Data-Driven Insights into Tourist Sentiments: A Study on Disneyland Visitor Reviews

Xiao Zhang
23015670
05/12/2023

Introduction

The project consisted of a comprehensive analysis of the review data for Disneyland, starting with loading a CSV file called DisneylandReviews.csv through Pandas and decoding it using cp1252 encoding. Rows containing 'missing' values were removed. Removed duplicate reviews. Sentiment analysis of the reviews was performed using NLTK's SentimentIntensityAnalyzer. Defined a function that assigns 'Positive', 'Negative' or 'Neutral' labels based on sentiment scores. A new column 'sentiment' was created to store the sentiment labels of the comments. And a bar chart showing the distribution of comment sentiment was drawn using Seaborn, and topic modelling (Latent Dirichlet Allocation) was performed using Gensim. A lexicon and corpus were created and 5 topics were extracted using LDA modelling. It was also interesting to use the WordCloud library to generate a word cloud of the comments, utilising the Disney Castle logo only for the mask map. Finally a map visualisation of the distribution of comments by country/region was produced using Plotly Express

Background

Mr Walt Disney said, "If you can dream it, you can do it. Remember that this whole thing started with a dream and a mouse." As the world's most successful IP manufacturing giant, Disney carries the childhood and beauty dreams of generations. Disneyland is one of the most recognisable and popular theme parks in the world, and its success is not only reflected in its financial prosperity, but also in its far-reaching impact on global culture. Behind this success are several intertwined factors that together make up the unique appeal of Disneyland. This dataset that I used is from Kaggle (https://www.kaggle.com/datasets/arushchillar/disneyland-reviews), which focuses on Reviews and Ratings of 3 Disneyland branches - California, Hong Kong and Paris. Which contains the Review_ID: unique id given to each review, Rating: ranging from 1 (unsatisfied) to 5 (satisfied), Year_Month: when the reviewer visited the theme park, Reviewer_Location: country of origin of visitor, Review_Text: comments made by visitor, Disneyland_Branch: location of Disneyland Park.

Firstly, by gaining insight into what visitors think about the parks, we are able to gain an intuitive understanding of the visitor experience. By mining the reviews for keywords and sentiments, we were able to identify the aspects of the park that guests cared about and appreciated the most, which provided the park with direction for improvement and enhancement.

Second, the project aims to provide strong support for improving the park experience. By analysing the distribution of ratings and sentiment analysis, we are able to determine how visitors feel about the park as a whole, which in turn allows us to identify priorities for improvement. For example, if negative reviews are focused on a particular aspect, such as service or facilities, the park management team can target improvements to enhance guest satisfaction.

Finally, by analysing park reviews in depth, we can provide recommendations for park operations. This includes understanding which specific aspects have been warmly received by guests and which areas may need more attention. Such insights can be instructive in developing long-term business strategies, launching new attractions or improving specific service areas.

Method

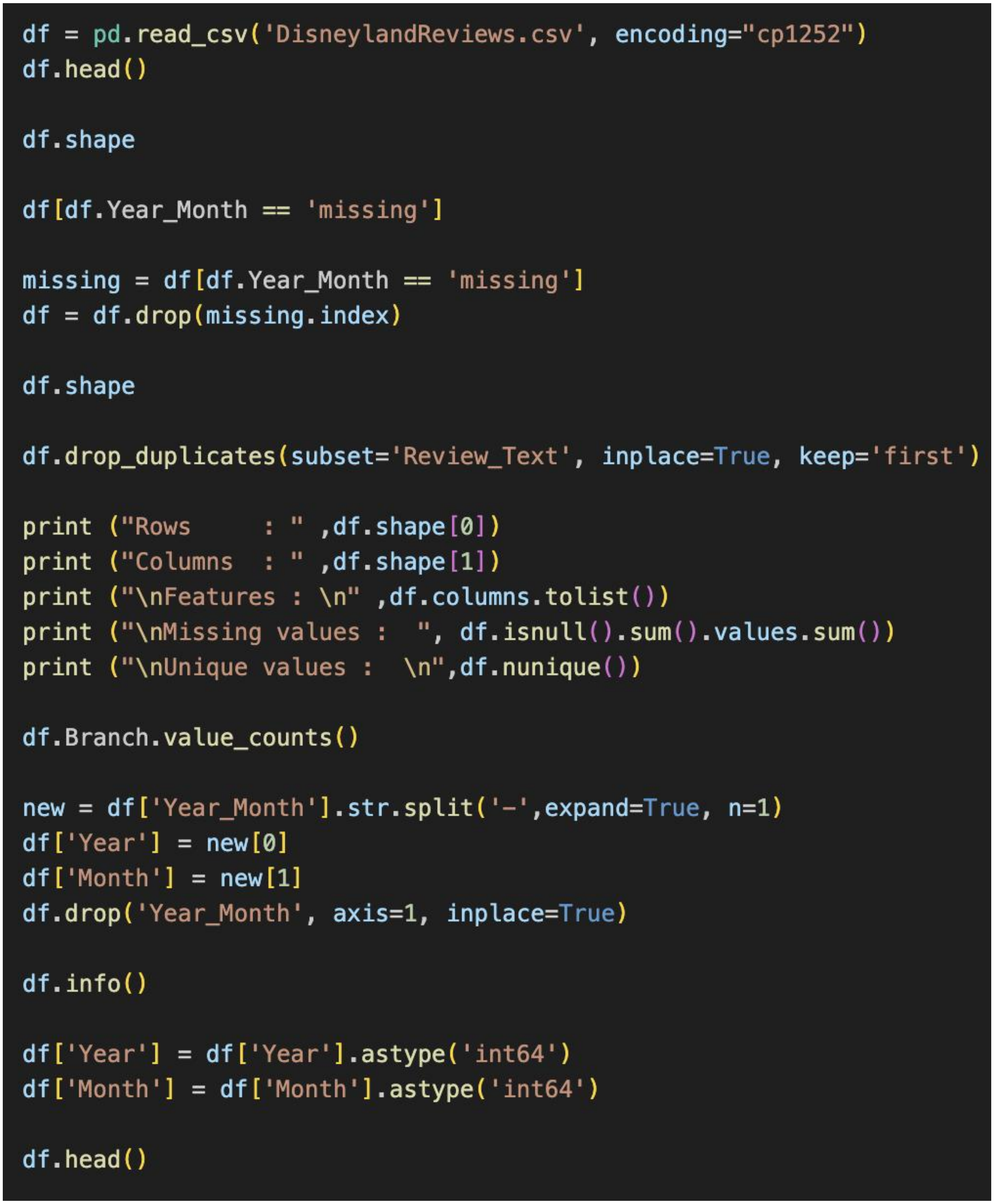
In this project, I conducted a comprehensive analysis of Disneyland's review text data, focusing on various aspects such as sentiment analysis, ratings distribution and theme modelling.

