Part 1: Analyzing Word Frequencies in Executive Order Disposition Notes and Titles

Aniket Kamat Yuyang Wu Brycen Manners
Soohyun Kim

Wednesday 17th December, 2025

We start by importing the necessary packages, reading in the data, and verifying/updating the format of the data.

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from collections import Counter
import spacy
```

```
EO = pd.read_csv("executive_orders.csv")
```

```
import EO_processing.verify as verify
import EO_processing.additions as add
```

```
count_type_incorrect = verify.verify_type(EO, 'Presidential Document')
if count_type_incorrect > 0:
    EO_type_verified = verify.fix_type(EO, 'Presidential Document')
    print('Dataframe modified to ensure type consistency.')
else:
    EO_type_verified = EO
    print('Types are consistent.')

count_subtype_incorrect = verify.verify_subtype(EO_type_verified, 'Executive Order')
if count_subtype_incorrect > 0:
    EO_subtype_verified = verify.fix_subtype(EO_type_verified, 'Executive Order')
    print('Dataframe modified to ensure subtype consistency.')
else:
    EO_subtype_verified = EO_type_verified
    print('Subtypes are consistent.')
EO_with_years = add.add_years(EO_subtype_verified, 'signing_date')
```

```
Types are consistent.
Subtypes are consistent.
```

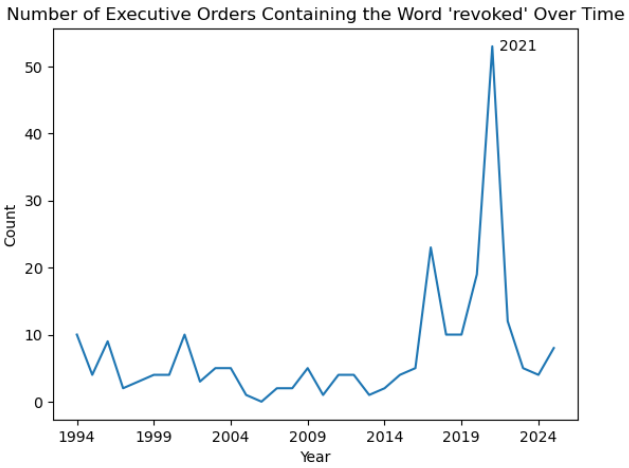
1 Frequency of “Revoked” in EO Disposition Notes

```
add.add_revoked_flag(EO_with_years)
count_revoked = EO_with_years.groupby("year")["is_revoked"].sum().reset_index()
count_revoked = count_revoked.reset_index()

max_revoke_count = count_revoked["is_revoked"].max()
max_revoked = count_revoked[count_revoked["is_revoked"] == max_revoke_count]

sns.lineplot(data=count_revoked, x="year", y="is_revoked")

plt.title("Number of Executive Orders Containing the Word 'revoked' Over Time")
plt.xlabel("Year")
plt.ylabel("Count")
plt.xticks(count_revoked['year'][::5])
plt.annotate(
    f"{int(max_revoked['year'].iloc[0])}",
    xy=(max_revoked['year'].iloc[0], max_revoke_count),
    xytext=(5, 5),
    textcoords="offset points",
    ha="left",
    va="top",
    fontsize=10,
    fontweight="regular"
)
plt.savefig("outputs/EO_revoked.png")
plt.show()
```



2 Frequency of “Amends” in EO Disposition Notes

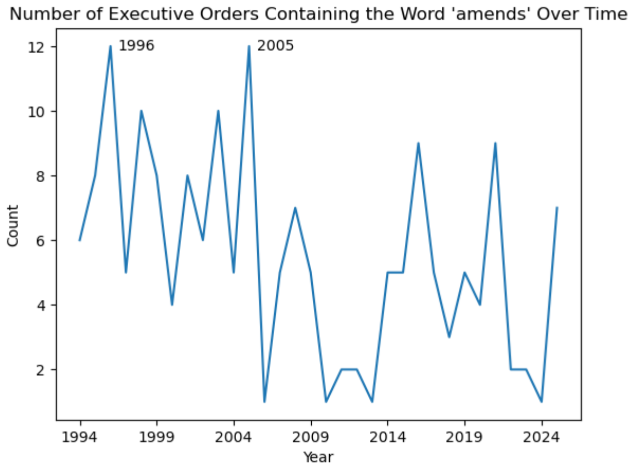
```
add.add_amends_flag(EO_with_years)
```

