	0.000000e+00	not normally distributed
4	price	43046.717756	0.000000e+00	not normally distributed
5	minimum_nights	22190.129178	0.000000e+00	not normally distributed
6	number_of_reviews	19163.172470	0.000000e+00	not normally distributed
7	reviews_per_month	15618.212372	0.000000e+00	not normally distributed
8	calculated_host_listings_count	16474.501173	0.000000e+00	not normally distributed
9	availability_365	12885.350519	0.000000e+00	not normally distributed
10	number_of_reviews_ltm	24895.698981	0.000000e+00	not normally distributed
11	bedroom	2088.624262	0.000000e+00	not normally distributed
12	distance_from_district_center	5453.881874	0.000000e+00	not normally distributed
13	distance_from_sub_district_center	4793.585683	0.000000e+00	not normally distributed

All numeric columns in this dataframe are **not normally distributed**. At a glance, the boxplot of the "availability_365" column appears to be roughly normal, but both the violin plot and histogram show a non-normal distribution. To confirm this, the normality will be tested using the normaltest.

Independent two-sample

T-test

	room_type_1	room_type_2	t_statistic	p_value	result
0	Entire home/apt	Private room	1.023683	3.060022e-01	No significant difference in average price
1	Entire home/apt	Hotel room	0.759066	4.479454e-01	No significant difference in average price
2	Entire home/apt	Shared room	7.664332	1.977270e-14	Significant difference in average price
3	Private room	Hotel room	0.060301	9.519286e-01	No significant difference in average price
4	Private room	Shared room	8.708035	3.948124e-18	Significant difference in average price
5	Hotel room	Shared room	4.387780	1.322175e-05	Significant difference in average price

This table shows t-test results **comparing average prices** between room types. **Significant price differences** were found between "**Entire home/apt**" and "**Shared room**," "**Private room**" and "**Shared room**," and "**Hotel room**" and "**Shared room**." No significant differences were observed in the other comparisons.

Mann-Whitney U test

	room_type_1	room_type_2	t_statistic	p_value	result
0	Entire home/apt	Private room	30317205.5	2.564177e-75	Significant difference in median price
1	Entire home/apt	Hotel room	2795910.0	1.769848e-01	No significant difference in median price
2	Entire home/apt	Shared room	4161821.0	5.076694e-201	Significant difference in median price
3	Private room	Hotel room	1529918.0	3.117582e-14	Significant difference in median price
4	Private room	Shared room	2562637.0	1.275102e-154	Significant difference in median price
5	Hotel room	Shared room	292717.5	6.966201e-102	Significant difference in median price

This table shows the results of tests **comparing median prices between room types**. **No significant difference** was observed between "**Entire home/apt**" and "**Hotel room**." For the other comparisons, significant median price differences were found.

Purwadhika
Digital Technology School

Correlation Testing

	column	spearman_correlation	p_value	result	correlation
0	id	0.077475	1.534336e-22	Has correlation	No correlation
1	host_id	0.025381	1.393711e-03	Has correlation	No correlation
2	latitude	-0.047731	1.825072e-09	Has correlation	No correlation
3	longitude	-0.069075	3.122815e-18	Has correlation	No correlation
4	minimum_nights	-0.102312	3.709713e-38	Has correlation	Weak correlation
5	number_of_reviews	-0.020784	8.872350e-03	Has correlation	No correlation
6	reviews_per_month	0.017104	3.128004e-02	Has correlation	No correlation
7	calculated_host_listings_count	0.084398	1.848012e-26	Has correlation	No correlation
8	availability_365	-0.000729	9.268605e-01	No correlation	No correlation
9	number_of_reviews_ltm	0.043644	3.854827e-08	Has correlation	No correlation
10	bedroom	0.395101	1.426715e-138	Has correlation	Moderate correlation
11	distance_from_district_center	-0.003017	7.040382e-01	No correlation	No correlation
12	distance_from_sub_district_center	-0.072255	8.335042e-20	Has correlation	No correlation

This table shows the **Spearman correlation** results for the **price column**. It lists the correlation values and their corresponding p-values, indicating whether there is a correlation with price. A "Weak Negative correlation" is found for the "**minimum_nights**" column, and a "**Moderate Positive correlation**" is found for the "**bedroom**" column. The rest of the columns either have "No correlation" or "Has correlation" but show very weak or no significant relationship with price.

Purwadhika
Digital Technology School

Data Visualization

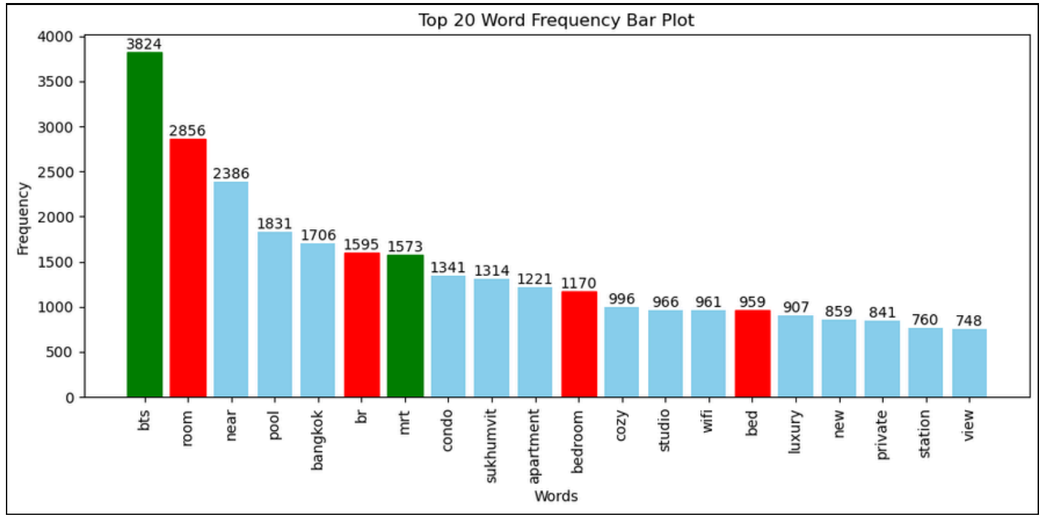
Data visualization is the graphical representation of data and information using charts, graphs, and other visual tools. It helps to communicate complex data patterns, trends, and insights in an easily understandable format, making it easier to interpret and analyze large datasets. By visualizing data, it becomes easier to identify relationships, patterns, and anomalies that might be missed in raw data.

