e-ISBN : 978-629-98755-x-x

SIG : e-Learning@CS
https://appspenang.uitm.edu.my/sigcs/
Publication Date : xx – xxx - 2024

DETERMINATION OF TEXT POLARITY CLASSIFICATION USING SENTIMENT ANALYSIS

Syarifah Adilah Mohamed Yusoff[1], *Jamal Othman[2], Mohd Saifulnizam Abu Bakar[3] and Arifah Fasha Rosmani[4]

*syarifah.adilah@uitm.edu.my[1], *jamalothman@uitm.edu.my[2], mohdsaiful071@uitm.edu.my[3], arifah840@usm.edu.my[4]*

[1,2,3,4]Jabatan Sains Komputer & Matematik (JSKM),
Universiti Teknologi MARA Cawangan Pulau Pinang, Malaysia

**Corresponding author*

ABSTRACT

To effectively, analyzing the unstructured data or text for reviewing purposes need a robust tool to process and represent the result in comprehensive data visualization. Texts reviews as responded from thousands of reviewers, customers' experiences or reputations and comments from the netizens are classified accordingly to derive overall perceptions on certain issues, products or views. The text reviews from the social media such as Facebook, twitter, Instagram or WhatsApp are the best platform to retrieve the specific field to perform the polarity classification. Polarity can be expressed as the numerical rating or sentiment score conveyed by a particular text, phrase or word. This paper will discuss phases that involves in Sentiment Analysis (SA) such as the Data Extraction, Data Pre-processing, Data Annotation, Polarity Detection, Evaluation and finally the Data Visualization. Two methods for data classification such as Machine Learning and Lexicon-based approaches have been employed to train the machine or tools to learn the data. Samples of python codes were provided at each phase of SA processes to demonstrate the classification and perform the data visualization based on the text reviews.

Keywords: classification, sentiment analysis, phyton, machine learning, lexicon-based

Introduction

With the growth of Internet infrastructure, users are increasingly engaging with social media platforms like Twitter, WhatsApp, Instagram, Facebook, and blogs to share reviews and comments. Today, when shopping online, you don't need to inquire about the quality of a product or service from others. Instead, you can rely on numerous product reviews available online, which provide detailed analyses to help you make informed decisions. In fact, Ghenie et al., 2025 has highlighted the crucial role of social media in modern business strategies, offering diverse perspectives on leveraging these platforms for feedback and engagement.

Sentiment Analysis (SA), or Opinion Mining, is a widely adopted approach by researchers and marketers to process and interpret significant volumes of public opinions and feedback on different subjects, products, and services. (Sánchez-Rada & Iglesias, 2019). This opinion mining involves the computational examination of feelings, opinions, and emotions conveyed through written text. This area has experienced considerable expansion in both scholarly research and practical applications within various industries. It falls under the category of Natural Language Processing(NLP), utilizes sophisticated computational techniques to categorize sentiments as positive, negative, or neutral (Neethu & Rajasree, 2013). Numerous studies have explored the application of Sentiment

Analysis to mine extensive datasets of reviews. For instance, Soleymani et al. (2017) and Yadav & Vishwakarma (2020) have published research on sentiment classification. Furthermore, Yue et al. (2019) and Liu et al. (2012) investigated the efficacy of online reviews. Jain et al. (2021) have discussed machine learning applications that utilize online reviews for sentiment categorization, predictive decision-making, and the identification of fraudulent reviews.

The process of extracting pertinent text reviews frequently entails managing noisy or ambiguous data. This noise may originate from diverse sources, including extraneous material, spam, or debates that are not pertinent to the analysis. A product review thread may contain irrelevant personal anecdotes, ads, or automated bot messaging, which might obfuscate the useful insights found inside authentic reviews. (Islam et al., 2024). Furthermore, the ambiguity in the data stems from the subjective characteristics of human language. Individuals articulate their viewpoints through various means, employing slang, abbreviations, emojis, and differing degrees of detail. This diversity complicates the uniform interpretation of the sentiment and significance of each review. A sarcastic remark may be misconstrued as affirmative feedback if the subtleties of sarcasm are not accurately recognised by the extraction algorithms (Almansour et al., 2022). The context of the reviews significantly influences their interpretation. A review pertinent in one context may be deemed irrelevant in another. A comprehensive analysis of a product's durability may be vital for a buyer seeking longevity, while meaningless for an individual focused on the product's aesthetic qualities. Consequently, comprehending the context and objective of the data extraction is crucial for precisely finding and employing pertinent reviews (Khalaf et al., 2024). In summary, extracting pertinent text reviews necessitates traversing an overwhelming amount of noisy and ambiguous data. It necessitates a synthesis of sophisticated NLP methodologies, ongoing model enhancement, and contextual comprehension to efficiently eliminate extraneous information and precisely extract key insights that appropriately characterise the data. (Almansour et al., 2022).

