```
Start coding or generate with AI.
```

1. Install required libraries and load the spaCy English model

```
!pip install spacy pandas numpy matplotlib seaborn scikit-learn emoji
!python -m spacy download en_core_web_sm
```

```
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.12/dist-packages (from
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.12/dist-packages (from s
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy
Requirement already satisfied: thinc<8.4.0,>=8.3.4 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.12/dist-packages (from spa
Requirement already satisfied: weasel<0.5.0,>=0.4.2 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: typer-slim<1.0.0,>=0.3.0 in /usr/local/lib/python3.12/dist-packages (from sp
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/python3.12/dist-packages (from spacy)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.12/dist-packages (from spa
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4 in /usr/local/lib/python3.12/dist-packa
Requirement already satisfied: jinja2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.1.6)
Requirement already satisfied: setuptools in /usr/local/lib/python3.12/dist-packages (from spacy) (75.2.0)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (25.
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.12/dist-packages (from panc
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.12/dist-packages (from pandas) (202
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (0
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.12/dist-packages (from matplotli
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotli
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (11.3
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.12/dist-packages (from scikit-learn)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.12/dist-packages (from scikit-learn)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.12/dist-packages (from scikit
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.12/dist-packages (from pyda
Requirement already satisfied: pydantic-core==2.41.4 in /usr/local/lib/python3.12/dist-packages (from pydar
Requirement already satisfied: typing-extensions>=4.14.1 in /usr/local/lib/python3.12/dist-packages (from p
Requirement already satisfied: typing-inspection>=0.4.2 in /usr/local/lib/python3.12/dist-packages (from py
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packages (from python-dateutil>=2
Requirement already satisfied: charset_normalizer<4,>=2 in /usr/local/lib/python3.12/dist-packages (from re
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.12/dist-packages (from requests
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.12/dist-packages (from requests
Requirement already satisfied: blis<1.4.0,>=1.3.0 in /usr/local/lib/python3.12/dist-packages (from thinc<8.
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/lib/python3.12/dist-packages (from th
Requirement already satisfied: click>=8.0.0 in /usr/local/lib/python3.12/dist-packages (from typer-slim<1.0
Requirement already satisfied: cloudpathlib<1.0.0,>=0.7.0 in /usr/local/lib/python3.12/dist-packages (from
Requirement already satisfied: smart-open<8.0.0,>=5.2.1 in /usr/local/lib/python3.12/dist-packages (from we
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.12/dist-packages (from jinja2->spa
Requirement already satisfied: wrapt in /usr/local/lib/python3.12/dist-packages (from smart-open<8.0.0,>=5.
Downloading emoji-2.15.0-py3-none-any.whl (608 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 608.4/608.4 kB 20.8 MB/s eta 0:00:00
Installing collected packages: emoji
Successfully installed emoji-2.15.0
Collecting en-core-web-sm==3.8.0
  Downloading https://github.com/explosion/spacy-models/releases/download/en_core_web_sm-3.8.0/en_core_web_
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.8/12.8 MB 26.3 MB/s eta 0:00:00
✓ Download and installation successful
You can now load the package via spacy.load('en_core_web_sm')
⚠ Restart to reload dependencies
If you are in a Jupyter or Colab notebook, you may need to restart Python in
order to load all the package's dependencies. You can do this by selecting the
'Restart kernel' or 'Restart runtime' option.
```

```
import spacy
import pandas as pd
```

```
import numpy as np
import re
import emoji
import matplotlib.pyplot as plt
import seaborn as sns
from collections import Counter
```

2. Load the Twitter US Airline Sentiment dataset

```
# Load dataset (update path if needed)
df = pd.read_csv("Tweets.csv")

df.head()
```

	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confi
0	570306133677760513	neutral	1.0000	NaN	
1	570301130888122368	positive	0.3486	NaN	(
2	570301083672813571	neutral	0.6837	NaN	
3	570301031407624196	negative	1.0000	Bad Flight	(
4	570300817074462722	negative	1.0000	Can't Tell	

Next steps: Generate code with df | New interactive sheet

3. Select tweet text and sentiment columns and remove missing values

```
df = df[['text', 'airline_sentiment']]
df.dropna(inplace=True)

print(df.shape)
```

```
(14640, 2)
```