Step for data visualization are:

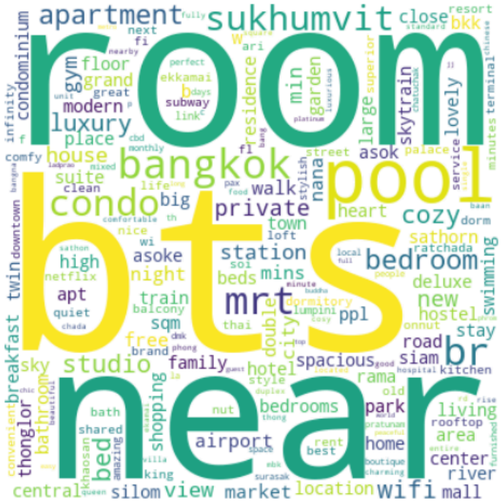
1. Univariate Analysis, and
2. Bivariate Analysis

Name Column

Word Count



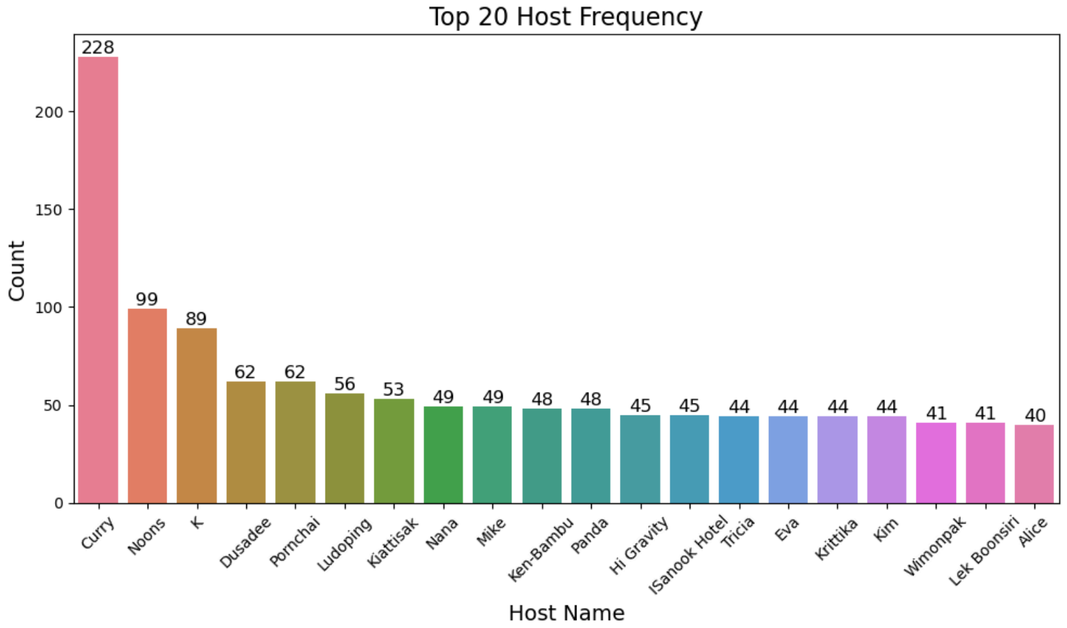
Word Cloud



This visualization shows the word count, from which we can identify two patterns to **extract as new features**: "**bedroom**," which contains terms like "room," "br," "bedroom," "bed," etc., and "**station**," which includes terms like "bts" and "mrt."

