ESSENTIALS OF DATA SCIENCE All DIVISIONS

Theory Activity No. 1

NAME : Krish Rajesh Tagram
DIVISION : ET2
ROLL NO : ET2-11
PRN : 202401070058
SUBJECT : EDS

Topic: Kaggle Text Classification Dataset

- Below is the link of Datasheet of the respective topic.

https://drive.google.com/file/d/1IqgEwwNCYQoqoQuu1rdkU3NL1iSQUr1t/view?usp=drive_link

Example Structure for Your Assignment:

Sr No.	Problem Statement	Code/Approach	Output/Explanation
1	Find the total number of rows and columns in the dataset.	df.shape	(10000, 3)
2	Display the first 5 entries of the dataset.	df.head()	(table shown)
3	Find the number of unique labels.	df[’label’].nunique()	5
4	Find the length of	df[’text_length’] = df[’text’].apply(len)	New column added.

Sr No.	Problem Statement	Code/Approach	Output/Explanation
	each text entry and store it in a new column called 'text_length'.		
5	Find the average length of text entries.	df['text_length'].mean()	134.5
6	Count how many texts belong to each label category.	df['label'].value_counts()	table shown
7	Find texts that have more than 100 words.	df[df['text'].apply(lambda x: len(x.split()) > 100)]	filtered dataframe
8	Find missing/null values in the dataset.	df.isnull().sum()	0
9	Replace missing text values with "No Content".	df['text'].fillna('No Content', inplace=True)	changes applied
10	Check duplicate text entries.	df['text'].duplicated().sum()	27 duplicates
11	Remove duplicate text entries.	df = df.drop_duplicates(subset='text')	cleaned dataset
12	Find the shortest text entry.	df['text_length'].min() and corresponding text	"Ok"
13	Find the longest text	df['text_length'].max() and corresponding text	(long text)

Sr No.	Problem Statement	Code/Approach	Output/Explanation
	entry.		
14	Create a new column ’word_count’ containing number of words in each text.	df[’word_count’] = df[’text’].apply(lambda x: len(x.split()))	new column
15	Find the average number of words per text.	df[’word_count’].mean()	23.4
16	List texts with label = ’sports’.	df[df[’label’]==’sports’]	filtered
17	Find the top 5 most common labels.	df[’label’].value_counts().head(5)	
18	Change all text to lowercase.	df[’text’] = df[’text’].str.lower()	cleaned text
19	Find the number of texts containing the word ″urgent″.	df[’text’].str.contains(’urgent’, case=False).sum()	45
20	Save the cleaned dataset to a new CSV file.	df.to_csv(’cleaned_text_dataset.csv’, index=False)	file saved

Code :

```
import pandas as pd

import numpy as np
```

```
# Load the dataset from the given path
df = pd.read_csv("/content/sample_data/ModelTrain.csv")

# 1. Total number of text entries
# Assuming 'Review' column contains the text data
total_texts = df['Review'].count()
print("1. Total Text Entries:", total_texts)

# 2. Average length of text
avg_text_length = df['Review'].apply(len).mean()
print("2. Average Text Length:", avg_text_length)

# 3. Count of unique categories (labels)
# Assuming 'Sentiment' column contains the labels
unique_labels = df['Sentiment'].nunique()
print("3. Unique Labels:", unique_labels)

# 4. Number of texts per category (label)
texts_per_label = df['Sentiment'].value_counts()
print("4. Texts per Category (Label):\n", texts_per_label)

# 5. Longest text entry (by character length)
longest_text = df['Review'].apply(len).max()
print("5. Longest Text Length:", longest_text)

# 6. Texts containing specific words (e.g., 'urgent')
texts_with_urgent = df[df['Review'].str.contains('urgent', case=False)]
print("6. Texts Containing 'urgent':\n", texts_with_urgent)

# 7. Create a new column 'text_length' containing the length of each text
df['text_length'] = df['Review'].apply(len)
```

```
# 8. Count missing values in the 'Review' column
missing_values = df['Review'].isnull().sum()
print("7. Missing Text Entries:", missing_values)

# 9. Most common label
most_common_label = df['Sentiment'].mode()[0]
print("8. Most Common Label:", most_common_label)

# 10. Check for duplicate texts
duplicate_texts = df['Review'].duplicated().sum()
print("9. Duplicate Text Entries:", duplicate_texts)

# 11. Remove duplicate text entries
df = df.drop_duplicates(subset='Review')

# 12. Count of texts with more than 100 words
texts_with_100_words = df[df['Review'].apply(lambda x: len(x.split()) > 100)]
print("10. Texts with More Than 100 Words:\n", texts_with_100_words)

# 13. Top 5 most frequent words (simplified approach - no advanced NLP libraries used)
from collections import Counter
word_counts = Counter(" ".join(df['Review']).split())
top_5_words = word_counts.most_common(5)
print("11. Top 5 Most Frequent Words:", top_5_words)

# 14. Check the distribution of text lengths (i.e., how long texts generally are)
text_length_distribution = df['text_length'].describe()
print("12. Text Length Distribution:\n", text_length_distribution)

# 15. Create a new column for word count
df['word_count'] = df['Review'].apply(lambda x: len(x.split()))
```

```
# 16. Average word count per text
avg_word_count = df['word_count'].mean()
print("13. Average Word Count per Text:", avg_word_count)

# 17. Find texts with the highest number of words
max_word_count = df['word_count'].max()
max_word_count_texts = df[df['word_count'] == max_word_count]
print("14. Texts with the Highest Word Count:\n", max_word_count_texts)

# 18. Texts that are less than 5 words long
short_texts = df[df['word_count'] < 5]
print("15. Texts with Less Than 5 Words:\n", short_texts)

# 19. Save the cleaned dataset (after removing duplicates and creating new columns)
df.to_csv('/content/sample_data/cleaned_ModelTrain.csv', index=False)
print("16. Cleaned Dataset Saved!")

# 20. Save the top 5 most frequent words to a CSV file
top_5_words_df = pd.DataFrame(top_5_words, columns=['Word', 'Frequency'])
top_5_words_df.to_csv('/content/sample_data/top_5_words.csv', index=False)
print("17. Top 5 Words Saved to CSV!")
```

