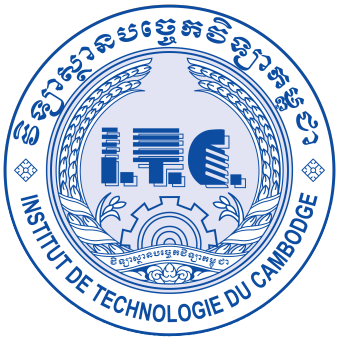
Institute of Technology of Cambodia

Department of Mathematics and Statistics



[Report Paper: Preprocessing on Tour Recommendation System]

by

Group 1:

Rith Chanthyda, ID:e20200612

Phun Sreypich, ID:e20200179

Phai Ratha, ID:e20200190

Kry Senghort, ID:e20200706

Mengheab Vathanak, ID:e20201145

Rithy Vira, ID:e20200978

A Project Report Submitted
in Partial Fulfilment of the requirements for
the Degree of Bachelor of
Data Science

Lacturer:

[Mr.CHAN Sophal]

May, 2023

Contents

Chapter 1

Introduction

1.1 Background

Tourism is a significant contributor to the global economy, and the tour industry has grown rapidly in recent years. With the rise of technology, many tour companies are using data-driven approaches to recommend personalized tours to users. However, creating an accurate tour recommendation system requires a thorough understanding of the data and effective data processing techniques.

1.2 Problem Statement

The tour recommendation system is designed to recommend tours to users based on their interests and preferences. However, the effectiveness of the system is largely dependent on the quality of the data used to train the model. Inaccurate, incomplete, and inconsistent data can lead to poor recommendations, resulting in low user satisfaction and decreased revenue for the tour company.

1.3 Main Objective

The main objective of this project is to develop a tour recommendation system that uses machine learning algorithms to recommend personalized tours to travelers. The system will use **Content-Based Filtering** and **Collaborative Filtering** techniques to recommend tours based on user preferences.The project will involve data cleaning, exploratory data analysis (EDA), feature engineering, and visualization to build a model that can accurately recommend tours to users.

1.3.1 Specific Objectives

The specific objectives of the study were:

- To perform data cleaning and data preprocessing to ensure data quality and accuracy.
- To conduct exploratory data analysis (EDA) to gain insights into the data and identify patterns and trends.
- To perform feature engineering to create a model that can accurately recommend tours to users.
- To visualize the data and model performance to facilitate interpretation and communication of the results

1.4 Scope of the study

The scope of this study is limited to developing a tour recommendation system that uses content-based filtering and collaborative filtering techniques. The system will be developed using a dataset of tours that includes information such as tour type, duration, price, location, and ratings. The study will focus on data cleaning, exploratory data analysis, feature engineering, and visualization techniques.

1.5 Significance of the study

The significance of this study lies in the development of a tour recommendation system that can enhance the travel experience for tourists. The system will provide personalized tour recommendations based on the traveler's preferences, resulting in a more efficient and enjoyable travel experience. Moreover, this study contributes to the field of recommendation systems by demonstrating the effectiveness of content-based filtering and collaborative filtering techniques in tour recommendation systems.

Chapter 2

Data Cleaning Process

2.1 Handling Missing Values

To handle missing values, we used the K-nearest neighbors (KNN) imputation method. First, we identified columns with missing values and calculated the percentage of missing values in each column. Next, we used KNN to impute the missing values based on the values of the nearest neighbors. This method was chosen because it can accurately estimate missing values based on the values of other similar observations in the dataset.

After applying the KNN imputation method to fill in the missing values, we compared the summary statistics table as shown in **Table 2.1** before and after the imputation to track down the effectiveness of the method. The results showed that the imputation successfully filled in all the missing values and improved the summary statistics table to an acceptable level, with minimal changes to the mean and standard deviation of the columns. Therefore, we can confidently proceed with the cleaned data for further analysis and modeling.