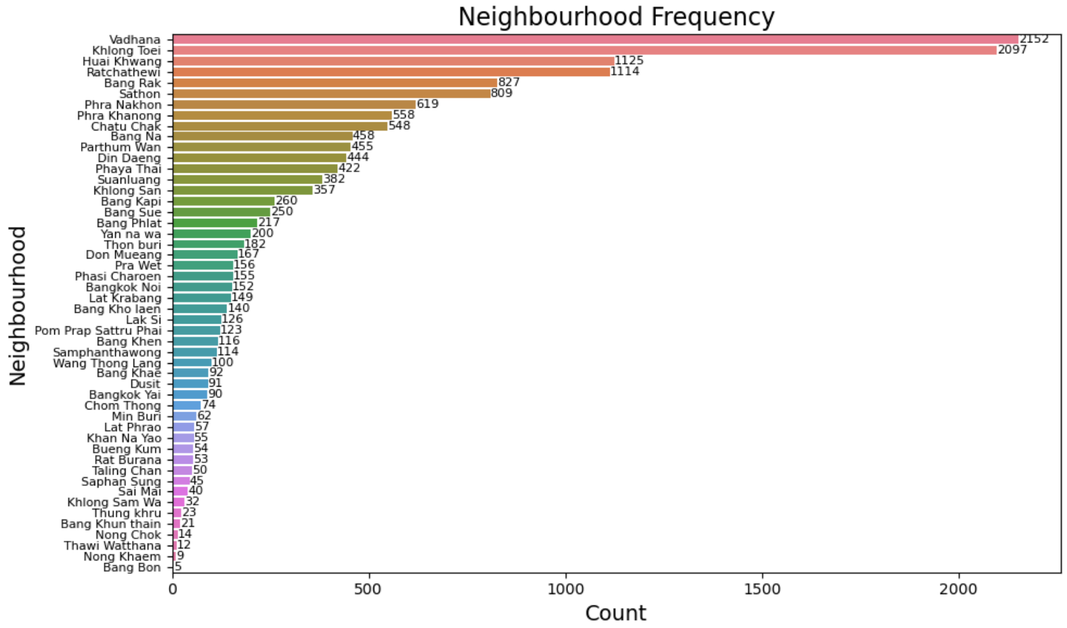
Host and Neighbourhood Column

Host



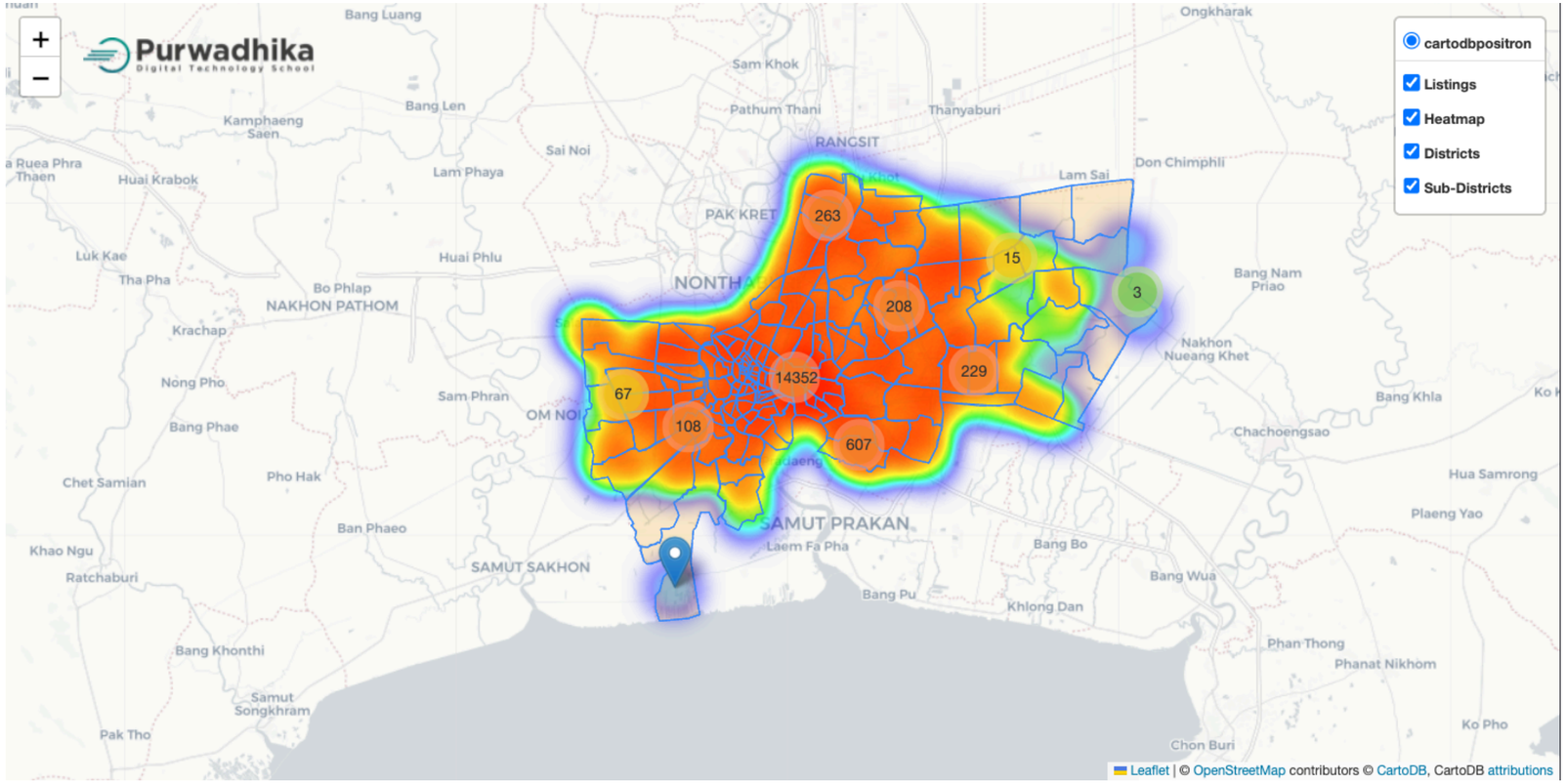
"**Curry**" stands out as the **top host with 228 listings**, followed by "**Noons**" and "**K**" with significantly fewer listings, indicating a concentration of properties among a few hosts

Neighbourhood



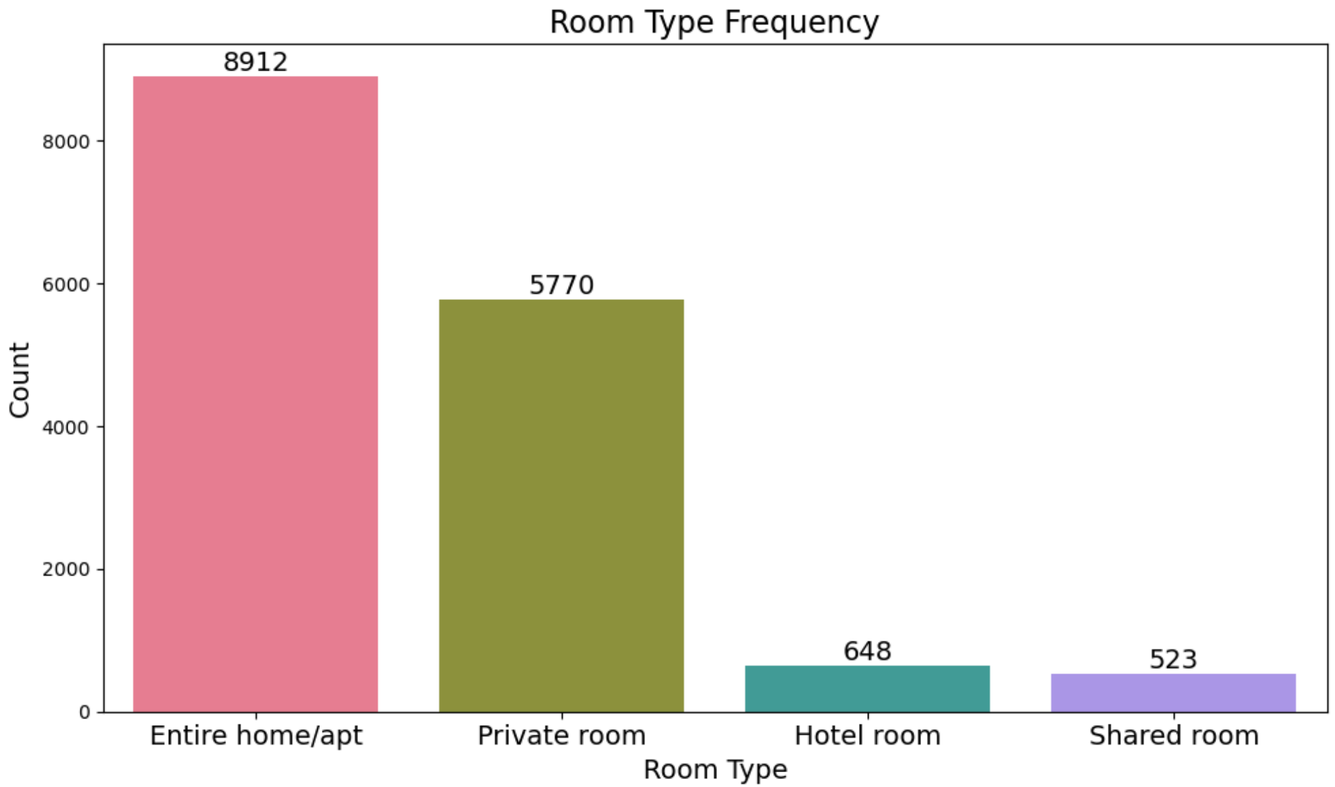
"**Watthana**" and "**Khlong Toei**" **dominate with over 2,000 listings** each, suggesting high demand in these areas, while other neighborhoods see a gradual decline in listings, reflecting lower popularity.

Geometry (Longitude & Latitude)



This heatmap indicates a high density of **listings concentrated in central Bangkok**, especially **around the district with 14,352 listings**. The intensity decreases as we move toward the outer districts, with significantly fewer listings on the outskirts. This pattern suggests that central areas are more popular for rentals, likely due to proximity to amenities and attractions.

