In [1]:
```python
import requests
import pandas as pd
```

In [2]:
```python
import requests
import pandas as pd

# API Keys (Replace with your own keys)
NEWSAPI_KEY = "3b983d060c2240ccaedcacdcdc7f29af"
MEDIASTACK_KEY = "d630631db8ad34e3ef4bc452b92009e8"

# List of Expanded Queries
QUERIES = [
    "smartphone charger removal",
    "no charger with phone",
    "Apple no charger policy",
    "Samsung charger removal",
    "e-waste and smartphone chargers",
    "sustainable smartphone accessories"
]
PAGE_SIZE = 50  # Fetch 50 results per page (can be adjusted)

def fetch_newsapi(query):
    articles = []
    for page in range(1, 3):  # Fetch up to 2 pages (Adjust as needed)
        url = f"https://newsapi.org/v2/everything?q={query}&language=en&pageSize={PAGE_SIZE}&page={page}&apiKey={NEWSA
PI_KEY}"
        response = requests.get(url).json()
        if "articles" in response:
            for article in response["articles"]:
                articles.append([
                    article.get("title", "No Title"),
                    article.get("description", "No Description"),
                    article["source"]["name"] if "source" in article else "Unknown",
                    article.get("publishedAt", "No Date"),
                    article.get("url", "No URL")
                ])
    return articles

def fetch_mediastack(query):
    url = f"http://api.mediastack.com/v1/news?access_key={MEDIASTACK_KEY}&languages=en&keywords={query}&limit=50"
    response = requests.get(url).json()
    articles = []
    if "data" in response:
        for article in response["data"]:
            articles.append([
                article.get("title", "No Title"),
                article.get("description", "No Description"),
                article.get("source", "Unknown"),
                article.get("published_at", "No Date"),
                article.get("url", "No URL")
            ])
    return articles

# Collecting articles from both APIs
all_articles = []
for query in QUERIES:
    all_articles.extend(fetch_newsapi(query))
    all_articles.extend(fetch_mediastack(query))

# Creating DataFrame
df_combined = pd.DataFrame(all_articles, columns=["title", "description", "source", "published_date", "url"]).drop_dup
licates()

# Save to CSV
df_combined.to_csv("smartphone_charger_news.csv", index=False)

# Show Summary
print(f"Total Articles Collected: {len(df_combined)}")
print(df_combined.head())

df_combined.info()
```

```
Total Articles Collected: 147
                                               title  \
0  Samsung reportedly had two paths to the Galaxy...
1  Qi2-ready wireless charging on the Samsung Gal...
2  Galaxy S25 Ultra S Pen Bluetooth removal makes...
3  Samsung Galaxy S25+ review: Samsung's transiti...
4  Galaxy S25 Ultra Review: You might like this p...

                                         description             source  \
0  The Galaxy S25 Edge almost had a different loo...     Android Police
1  From protective cases and chargers to ring hol...  Android Authority
2  The Galaxy S25 Ultra has brought many great im...          SamMobile
3  The Samsung you know is gone, replaced by a co...     Android Police
4  Samsung’s flagship smartphones don’t live near...     9to5google.com

         published_date                                                url
0  2025-02-08T22:45:58Z  https://www.androidpolice.com/samsung-galaxy-s...
1  2025-01-24T15:08:54Z  https://www.androidauthority.com/esr-accessori...
2  2025-01-23T15:26:55Z  https://www.sammobile.com/opinion/galaxy-s25-u...
3  2025-02-08T23:00:10Z  https://www.androidpolice.com/samsung-galaxy-s...
4  2025-02-05T20:00:00Z  http://9to5google.com/2025/02/05/samsung-galax...
<class 'pandas.core.frame.DataFrame'>
Index: 147 entries, 0 to 165
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   title           147 non-null    object
 1   description     147 non-null    object
 2   source          147 non-null    object
 3   published_date  147 non-null    object
 4   url             147 non-null    object
dtypes: object(5)
memory usage: 6.9+ KB
```