1.1 Data loading and cleaning:

Load the CSV file using pandas.

Remove rows with 'missing' in the Year_Month column.

Remove duplicate comment text lines.



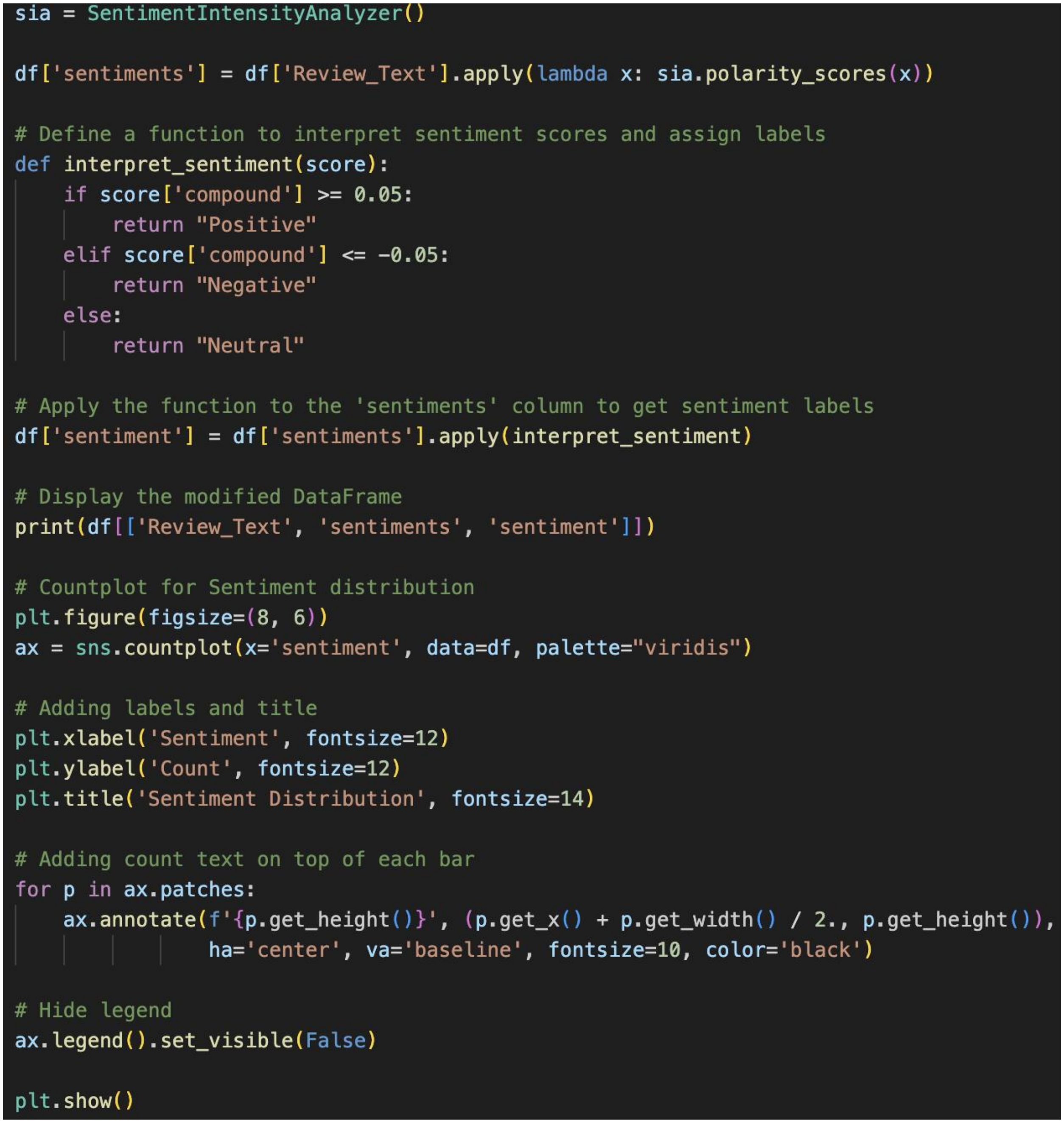
```
df = pd.read_csv('DisneylandReviews.csv', encoding="cp1252")
df.head()

df.shape

df[df.Year_Month == 'missing']

missing = df[df.Year_Month == 'missing']
df = df.drop(missing.index)

df.shape

df.drop_duplicates(subset='Review_Text', inplace=True, keep='first')

print ("Rows     : " ,df.shape[0])
print ("Columns  : " ,df.shape[1])
print ("\nFeatures : \n" ,df.columns.tolist())
print ("\nMissing values :  ", df.isnull().sum().values.sum())
print ("\nUnique values :  \n",df.nunique())

df.Branch.value_counts()

new = df['Year_Month'].str.split('-',expand=True, n=1)
df['Year'] = new[0]
df['Month'] = new[1]
df.drop('Year_Month', axis=1, inplace=True)

df.info()

df['Year'] = df['Year'].astype('int64')
df['Month'] = df['Month'].astype('int64')

df.head()
```