Original dataset summary statistics:

	Index	Number_of_Reviewer	Rating	Duration	Price
count	4785.000000	2095.000000	2095.000000	4785.000000	4352.000000
mean	2393.000000	18.020525	4.836993	20.350679	138.641540
std	1381.454849	72.065230	0.551581	39.070865	237.163299
min	1.000000	1.000000	1.000000	1.000000	5.000000
25%	1197.000000	1.000000	5.000000	5.000000	45.000000
50%	2393.000000	3.000000	5.000000	7.000000	74.100000
75%	3589.000000	11.000000	5.000000	10.000000	140.000000
max	4785.000000	1289.000000	5.000000	1182.000000	3665.000000

Imputed dataset summary statistics:

	Index	Number_of_Reviewer	Rating	Duration	Price
count	4785.000000	4785.000000	4785.000000	4785.000000	4785.000000
mean	2393.000000	10.680627	4.581818	20.350679	136.209864
std	1381.454849	48.424251	0.678732	39.070865	228.511838
min	1.000000	1.000000	1.000000	1.000000	5.000000
25%	1197.000000	1.400000	4.300000	5.000000	46.000000
50%	2393.000000	2.600000	5.000000	7.000000	75.000000
75%	3589.000000	6.800000	5.000000	10.000000	138.470000
max	4785.000000	1289.000000	5.000000	1182.000000	3665.000000

Table 2.1: Summariy Statistics table between Original data and Impute data

2.2 Identifying Outliers

To identify outliers, we used boxplots. We created a boxplot as shown in **Figure 2.1**, for each column in the dataset and examined any observations that fell outside the whiskers of the boxplot. Any observations that were considered outliers in datasets.

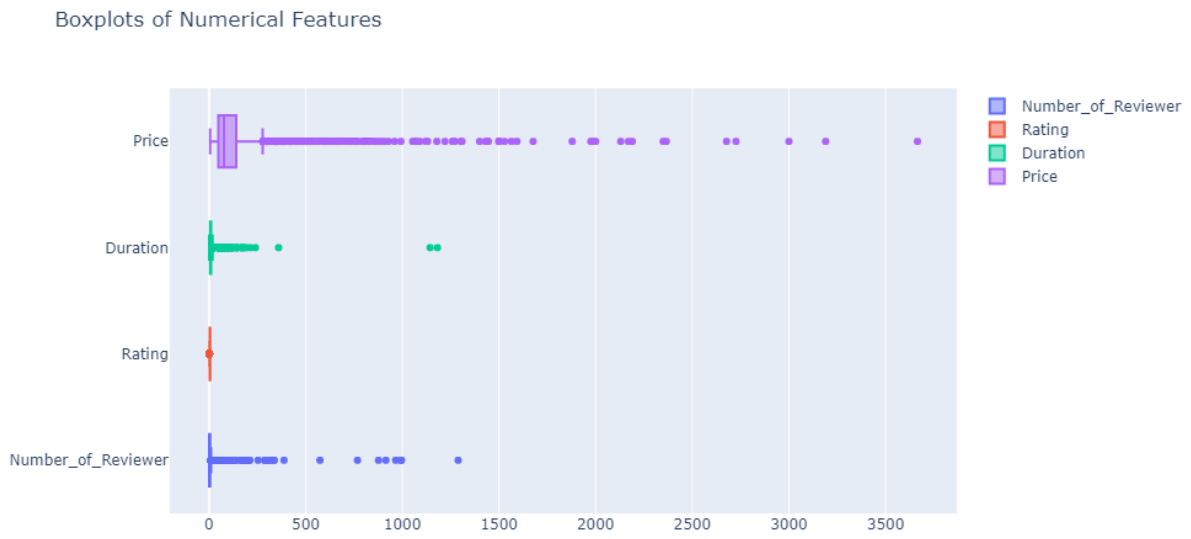


Figure 2.1: Summariy Statistics table between Original data and Impute data

Finding:

- Base on **Figure 2.1**, there are many high or low values in the Number of Reviewer column which may indicate highly popular or unpopular tours.
- There are also many high or low rating in the Rating column which may indicate the satification of the tours
- Base on Price and Duration column , there are also many high or low cost depend on duration.