In [3]:
```
!pip install feedparser
```

```
Collecting feedparser
  Downloading feedparser-6.0.11-py3-none-any.whl.metadata (2.4 kB)
Collecting sgmllib3k (from feedparser)
  Downloading sgmllib3k-1.0.0.tar.gz (5.8 kB)
  Preparing metadata (setup.py) ... done
Downloading feedparser-6.0.11-py3-none-any.whl (81 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 81.3/81.3 kB 1.2 MB/s eta 0:00:00
Building wheels for collected packages: sgmllib3k
  Building wheel for sgmllib3k (setup.py) ... done
  Created wheel for sgmllib3k: filename=sgmllib3k-1.0.0-py3-none-any.whl size=6047 sha256=423d83962e4c3f50126e8a46fe1
1e9d70d71ee58510d156b78f83f9160a615dc
  Stored in directory: /root/.cache/pip/wheels/3b/25/2a/105d6a15df6914f4d15047691c6c28f9052cc1173e40285d03
Successfully built sgmllib3k
Installing collected packages: sgmllib3k, feedparser
Successfully installed feedparser-6.0.11 sgmllib3k-1.0.0
```

In [4]:
```python
import feedparser
import pandas as pd

# Define multiple RSS feed URLs with different queries
QUERIES = [
    "smartphone charger removal",
    "no charger with phone",
    "Apple no charger policy",
    "Samsung charger removal"
]

BASE_URL = "https://news.google.com/rss/search?q={}&hl=en-GB&gl=GB&ceid=GB:en"

# Collect articles from all queries
news_list = []
for query in QUERIES:
    RSS_FEED_URL = BASE_URL.format(query.replace(" ", "+"))
    feed = feedparser.parse(RSS_FEED_URL)

    for entry in feed.entries:
        title = entry.title
        link = entry.link
        published = entry.published
        news_list.append([title, published, link, query])  # Add query for reference

# Convert to DataFrame
df_google_news = pd.DataFrame(news_list, columns=["Title", "Published Date", "URL", "Query"])

# Save to CSV
df_google_news.to_csv("google_news_expanded.csv", index=False)

# Print sample data
print(df_google_news.head())
```

```
                                               Title  \
0  Charger not included: why aren’t phone charger...
1  Xiaomi to join Apple and Samsung in removing c...
2  How much Apple saved by removing charger, Earp...
3  Apple, phone chargers, and the art of greenwas...
4  EU Commission proposes removing chargers from ...

                  Published Date  \
0  Mon, 08 Apr 2024 07:00:00 GMT
1  Sun, 22 Sep 2024 07:00:00 GMT
2  Fri, 10 Jan 2025 08:00:00 GMT
3  Sat, 16 Apr 2022 07:00:00 GMT
4  Fri, 24 Sep 2021 07:00:00 GMT

                                                 URL  \
0  https://news.google.com/rss/articles/CBMif0FVX...
1  https://news.google.com/rss/articles/CBMi_gFBV...
2  https://news.google.com/rss/articles/CBMiywFBV...
3  https://news.google.com/rss/articles/CBMihgFBV...
4  https://news.google.com/rss/articles/CBMiqgFBV...

                        Query
0  smartphone charger removal
1  smartphone charger removal
2  smartphone charger removal
3  smartphone charger removal
4  smartphone charger removal
```