Output:

```
1. Total Text Entries: 8074

2. Average Text Length: 638.3526133267278

3. Unique Labels: 1

4. Texts per Category (Label):
 Sentiment
NEGATIVE    8073
Name: count, dtype: int64

5. Longest Text Length: 6931

6. Texts Containing 'urgent':
                                    Review Sentiment
269   great neglect come back day stay hebei guest h...  NEGATIVE
2374  good value really enjoy every interaction staf...  NEGATIVE
4347  try something else decide try hotwire first ti...  NEGATIVE
4580  get pay whitehall chicago truly independent bo...  NEGATIVE
5295  cockroach awful experience base expedia rating...  NEGATIVE
5373  not great staff make even worse first lobby sm...  NEGATIVE
5537  kid look luxury extra come dubai marine beach ...  NEGATIVE
5742  urgent please read british muslim info give in...  NEGATIVE
5934  not hilton standard stay stopover bhutan would...  NEGATIVE
6064  pleasant glitch please avoid hair salon cost p...  NEGATIVE
6390  stay somewhere else bad smell arrive furniture...  NEGATIVE
6484  far customer service day rate aed check become...  NEGATIVE
6694  bad ever avoid visit dubai urgent need update ...  NEGATIVE

7. Missing Text Entries: 0

8. Most Common Label: NEGATIVE

9. Duplicate Text Entries: 43

10. Texts with More Than 100 Words:
                                    Review Sentiment  text_length
0     stylish clean reasonable value poor glad first...  NEGATIVE        1145
```

```
1     clean good poor service check friend arrive di...  NEGATIVE        823
3     nice apartment stay bedroom home away home nic...  NEGATIVE        935
4     avoid plan laundry place stay family read prev...  NEGATIVE        882
5     really good alternative accomodation beijing f...  NEGATIVE        950
...                                                ...       ...         ...
8062  much arrive checkin caesar palace long line pe...  NEGATIVE        719
8064  nice get right friend mine check day request r...  NEGATIVE        878
8066  need air filter system thorough clean husband ...  NEGATIVE        997
8071  clean decent place stay use travelzoo special ...  NEGATIVE        704
8072  employee bad attitude come back th vegas trip ...  NEGATIVE        682

[3116 rows x 3 columns]
```

```
11. Top 5 Most Frequent Words: [('not', 27998), ('stay', 11583), ('would', 8536), ('get', 7960), ('no', 6913)]
```

```
12. Text Length Distribution:
 count    8031.000000
mean      641.727680
std       516.974983
min         1.000000
25%       315.500000
50%       521.000000
75%       836.500000
max      6931.000000
Name: text_length, dtype: float64
```

```
13. Average Word Count per Text: 101.72805379155771
```

```
14. Texts with the Highest Word Count:
                                                Review Sentiment  \
3721  decent great terrible service stay four day co...  NEGATIVE
```

```
      text_length  word_count
3721         6931        1088

15. Texts with Less Than 5 Words:
                              Review Sentiment  text_length  word_count
8                    would not stay  NEGATIVE           14           3
10    neat bamboo garden questionable  NEGATIVE           31           4
56                   well keep secret  NEGATIVE           16           3
74                          not seem  NEGATIVE            8           2
75               capital capital form  NEGATIVE           20           3
...                               ...       ...          ...         ...
7804                        bad b w  NEGATIVE            7           3
7805                          gross  NEGATIVE            5           1
7807                         beware  NEGATIVE            6           1
7808             place pitt rude staff  NEGATIVE           21           4
7906         not vegas experience want  NEGATIVE           25           4

[341 rows x 4 columns]

16. Cleaned Dataset Saved!

17. Top 5 Words Saved to CSV!
```