Overall, it may not be necessary to drop outliers in our recommendation system as they could potentially reveal valuable patterns and insights. By including outliers in our analysis, we can gain a better understanding of the characteristics and preferences of our users, which could lead to better tour recommendations. Therefore, rather than simply removing outliers, we should carefully consider their impact on our analysis and take advantage of the valuable information they may provide.

2.3 Handling Duplicates

To handle duplicates, we first identified columns that contained unique identifiers for each tour, such as tour name and Price. Then, we used pandas library in Python to drop the duplicate rows based on these columns. This was done to ensure that each tour was only represented once in the dataset, and to avoid biasing the recommendation system towards certain tours.

Chapter 3

Data Exploration & Data Visualization

EDA is a crucial step in the development of a tour recommendation system. It helps in identifying patterns, trends, and insights that can be used to develop a more accurate recommendation system. In this project, EDA involved analyzing the distribution of variables, detecting correlations between variables, and identifying features that were most predictive of user preferences. Visualization techniques such as scatter plots, histograms, and heatmaps were used to help identify these patterns and trends.

3.1 Analyzing the distribution of variables

Analyzing the distribution of variables is crucial for building an effective tour recommendation system. By understanding the distribution of important variables like tour type, rating, and number of ratings, we can gain insights into user preferences and identify any patterns or trends that may exist in the data. In this section, we will explore the distribution of variables and discuss our findings.

3.1.1 Rating distribution

Base on **Figure 3.1**, the histogram shows that the distribution of ratings is positively skewed, indicating that the majority of the tours have a higher rating.

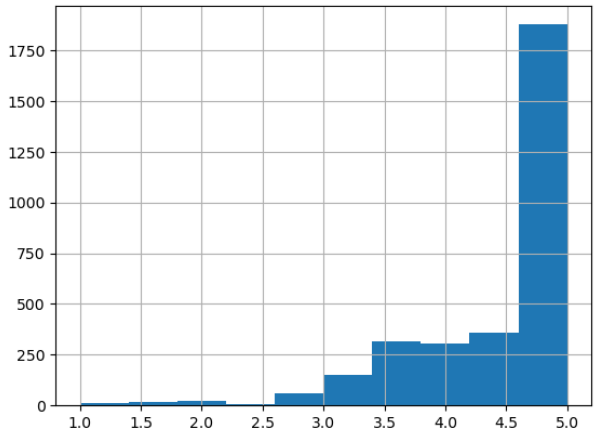


Figure 3.1: Histogram of Rating

3.1.2 Tour Type distribution

After analyzing the distribution of variables in , the following insights were found:

- Bus Tours have the highest number of packages with a total of 911 products, covering 35.19
- Full-day Tours is the second most popular tour type with 736 products, covering 28.43
- Circuit and some other tour types have only one package, covering 0.04