1.2. Sentiment Analysis:

Use SentimentIntensityAnalyzer from the nltk library for sentiment analysis.

A function was defined to interpret the sentiment scores and assign labels (positive, negative, neutral) to the comments based on the sentiment scores.

A count plot of the sentiment distribution was drawn using seaborn.



```python
sia = SentimentIntensityAnalyzer()

df['sentiments'] = df['Review_Text'].apply(lambda x: sia.polarity_scores(x))

# Define a function to interpret sentiment scores and assign labels
def interpret_sentiment(score):
    if score['compound'] >= 0.05:
        return "Positive"
    elif score['compound'] <= -0.05:
        return "Negative"
    else:
        return "Neutral"

# Apply the function to the 'sentiments' column to get sentiment labels
df['sentiment'] = df['sentiments'].apply(interpret_sentiment)

# Display the modified DataFrame
print(df[['Review_Text', 'sentiments', 'sentiment']])

# Countplot for Sentiment distribution
plt.figure(figsize=(8, 6))
ax = sns.countplot(x='sentiment', data=df, palette="viridis")

# Adding labels and title
plt.xlabel('Sentiment', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.title('Sentiment Distribution', fontsize=14)

# Adding count text on top of each bar
for p in ax.patches:
    ax.annotate(f'{p.get_height()}', (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='baseline', fontsize=10, color='black')

# Hide legend
ax.legend().set_visible(False)

plt.show()
```

1.3. topic modelling:

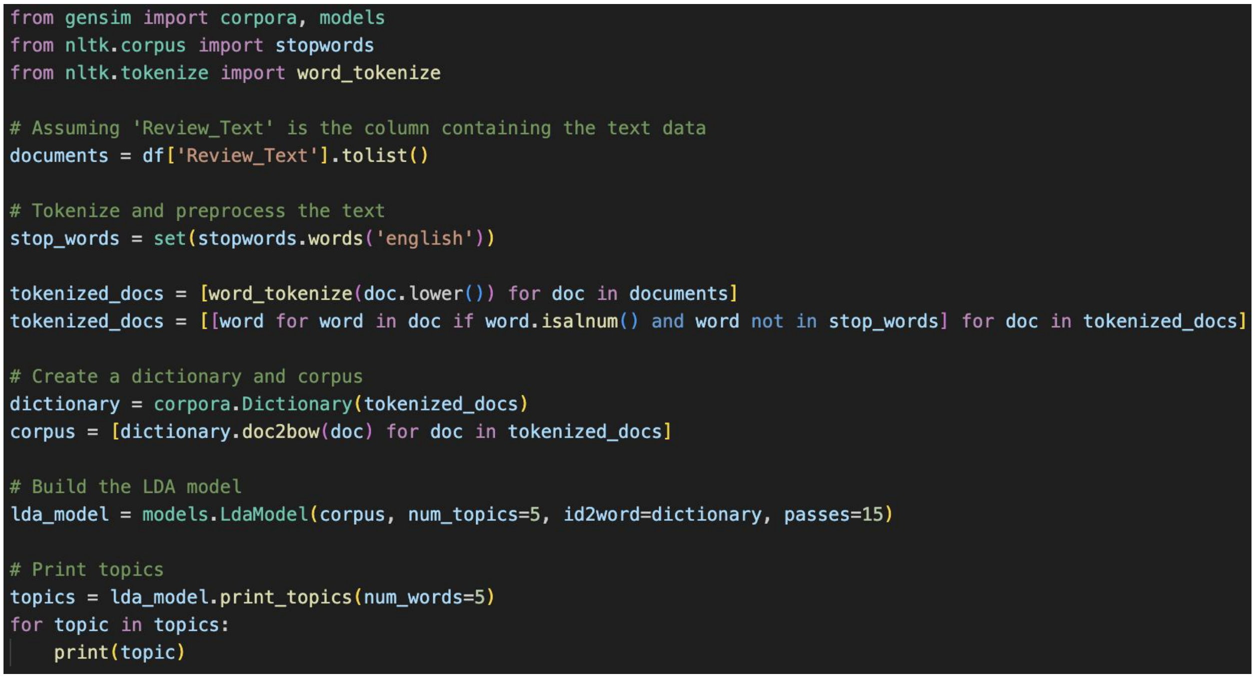
Theme modelling was performed using the gensim library.

Preprocessed text data, including word splitting and deactivation.