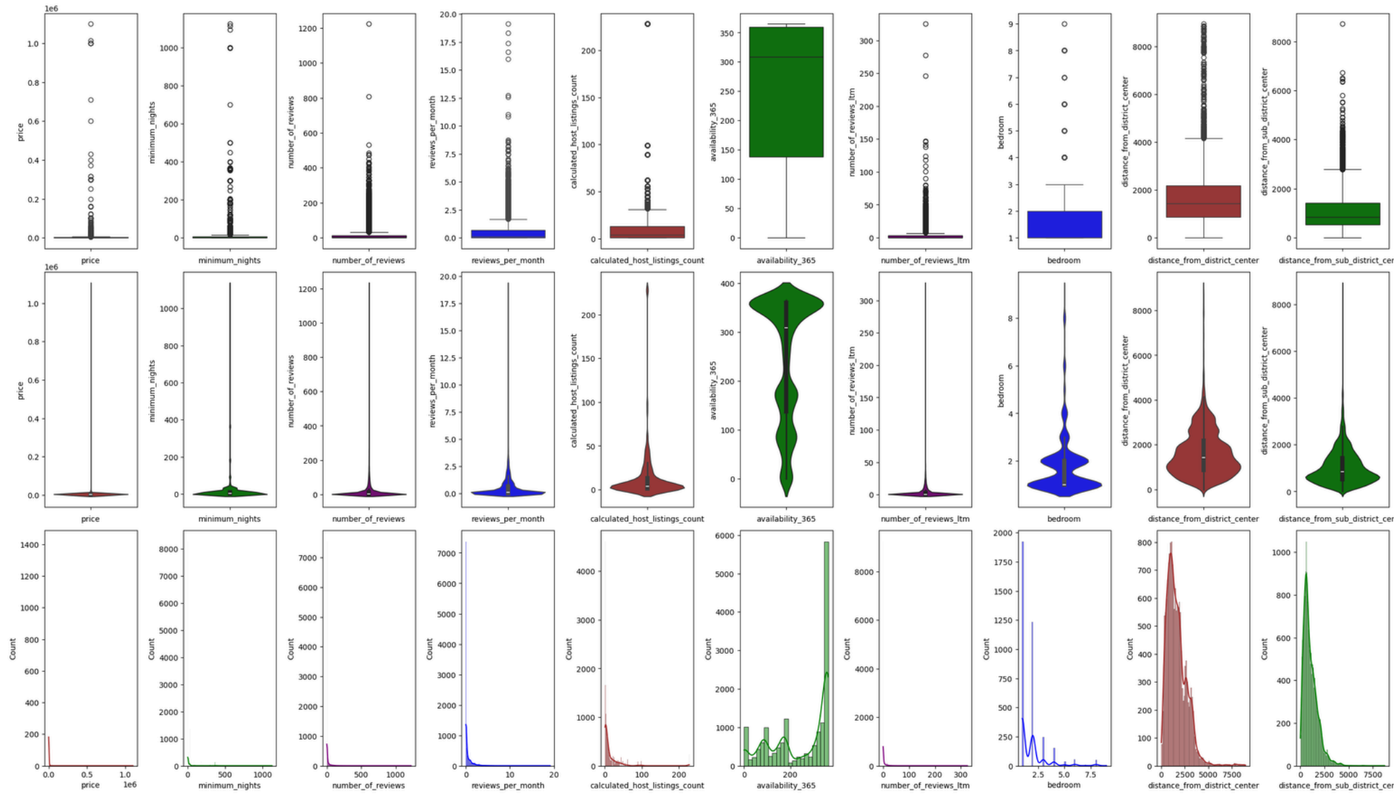
Room Type



"**Entire home/apt**" **is the most popular room type, with 8,912 listings**, followed by **"Private room" with 5,770 listings**. "Hotel room" and "Shared room" are much less common, with only 648 and 523 listings, respectively. This suggests that travelers prefer more private accommodations, with entire homes or apartments being the top choice.