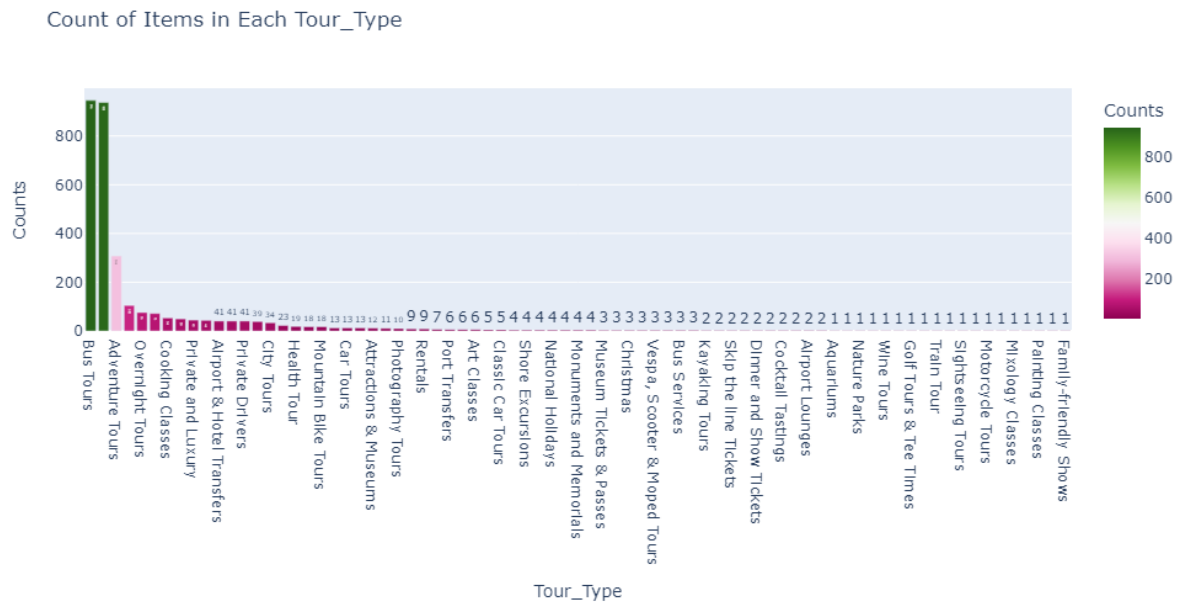


Figure 3.2: Bar Chart of Tour Type

3.2 Popularity Analysis

In this section, we analyze popular **Tour Type** base on **Number of Reviewe**r. And Base on **Figure 3.3**, the following insights were found:

- **Full-day Tours** have 11.204K reviewers. It covers 39.62% of the total reviewer.
- Next, the best tour is **Bus Tours**, which has 5931 reviewers. It covers 20.97% of the total reviewer.
- **Mountain Bike Tours** has the lowest reviewer among the top 10.

Base on this we can assume that this tour are the most solve packages since there are many reviwer, we can also assume that this type of ture are popular among the user.

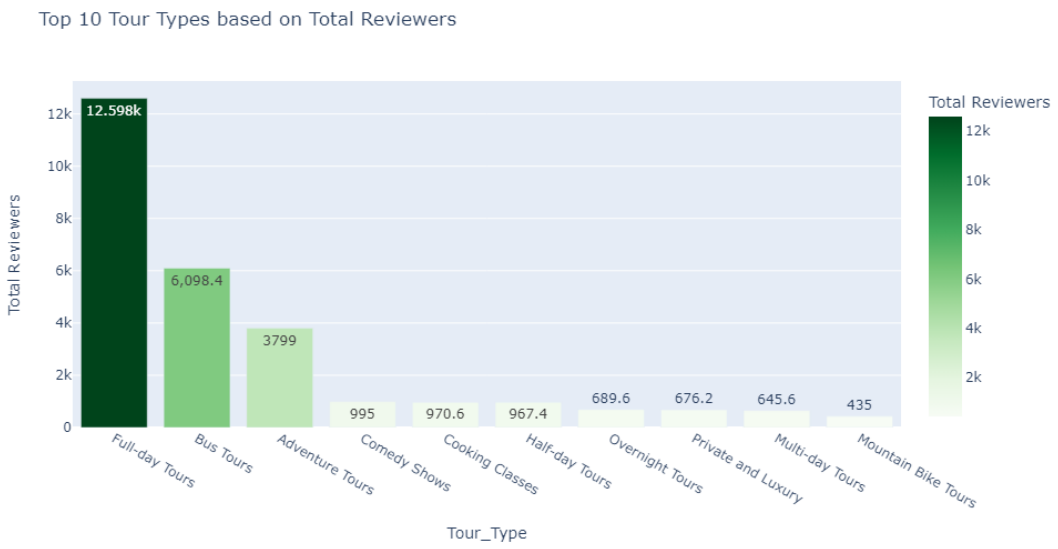


Figure 3.3: Bar Chart of Tour Type

3.3 Correlation Analysis

Figure 3.4 displays a scatterplot matrix, which shows the relationship between each variable in the dataset. It is observed that there are only a few variables that exhibit a strong correlation with each other, with the most notable relationship being between **Price** and **Duration**. Other variables do not seem to have a significant correlation, suggesting that they are independent of each other and can provide unique information for the recommendation system.