In [5]:
```python
import pandas as pd

# Load the datasets
file_google_news = "google_news_expanded.csv"
file_smartphone_news = "smartphone_charger_news.csv"

df_google_news = pd.read_csv(file_google_news)
df_smartphone_news = pd.read_csv(file_smartphone_news)

# Strip spaces from column names to ensure consistency
df_google_news.columns = df_google_news.columns.str.strip()
df_smartphone_news.columns = df_smartphone_news.columns.str.strip()

# Print available columns before selecting
print("Google News Columns:", df_google_news.columns.tolist())
print("Smartphone News Columns:", df_smartphone_news.columns.tolist())

# Select only relevant columns for merging
columns = ["Title", "Source", "Published Date", "URL"]

# Filter only available columns to avoid KeyError
df_google_news = df_google_news[[col for col in columns if col in df_google_news.columns]]
df_smartphone_news = df_smartphone_news[[col for col in columns if col in df_smartphone_news.columns]]

# Merge datasets and remove duplicates
df_combined = pd.concat([df_google_news, df_smartphone_news], ignore_index=True).drop_duplicates()

# Save the merged dataset
df_combined.to_csv("merged_labeled_news_data.csv", index=False)

# Show dataset summary
print(f"✅ Merged dataset saved as 'merged_labeled_news_data.csv' with {len(df_combined)} articles.")
print(df_combined.head())
```

```
Google News Columns: ['Title', 'Published Date', 'URL', 'Query']
Smartphone News Columns: ['title', 'description', 'source', 'published_date', 'url']
✅ Merged dataset saved as 'merged_labeled_news_data.csv' with 253 articles.
                                               Title  \
0  Charger not included: why aren’t phone charger...
1  Xiaomi to join Apple and Samsung in removing c...
2  How much Apple saved by removing charger, Earp...
3  Apple, phone chargers, and the art of greenwas...
4  EU Commission proposes removing chargers from ...

                  Published Date  \
0  Mon, 08 Apr 2024 07:00:00 GMT
1  Sun, 22 Sep 2024 07:00:00 GMT
2  Fri, 10 Jan 2025 08:00:00 GMT
3  Sat, 16 Apr 2022 07:00:00 GMT
4  Fri, 24 Sep 2021 07:00:00 GMT

                                                 URL
0  https://news.google.com/rss/articles/CBMif0FVX...
1  https://news.google.com/rss/articles/CBMi_gFBV...
2  https://news.google.com/rss/articles/CBMiywFBV...
3  https://news.google.com/rss/articles/CBMihgFBV...
4  https://news.google.com/rss/articles/CBMiqgFBV...
```

In [6]:
```
pip install vaderSentiment
```

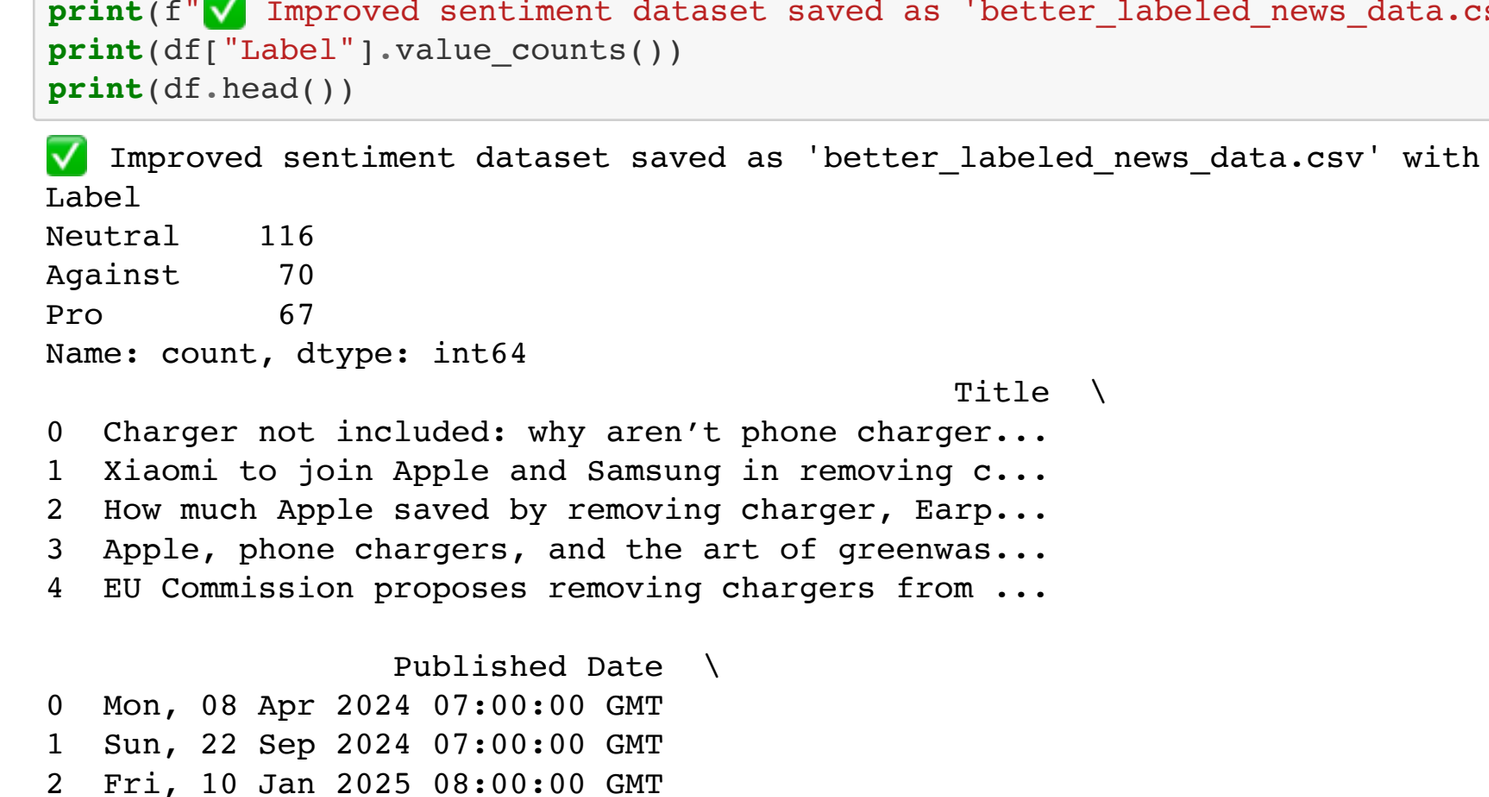
```
Collecting vaderSentiment
  Downloading vaderSentiment-3.3.2-py2.py3-none-any.whl.metadata (572 bytes)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from vaderSentiment) (2.32.3)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests->va
derSentiment) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests->vaderSentimen
t) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests->vaderSen
timent) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests->vaderSen
timent) (2025.1.31)
Downloading vaderSentiment-3.3.2-py2.py3-none-any.whl (125 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 126.0/126.0 kB 6.3 MB/s eta 0:00:00
Installing collected packages: vaderSentiment
Successfully installed vaderSentiment-3.3.2
```