This study explores the methodological framework for conducting Sentiment Analysis, outlining the necessary procedures to assess classification polarity through the use of the Python programming language. The phases of Sentiment Analysis encompass Data Extraction, Data Pre-processing, Data Labelling, Polarity Detection, Evaluation, and Data Visualisation. The study utilised further polarity metrics, lexicon-based methods, and machine learning models to identify sentiment analysis in airline reviews.

Methodology

Sentiment Analysis (SA) in text reviews or opinion requires the execution of several critical phases in order to produce accurate results. These phases include Data Extraction, Data Pre-processing, Data Classification, Polarity Detection, and Evaluation. The ultimate objective of this process is to achieve

comprehensive Data Visualization (Aqlan et. al, 2019) as has been illustrated in Fig.1.

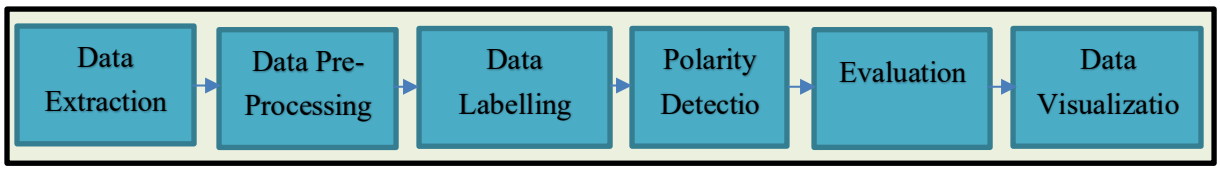


Figure1: sentiment analysis phases

The implementation was executed using the Python programming language within the Anaconda3 software platform. This platform offers a comprehensive suite of libraries and tools specifically designed for data preprocessing and classification in Sentiment Analysis (SA). Python's versatility makes it an effective tool for data classification, leveraging both machine learning and lexicon-based approaches through the implementation of specialized libraries available in Anaconda3.

The dataset used for this study, titled **"Airlines Reviews and Rating,"** was sourced from Kaggle and comprises an extensive collection of reviews and ratings from passengers across various airlines. It includes multiple columns containing different data types, such as textual reviews, numerical ratings, travel dates, and other relevant information. The dataset is rich in information and offers valuable insights into passengers' experiences with different airlines. However, they also pose challenges in terms of data quality, completeness, and the need for suitable preprocessing to ensure accurate analysis and visualization. Data preprocessing, or data cleansing, was applied to improve data quality and minimize noise, with the primary objective of ensuring the data was consistent and free from errors. Cleaned data enhances the effectiveness of data visualization, enabling the quick identification of patterns and making it easier to forecast or predict future trends.

Data extraction phase

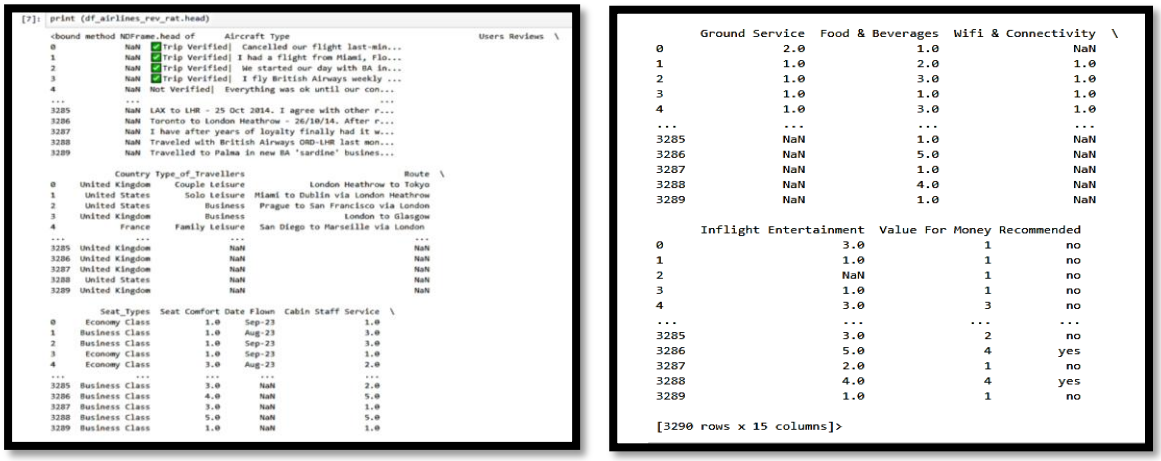
The dataset on airline reviews and rating has been downloaded from the Kaggle website (https://www.kaggle.com/datasets/anandshaw2001/airlines-reviews-and-rating). This source file was used to determine the polarity classification using the Sentiment Analysis approach. The downloaded data source file was imported to Data Frame of Python programming language using Pandas library in the Anaconda3 as shown in the following Fig. 2. The created Data Frame was used for further processing after all records from the data source file have been loaded into the memory.

```
import pandas as pd

df_airlines_rev_rat = pd.read_csv("Airlines Review and Rating.csv")
```

Figure 2: load all records from the data source into the data frame using pandas.