Purwadhika
Digital Technology School

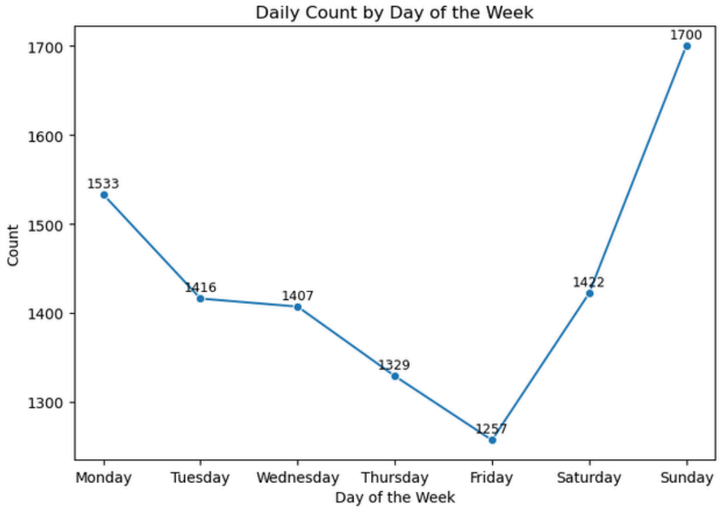
Numeric Dtype

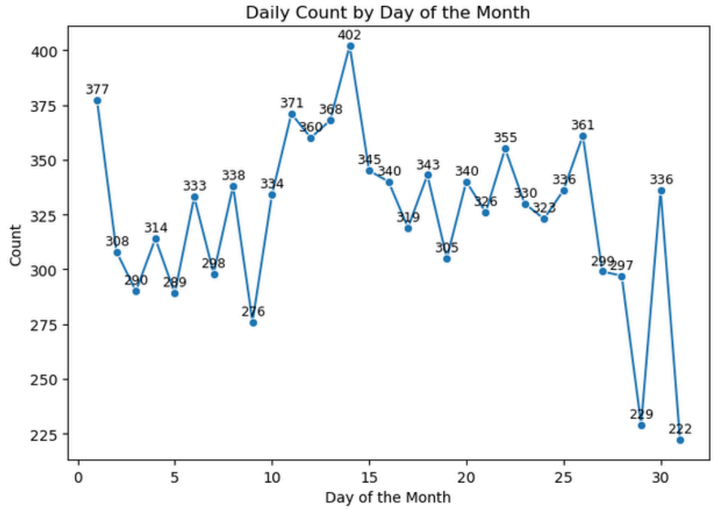


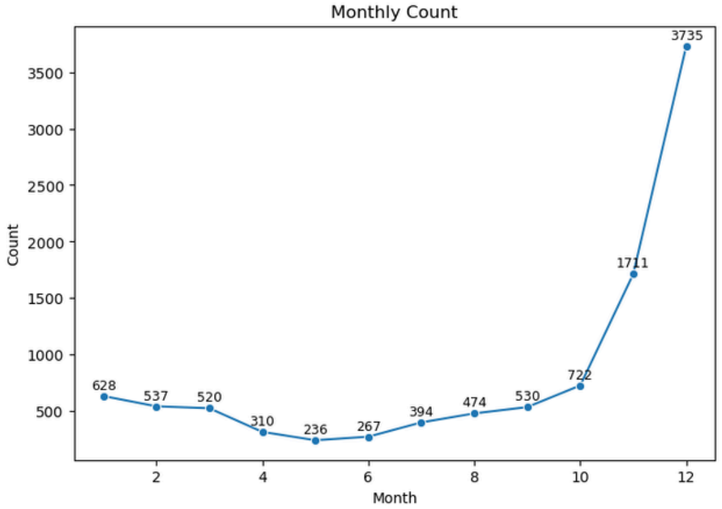

```
'price' has 1403 or 8.85% outliers
'minimum_nights' has 3168 or 19.98% outliers
'number_of_reviews' has 2240 or 14.13% outliers
'reviews_per_month' has 1471 or 9.28% outliers
'calculated_host_listings_count' has 1832 or 11.56% outliers
'availability_365' has 0 or 0.00% outliers
'number_of_reviews_ltm' has 2219 or 14.00% outliers
'bedroom' has 299 or 1.89% outliers
'distance_from_district_center' has 291 or 1.84% outliers
'distance_from_sub_district_center' has 466 or 2.94% outliers
```

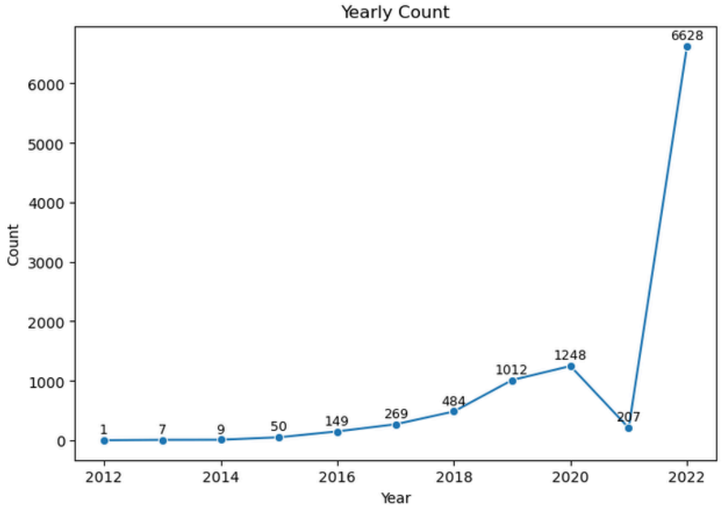
All numeric columns in this dataframe **are not normally distributed**. The "availability_365" column appears roughly normal in the boxplot but shows a non-normal distribution in the violin plot and histogram. Several columns have outliers, with "minimum_nights" having the highest percentage at 19.98%, followed by "number_of_reviews" (14.13%) and "number_of_reviews_ltm" (14.00%). Other columns like "bedroom" (1.89%) and "distance_from_sub_district_center" (2.94%) also contain outliers, while "availability_365" has no outliers.

