DECISION SCIENCES

DA3110 - Data Wrangling
In 20 - Semester 5
Final Group Assignment

Group No:	08				
18Index No(s):	206124J	206132G	206037U	206121X	206077P

Word Count: 2526 No. Pages: 12 No. Figures: 18 No. Tables: 01

Executive Summary

This report describes the Python data wrangling technique used to clean and prepare five interrelated datasets for analysis. Facebook-Google+-and-NewsStory, Missing Publication Date, peer-review-patent-and-policy data, Subject codes, and Tweet Data were among the datasets. The procedure included dealing with missing data, imputation, reduction, and combining datasets that missing publication dates. Enhancing understanding and analytical potential required the transformation of the subject codes into subject names. Improved data quality was achieved through the use of specialized approaches for smooth integration. The data wrangling method included analysis of important factors including structure, granularity, correctness, temporality, and scope. For the following study to provide reliable insights, it was essential that the datasets meet these quality criteria.

The article provides an overview of the complex data wrangling process, highlighting the challenges that faced and the methods used to enhance data quality. The project is considered successful when it results in a single CSV file with consistent headers that is ready for additional analysis and decision-making. Researchers and stakeholders looking to maximize the potential of these datasets will find the insights gathered from this study to be quite helpful.

1. Nature & Key Characteristics of data

Five separate csv files were used in this task and to make the wrangling part more easier, we convert these data sets into excel files. Following are the names of the data sets and the nature of each data set.

1.1. Data set 1 – Facebook-Google+andNewsStory.csv

```
In [19]: print("Structure of the fb dataset\n")
         fb.info()

Structure of the fb dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3826 entries, 0 to 3825
Data columns (total 27 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Mention Type               3826 non-null   object
 1   Mention Date               3826 non-null   object
 2   Outlet or Author           2588 non-null   object
 3   Mention Title              3634 non-null   object
 4   Country                    1935 non-null   object
 5   External Mention ID        0 non-null      float64
 6   Mention URL                3754 non-null   object
 7   Research Output Title      3826 non-null   object
 8   Journal/Collection Title   3598 non-null   object
 9   Output Type                3826 non-null   object
 10  SubjectCode1               2991 non-null   float64
 11  SubjectCode2               2968 non-null   float64
 12  SubjectCode3               479 non-null    float64
 13  Publication Date           3711 non-null   object
 14  Altmetric Attention Score  3826 non-null   int64
 15  DOI                        3133 non-null   object
 16  ISBN                       94 non-null     float64
 17  National Clinical Trial ID 1 non-null      object
 18  URI                        754 non-null    object
 19  PubMed ID                  2051 non-null   float64
 20  PubMedCentral ID           61 non-null     object
 21  Handle.net IDs             91 non-null     object
 22  ADS Bibcode                73 non-null     object
 23  arXiv ID                   1 non-null      float64
 24  RePEc ID                   4 non-null      object
 25  SSRN                       62 non-null     float64
 26  URN                        3 non-null      object
dtypes: float64(8), int64(1), object(18)
memory usage: 807.2+ KB
```

Figure 1: info of fb data set

The provided dataset is a collection of mentions or references related to research outputs. It includes various information such as the type of mention, mention date, outlet or author, research output title, country, publication date, DOI (Digital Object Identifier), Altmetric attention score, and other identifiers. The dataset is diverse, containing both categorical and numeric data, making it valuable for various analyses related to research mentions and their characteristics. We can explore this dataset to gain insights about the popularity and impact of different research outputs across different outlets, countries, and subjects.