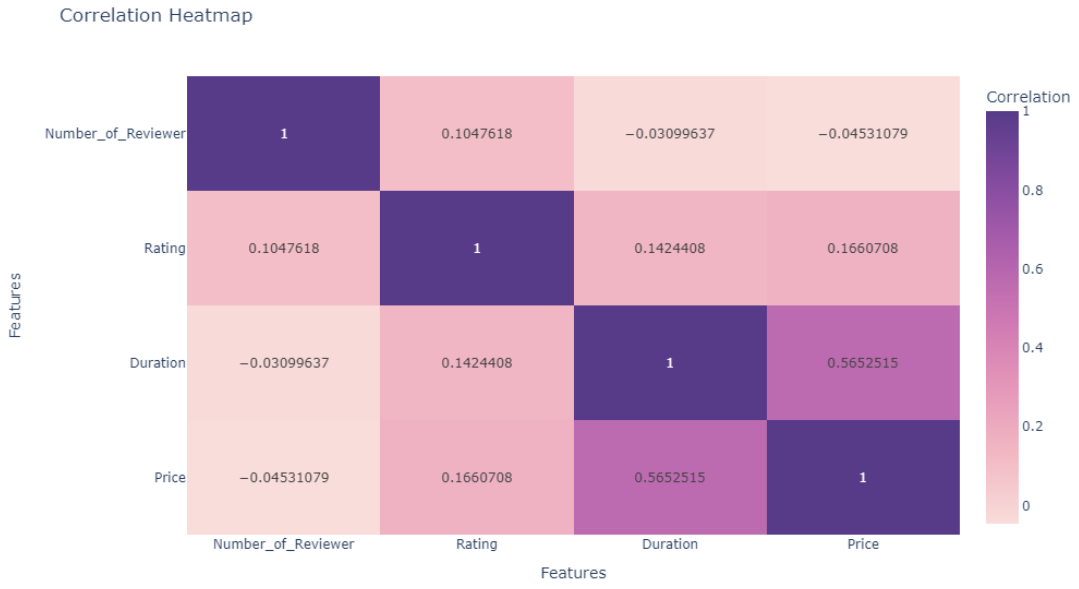


Figure 3.4: Heatmap of Numerical Variables

Based on **Figure 3.5**, there appears to be a slight positive correlation between tour duration and tour price, as the trendline slopes upwards. Additionally, the plot shows that there are a variety of ratings for the tours, as seen by the different colors. There does not appear to be any distinct patterns based on rating, as there are highly rated tours at various price and duration levels. The correlation coefficient of 0.56 also supports the idea of a positive correlation, albeit a moderate one.