```
count_amends = EO_with_years.groupby("year")["is_amendment"].sum().reset_index()
count_amends = count_amends.reset_index()

max_amends_count = count_amends["is_amendment"].max()
max_amends = count_amends[count_amends["is_amendment"] == max_amends_count]

sns.lineplot(data=count_amends, x="year", y="is_amendment")

plt.title("Number of Executive Orders Containing the Word 'amends' Over Time")
plt.xlabel("Year")
plt.ylabel("Count")
plt.xticks(count_amends['year'][::5])
for i in range(len(max_amends)):
    plt.annotate(
        f"{int(max_amends['year'].iloc[i - 1])}",
        xy=(max_amends['year'].iloc[i - 1], max_amends_count),
        xytext=(5, 5),
        textcoords="offset points",
        ha="left",
        va="top",
        fontsize=10,
        fontweight="regular"
    )
plt.savefig("outputs/EO_amends.png")
plt.show()
```

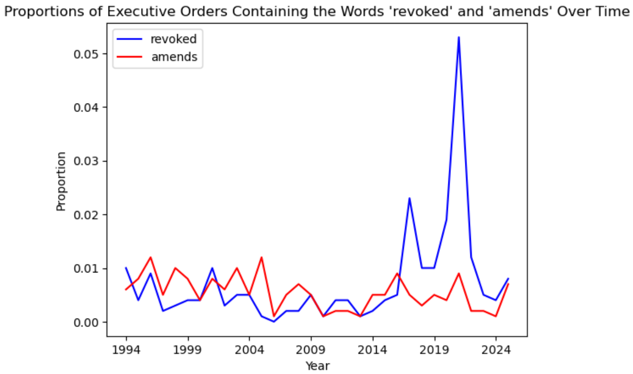


3 Frequency of Amends vs Revoked over time

```
count_revoked['prop_revoked'] = count_revoked['is_revoked'] / len(EO_with_years)
count_amends['prop_amends'] = count_amends['is_amendment'] / len(EO_with_years)
```

```
sns.lineplot(data=count_revoked, x="year", y="prop_revoked", color="blue", label="revoked")
sns.lineplot(data=count_amends, x="year", y="prop_amends", color="red", label="amends")

plt.title("Proportions of Executive Orders Containing the Words 'revoked' and 'amends' Over
plt.xlabel("Year")
plt.ylabel("Proportion")
plt.xticks(count_revoked['year'][::5])
plt.savefig(
    "outputs/EO_proportion_revoked_amends.png",
    bbox_inches="tight"
)
plt.show()
```



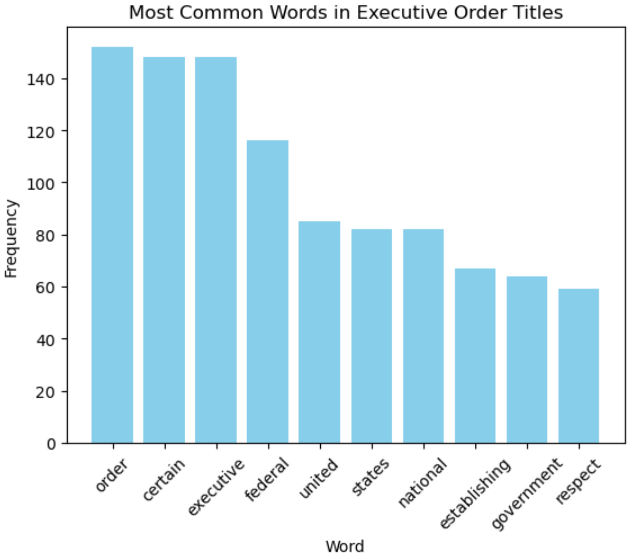
4 Most Common Tokens in EO Titles

```
nlp = spacy.load("en_core_web_sm")

all_titles = " ".join(EO['title'].tolist())

doc = nlp(all_titles)
words = [token.text.lower() for token in doc if not token.is_stop and token.is_alpha]
word_counts = Counter(words)
common_words = word_counts.most_common(10)
words, counts = zip(*common_words)

plt.bar(words, counts, color='skyblue')
plt.xlabel('Word')
plt.ylabel('Frequency')
plt.title('Most Common Words in Executive Order Titles')
plt.xticks(rotation=45)
plt.show()
```


Most Common Words in Executive Order Titles
Frequency
0
20
40
60
80
100
120
140
order
certain
executive
federal
united
states
national
establishing
government
respect
Word