4. Clean tweets

```
def clean_tweet(text):
    text = text.lower()
    text = re.sub(r"http\S+|www\S+", "", text)      # remove URLs
    text = re.sub(r"@\w+", "", text)                # remove mentions
    text = emoji.replace_emoji(text, replace="")   # remove emojis
    text = re.sub(r"#", "", text)                   # remove hashtag symbol
    text = re.sub(r"[^a-z\s]", "", text)            # remove special characters
    text = re.sub(r"\s+", " ", text).strip()
```

```
    return text

df['clean_text'] = df['text'].apply(clean_tweet)
```

5. Create a cleaned tweet corpus

```
cleaned_corpus = df['clean_text'].tolist()

cleaned_corpus[:5]
```

```
['what said',
 'plus youve added commercials to the experience tacky',
 'i didnt today must mean i need to take another trip',
 'its really aggressive to blast obnoxious entertainment in your guests faces amp they have little
recourse',
 'and its a really big bad thing about it']
```

6. Initialize the spaCy NLP pipeline

```
nlp = spacy.load("en_core_web_sm", disable=["ner", "parser"])
```

7. Create and add a custom spaCy pipeline component to detect hashtags

```
from spacy.language import Language

@Language.component("hashtag_detector")
def hashtag_detector(doc):
    doc._.hashtags = [token.text for token in doc if token.text.startswith("#")]
    return doc

# Add extension attribute
spacy.tokens.Doc.set_extension("hashtags", default=[], force=True)

# Add component to pipeline
nlp.add_pipe("hashtag_detector", last=True)

print(nlp.pipe_names)
```

```
['tok2vec', 'tagger', 'attribute_ruler', 'lemmatizer', 'hashtag_detector']
```

8. Process the cleaned tweets using the customized spaCy pipeline

```
docs = list(nlp.pipe(cleaned_corpus, batch_size=1000))
```

9. Extract lemmas and part-of-speech tags from processed tweets

```
lemma_pos_data = []

for doc in docs:
    lemma_pos_data.append([
        (token.lemma_, token.pos_)
        for token in doc
        if not token.is_stop and not token.is_punct
    ])
```

```
lemma_pos_data[:2]
```

```
[[('say', 'VERB')],
 [('plus', 'CCONJ'),
  ('ve', 'VERB'),
  ('add', 'VERB'),
  ('commercial', 'NOUN'),
  ('experience', 'NOUN'),
  ('tacky', 'ADV')]]
```

10. Extract hashtags from original tweets and compute their frequencies

```
def extract_hashtags(text):
    return re.findall(r"#\w+", text.lower())

all_hashtags = []

for tweet in df['text']:
    all_hashtags.extend(extract_hashtags(tweet))

hashtag_freq = Counter(all_hashtags)
hashtag_freq.most_common(10)
```

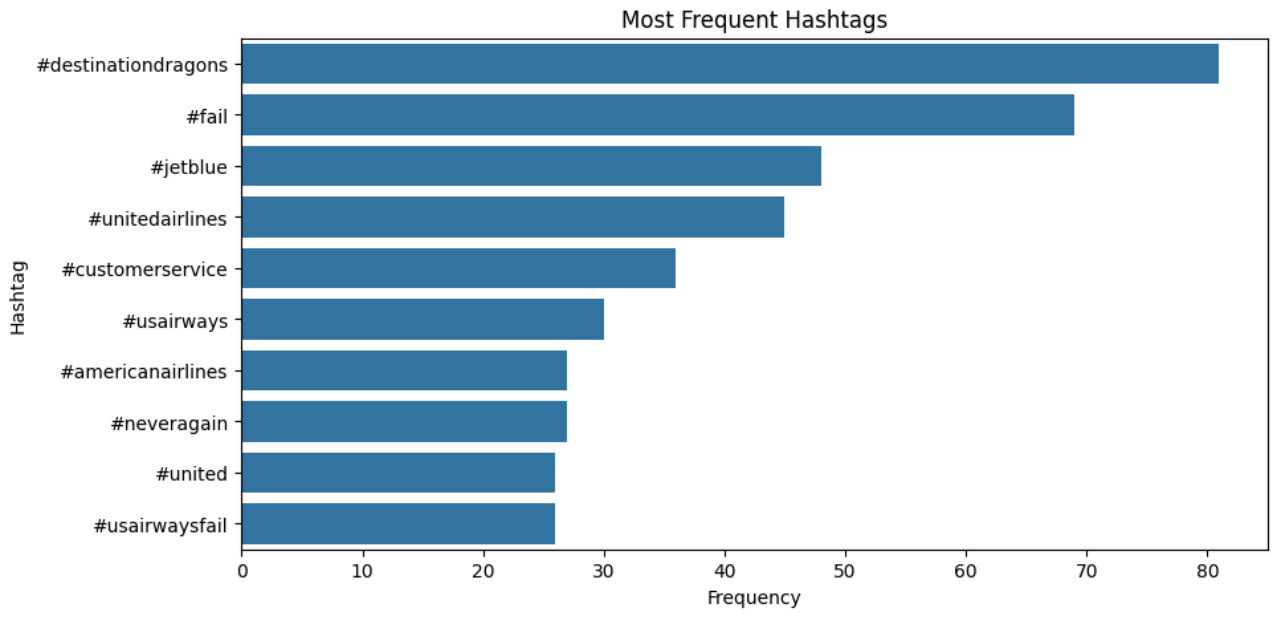
```
[('#destinationdragons', 81),
 ('#fail', 69),
 ('#jetblue', 48),
 ('#unitedairlines', 45),
 ('#customerservice', 36),
 ('#usairways', 30),
 ('#americanairlines', 27),
 ('#neveragain', 27),
 ('#united', 26),
 ('#usairwaysfail', 26)]
```