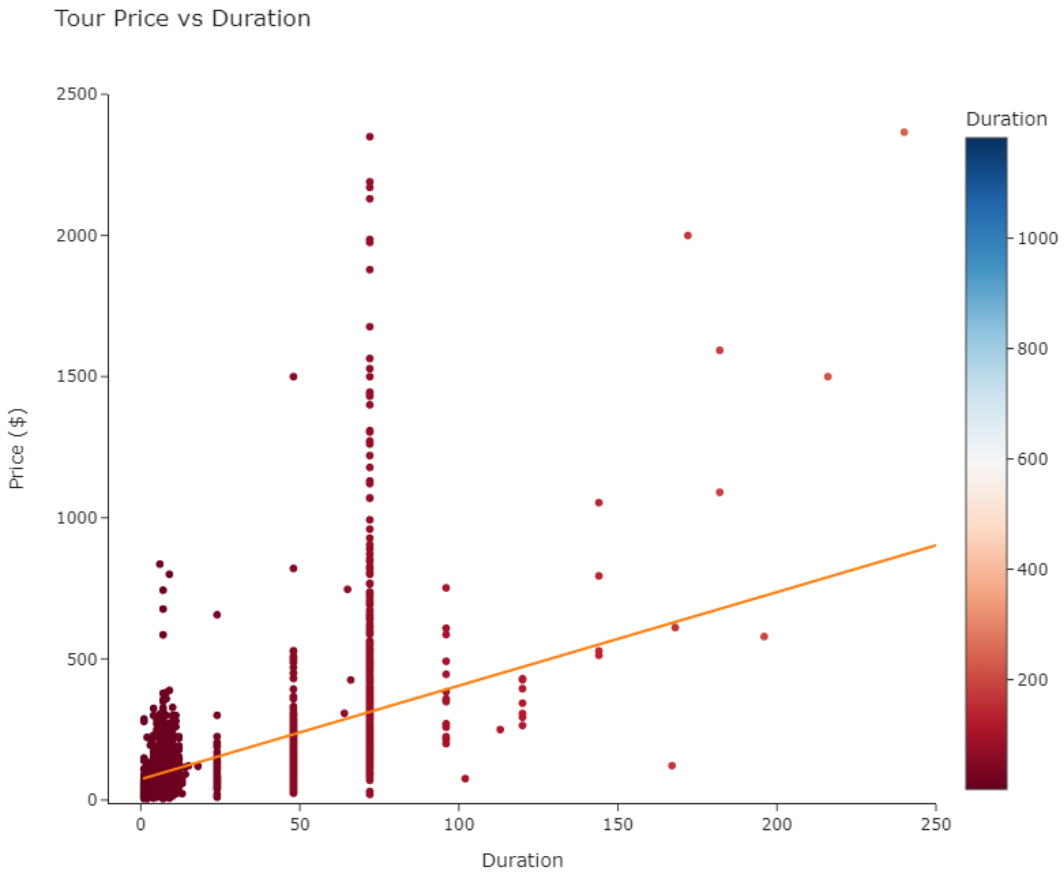


Figure 3.5: Scatter Plot between **Price** and **Duration**

Chapter 4

Feature Engineering

In order to build an effective tour recommendation system, feature engineering is a crucial step. Feature engineering involves selecting and transforming the relevant features in the dataset to improve the accuracy of the model. In this section, we will discuss the features we used for our tour recommendation system, why we selected them, and the data transformations we applied.

4.1 Text Preprocessing

It should be noted that for our category variables, we will only use the **Tour_Name**, **Tour_Type**, **Highlight**, and **Location** columns to build the model. To transform the text data into numerical format, we will use CountVectorizer, which converts text data into a matrix of token counts. By analyzing the feature distributions and applying the necessary transformations, we can improve the accuracy of our tour recommendation system.

4.1.1 What is CountVectorizer, and how does it work?

CountVectorizer is a text preprocessing technique used to transform textual data into numerical vectors. It works by converting a collection of text documents to a matrix of token counts, where each row represents a document and each column represents a unique word in the corpus. The CountVectorizer object first tokenizes the text into individual words or tokens and then counts the frequency of each token in each document. The resulting matrix can be used as input to machine learning models.

For example, suppose we have a collection of three documents:

1. "The quick brown fox jumps over the lazy dog."
2. "The dog chased the fox, but the fox was too quick."
3. "The lazy dog slept in the sun, but the quick brown fox hid behind a tree."

To use CountVectorizer, we first need to tokenize the text by splitting it into individual words and removing any punctuation or stop words. The resulting vocabulary might look something like this:

['brown', 'chased', 'dog', 'fox', 'hid', 'jumps', 'lazy', 'quick', 'slept', 'sun', 'tree']

Next, we count the frequency of each word in each document and create a matrix with rows corresponding to the documents and columns corresponding to the words in the vocabulary. The resulting matrix might look something like this:

	brown	chased	dog	fox	hid	jumps	lazy	quick	slept	sun
height1	1	1	0	1	1	0	1	1	1	0
height2	0	1	1	2	0	0	0	1	1	0
height3	1	1	0	1	1	1	0	1	1	1

In this way, each document is represented by a numerical vector that captures the frequency of each word in the vocabulary. This matrix can be used as input to machine learning algorithms for tasks such as text classification or clustering., change exaple to relate to tour