Last Review



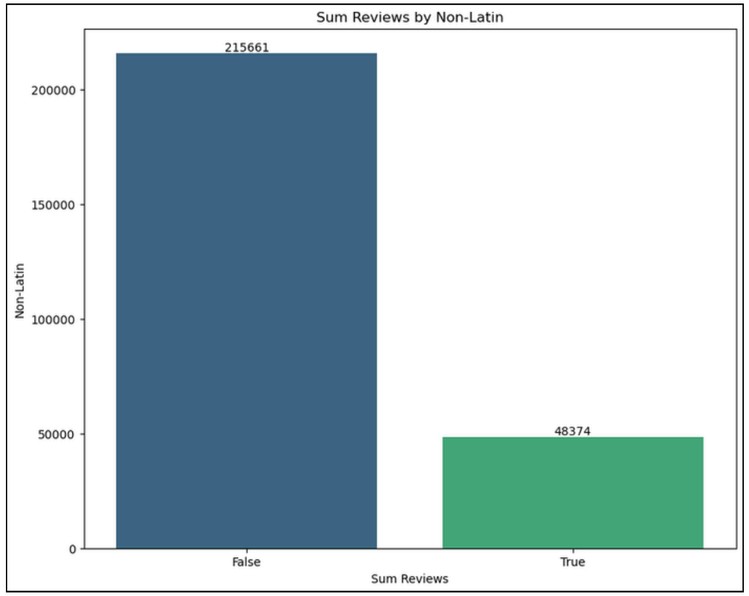


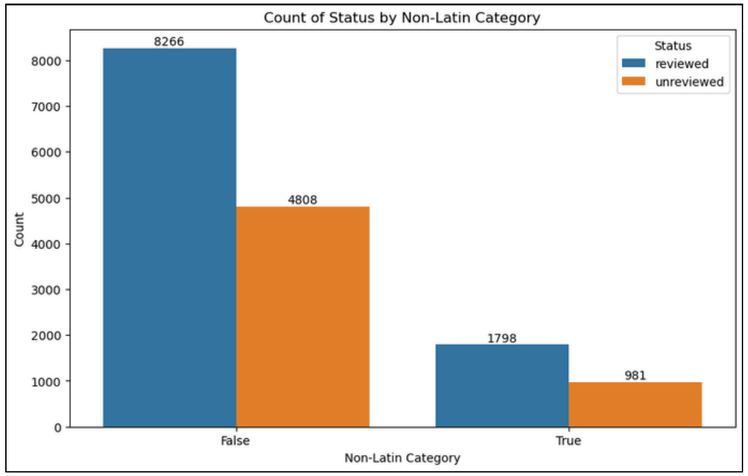


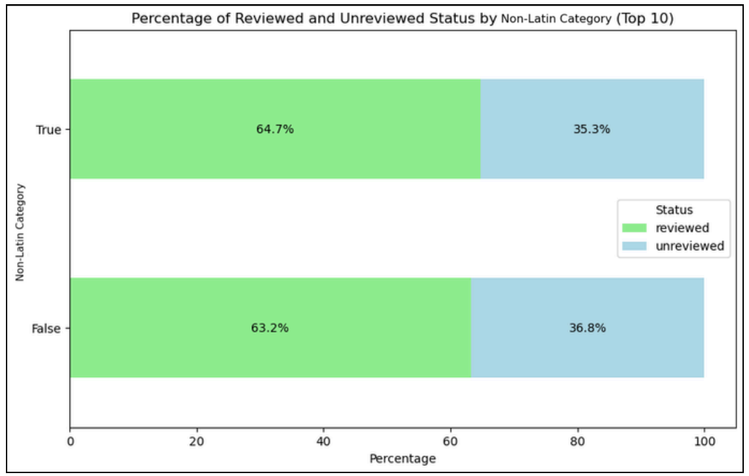


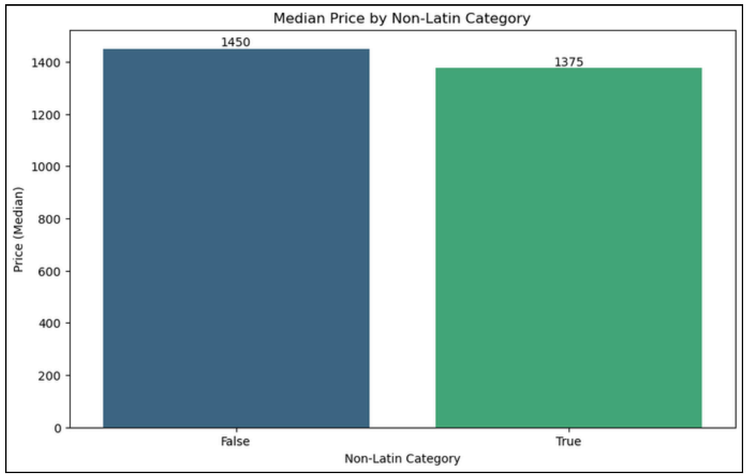
activity **peaks on Sundays** and is **lowest on Fridays**, with noticeable end-of-month spikes. **December has a sharp increase in activity,** suggesting seasonality, while **2022 shows a major surge**, indicating recent growth in data collection or activity levels.

Name Column



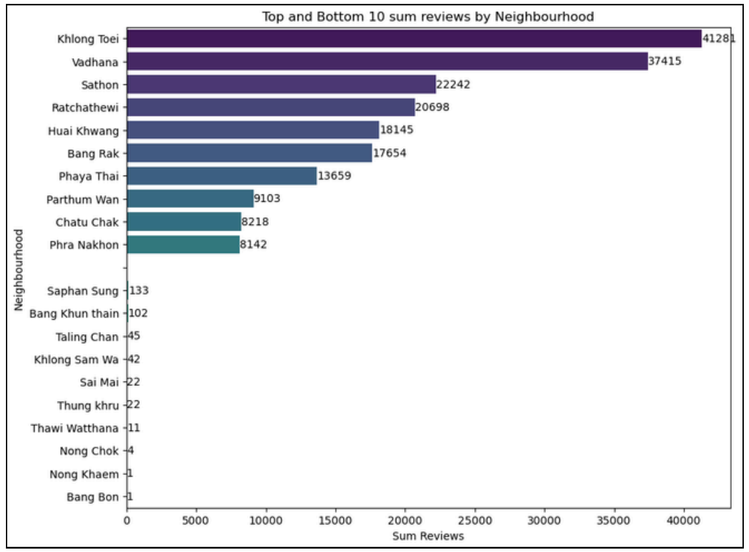


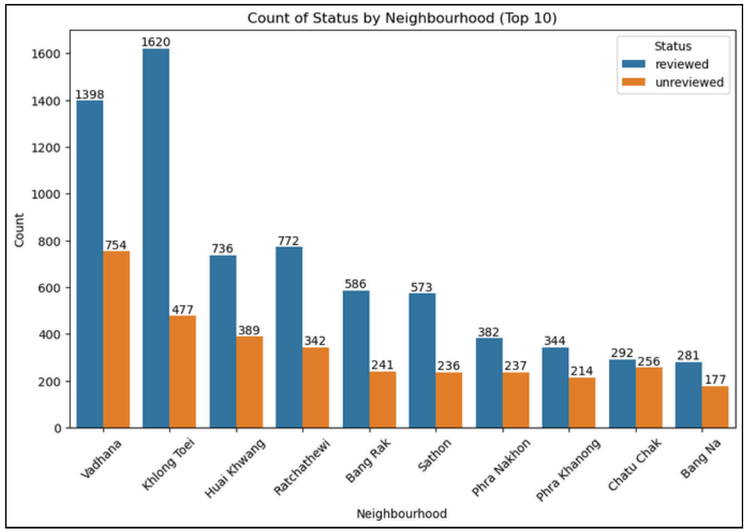


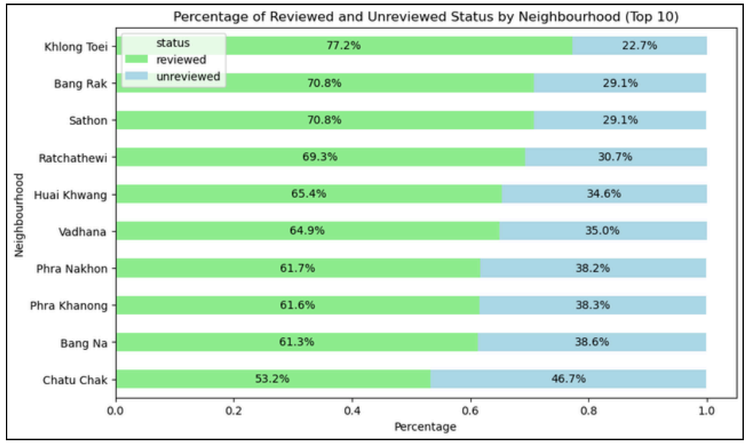


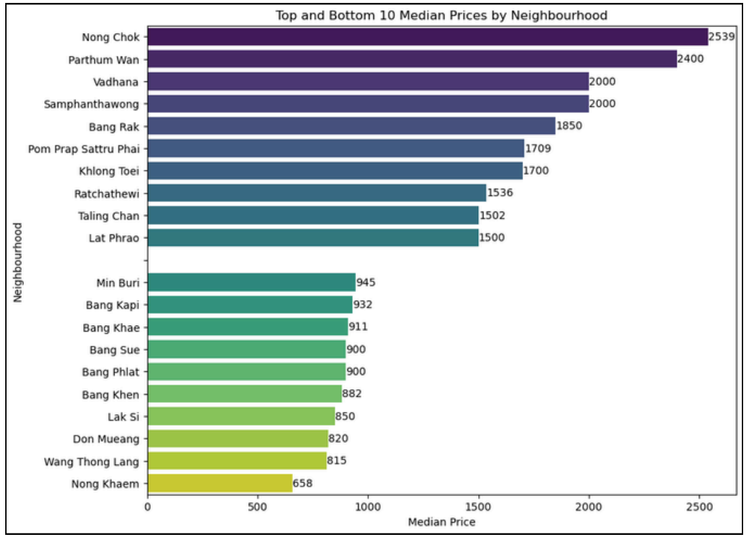
Non-Latin reviews are much fewer (48,374) than **Latin-based ones (215,661)**, with more unreviewed listings. Reviewed status is evenly split by Non-Latin, and median prices are similar, with Latin-based **slightly higher (1,450 vs. 1,375)**.