1.2. Data set 2 - Missing Publication Date.csv

```
print("Structure of the missing dataset\n")
missing.info()

Structure of the missing dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2751 entries, 0 to 2750
Data columns (total 26 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Mention Type               2751 non-null   object
 1   Mention Date               153 non-null    object
 2   Outlet or Author           2722 non-null   object
 3   Mention Title              150 non-null    object
 4   Country                    101 non-null    object
 5   External Mention ID        2598 non-null   float64
 6   Mention URL                151 non-null    object
 7   Research Output Title      2751 non-null   object
 8   Journal/Collection Title   1012 non-null   object
 9   Output Type                2751 non-null   object
 10  SubjectCode1               1001 non-null   float64
 11  SubjectCode2               1001 non-null   float64
 12  SubjectCode3               32 non-null     float64
 13  Altmetric Attention Score  2751 non-null   int64
 14  DOI                        1043 non-null   object
 15  ISBN                       14 non-null     float64
 16  National Clinical Trial ID 0 non-null      float64
 17  URI                        1650 non-null   object
 18  PubMed ID                  8 non-null      float64
 19  PubMedCentral ID           0 non-null      float64
 20  Handle.net IDs             45 non-null     object
 21  ADS Bibcode                0 non-null      float64
 22  arXiv ID                   0 non-null      float64
 23  RePEc ID                   14 non-null     object
 24  SSRN                       89 non-null     float64
 25  URN                        2 non-null      object
dtypes: float64(11), int64(1), object(14)
memory usage: 558.9+ KB
```

Figure 2: info of missing data set

This dataset contains information about mentions of various research outputs, including details such as mention date, outlet or author, subject codes, Altmetric attention score, and various identifiers like DOI, ISBN, and RePEc ID. It provides valuable insights into the attention and dissemination of research outputs across different channels.

1.3. Data set 3 - Peerreview-patent and policy data.csv

```
print("Structure of the pereview dataset\n")
pereview.info()
```

```
Structure of the pereview dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1985 entries, 0 to 1984
Data columns (total 27 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Mention Type                1985 non-null   object
 1   Mention Date                1985 non-null   object
 2   Outlet or Author            1769 non-null   object
 3   Mention Title               1983 non-null   object
 4   Country                     1769 non-null   object
 5   External Mention ID         0 non-null      float64
 6   Mention URL                 1985 non-null   object
 7   Research Output Title       1985 non-null   object
 8   Journal/Collection Title    1860 non-null   object
 9   Output Type                 1985 non-null   object
 10  SubjectCode1                1780 non-null   float64
 11  SubjectCode2                1732 non-null   float64
 12  SubjectCode3                420 non-null    float64
 13  Publication Date            1980 non-null   object
 14  Altmetric Attention Score   1985 non-null   int64
 15  DOI                         1905 non-null   object
 16  ISBN                        44 non-null     float64
 17  National Clinical Trial ID  1 non-null      object
 18  URI                         482 non-null    object
 19  PubMed ID                   910 non-null    float64
 20  PubMedCentral ID            132 non-null    object
 21  Handle.net IDs              77 non-null     object
 22  ADS Bibcode                 47 non-null     object
 23  arXiv ID                    0 non-null      float64
 24  RePEc ID                    5 non-null      object
 25  SSRN                        4 non-null      float64
 26  URN                         0 non-null      float64
dtypes: float64(9), int64(1), object(17)
memory usage: 418.8+ KB
```

Figure 3: info of preview data set

The dataset contains a variety of information related to research mentions and outputs, including details about their subjects, authors, publication dates, attention scores, and various identifiers such as DOI and PubMed IDs. The dataset's wide range of fields allows for diverse analyses related to research mentions and outputs.

1.4. Data set 4 - Subject codes.xlsx

```
print("Structure of the subcode dataset\n")
subcode.info()
```

```
Structure of the subcode dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 140 entries, 0 to 139
Data columns (total 2 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Subject Code  140 non-null    int64
 1   Subject       140 non-null    object
dtypes: int64(1), object(1)
memory usage: 2.3+ KB
```

Figure 4: info of subcode data set

The dataset seems to offer data about topics or courses, presumably in an academic or educational setting. The following are the columns in the dataset:

1. Subject Code: The courses' respective subject codes are shown in this column. The subject codes are represented as integer values because this column's data type is int64.

2. Subject: The names or course titles are listed in this column. The subject names are likely represented as strings because the data type for this column is object.

1.5. Data set 5 – Tweet Data.csv

```
print("Structure of the tweet dataset\n")
tweet.info()
```

```
Structure of the tweet dataset

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18487 entries, 0 to 18486
Data columns (total 10 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Outlet or Author        18479 non-null  float64
 1   External Mention ID     18487 non-null  float64
 2   Research Output Title   18487 non-null  object
 3   Journal/Collection Title 15714 non-null  object
 4   Output Type             18487 non-null  object
 5   SubjectCode1            13284 non-null  float64
 6   SubjectCode2            12969 non-null  float64
 7   SubjectCode3            3932 non-null   float64
 8   Publication Date        15889 non-null  object
 9   DOI                     14133 non-null  object
dtypes: float64(5), object(5)
memory usage: 1.4+ MB
```

Figure 5: info of tweet data set

The "Tweet Dataset" is a dataset with 10 columns and 18,487 items. Each element in the collection is a tweet, a brief statement published on a social networking site.

1.6. Data set 6 – Wrangled data set: Final_df

```
print("Structure of the Final dataset\n")
Final.info()
```

```
Structure of the Final dataset

<class 'pandas.core.frame.DataFrame'>
Int64Index: 4863 entries, 2 to 18456
Data columns (total 10 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Mention Type      931 non-null    object
 1   Mention Date      907 non-null    object
 2   Mention Time      907 non-null    object
 3   Outlet or Author  4796 non-null   object
 4   Title             862 non-null    object
 5   Country           743 non-null    object
 6   Subjects          4863 non-null   object
 7   Publication Date  4804 non-null   object
 8   DOI               4858 non-null   object
 9   URL               891 non-null    object
dtypes: object(10)
memory usage: 417.9+ KB
```

Figure 6: info of Final_df

A composite dataset known as the "Final Dataset" was produced by merging information from many sources, including the "Pereview Dataset," "Subcode Dataset," "Tweet Dataset," "FB Dataset," "Review Dataset," and a "Missing Dataset."

It has 10 columns and 4,863 entries, all of which have the data type "object," indicating that the dataset largely consists of textual or category information. The dataset uses 471.9KB of memory, which shows that it is a reasonable size for most computing systems and is of a moderate size.

Utilizing data from many datasets, the "Final Dataset" presents a more thorough perspective of the underlying data. Data integration, data cleaning to handle missing or incorrect values, exploratory data analysis (EDA) to gain insights and spot patterns, feature engineering to boost predictive power, and possibly developing machine learning models or performing statistical analyses for decision-making are all necessary steps to effectively work with it.

2. Logical sequence to wrangle.

In order to prepare raw data for analysis, the data wrangling procedure is a crucial step. Cleaning, transforming, and structuring the data into a more accessible and useful manner demands a systematic and well-organized series of processes. The actions taken to get the data for this analysis can be seen in the logical order listed below:

2.1. Integrating and loading data

Loading the libraries that are necessary and importing all five datasets are the first steps in the data-wrangling procedure. Each dataset includes particular data that is important to our study. Then, in order to enable unified processing, we merge each of these datasets into a single, cohesive dataset called merge_df.

```
import pandas as pd

# Import data set
fb = pd.read_excel('D:/Uni/5 Semester/Data Wrangling/final/Facebook-Google+andNewsStory.xlsx')
missing = pd.read_excel('D:/Uni/5 Semester/Data Wrangling/final/Missing Publication Date.xlsx')
pereview = pd.read_excel('D:/Uni/5 Semester/Data Wrangling/final/peerreview-patent and policy data.xlsx')
subcode = pd.read_excel('D:/Uni/5 Semester/Data Wrangling/final/Subject codes.xlsx')
tweet = pd.read_excel('D:/Uni/5 Semester/Data Wrangling/final/Tweet Data.xlsx')
```

Figure 7: code: import pandas & data sets

```
# Merge data sets into one data frame
merge_df = pd.concat([fb, missing, pereview, subcode, tweet])
print("Column names of Merge data set \n", merge_df.columns.tolist())

print(merge_df.shape[0])
```

Figure 8:codes: merge data set into one data set as merge_df

2.2. Remove missing values in SubjectCode columns.

We identify the rows of the columns SubjectCode1, SubjectCode2, and SubjectCode3 that have missing data. We clean the data by removing rows that contain missing values in any of these columns because they are critical for further analysis.

```
columns_to_clean = ['SubjectCode1', 'SubjectCode2', 'SubjectCode3']
merge_df.dropna(subset=columns_to_clean, inplace=True)

merge_df.to_csv('clean_null.csv', index=False)
print(merge_df.head)
print(merge_df.shape[0])
```

Figure 9: codes: clean null values in subject codes columns

2.3. Replace subject codes with subject names.

Columns SubjectCode1, SubjectCode2, and SubjectCode3's topic code values are changed to their respective subject names to make the dataset easier to understand and rename the dataset as rename_df.

```
# Function to replace SubjectCode with Subject names
def replace_subject_code(row):
    try:
        row['SubjectCode1'] = subcode.loc[subcode['Subject Code'] == row['SubjectCode1'], 'Subject'].iloc[0]
    except IndexError:
        row['SubjectCode1'] = None

    try:
        row['SubjectCode2'] = subcode.loc[subcode['Subject Code'] == row['SubjectCode2'], 'Subject'].iloc[0]
    except IndexError:
        row['SubjectCode2'] = None

    try:
        row['SubjectCode3'] = subcode.loc[subcode['Subject Code'] == row['SubjectCode3'], 'Subject'].iloc[0]
    except IndexError:
        row['SubjectCode3'] = None

    return row

# Perform the join and replace operation
rename_df = merge_df.apply(replace_subject_code, axis=1)

# Csv file
rename_df.to_csv('rename_df.csv', index=False)
print("Column names of Rename data set \n", rename_df.columns.tolist())

print(rename_df.shape[0])
```

Figure 10: codes: replace subject codes with subject names & rename data set as rename_df

2.4. Merge subject codes columns into single column as Subjects

All the subject code columns are combined into a single column named Subjects to further simplify the dataset. To maintain clarity, a semicolon is used to divide the subject name. The SubjectCode1, SubjectCode2, and SubjectCode3 columns are then deleted from the dataset since they are no longer needed.

```
# Define a custom function to merge the subject codes
def merge_subject_codes(row):
    codes = [str(row["SubjectCode1"]), str(row["SubjectCode2"]), str(row["SubjectCode3"])]
    merged_codes = "; ".join(code for code in codes if code != "")
    return merged_codes

# Apply the custom function to create the new merged column
rename_df["Subjects"] = rename_df.apply(merge_subject_codes,axis=1)

rename_df.to_csv('rename_merge.csv', index=False)
print("Column names of Rename data set \n", rename_df.columns.tolist())
print(rename_df.shape[0])
```

Figure 11: codes: merge subject code columns into one as subjects

```python
# Define a custom function to merge the subject codes
def merge_subject_codes(row):
    codes = [str(row["SubjectCode1"]), str(row["SubjectCode2"]), str(row["SubjectCode3"])]
    merged_codes = "; ".join(code for code in codes if code != "")
    return merged_codes

# Apply the custom function to create the new merged column
rename_df["Subjects"] = rename_df.apply(merge_subject_codes,axis=1)

rename_df.to_csv('rename_merge.csv', index=False)
print("Column names of Rename data set \n", rename_df.columns.tolist())
print(rename_df.shape[0])
```

Figure 12: codes: separate each subject using ";"

2.5. Separate mention date & time.

In the previous merge_df dataset mention date and time are combined. To make the analysis easier we separate those two into two different columns as Mention Date and Mention Time.

```python
# Create a new column of mention timme
rename_df['Mention Time'] = pd.to_datetime(rename_df['Mention Date']).dt.time
# Remove the time portion from the "Mention Date" column
rename_df['Mention Date'] = pd.to_datetime(rename_df['Mention Date']).dt.date

print(rename_df.shape[0])
rename_df.head()
```

Figure 13: codes: separate mention date & time

2.6. Create the final data set with relevant columns.

As the last step in the logical sequence, we only keep the columns that are necessary for our analysis, and save the final data set as Final_df. This is the wrangled data set of this study.

```python
# Rename the columns
rename_df.rename(columns={'Mention Title': 'Title', 'Mention URL': 'URL', 'Old_Column3': 'New_Column3'}, inplace=True)

# List of column names to save
columns_to_save = ["Mention Type","Mention Date","Mention Time","Outlet or Author","Title","Country","Subjects","Publication Date","DOI","URL"]

# Create a new DataFrame with only the columns to save
Final = rename_df[columns_to_save]
Final

Final.to_csv('final.csv', index=False)
print("Column names of Rename data set \n", Final.columns.tolist())
Final.info()
Final.head()
```

Figure 14: codes: create final data set

	Mention Type	Mention Date	Mention Time	Outlet or Author	Title	Country	Subjects	Publication Date	DOI	URL
2	News story	2022-06-07	14:00:00	PLoS Medicine	What influences Bangladeshi Boro rice farmers'...	United Kingdom	Earth Sciences; Geology; Environmental Sciences	2021-03-01	10.1016/j.gfs.2020.100464	http://ct.moreover.com/?a=47930435815&p=1pl&v=...
38	News story	2022-05-19	19:28:00	The Conversation	Sri Lanka : de la crise économique à la crise ...	Australia	Environmental Sciences; Environmental Science ...	2008-04-08	10.4000/echogeo.2543	http://ct.moreover.com/?a=47784917179&p=1pl&v=...
39	News story	2022-05-19	19:28:00	Yahoo! News	De la crise économique à la crise politique : ...	United States	Environmental Sciences; Environmental Science ...	2008-04-08	10.4000/echogeo.2543	http://ct.moreover.com/?a=47782446217&p=1pl&v=...
45	News story	2022-05-18	06:16:00	MSN	Tractations politiques et alliances électorale...	United States	Studies In Human Society; Political Science; S...	2020-11-26	10.1017/s1743923x20000471	http://ct.moreover.com/?a=47766804818&p=1pl&v=...
47	News story	2022-05-18	00:00:00	La Tribune	Tractations politiques et alliances électorale...	France	Studies In Human Society; Political Science; S...	2020-11-26	10.1017/s1743923x20000471	http://ct.moreover.com/?a=47766626914&p=1pl&v=...

Table 1: wrangled data set

The above-mentioned logical sequence provides a structured and logical method for wrangling data, which guarantees the final dataset, is fit for further analysis.

3. Automation of the workflow

Although the current study focuses primarily on the logical order of data-wrangling processes, we recognize the importance of automation in today's data analysis and its possible utilization in managing massive and repetitive data-wrangling operations. Automation significantly contributes to the consistency of data transformation, a decrease in manual effort, and the streamlining of the data preparation process. We can gain from the following by encoding data wrangling steps into functions and scripts:

Time Efficiency: We can automate time-consuming data preparation processes so that they can be completed with less or zero manual input. The importance of this increases when working with large datasets or when frequent updates are necessary.

Reproducibility & Consistency: Automated processes provide reproducibility and the consistent application of data-wrangling steps, lowering the possibility of mistakes that can occur due to manually wrangling data. This is necessary to preserve data reliability and accuracy in analytical outputs.

Scalability: Manually data wrangling becomes become impractical as datasets increase in size and complexity. We can scale data-wrangling operations effectively and manage vast amounts of data with ease due to automation.

Reusability & Modularity: We can reuse data wrangling stages across several projects by developing modular functions and scripts, leading to a more effective and standardized data preparation process.

Error Handling: We can develop effective error-handling procedures through automation, which gives us more depth into data-related issues and speeds up error resolution.

When working on data-intensive tasks, it is crucial to take into account the structure of automation functions and scripts and the incorporation of error-handling systems.

4. Data Wrangling Dynamics

Data wrangling dynamics involve addressing the challenges and opportunities presented by the structure, granularity, accuracy, temporality, and scope of the dataset. Throughout the data wrangling process, we perform operations such as data cleaning, merging, transformation, and handling missing values to ensure the data's usability and reliability for further analysis.

Structure

The main datasets are 'fb', 'missing', 'preview', 'subcode', and 'tweet', which are initially loaded into separate Pandas Data Frames. These datasets have varying numbers of rows and columns, and each column represented different attributes of the research outputs, mentions, and associated metadata. During data wrangling, we performed several operations to combine and clean the data, ensuring a coherent and unified structure. We dropped irrelevant columns, merged the datasets, and created new columns to enhance data insights. The dataset contains a total of 4,863 entries (rows) and 10 columns after the data-wrangling process. The columns include attributes such as Mention Type, Mention Date, Mention Time, Outlet or Author, Title, Country, Subjects, Publication Date, DOI, and URL.

Granularity

The dataset's granularity is at the level of individual mentions of research outputs. Each row represents a specific mention with various attributes such as the type of mention, date, author, title, country, subjects, publication date, DOI, and URL. We aimed to maintain a consistent level of granularity by handling missing values, merging data based on common identifiers, and ensuring that each row represented a unique mention of the research output.

Accuracy

The accuracy of the dataset depends on the quality of the original data sources and the data-wrangling process applied. During data-wrangling, missing values were handled by dropping rows with missing subject codes. Additionally, subject codes were replaced with subject names using a mapping provided in the "Subject Code" dataset.

Temporality

The dataset contains temporal information such as the "Mention Date" and "Mention Time.". However, it's important to note that the "Mention Date" was split into a separate column for the date part and time part during data wrangling. The "Publication Date" also provides information about the date when the research output was published.

Scope

The scope of the dataset includes various mentions related to research outputs, along with information about the outlets, authors, and subject codes associated with the research outputs. The dataset provides valuable insights into the distribution of mentions across different subjects and outlets.

5. Types of transformation that would improve the quality of this dataset.

During the data wrangling process, several modifications were used to enhance the dataset's quality and usefulness. These changes meant to improve the completeness, validity, and readability of the data. The main modifications that improved the quality of the dataset are as follows:

1. **Remove the records with the missing subject codes:**

To ensure that the dataset only contained occurrences with complete and appropriate data, entries with missing the subject codes were eliminated. Potential bias or errors based on by missing data were thereby reduced. There were three columns (SubjectCode1, SubjectCode2, SubjectCode3) which contain subject codes. So, we remove all null values in those columns because the data analyzing is highly based on those data. So, the null values in other columns can be remain because those are not very important.

2. **Splitting Mention Date into Separate Columns:**

In the merged data set, the “Mention Date” column consists with both date and time. So, we split the time & date and include time as a separate column. Then the time portion in date column is deleted. The original "Mention Date" column was divided into the two columns "Mention Date" and "Mention Time," which made it possible to analyze mentions over time at a more detailed level. This transition made it easier to analyze trends and obtain time-based insights.

3. **Renaming Columns with Relevant Names:**

The column names in the dataset were changed to be clearer and understandable, which has improved data clarity and made it easier to understand the data's content. Column names that are meaningful improve the dataset's applicability for reporting and analysis.

4. **Remove Un useful Columns:**

The first data set consist with around 30 columns. As the final one we got 10 columns which has the most relevant data. Reducing data noise and sharpening data by removing unimportant or unnecessary columns that have little or no impact on the analysis. This change reduced the size of the dataset and facilitated further data processing.

5. **Convert the subject code into subject names:**

The three columns with subject codes converted that represent the subject names. After that the dataset was made easier to understand and utilize by mapping subject codes to their respective subject names. This modification made it possible for data users to quickly recognize and understand the many subjects stated in different contexts.

6. **Merge three subject code name column in to one column:**

The data was better organized and column redundancy reduced through merging subject names from three columns into a single column and separating them with semicolons. This modification made it easier to summarize the various topics connected to each mention.

7. **Ordering data columns:**

The data looks better, and consistency maintained throughout the dataset by rearranging the columns in an ordered and organized way. When working with the data, a structured column structure makes easy reference and navigation simple.
In conclusion, the changes indicated above considerably improved the dataset's usability and quality. They addressed data accuracy, enhanced data representation and structure, and enabled greater insight analysis. The dataset is now more capable for next analysis and reporting of data activities as a result of these modifications.