4.1.2 Clean String Columns

Before converting the data into a word vector, we need to clean it.

```
import re
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity

df2 = df.copy()
rmv_spc = lambda a: a.strip()
get_list = lambda a: list(map(rmv_spc, re.split('&|,|\*|n', a)))
for col in ['Tour_Name', 'Tour_Type', 'Highlight', 'Location']:
    df2[col] = df2[col].apply(get_list)

def cleaner(x):
    if isinstance(x, list):
        return [str.lower(i.replace(" ", "")) for i in x]
    else:
        if isinstance(x, str):
            return str.lower(x.replace(" ", ""))
        else:
            return ''

for col in ['Tour_Name', 'Tour_Type', 'Highlight']:
    df2[col] = df2[col].apply(cleaner)

def couple(x):
    return ' '.join(x['Tour_Name']) + ' ' + ' '.join(x['Tour_Type']) + ' ' + ' '.join(x['Highlight'])+ ' '.join(x['Location'])

df2['classification_features'] = df2.apply(couple, axis=1)
```

Finally, we can apply **Scikit-Learn's CountVectorizer()** on the combined text data.The variable "vectorized" is a sparse matrix with a numeric representation of the strings we extracted.

```
In [94]: count = CountVectorizer(stop_words='english')
         count_matrix = count.fit_transform(df2['classification_features'])
         df2['classification_features']

Out[94]: 0       phare:thecambodia circusshowi siemreap comedys...
         1       khmergourmetcooki gclass cooki gclasses i -per...
         2       kohker be gmealeafull-dayjoi -i tour full-dayt...
         3       kampotdaytour"bokornatio alpark" adve turetour...
         4       bikethesiemreapcou trysidewithlocalexpert adve...
                                       ...
         4778    ph omkule tour waterfalls 1000li gariver recli...
         4780    kohkhebe gmealealesscrowdedprivatetour museumt...
         4781    speciala gkorsu rise su settour full-daytours ...
         4782    mostamazi ga gkortour adve turetours guidedtou...
         4783    amazi gkohkera dbe gmealeatour full-daytours e...
         Name: classification_features, Length: 3128, dtype: object
```

Chapter 5

Results

5.1 Results

Based on the analysis of the tour recommendation system, it was observed that the majority of tours had higher ratings, with a positively skewed distribution. This suggests that users were generally satisfied with the tours and rated them positively.

Further analysis revealed that **bus tours had the highest number of packages, covering 35.19% of the total packages**, followed by **full-day tours covering 28.43%** of the total packages. Among the top 10 tours, **mountain bike tours had the lowest reviewer count**, indicating that they were less popular among users.

Additionally, **full-day tours were found to be the most popular among users, with 11,204 reviewers, covering 39.62% of the total reviewers**. **Bus tours had 5,931 reviewers, covering 20.97%** of the total reviewers. This information can be used to tailor the recommendation system to suggest more full-day tours, given their popularity among users.

Furthermore, correlation analysis revealed a **slight positive correlation of 0.56 between tour duration and price**, indicating that longer tours tend to be more expensive. However, other variables did not show any significant correlation, which suggests that they can provide unique information for the recommendation system. This insight can be leveraged to design a recommendation system that takes into account various factors beyond just tour duration and price.

Overall, these findings provide valuable insights into the preferences of users and the characteristics of the tours in the system. By leveraging these insights, the recommendation system can be designed to suggest tours that are more likely to be preferred by users, thus improving user satisfaction and increasing revenue for the tour providers.

Chapter 6

Conclusion

6.1 Conclusion

The tour recommendation system can make use of the positively skewed distribution of ratings to recommend tours with higher ratings. The most popular tour types, **full-day and bus tours**, can also be recommended based on the number of reviewers they have. However, there is a need to further investigate the popularity of mountain bike tours. The slight positive correlation of **0.56** between tour duration and price provides an opportunity for the recommendation system to recommend tours with higher durations and prices based on the user's preferences. The system can benefit from feature engineering techniques such as **CountVectorizer** to transform text data into numerical vectors to improve the accuracy of the model. Future work can involve incorporating other variables to the model to provide more accurate recommendations.