Neighbourhood Column



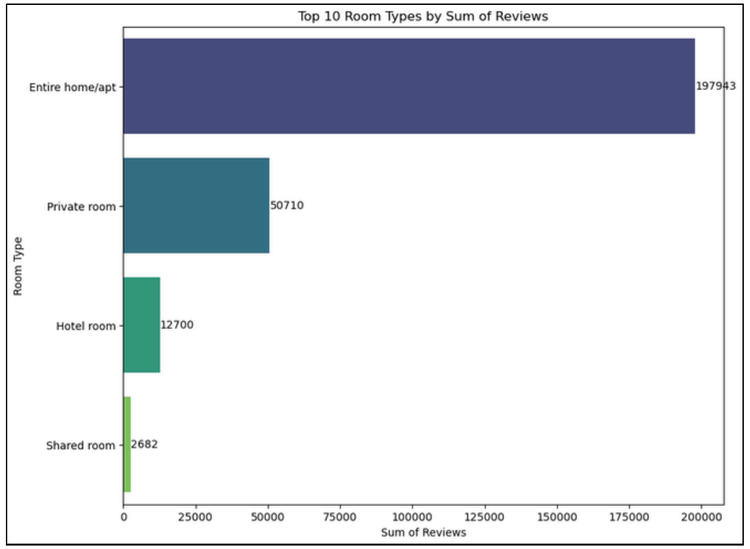


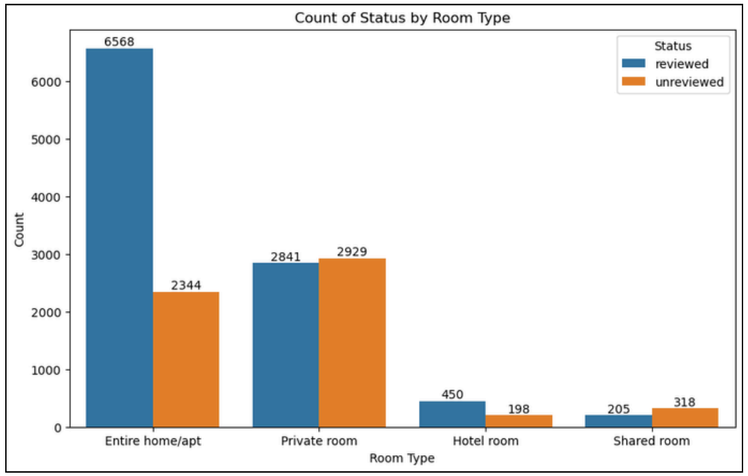


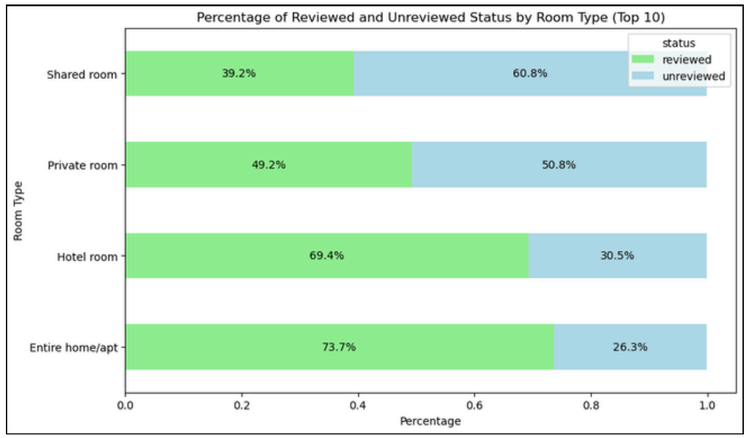


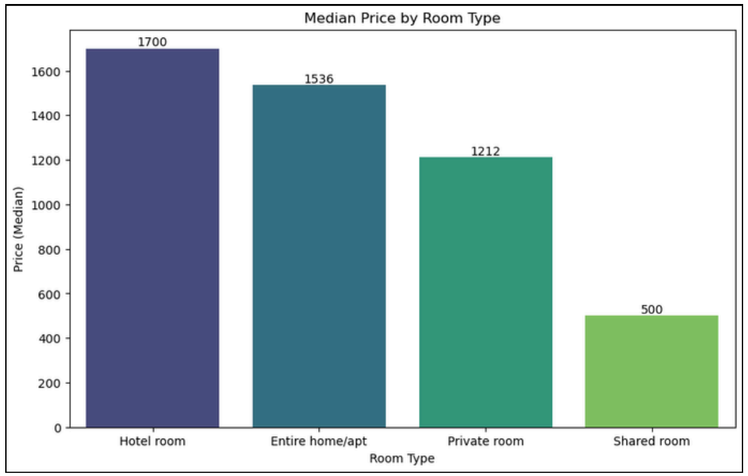
Khlong Toei and Watthana lead in **reviews (41,287 and 37,415)**, with high engagement, while **Nong Chok is the cheapest (658)** and **Nong Chok the priciest (2,539)**, highlighting varied appeal and market segments across neighborhoods.