Creation of dictionary and corpus.

Constructed LDA (Latent Dirichlet Allocation) topic models.

Printed the first five words for each topic.



```
from gensim import corpora, models
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

# Assuming 'Review_Text' is the column containing the text data
documents = df['Review_Text'].tolist()

# Tokenize and preprocess the text
stop_words = set(stopwords.words('english'))

tokenized_docs = [word_tokenize(doc.lower()) for doc in documents]
tokenized_docs = [[word for word in doc if word.isalnum() and word not in stop_words] for doc in tokenized_docs]

# Create a dictionary and corpus
dictionary = corpora.Dictionary(tokenized_docs)
corpus = [dictionary.doc2bow(doc) for doc in tokenized_docs]

# Build the LDA model
lda_model = models.LdaModel(corpus, num_topics=5, id2word=dictionary, passes=15)

# Print topics
topics = lda_model.print_topics(num_words=5)
for topic in topics:
    print(topic)
```

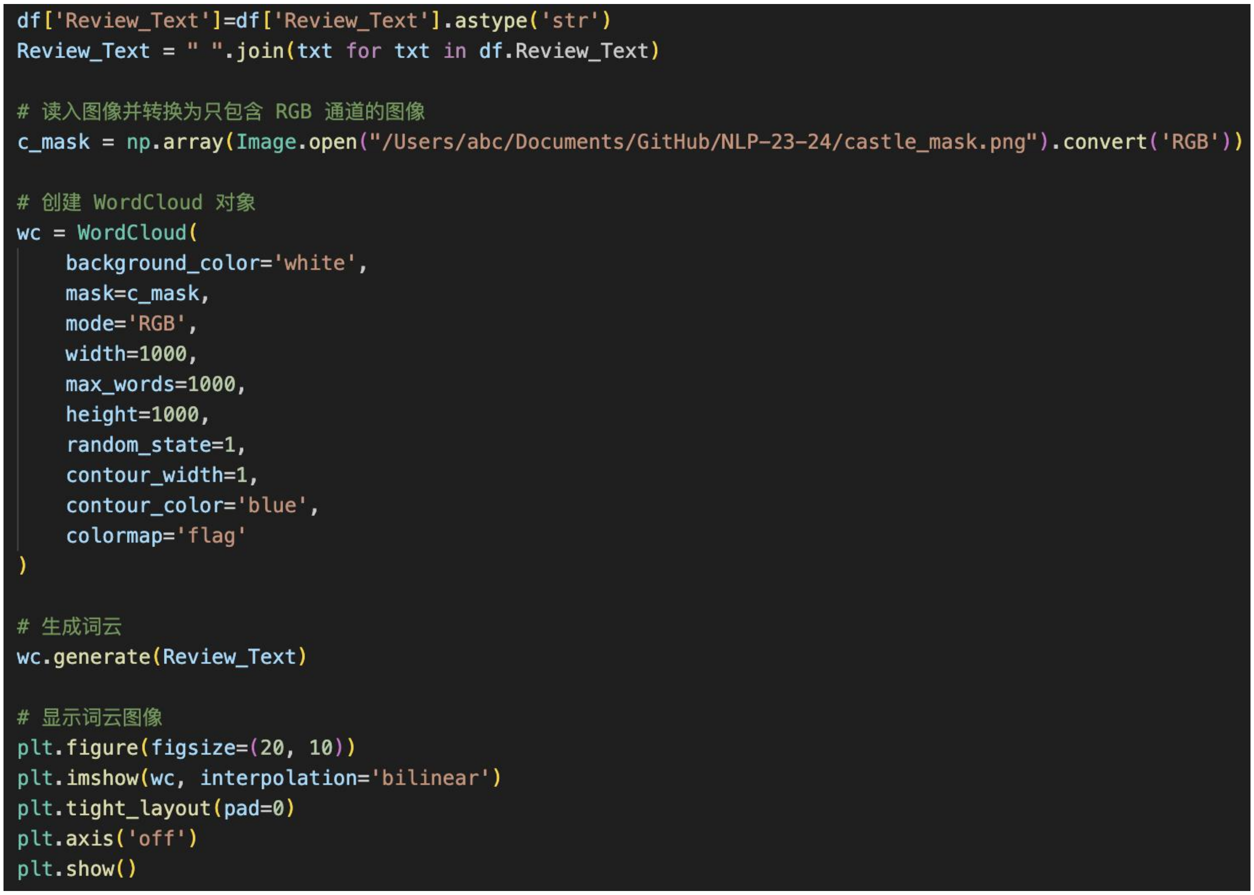
1.4. word cloud generation:

Word clouds were generated using WordCloud and matplotlib libraries.

An image was read and converted to contain only RGB channels.

A word cloud object was created and parameters like background colour, mask map, colour etc. were set.