6. Data visualization & Insights

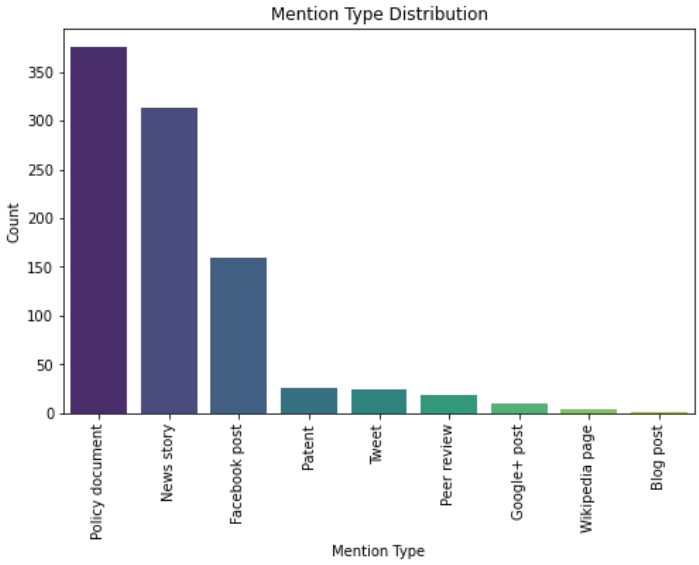
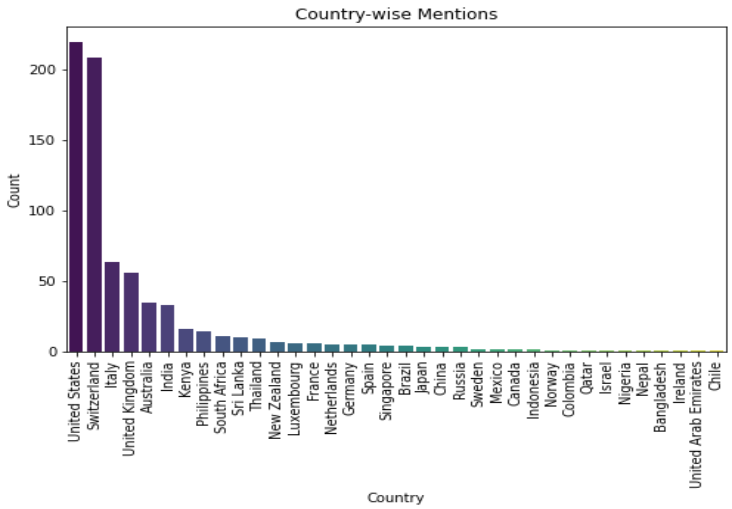


Figure 15: Bar graph: mention type

By using this visualization, we can see that, “Policy document” is the prevalent mention type and “Blog post” is the least mention type.



This bar graph shows about the country wise mention. This would help us to get an idea about how mentions are made based on the country. Most number of mentions are done by United Staes and least number of mentions are made by Chile.

Figure 16: Bar graph: mentions-country wise

```
# Calculate the Mention Type counts
mention_type_counts = Final['Mention Type'].value_counts()

# Display the Top N Mention Types and their Counts
N = 5  # Change this value to show the top N mention types
top_mention_types = mention_type_counts.head(N)
print(top_mention_types)
```

```
Policy document    376
News story         314
Facebook post      159
Patent              25
Tweet               24
Name: Mention Type, dtype: int64
```

Figure 17: codes: top 5 mention types

```
# Calculate the Subject counts
subject_counts = Final['Subjects'].value_counts()

# Display the Subject counts
print(subject_counts)
```

```
Medical And Health Sciences; Clinical Sciences; Public Health And Health Services                    494
Medical And Health Sciences; Cardiorespiratory Medicine And Haematology; Clinical Sciences           398
Environmental Sciences; Environmental Science And Management; Biological Sciences                   356
Studies In Human Society; Political Science; Sociology                                              242
Medical And Health Sciences; Paediatrics And Reproductive Medicine; Public Health And Health Services  212
                                                                                                    ...
Environmental Sciences; Soil Sciences; Agricultural And Veterinary Sciences                           1
Environmental Sciences; Soil Sciences; Biological Sciences                                            1
Earth Sciences; Environmental Sciences; Medical And Health Sciences                                   1
Engineering; Electrical And Electronic Engineering; Mechanical Engineering                            1
Studies In Human Society; Criminology; Policy And Administration                                      1
Name: Subjects, Length: 220, dtype: int64
```

Figure 18: codes: subject counts