Room Type Column



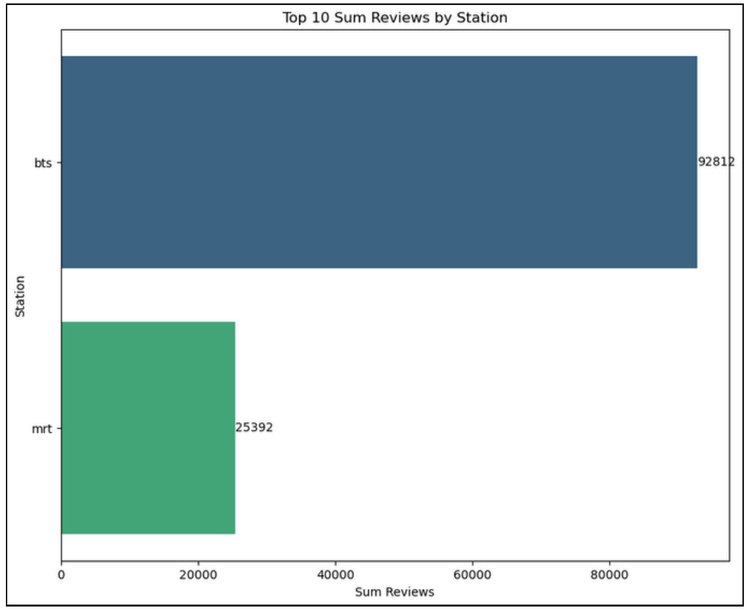


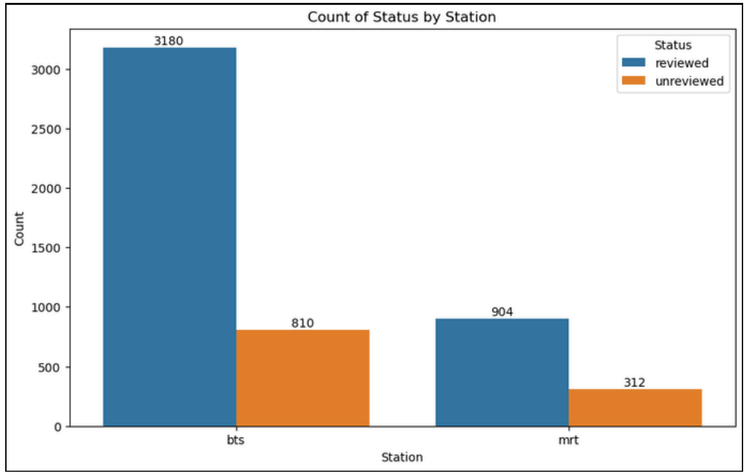


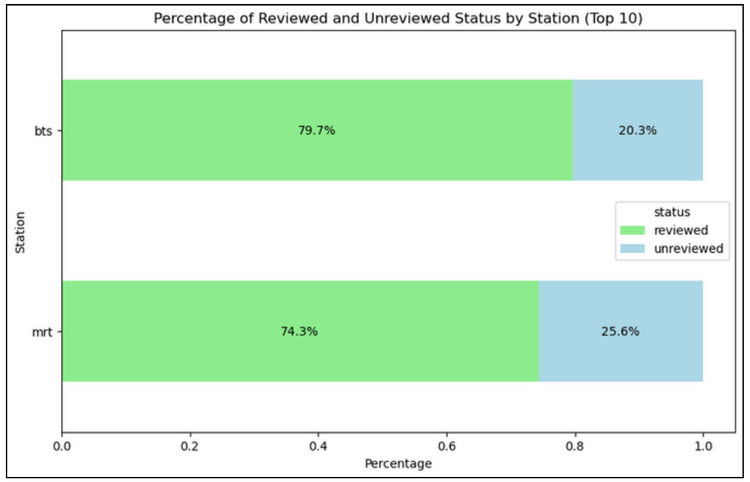


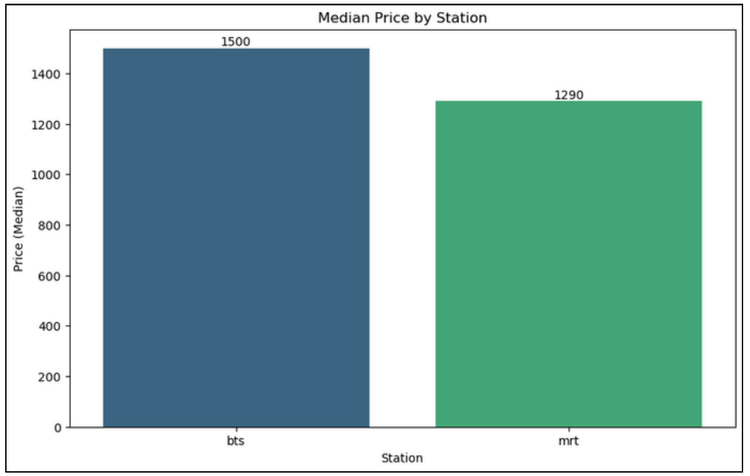
"**Entire home/apt**" **dominates in reviews (197,913)** and **has a high review rate (73.7%)**, while "**Shared room**" **has the lowest at 39.2%**. "Hotel room" is the priciest (1,700), with "Shared room" as the cheapest (500), indicating varied cost-benefit across room types.

Station Column









BTS stations **have more reviews (92,812)** and a **higher review rate (79.7%)** than **MRT (25,392 and 74.3%)**, with fewer unreviewed statuses. **BTS** also **has a higher median price (1,500 vs. 1,290)**, suggesting perceived higher value.

Conclusion

- Concentration Among Hosts: A few hosts, particularly "Curry" with 228 listings, hold a large portion of the total listings, indicating a potential monopolization or reliance on large-scale hosts rather than individual hosts.

- High Demand in Central Bangkok: Central districts, particularly "Watthana" and "Khlong Toei," dominate in listing density, reflecting a strong demand in these neighborhoods likely due to their proximity to key amenities, attractions, and transit stations. Listings and activity drop significantly as one moves away from central areas, indicating lower demand in the outskirts.

- Preference for Entire Properties: "Entire home/apt" listings are highly preferred, with the highest count and review rate, suggesting that guests prioritize privacy and autonomy over shared or hotel room options.

- Seasonal and Weekly Trends: Activity peaks on Sundays and around the end of the month, with a notable surge in December, suggesting both weekly and seasonal booking patterns. This seasonality is crucial for planning pricing and promotions.

- Varied Market Appeal by Location: Neighborhoods show diverse appeal, with "Nong Chok" offering the most affordable options and "Pathum Wan" the priciest. Listings near BTS stations tend to attract more reviews and have a higher median price, indicating a perceived higher value and possibly greater accessibility.

Recommendation

- Encourage Diverse Hosting: Airbnb could encourage a broader distribution of hosts by providing incentives for individual or smaller-scale hosts. This approach may increase listing variety and appeal to different guest demographics.

- Focus Marketing on Central Districts: For continued growth, Airbnb should target "Watthana" and "Khlong Toei" in promotional efforts, as these neighborhoods already show strong demand. However, marketing campaigns could also highlight less popular neighborhoods to spread demand and showcase unique, lower-cost areas.

- Highlight Entire Homes and Privacy Options: Given the preference for entire homes/apartments, Airbnb could emphasize listings that offer privacy and exclusive access, especially during peak booking periods, to cater to this popular guest preference.

- Seasonal Pricing and Promotions: Hosts could use dynamic pricing strategies, increasing rates during December and end-of-month peak times to maximize revenue. Airbnb can support hosts with tools to automate this pricing adjustment based on historical activity data.

- Promote Listings Near BTS Stations: Since BTS-accessible listings have higher reviews and perceived value, Airbnb should emphasize these locations in their Bangkok listings and encourage hosts to highlight proximity to BTS in their profiles.

Purwadhika
Digital Technology School

Thank You!

brillian adhiyaksa

brillian.adhiyaksa@gmail.com

@brillian.adhiyaksa

@BrillianAK