11. Visualize the most frequent hashtags

```
top_hashtags = hashtag_freq.most_common(10)

hashtags, counts = zip(*top_hashtags)

plt.figure(figsize=(10,5))
sns.barplot(x=list(counts), y=list(hashtags))
plt.title("Most Frequent Hashtags")
plt.xlabel("Frequency")
plt.ylabel("Hashtag")
plt.show()
```



12. Filter negative tweets and visualize their POS tag distribution

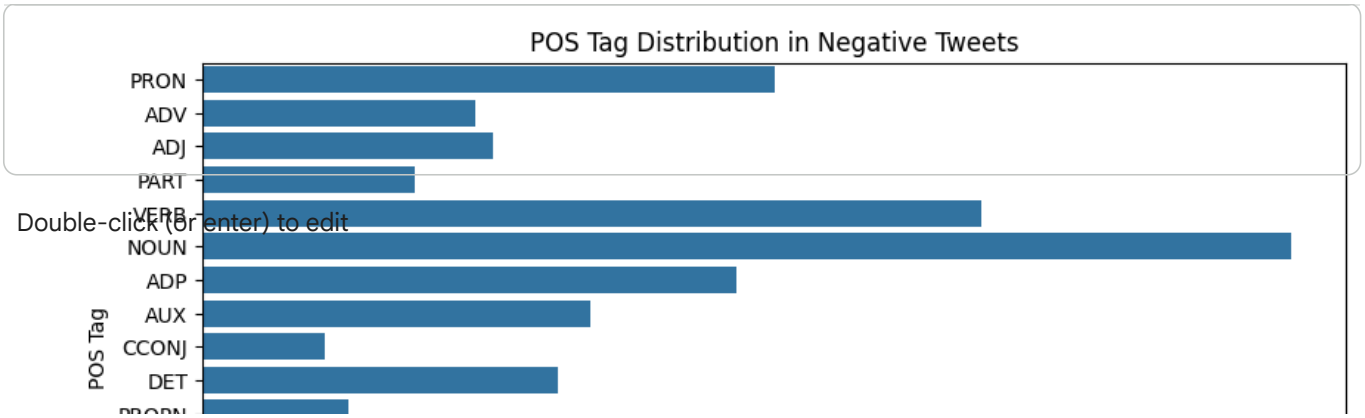
```
negative_docs = [
    doc for doc, sentiment in zip(docs, df['airline_sentiment'])
    if sentiment == "negative"
]

pos_tags = []

for doc in negative_docs:
    pos_tags.extend([token.pos_ for token in doc if not token.is_punct])

pos_freq = Counter(pos_tags)

plt.figure(figsize=(10,5))
sns.barplot(x=list(pos_freq.values()), y=list(pos_freq.keys()))
plt.title("POS Tag Distribution in Negative Tweets")
plt.xlabel("Frequency")
plt.ylabel("POS Tag")
plt.show()
```



Double-click (or enter) to edit