In [7]:
```python
import pandas as pd
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

# Load dataset
file_path = "/content/merged_labeled_news_data.csv"
df = pd.read_csv(file_path)

# Initialize VADER
analyzer = SentimentIntensityAnalyzer()

# Function to analyze sentiment
def get_vader_sentiment(text):
    sentiment_score = analyzer.polarity_scores(str(text))["compound"]  # Get overall sentiment score

    if sentiment_score >= 0.05:
        return "Pro"  # Positive sentiment
    elif sentiment_score <= -0.05:
        return "Against"  # Negative sentiment
    else:
        return "Neutral"  # Neutral sentiment

# Apply VADER sentiment analysis
df["Label"] = df["Title"].apply(get_vader_sentiment)

# Save the improved labeled dataset
df.to_csv("vader_news_data.csv", index=False)

# Show summary
print(f"✅ Improved sentiment dataset saved as 'better_labeled_news_data.csv' with {len(df)} articles.")
print(df["Label"].value_counts())
print(df.head())
```

```
✅ Improved sentiment dataset saved as 'better_labeled_news_data.csv' with 253 articles.
Label
Neutral    116
Against     70
Pro         67
Name: count, dtype: int64
                                               Title  \
0  Charger not included: why aren’t phone charger...
1  Xiaomi to join Apple and Samsung in removing c...
2  How much Apple saved by removing charger, Earp...
3  Apple, phone chargers, and the art of greenwas...
4  EU Commission proposes removing chargers from ...

                  Published Date  \
0  Mon, 08 Apr 2024 07:00:00 GMT
1  Sun, 22 Sep 2024 07:00:00 GMT
2  Fri, 10 Jan 2025 08:00:00 GMT
3  Sat, 16 Apr 2022 07:00:00 GMT
4  Fri, 24 Sep 2021 07:00:00 GMT

                                                 URL    Label
0  https://news.google.com/rss/articles/CBMif0FVX...  Neutral
1  https://news.google.com/rss/articles/CBMi_gFBV...      Pro
2  https://news.google.com/rss/articles/CBMiywFBV...      Pro
3  https://news.google.com/rss/articles/CBMihgFBV...  Against
4  https://news.google.com/rss/articles/CBMiqgFBV...  Neutral
```