The dataset's content was organised into columns and rows, representing fields and records, respectively. The *print()* command was employed to display the records, revealing the initial five records for each column or field. The following Fig. 3 shows the commands in Python and the output. The data source consists of 3290 rows or records and 15 columns.



```
[7]: print (df_airlines_rev_rat.head)

<bound method NDFrame.head of      Aircraft Type                                                Users Reviews  \
0              NaN  ✅Trip Verified|  Cancelled our flight last-min...
1              NaN  ✅Trip Verified| I had a flight from Miami, Flo...
2              NaN  ✅Trip Verified|  We started our day with BA in...
3              NaN  ✅Trip Verified|  I fly British Airways weekly ...
4              NaN  Not Verified|  Everything was ok until our con...
...            ...                                                ...
3285           NaN  LAX to LHR - 25 Oct 2014. I agree with other r...
3286           NaN  Toronto to London Heathrow - 26/10/14. After r...
3287           NaN  I have after years of loyalty finally had it w...
3288           NaN  Traveled with British Airways ORD-LHR last mon...
3289           NaN  Travelled to Palma in new BA 'sardine' busines...

            Country Type_of_Travellers                          Route  \
0    United Kingdom     Couple Leisure       London Heathrow to Tokyo
1     United States      Solo Leisure  Miami to Dublin via London Heathrow
2     United States          Business  Prague to San Francisco via London
3    United Kingdom          Business              London to Glasgow
4            France    Family Leisure  San Diego to Marseille via London
...             ...               ...                             ...
3285 United Kingdom               NaN                             NaN
3286 United Kingdom               NaN                             NaN
3287 United Kingdom               NaN                             NaN
3288  United States               NaN                             NaN
3289 United Kingdom               NaN                             NaN

         Seat_Types  Seat Comfort Date Flown  Cabin Staff Service  \
0     Economy Class           1.0     Sep-23                  1.0
1    Business Class           1.0     Aug-23                  3.0
2    Business Class           1.0     Sep-23                  3.0
3     Economy Class           1.0     Sep-23                  1.0
4     Economy Class           3.0     Aug-23                  2.0
...             ...           ...        ...                  ...
3285 Business Class           3.0        NaN                  2.0
3286 Business Class           4.0        NaN                  5.0
3287 Business Class           3.0        NaN                  1.0
3288 Business Class           5.0        NaN                  5.0
3289 Business Class           1.0        NaN                  1.0
```

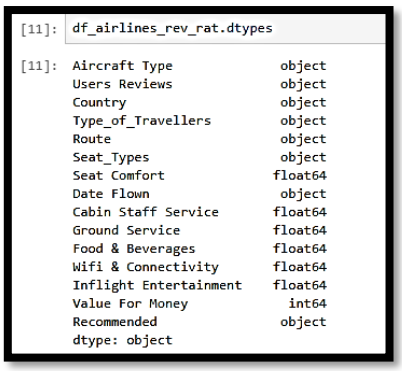
```
      Ground Service  Food & Beverages  Wifi & Connectivity  \
0                2.0              1.0                  NaN
1                1.0              2.0                  1.0
2                1.0              3.0                  1.0
3                1.0              1.0                  1.0
4                1.0              3.0                  1.0
...              ...              ...                  ...
3285             NaN              1.0                  NaN
3286             NaN              5.0                  NaN
3287             NaN              1.0                  NaN
3288             NaN              4.0                  NaN
3289             NaN              1.0                  NaN

      Inflight Entertainment  Value For Money Recommended
0                        3.0                1          no
1                        1.0                1          no
2                        NaN                1          no
3                        1.0                1          no
4                        3.0                3          no
...                      ...              ...         ...
3285                     3.0                2          no
3286                     5.0                4         yes
3287                     2.0                1          no
3288                     4.0                4         yes
3289                     1.0                1          no

[3290 rows x 15 columns]>
```

Figure 3: display the first five records for each column from the data frame.

The `dtypes` function in a Pandas data frame, was used to determine the type and names of all fields or columns that exists in the data source file as illustrated in Fig. 4.



```
[11]: df_airlines_rev_rat.dtypes

[11]: Aircraft Type              object
      Users Reviews              object
      Country                    object
      Type_of_Travellers         object
      Route                      object
      Seat_Types                 object
      Seat Comfort              float64
      Date Flown                 object
      Cabin Staff Service       float64
      Ground Service            float64
      Food & Beverages          float64
      Wifi & Connectivity       float64
      Inflight Entertainment    float64
      Value For Money             int64
      Recommended                object
      dtype: object
```

Figure 4: List of fields of the data source file in the data frame

Before determining which columns and rows to extract for further preprocessing and classification, it is crucial to examine the presence of null data in each column. Fig. 5 below provides a detailed overview of null occurrences per column. Notably, only 20% of the columns, including "user reviews," "value for money," and "recommended," contain no null values.

```
print(df_airlines_rev_rat.isnull().sum())

Aircraft Type             1394
Users Reviews                0
Country                      1
Type_of_Travellers         403
Route                      407
Seat_Types                   3
Seat Comfort               114
Date Flown                 410
Cabin Staff Service        125
Ground Service             478
Food & Beverages           379
Wifi & Connectivity       2698
Inflight Entertainment    1119
Value For Money              0
Recommended                  0
dtype: int64
```

Figure 5: number of null values occurrences *for* each column.

Focusing to the aircraft type under consideration was 'A320', Fig. 6 presents the comprehensive code for the extraction. The code filters the 'A320' aircraft type from the data frame and assigned the result into a new data frame called a320_df. Subsequently, the number of records and the count of null values in each column or attribute were recorded, culminating in a total extraction of 352 rows.

```
filter = = df_airlines_rev_rat['Aircraft Type']=='A320'

a320_df = df_airlines_rev_rat[filter]

a320_df.shape
```

Figure 6: codes to filter and store into new data frame specifically for ‘A320’ aircraft.

Next, the distribution of null values across all attributes using a simple command were depicted in a presentable bar chart, as shown in Fig. 7. The chart reveals the count of NaN or null values across attributes as follows: Seat Comfort (7), Date Flown (1), Cabin Staff Service (7), Ground Service (1), Food & Beverages (51), Wifi & Connectivity (300), and Inflight Entertainment (288), with the number of null values indicated in parentheses.

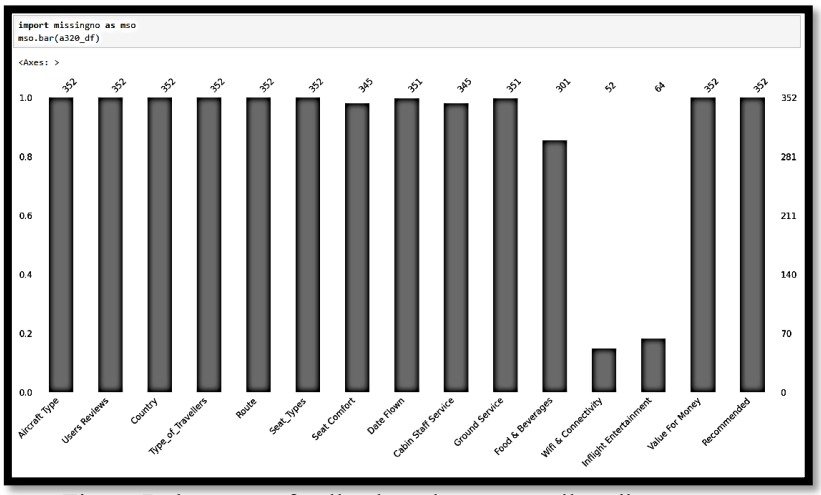


Figure 7: the count of null values data across all attributes.

Data preprocessing phase

Using the a320_df data frame, a simple Sentiment Analysis (SA) was conducted to assess the polarity of passengers' reviews on seat comfort and cabin staff service. First, we need to remove unrelated attributes to the sentiment analysis, such as Date Flown, Ground Service, Food & Beverages, WIFI & Connectivity, and Inflight Entertainment. Fig. 8 illustrates the method used to drop these columns from the data frame.

```
a320_df.columns

Index(['Aircraft Type', 'Users Reviews', 'Country', 'Type_of_Travellers',
       'Route', 'Seat_Types', 'Seat Comfort', 'Cabin Staff Service'],
      dtype='object')

a320_df = a320_df.drop(columns= ["Date Flown","Ground Service", "Food & Beverages",
       "Wifi & Connectivity", "Inflight Entertainment", "Value For Money",
       "Recommended"])

a320_df.columns

Index(['Aircraft Type', 'Users Reviews', 'Country', 'Type_of_Travellers',
       'Route', 'Seat_Types', 'Seat Comfort', 'Cabin Staff Service'],
      dtype='object')
```

Figure 8: dropping unrelated columns from the data frame before further pre-processing.

Next, identified the columns that still contain null values and apply the dropna function to remove the corresponding rows from the a320_df data frame. Fig. 9 demonstrates the implementation of the dropna function to eliminate records with null values. Previously, the data frame contained 352 rows, but 7 of these rows had null values in the "seat comfort" and "cabin staff service" columns were eliminated. After this step, the data frame contains only the clean data with no null values remaining in any column.

```
print(a320_df.isnull().sum())

Aircraft Type          0
Users Reviews          0
Country                0
Type_of_Travellers     0
Route                  0
Seat_Types             0
Seat Comfort           7
Cabin Staff Service    7
dtype: int64

a320_df.dropna(inplace=True)

print(a320_df.isnull().sum())

Aircraft Type          0
Users Reviews          0
Country                0
Type_of_Travellers     0
Route                  0
Seat_Types             0
Seat Comfort           0
Cabin Staff Service    0
dtype: int64

a320_df.shape

(345, 8)
```

Figure 9: `dropna` function will remove those records with null values.

Currently, a320_df data frame consists of two columns which were rated by the passengers. The columns are “Seat Comfort” and “Cabin Staff Service”. The data underwent preprocessing steps to calculate the average rate of these two columns and the result have been added as new column in the

data frame as “Average Rate” as illustrated in Fig. 10.

```
a320_df['Average Rate'] = (a320_df['Seat Comfort'] + a320_df['Cabin Staff Service'])/2

print(a320_df[['Seat Comfort','Cabin Staff Service','Average Rate']])

      Seat Comfort  Cabin Staff Service  Average Rate
7              3.0                  4.0           3.5
14             1.0                  3.0           2.0
23             1.0                  1.0           1.0
25             1.0                  3.0           2.0
26             2.0                  2.0           2.0
...            ...                  ...           ...
2860           2.0                  5.0           3.5
2863           3.0                  3.0           3.0
2864           2.0                  4.0           3.0
2870           4.0                  5.0           4.5
2875           3.0                  5.0           4.0
```

Figure 10: the new column average rate calculated and added in the data frame.

Data labelling phase

Polarity of text commonly given in decimal value in range of [-1,1], where positive sentiment when polarity >0; negative sentiment when polarity <0; and neutral when polarity=0. Using the data from the a320_df, the "average rate" column was considered as polarity measure where an average rate below 2.99 is classified as negative, 3.00 as neutral, and above 3.0 as positive. Fig. 11 below illustrates the commands used to carry out the process of polarity rating.

```
a320_df['Polarity Rating'] = a320_df['Average Rate'].apply(lambda x: 'Positive' if x > 3 else('Neutral' if x == 3 else 'Negative'))

print(a320_df[['Average Rate','Polarity Rating']])

      Average Rate Polarity Rating
7              3.5        Positive
14             2.0        Negative
23             1.0        Negative
25             2.0        Negative
26             2.0        Negative
...            ...             ...
2860           3.5        Positive
2863           3.0         Neutral
2864           3.0         Neutral
2870           4.5        Positive
2875           4.0        Positive
```

Figure 11: the new column of polarity rating created after the classification has been performed.

The next step involves conducting a basic Sentiment Analysis (SA) and visualizing the results using the matplotlib and seaborn libraries. Fig. 12 illustrates the outcome, presented in a comprehensive graph. Based on the findings, the service provided by the cabin staff and the seat comfort of the A320 aircraft are not particularly well-received, as the levels of negative and positive sentiment are nearly equal. Consequently, the airline should consider strategies to enhance passenger satisfaction in these

areas to improve overall ratings.

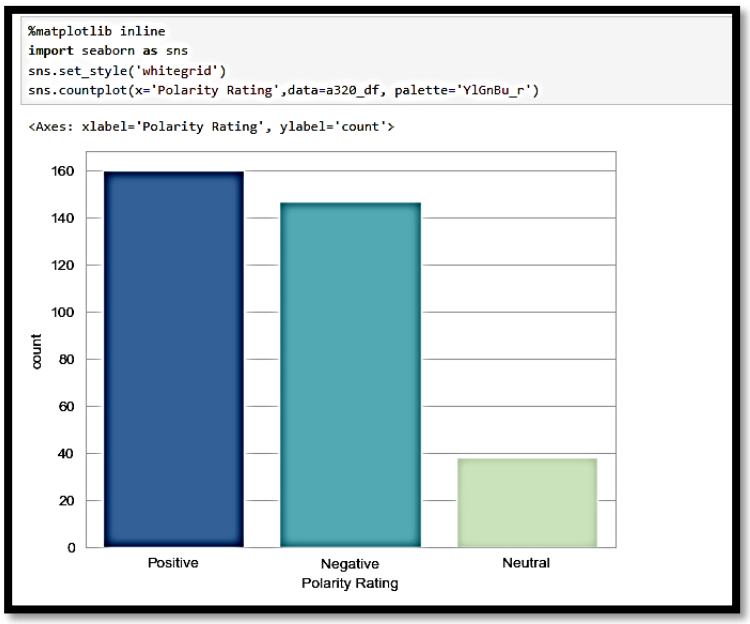


Figure 12: the bar chart graph of polarity rating among the passengers.

Polarity detection and evaluation phase

The previous example, as illustrated in Fig. 12, demonstrates a basic application of Sentiment Analysis (SA). In the next example, we will utilize a specialized library called AFINN to classify and conclude the sentiment in the user reviews column. The AFINN library is a dictionary that contains words which help to identify whether each word in a users' reviews column is matched with the words stored in the AFINN dictionary to determine the corresponding score either negative or positive and finally conclude the sentiment. This method of classification is known as lexicon-based classification.

The implementation of lexicon-based classification for the same a320_df data was focusing on the attribute column "Users Reviews". Fig. 13 presents the user reviews column for the A320 aircraft.

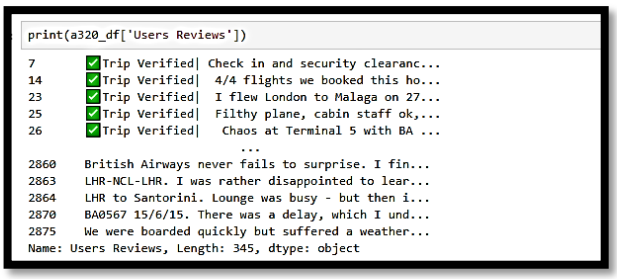


Figure 13: the users' reviews on the aircraft A320

The code in Fig. 14 illustrates how to extract the users' reviews column from the a320_df data

frame and copied into the new data frame named users_review_df. Next, determine the score and polarity sentiment and the result will be added as separate column in the same data frame users_review_df, next to the column of users' reviews.

```
from afinn import Afinn
#instantiate afinn
afn = Afinn()
#creating list sentences
users_reviews_df = a320_df['Users Reviews']
# compute scores (polarity) and labels
scores = [afn.score(article) for article in users_reviews_df]
sentiment = ['positive' if score > 0
                        else 'negative' if score < 0
                            else 'neutral'
                               for score in scores]
#sentimen & lexicon

# dataframe creation
a320_sa_df = pd.DataFrame()
a320_sa_df['Users Reviews'] = users_reviews_df
a320_sa_df['scores'] = scores
a320_sa_df['sentiments'] = sentiment
print(a320_sa_df)

#plot the bar chart graph
%matplotlib inline
sns.set_style('whitegrid')
sns.countplot(x='sentiments',data=a320_sa_df, palette='YlGnBu_r')
```

Figure 14: codes in Python to determine the score and sentiment based on the users' reviews.

The following Fig. 15 depicts the bar chart shows that overall sentiment on the users' reviews toward the A320 aircraft. From the reported graph almost 40% of the passengers are not happy with the service on A320 aircraft and the same result was also reported in the previous sentiment analysis in Fig. 12. Further investigation needs to be done to scrutinize the reasons and plan the new strategies to improve the services to passengers.

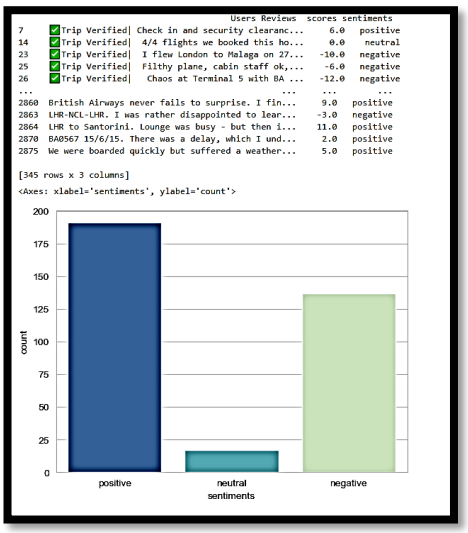


Figure 15: the bar chart of A320 sentiment by using the lexicon-based approach.

Evaluation and Discussion

Finally at this phase, this study has conducted sentiment analysis on passenger reviews using a machine learning technique. Machine learning leverages algorithms to learn from data, which is divided into training and testing subsets. It trains on the training data to learn patterns and uses the learning model to classify sentiment. In the example provided in Fig. 16, this study utilized a cleansed dataset that exclusively features records from 'Malaysia Airlines' (https://www.kaggle.com/datasets/khushipitroda/airline-reviews). This approach specifically targets the reviews column, applying machine learning techniques to derive insights. Fig. 17 presents the results of the machine learning approach utilizing the logistic regression algorithm.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix

# Load the dataset
df_airline_reviews = pd.read_csv('airline_reviews.csv')

#filter the 'Malaysia Airlines' from the airline name column
filter = df_airline_reviews['Airline Name']=='Malaysia Airlines'
df_malaysia_airlines = df_airline_reviews[filter]

# Add a 'Sentiment' column based on the 'Recommended' column
# Assuming 'Recommended' 1 as positive sentiment and 0 as negative sentiment
df_malaysia_airlines['Sentiment'] = df_malaysia_airlines['Recommended']

# Text preprocessing
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
import re

corpus = []
ps = PorterStemmer()
for review in df_malaysia_airlines['Review']:
    review = re.sub('[^a-zA-Z]', ' ', review)
    review = review.lower()
    review = review.split()
    review = [ps.stem(word) for word in review if not word in set(stopwords.words('english'))]
    review = ' '.join(review)
    corpus.append(review)

# Convert text to features
tfidf = TfidfVectorizer(max_features=5000)
X = tfidf.fit_transform(corpus).toarray()
y = df_malaysia_airlines['Sentiment'].values
```

```
# Split the data
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

# Train the model
classifier = LogisticRegression()
classifier.fit(X_train, y_train)

# Make predictions
y_pred = classifier.predict(X_test)

# Evaluate the model
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
print("\nConfusion Matrix:\n", confusion_matrix(y_test, y_pred))

# Save the model and vectorizer for future use
import joblib
joblib.dump(classifier, 'logistic_regression_model.pkl')
joblib.dump(tfidf, 'tfidf_vectorizer.pkl')
```

Figure 16: apply the logistic regression algorithm in machine learning approach for sentiment analysis.

```
Accuracy: 0.6

Classification Report:
              precision    recall  f1-score   support

           0       0.60      1.00      0.75        12
           1       0.00      0.00      0.00         8

    accuracy                           0.60        20
   macro avg       0.30      0.50      0.37        20
weighted avg       0.36      0.60      0.45        20


Confusion Matrix:
 [[12  0]
 [ 8  0]]
```

Figure 17: result of sentiment analysis processing by using the machine learning approach with the logistic regression algorithm.

The model achieved an overall accuracy of 0.6, indicating that 60% of the instances in the dataset were correctly classified. The classification report provides key metrics such as precision, recall, and F1-score for both class 0 and class 1. For class 0, the model achieved a precision of 60%, meaning that 60% of the instances predicted as class 0 were correct. The recall for class 0 was 100%, indicating that all instances of class 0 were accurately identified. The F1-score for class 0 was 0.75, representing a balance between precision and recall. The support for class 0 is 12, reflecting the 12 instances of class 0 in the dataset.

In contrast, the model performed poorly for class 1. The precision for class 1 was 0%, indicating that the model failed to correctly predict any instances of class 1. Similarly, the recall for class 1 was 0%, meaning none of the actual class 1 instances were identified. The F1-score was also 0, reflecting the poor performance in both precision and recall. There were 8 instances of class 1 in the dataset, as

indicated by the support value.

The macro average provides an unweighted mean score for each metric, treating each class equally. The weighted average, however, accounts for the support, or the number of true instances for each class, to provide a weighted mean score. The confusion matrix shows that all 12 instances of class 0 were correctly classified as 0, while all 8 instances of class 1 were misclassified as class 0. There were no correct or incorrect classifications of class 1.

Based on the Fig. 16 results indicate that the model was unable to distinguish instances of class 1, likely due to class imbalance or a lack of distinguishing features for class 1. To enhance the model's performance, particularly for class 1, it might be beneficial to balance the dataset through resampling (either by oversampling class 1 or under sampling class 0), adjust class weights to penalize misclassification of the minority class, or explore alternative algorithms or model architecture. Without addressing these issues, the model remained biased towards predicting class 0, limiting its overall effectiveness.

Data visualisation is the depiction of results through charts or specialised graphics. The results have been presented in a comprehensible way that aids strategic management in decision-making. This study demonstrated Sentiment Analysis processing with specialised graphs, which were updated in real-time using Python. Fig. 17 depicts a radar graph that shows the positive or negative reviews given to each airline from South East Asia airlines. It shows that highest positive reviews were given to Garuda Indonesia.

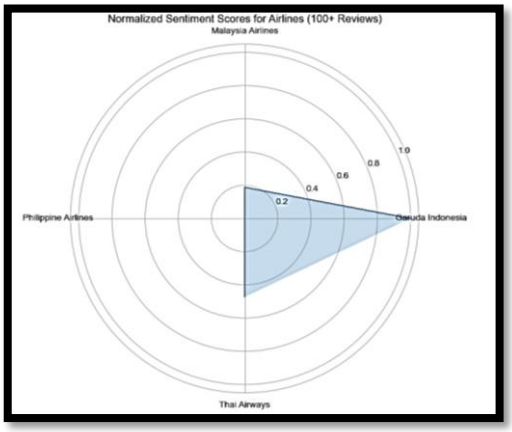


Figure 17: radar graph.

Next, Fig. 18 shows the usage of word cloud specifically for the Malaysia Airline sentiment. The words were extracted from the users' reviews. Meanwhile Fig. 19 depicts the bar chart that visualized the rates given by the passengers of Malaysia Airline specifically on the seat comfort.

Figure 18: word cloud graph.

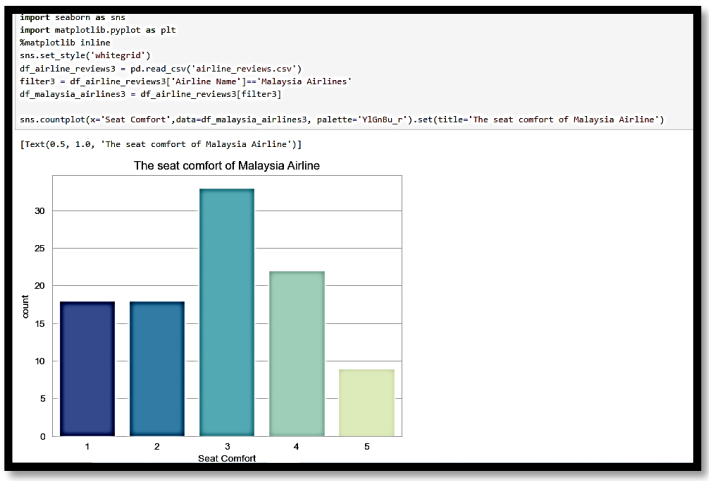


Figure 19: bar chart graph

Python programming provides a diverse array of graph types, enabling researchers to choose and implement the most appropriate chart for presenting elegant reports in a straightforward and visually appealing manner.

Conclusion

Sentiment analysis is a powerful tool used to interpret and classify emotions expressed in textual data, offering valuable insights into public opinion, customer feedback, or social media trends. By analyzing the tones of reviews, comments or feedbacks either positive, negative or neutral, this technique helps businesses and organizations make informed decisions, improve customer satisfaction, and gauge overall sentiment towards products, services, or topics. With advancements in natural language processing and machine learning, sentiment analysis continues to evolve, enabling more accurate and nuanced interpretations of complex human emotions in large datasets. The sentiment analysis process begins with data extraction and culminates in presenting the results through the most suitable and visually appealing data visualization methods. Many practitioners now utilize the concept of a

dashboard as a one-stop centre, where all key results are displayed in one place. This approach makes it easier for users to quickly grasp the latest information or trends, facilitating informed decision-making.

References:

Almansour, A., Alotaibi, R., & Alharbi, H. (2022). Text-rating review discrepancy (TRRD): an integrative review and implications for research. *Future Business Journal*, *8*(1). https://doi.org/10.1186/s43093-022-00114-y

Aqlan, A., Bairam, M., & Naik, R. L. (2019). A study of sentiment analysis: Concepts, techniques, and challenges. In Advances in science, technology & innovation (pp. 147–162). Springer. https://doi.org/10.1007/978-981-13-6459-4_16

Ghenie, D.-Ș., Şoncutean, V.-V., & Sitar-Tăut, D.-A. (2025). Driving Factors of Social Commerce Intention: The Role of Social Commerce Constructs, Social Influence, and Trust. *Smart Innovation, Systems and Technologies*, 107–117. https://doi.org/10.1007/978-981-96-0161-5_10

Islam, Md. S., Muhammad Nomani Kabir, Ngahzaifa Ab Ghani, Kamal Zuhairi Zamli, Nor, Md. Mustafizur Rahman, & Mohammad Ali Moni. (2024). "Challenges and future in deep learning for sentiment analysis: a comprehensive review and a proposed novel hybrid approach." *Artificial Intelligence Review*, *57*(3). https://doi.org/10.1007/s10462-023-10651-9

Jain PK, Pamula R, Ansari S (2021) A supervised machine learning approach for the credibility assessment of user-generated content. Wirel Pers Commun 118(4):2469–2485

Khalaf, I., Mohammad-Reza Feizi-Derakhshi, Saeed Pashazadeh, & Asadpour, M. (2024). A comprehensive review of visual–textual sentiment analysis from social media networks. *Journal of Computational Social Science*. https://doi.org/10.1007/s42001-024-00326-y

Liu B (2012) Sentiment analysis and opinion mining. Synth Lect Hum Lang Technol 5(1):1–167

Neethu J., Rajasree R. (2013). Sentiment Analysis in Twitter using Machine Learning Techniques. IEEE - 31661, 4th ICCCNT 2013 July 4 - 6, 2013, Tiruchengode, India

Sánchez-Rada J.F., Iglesias C.A. (2019). Social context in sentiment analysis: formal definition, overview of current trends and framework for comparison. Inf Fusion 52:344–356

Soleymani M, Garcia D, Jou B, Schuller B, Chang SF, Pantic M (2017) A survey of multimodal sentiment analysis. Image Vis Comput 65:3–14

Yadav A, Vishwakarma DK (2020) Sentiment analysis using deep learning architectures: a review. Artif Intell Rev 53(6):4335–4385

Yue L, Chen W, Li X, Zuo W, Yin M (2019) A survey of sentiment analysis in social media. Knowl Inf Syst 60(2):617–663