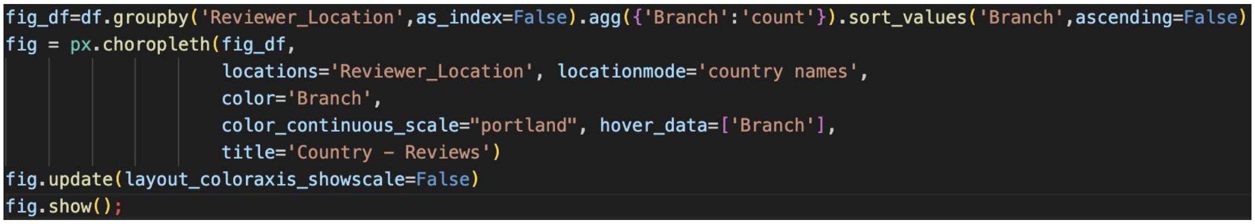
The word cloud image was generated and displayed.



```
df['Review_Text']=df['Review_Text'].astype('str')
Review_Text = " ".join(txt for txt in df.Review_Text)

# 读入图像并转换为只包含 RGB 通道的图像
c_mask = np.array(Image.open("/Users/abc/Documents/GitHub/NLP-23-24/castle_mask.png").convert('RGB'))

# 创建 WordCloud 对象
wc = WordCloud(
    background_color='white',
    mask=c_mask,
    mode='RGB',
    width=1000,
    max_words=1000,
    height=1000,
    random_state=1,
    contour_width=1,
    contour_color='blue',
    colormap='flag'
)

# 生成词云
wc.generate(Review_Text)

# 显示词云图像
plt.figure(figsize=(20, 10))
plt.imshow(wc, interpolation='bilinear')
plt.tight_layout(pad=0)
plt.axis('off')
plt.show()
```

1.5. country comment distribution visualisation:

Choropleth plots of country comment distribution were created using plotly.express.



```
fig_df=df.groupby('Reviewer_Location',as_index=False).agg({'Branch':'count'}).sort_values('Branch',ascending=False)
fig = px.choropleth(fig_df,
                    locations='Reviewer_Location', locationmode='country names',
                    color='Branch',
                    color_continuous_scale="portland", hover_data=['Branch'],
                    title='Country - Reviews')
fig.update(layout_coloraxis_showscale=False)
fig.show();
```

2. data preprocessing:

Split Year_Month into Year and Month columns.

Split Year_Month into Year and Month columns. Convert the data types of the Year and Month columns to integers.

3. model parameters and experimental details:

3.1. Sentiment Analysis Experiment:

Sentiment analysis is performed using SentimentIntensityAnalyzer.

The comments were classified as positive, negative and neutral by modifying the threshold of compound sentiment score.

3.2. Topic modelling experiment:

Using the LDA model, the number of topics was set to 5 and the number of iterations was 15.

4. word cloud generation experiment:

WordCloud was used to generate word clouds, with parameters such as background colour, mask map, and colour set.

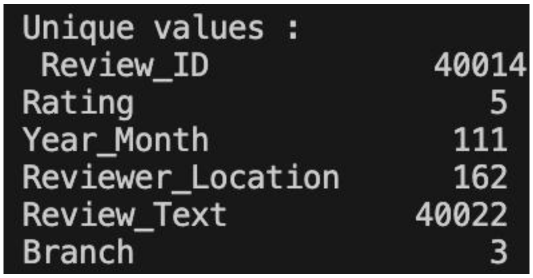
5. country comment distribution visualisation experiment:

Used plotly.express to create a Choropleth graph showing the distribution of comments in different countries.

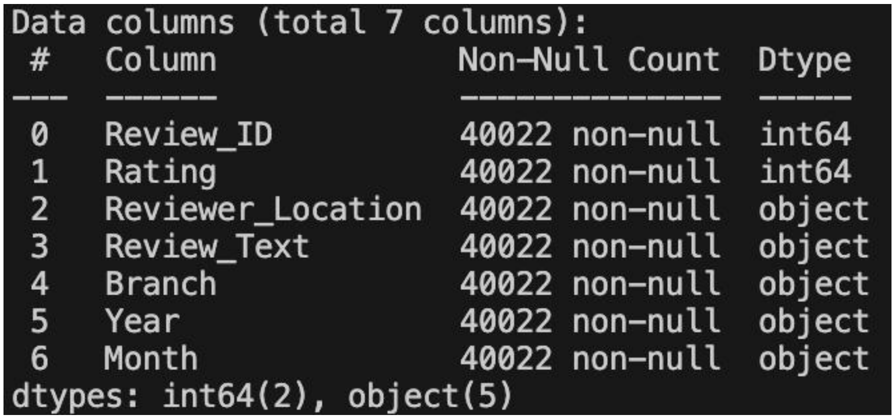
Results

This is where you share the results from your work. You do not need to share every output or figure, just the most important results. This could be good and bad examples from a generative algorithm. These could be examples of individual texts (documents) that have been collected using web scraping, or some charts showing how much data you have collected. This could also be the most important words for the topics that are most meaningful, or a table with various classification accuracy results, like the following:

The following is the sample size of the data for this analysis (after removing invalid data)

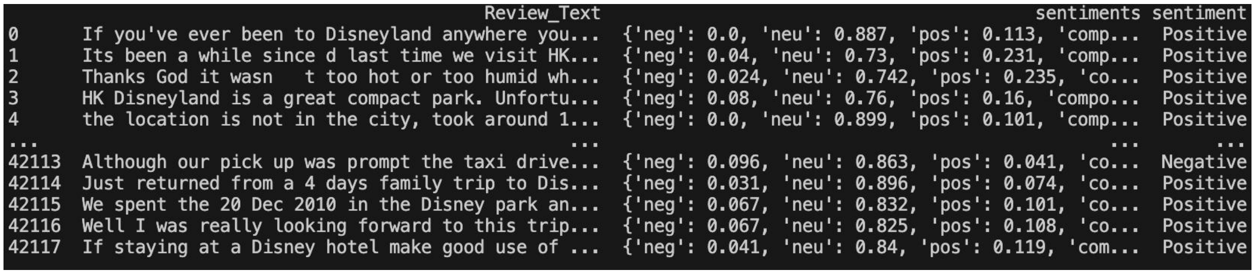


```
Unique values :
 Review_ID            40014
Rating                   5
Year_Month             111
Reviewer_Location      162
Review_Text          40022
Branch                   3
```

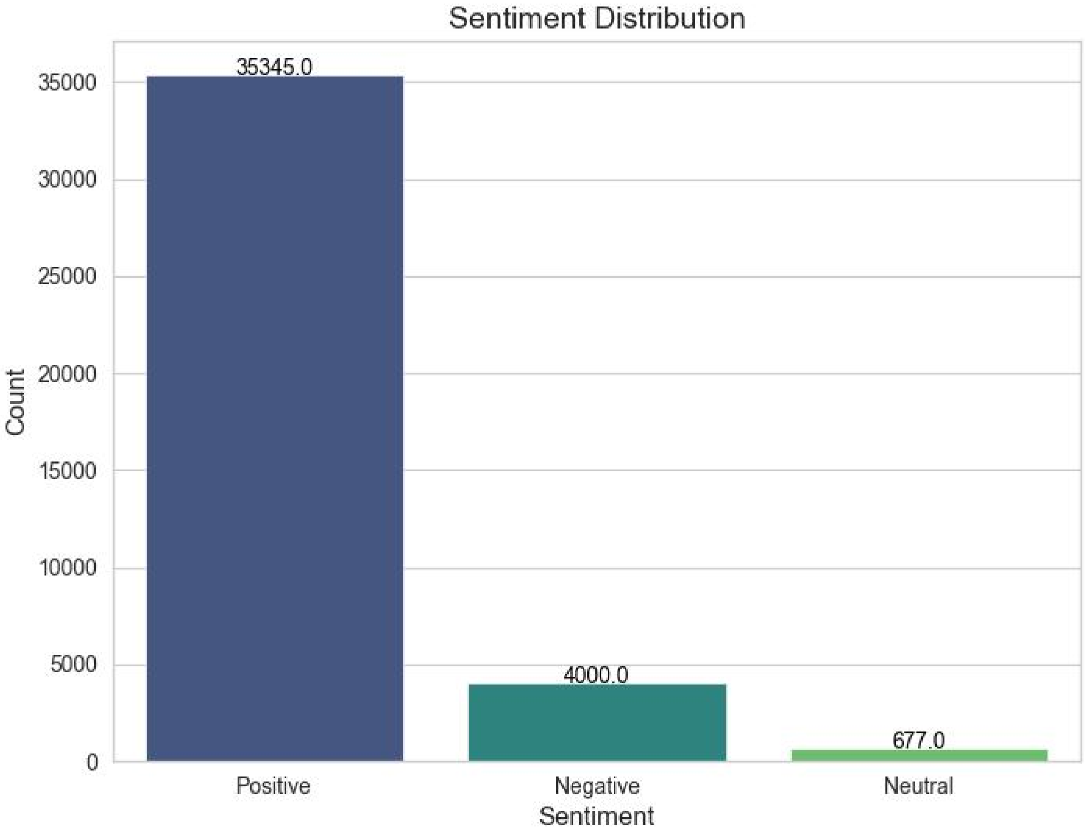


```
Data columns (total 7 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Review_ID           40022 non-null  int64
 1   Rating              40022 non-null  int64
 2   Reviewer_Location   40022 non-null  object
 3   Review_Text         40022 non-null  object
 4   Branch              40022 non-null  object
 5   Year                40022 non-null  object
 6   Month               40022 non-null  object
dtypes: int64(2), object(5)
```

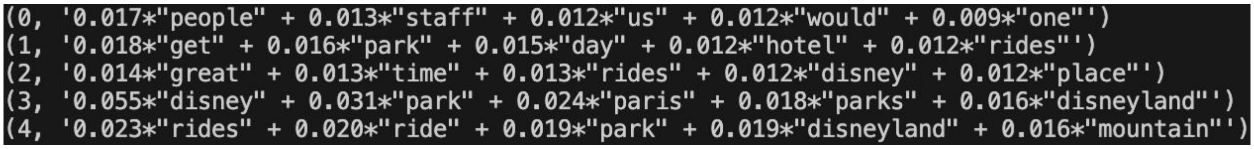
The output of this code consists of a modified DataFrame and a bar graph showing the distribution of sentiment.



```
                                            Review_Text                                         sentiments sentiment
0      If you've ever been to Disneyland anywhere you...  {'neg': 0.0, 'neu': 0.887, 'pos': 0.113, 'comp...  Positive
1      Its been a while since d last time we visit HK...  {'neg': 0.04, 'neu': 0.73, 'pos': 0.231, 'comp...  Positive
2      Thanks God it wasn   t too hot or too humid wh...  {'neg': 0.024, 'neu': 0.742, 'pos': 0.235, 'co...  Positive
3      HK Disneyland is a great compact park. Unfortu...  {'neg': 0.08, 'neu': 0.76, 'pos': 0.16, 'compo...  Positive
4      the location is not in the city, took around 1...  {'neg': 0.0, 'neu': 0.899, 'pos': 0.101, 'comp...  Positive
...                                                  ...                                                ...       ...
42113  Although our pick up was prompt the taxi drive...  {'neg': 0.096, 'neu': 0.863, 'pos': 0.041, 'co...  Negative
42114  Just returned from a 4 days family trip to Dis...  {'neg': 0.031, 'neu': 0.896, 'pos': 0.074, 'co...  Positive
42115  We spent the 20 Dec 2010 in the Disney park an...  {'neg': 0.067, 'neu': 0.832, 'pos': 0.101, 'co...  Positive
42116  Well I was really looking forward to this trip...  {'neg': 0.067, 'neu': 0.825, 'pos': 0.108, 'co...  Positive
42117  If staying at a Disney hotel make good use of ...  {'neg': 0.041, 'neu': 0.84, 'pos': 0.119, 'com...  Positive
```

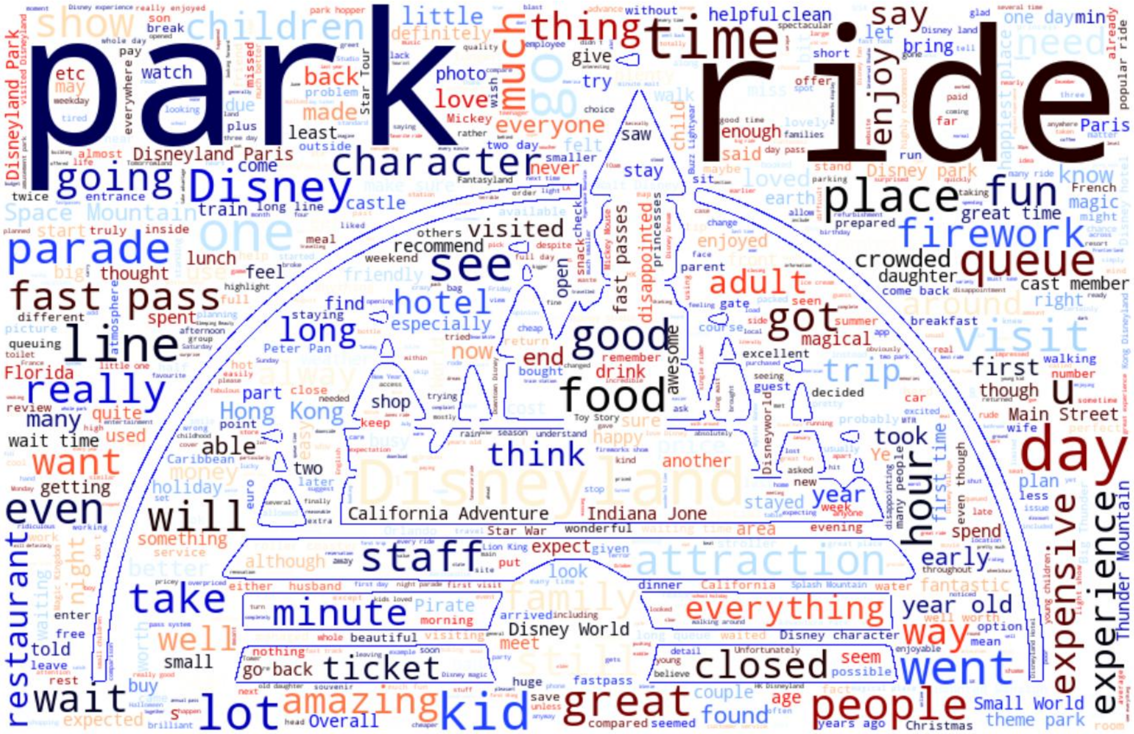


This code discovers the structure of topics in the text data through an LDA model, where each topic is represented by a set of words. The printed output displays each topic and its corresponding top five representative words.

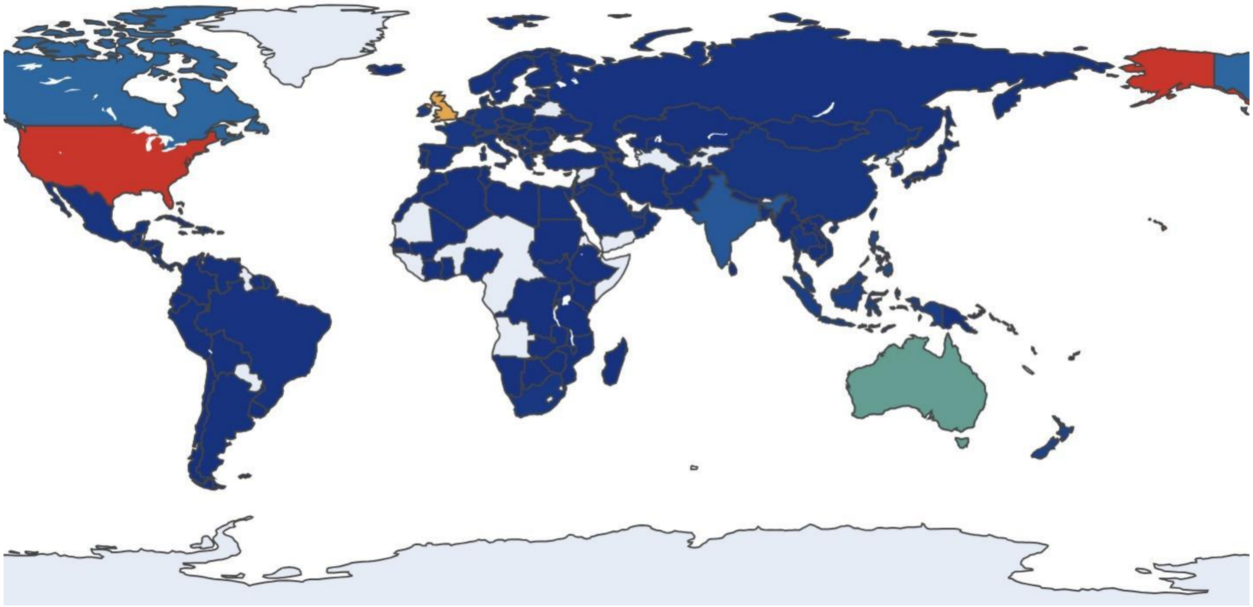


```
(0, '0.017*"people" + 0.013*"staff" + 0.012*"us" + 0.012*"would" + 0.009*"one"')
(1, '0.018*"get" + 0.016*"park" + 0.015*"day" + 0.012*"hotel" + 0.012*"rides"')
(2, '0.014*"great" + 0.013*"time" + 0.013*"rides" + 0.012*"disney" + 0.012*"place"')
(3, '0.055*"disney" + 0.031*"park" + 0.024*"paris" + 0.018*"parks" + 0.016*"disneyland"')
(4, '0.023*"rides" + 0.020*"ride" + 0.019*"park" + 0.019*"disneyland" + 0.016*"mountain"')
```

This code uses the WordCloud library to create a word cloud based on the text of the comment, and uses the provided Disney castle shape as a mask



This code uses the plotly.express library to create a heat map of countries based on the location of commenters, where the colour shade indicates the number of comments received for that country, which can be viewed by hovering over the country on the map.



Discussion

The Walt Disney Company is celebrating its 100th anniversary this year. From movies and TV shows to theme parks, the company has continued to captivate children and adults with its magic for over a century. Disney's six Disneyland parks around the world continue to make up losses, with a combined total of more than 100 million visitors each year. Disney theme parks made an operating profit of about US$2,425 million from April to June this year, accounting for 70 per cent of the total profit. In the past hundred years, Disney has worked hard to protect everyone's childhood, but at the same time, this "century-old shop" is also experiencing its own "old age crisis". In recent years, Disney has experienced different degrees of layoffs, losses, poor box office, price hikes and loss of Disney+ paid members in different regions of the world. So understanding what visitors say and think about Disney parks plays an important role for Disney, it not only helps the business to track visitors' opinions, but also helps the business to discover ways to improve the vitality of the parks. I hope that methods such as Sentiment Analysis and Theme Modelling can be used to categorise these comments, find common themes, understand the aspects that visitors are concerned about and what may be the trending topics that will help me to have a more comprehensive understanding of the guest feedback and provide valuable insights for Disney to improve services, enhance the guest experience, and develop marketing strategies. I hope that Disney will continue to provide a better childhood and life for audiences from all over the world in its next 100 years.

But sadly, my analyses of this dataset still leave a lot to be desired.

The reviews collected so far only last until 2019, and after three years of global uncertainties such as epidemics, the data is bound to have some ups and downs, which can make the results of the analyses less realistic. Also in the sentiment analysis section randomly generated sentiment labels were used, I wanted to use more complex sentiment analysis models such as deep learning models to improve the accuracy of the sentiment analysis but this was not possible due to technical force. The visualisation part of the analysis was also not comprehensive enough to help communicate my results more clearly.

Conclusion

For this project, I conducted a comprehensive analysis of review data for Disney theme parks. Firstly, I cleaned the data, dealt with missing values and duplicates, and conducted an initial exploratory analysis of the data, including the distribution of the number of reviews and the popularity of the theme parks. Subsequently, I performed sentiment analysis, using sentiment scores and randomly generated sentiment labels to represent the emotional tendencies of the reviews. Next, I performed topic modelling using natural language processing techniques to identify key themes in the reviews. Finally, through word clouds and geospatial visualisations, I showed a visual representation of the content of the reviews and attempted to relate the data to the Disney centenary celebrations. This project not only provided insights into user sentiment and themes, but also provided initial data to support further research and the Disney Centennial celebration. In the future, more refined analyses and further data collection can be used to deepen the understanding of user experience and emotional feedback.

Ethical considerations

Ethical considerations are critical in any data-based project, with datasets and models created intended for research and exploratory analysis. The results of topic modelling provide insights into the topic, but interpretations can be subjective and should not be taken as an absolute representation of the user's opinion. The dataset is derived from Disneyland reviews and ethical consideration should be given to the privacy and consent of the individuals providing the reviews, and insights derived from the analysis should not be misused to manipulate public opinion or make misleading statements. By considering these ethical aspects, the project aims to provide valuable insights while respecting the privacy and rights of the individuals who provided comments.

LLM disclaimer

Wrote code and modified adjustments

Getting help by providing code snippets and asking questions

Translating articles

Summarise internet hotspots

Text language may be translated from another language

Bibliography

[1] Luo, Jian Ming, et al. "Topic modelling for theme park online reviews: Analysis of Disneyland." Journal of Travel & Tourism Marketing 37.2 (2020): 272-285.

[2] ABisong, Ekaba, and Ekaba Bisong. "Matplotlib and seaborn." Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners (2019): 151-165.

[3] AHMED ASHOUR (2020) Disneyland Reviews NLP +Sentiment Analysis. Available at: https://www.kaggle.com/code/ahmedterry/disneyland-reviews-nlp-sentiment-analysis/notebook (Accessed: 5/12/2023).