In [8]:
```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the labeled dataset
file_path = "vader_news_data.csv"
df = pd.read_csv(file_path)

# Ensure "Label" column exists
if "Label" in df.columns:
    # Plot Sentiment Distribution
    plt.figure(figsize=(8,5))
    sns.countplot(x="Label", data=df, palette="coolwarm")
    plt.title("Sentiment Distribution in News Articles")
    plt.xlabel("Sentiment Category")
    plt.ylabel("Number of Articles")
    plt.show()
else:
    print("⚠️ 'Label' column not found in the dataset. Please check the data.")
```

```
<ipython-input-8-5f09825ad59c>:13: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `h
ue` and set `legend=False` for the same effect.

  sns.countplot(x="Label", data=df, palette="coolwarm")
```



In [9]:
```python
import nltk
nltk.download('punkt')
nltk.download('punkt_tab')
nltk.download('wordnet')
nltk.download('stopwords')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

Out[9]: True

In [11]:
```python
import pandas as pd
import re
import nltk
from nltk.tokenize import word_tokenize
from nltk.stem import PorterStemmer, WordNetLemmatizer
from nltk.corpus import stopwords
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer

# Load the dataset
file_path = "vader_news_data.csv"
df = pd.read_csv(file_path)

# Initialize preprocessing tools
stemmer = PorterStemmer()
lemmatizer = WordNetLemmatizer()
stop_words = set(stopwords.words('english'))

# Function to clean text
def clean_text(text):
    text = re.sub(r'\W', ' ', str(text))  # Remove special characters
    text = re.sub(r'\d+', '', text)  # Remove numbers
    text = text.lower().strip()  # Convert to lowercase
    return text

# Function for Stemming
def apply_stemming(text):
    words = word_tokenize(clean_text(text))  # Tokenize & clean text
    words = [stemmer.stem(word) for word in words if word not in stop_words]
    return " ".join(words)

# Function for Lemmatization
def apply_lemmatization(text):
    words = word_tokenize(clean_text(text))  # Tokenize & clean text
    words = [lemmatizer.lemmatize(word) for word in words if word not in stop_words]
    return " ".join(words)

# Apply Stemming & Lemmatization
df["Stemmed_Title"] = df["Title"].apply(apply_stemming)
df["Lemmatized_Title"] = df["Title"].apply(apply_lemmatization)
df[["Title", "Stemmed_Title", "Label"]].to_csv("stemmed_news_data.csv", index=False)
df[["Title", "Lemmatized_Title", "Label"]].to_csv("lemmatized_news_data.csv", index=False)

# Save cleaned dataset
df.to_csv("preprocessed_news_data.csv", index=False)

# Initialize Vectorizers
count_vectorizer = CountVectorizer(max_df=0.9, min_df=5, max_features=5000)
tfidf_vectorizer = TfidfVectorizer(max_df=0.9, min_df=5, max_features=5000)

# Apply Vectorization on Lemmatized Titles
count_matrix = count_vectorizer.fit_transform(df["Lemmatized_Title"])
tfidf_matrix = tfidf_vectorizer.fit_transform(df["Lemmatized_Title"])

# Convert to DataFrames
df_count = pd.DataFrame(count_matrix.toarray(), columns=count_vectorizer.get_feature_names_out())
df_tfidf = pd.DataFrame(tfidf_matrix.toarray(), columns=tfidf_vectorizer.get_feature_names_out())

# Save vectorized datasets
df_count.to_csv("count_vectorized_news.csv", index=False)
df_tfidf.to_csv("tfidf_vectorized_news.csv", index=False)

# Print confirmation
print("✅ Preprocessing Complete! Datasets saved.")
```

```
✅ Preprocessing Complete! Datasets saved.
```

In [ ]: