Quinn Underriner HW3

In [194]:
```
import pandas as pd
import numpy as np
import nltk
import unicodedata
import sys
import re
import os
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.decomposition import LatentDirichletAllocation as LDA
from nltk.stem import PorterStemmer
from afinn import Afinn

from sklearn.feature_extraction.text import CountVectorizer
import seaborn as sns

from PIL import Image
import matplotlib.pyplot as plt
```

1. Load the platforms.csv file containing the 2016 Democratic and Republican party platforms. Note the 2X2 format, where each row is a document, with the party recorded as a separate feature. Also, load the individual party .txt files as a corpus.

In [195]:
```
platforms = pd.read_csv("platforms.csv")
```

In [315]:
```
with open("d16.txt") as f:
    dems_raw = f.read()
```

In [316]:
```
with open("r16.txt") as f:
    repubs_raw = f.read()
```

1. Create a document-term matrix and preprocess the platforms by the following criteria (at a minimum): a. Convert to lowercase b. Remove the stopwords c. Remove the numbers d. Remove all punctuation

In [220]:
```
def keep_chr(char):
    return (unicodedata.category(char).startswith('P'))

PUNCTUATION = " ".join(
    [chr(i) for i in range(sys.maxunicode) if keep_chr(chr(i))])

stop_words = set(stopwords.words('english'))
```

In [223]:
```
def preprocess(df):
    df = df.lower()
    df = re.sub(r'\d+', '', df)
    porter = PorterStemmer()

    clean_df = []
    df_split = df.split()
    for i in df_split:
        i = i.strip(PUNCTUATION)
        if i in stop_words:
            continue
        if len(i) == 0:
            continue
        i = porter.stem(i)
        clean_df.append(i)
    return clean_df
```

In [317]:
```
dems = preprocess(dems_raw)
```

In [318]:
```
repubs = preprocess(repubs_raw)
```

In [319]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
import pandas as pd

vect = TfidfVectorizer()
tfidf_matrix = vect.fit_transform(dems)
df = pd.DataFrame(tfidf_matrix.toarray(), columns = vect.get_feature_nam
es())
print(df)
```

```
        aapi  abandon  abid  abil  abl  abolish  aborigin  abort  aborti
on  \
0        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
1        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
2        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
3        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
4        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
5        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
6        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
7        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
8        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
9        0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
10       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
11       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
12       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
13       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
14       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
15       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
16       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
17       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
18       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
19       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
20       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
21       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
22       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
23       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
24       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
25       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
26       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
27       0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
```

```
0.0
28      0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
29      0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
...     ...     ...   ...   ...  ...     ...      ...    ...
...
15325   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15326   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15327   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15328   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15329   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15330   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15331   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15332   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15333   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15334   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15335   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15336   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15337   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15338   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15339   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15340   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15341   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15342   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15343   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15344   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15345   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15346   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15347   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15348   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
15349   0.0     0.0   0.0   0.0  0.0     0.0      0.0    0.0
0.0
```

```
15350   0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
15351   0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
15352   0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
15353   0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0
15354   0.0      0.0   0.0   0.0  0.0      0.0       0.0    0.0
0.0

       abroad  ...  yezidi  yield  york  young  youth  zealand  zero  z
ika  \
0         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
1         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
2         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
3         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
4         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
5         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
6         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
7         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
8         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
9         0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
10        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
11        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
12        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
13        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
14        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
15        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
16        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
17        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
18        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
19        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
20        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
21        0.0  ...     0.0    0.0   0.0    0.0    0.0      0.0   0.0
0.0
```

```
22          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
23          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
24          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
25          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
26          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
27          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
28          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
29          0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
...         ...  ...     ...     ...    ...     ...     ...       ...    ...
...
15325       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15326       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15327       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15328       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15329       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15330       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15331       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15332       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15333       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15334       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15335       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15336       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15337       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15338       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15339       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15340       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15341       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15342       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15343       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15344       0.0  ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
```

```
0.0
15345      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15346      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15347      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15348      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15349      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15350      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15351      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15352      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15353      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0
15354      0.0   ...     0.0     0.0    0.0     0.0     0.0       0.0    0.0
0.0

       zip  zone
0      0.0   0.0
1      0.0   0.0
2      0.0   0.0
3      0.0   0.0
4      0.0   0.0
5      0.0   0.0
6      0.0   0.0
7      0.0   0.0
8      0.0   0.0
9      0.0   0.0
10     0.0   0.0
11     0.0   0.0
12     0.0   0.0
13     0.0   0.0
14     0.0   0.0
15     0.0   0.0
16     0.0   0.0
17     0.0   0.0
18     0.0   0.0
19     0.0   0.0
20     0.0   0.0
21     0.0   0.0
22     0.0   0.0
23     0.0   0.0
24     0.0   0.0
25     0.0   0.0
26     0.0   0.0
27     0.0   0.0
28     0.0   0.0
29     0.0   0.0
...    ...   ...
15325  0.0   0.0
15326  0.0   0.0
15327  0.0   0.0
```

```
15328  0.0   0.0
15329  0.0   0.0
15330  0.0   0.0
15331  0.0   0.0
15332  0.0   0.0
15333  0.0   0.0
15334  0.0   0.0
15335  0.0   0.0
15336  0.0   0.0
15337  0.0   0.0
15338  0.0   0.0
15339  0.0   0.0
15340  0.0   0.0
15341  0.0   0.0
15342  0.0   0.0
15343  0.0   0.0
15344  0.0   0.0
15345  0.0   0.0
15346  0.0   0.0
15347  0.0   0.0
15348  0.0   0.0
15349  0.0   0.0
15350  0.0   0.0
15351  0.0   0.0
15352  0.0   0.0
15353  0.0   0.0
15354  0.0   0.0

[15355 rows x 2688 columns]
```

In [320]:
```
vect = TfidfVectorizer()
tfidf_matrix = vect.fit_transform(repubs)
df = pd.DataFrame(tfidf_matrix.toarray(), columns = vect.get_feature_nam
es())
print(df)
```

```
       abandon  abet  abhor  abid  abil  abl  able  abolish  abort  \
0          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
1          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
2          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
3          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
4          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
5          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
6          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
7          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
8          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
9          0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
10         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
11         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
12         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
13         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
14         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
15         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
16         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
17         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
18         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
19         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
21         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
22         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
23         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
24         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
25         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
26         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
27         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
28         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
29         0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
...        ...   ...    ...   ...   ...  ...   ...      ...    ...
20202      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20203      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20204      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20205      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20206      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20207      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20208      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20209      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20210      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20211      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20212      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20213      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20214      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20215      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20216      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20217      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20218      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20219      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20220      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20221      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20222      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20223      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20224      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20225      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20226      0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
```

```
20227       0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20228       0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20229       0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20230       0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0
20231       0.0   0.0    0.0   0.0   0.0  0.0   0.0      0.0    0.0

       abortifaci  ...  yesterday  yet  yield  young  younger  youngste
r  \
0             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
1             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
2             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
3             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
4             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
5             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
6             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
7             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
8             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
9             0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
10            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
11            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
12            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
13            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
14            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
15            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
16            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
17            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
18            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
19            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
21            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
22            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
23            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
24            0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
```

```
0
25          0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
26          0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
27          0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
28          0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
29          0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
...         ...  ...     ...  ...   ...   ...    ...
...
20202       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20203       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20204       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20205       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20206       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20207       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20208       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20209       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20210       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20211       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20212       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20213       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20214       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20215       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20216       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20217       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20218       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20219       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20220       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20221       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20222       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
20223       0.0  ...     0.0  0.0   0.0   0.0    0.0      0.
0
```

```
20224         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20225         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20226         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20227         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20228         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20229         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20230         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0
20231         0.0  ...        0.0  0.0    0.0    0.0      0.0        0.
0

       youth  zika  zip  zone
0        0.0   0.0  0.0   0.0
1        0.0   0.0  0.0   0.0
2        0.0   0.0  0.0   0.0
3        0.0   0.0  0.0   0.0
4        0.0   0.0  0.0   0.0
5        0.0   0.0  0.0   0.0
6        0.0   0.0  0.0   0.0
7        0.0   0.0  0.0   0.0
8        0.0   0.0  0.0   0.0
9        0.0   0.0  0.0   0.0
10       0.0   0.0  0.0   0.0
11       0.0   0.0  0.0   0.0
12       0.0   0.0  0.0   0.0
13       0.0   0.0  0.0   0.0
14       0.0   0.0  0.0   0.0
15       0.0   0.0  0.0   0.0
16       0.0   0.0  0.0   0.0
17       0.0   0.0  0.0   0.0
18       0.0   0.0  0.0   0.0
19       0.0   0.0  0.0   0.0
20       0.0   0.0  0.0   0.0
21       0.0   0.0  0.0   0.0
22       0.0   0.0  0.0   0.0
23       0.0   0.0  0.0   0.0
24       0.0   0.0  0.0   0.0
25       0.0   0.0  0.0   0.0
26       0.0   0.0  0.0   0.0
27       0.0   0.0  0.0   0.0
28       0.0   0.0  0.0   0.0
29       0.0   0.0  0.0   0.0
...      ...   ...  ...   ...
20202    0.0   0.0  0.0   0.0
20203    0.0   0.0  0.0   0.0
20204    0.0   0.0  0.0   0.0
20205    0.0   0.0  0.0   0.0
20206    0.0   0.0  0.0   0.0
20207    0.0   0.0  0.0   0.0
20208    0.0   0.0  0.0   0.0
20209    0.0   0.0  0.0   0.0
```

```
20210    0.0    0.0  0.0    0.0
20211    0.0    0.0  0.0    0.0
20212    0.0    0.0  0.0    0.0
20213    0.0    0.0  0.0    0.0
20214    0.0    0.0  0.0    0.0
20215    0.0    0.0  0.0    0.0
20216    0.0    0.0  0.0    0.0
20217    0.0    0.0  0.0    0.0
20218    0.0    0.0  0.0    0.0
20219    0.0    0.0  0.0    0.0
20220    0.0    0.0  0.0    0.0
20221    0.0    0.0  0.0    0.0
20222    0.0    0.0  0.0    0.0
20223    0.0    0.0  0.0    0.0
20224    0.0    0.0  0.0    0.0
20225    0.0    0.0  0.0    0.0
20226    0.0    0.0  0.0    0.0
20227    0.0    0.0  0.0    0.0
20228    0.0    0.0  0.0    0.0
20229    0.0    0.0  0.0    0.0
20230    0.0    0.0  0.0    0.0
20231    0.0    0.0  0.0    0.0

[20232 rows x 3472 columns]
```

1. Visually inspect your cleaned documents by creating a wordcloud for each major party's platform. Based on this naive visualization, offer a few-sentence-description of general patterns you see (e.g., What are commonly used words? What are less commonly used words? Can you get a sense of differences between the parties at this early stage?

In [321]:
```
def create_cloud(df):
    # https://www.datacamp.com/community/tutorials/wordcloud-python
    stop_words = set(stopwords.words('english'))



    text = " ".join(df)

    usa_mask = np.array(Image.open("US_img.png"))
    wordcloud = WordCloud(max_words=100, stopwords=stop_words, \
                  background_color="white", collocations=False,\
                  mask=usa_mask, contour_width=2, contour_color='steelblu
e').generate(text)

    plt.figure(figsize=(25,20))

    plt.imshow(wordcloud, interpolation="bilinear")
    plt.axis("off")
    plt.show()
```

In [228]:
```
#Democratic Cloud
create_cloud(dems)
```

In [229]:
```
#Republican Cloud

create_cloud(repubs)
```

In the Dem word cloud its clear that they have a focus on "health"(care) that one does not see in the republican word cloud. This makes sense as medicare for all (or at least the expansion of health care to more people) is a key platform of the Dem party.

In the republican we see lots of usage of "state" and "feder", which makes sense given conservatives focus on the differences between state and local governmental responsibilities, they would want to be repeatedly delineating these (to a lesser extent, we see this with "private" and "public". In the dem cloud we see a greater emphasis on just public.

Republicans talk about relgion and abortion, while these terms are not common for Dems.

SENTIMENT ANALYSIS

1. Use the “Bing” and “AFINN” dictionaries to calculate the sentiment of each cleaned party platform. Present the results however you’d like (e.g., visually and/or numerically).

In [230]:
```
afinn = Afinn()
dems_sentiment = ' '.join(word for word in dems)
afinn.score(dems_sentiment)
```

Out[230]: 964.0

In [231]:
```
repubs_sentiment = ' '.join(word for word in repubs)
afinn.score(repubs_sentiment)
```

Out[231]: 648.0

In [371]:
```
g = 0
score_list_dems = []
for i in dems:


    g += afinn.score(i)
    score_list_dems.append(afinn.score(i))

score_list_dems[0:25]
```

Out[371]:
```
[0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 1.0,
 2.0,
 -1.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0,
 0.0]
```

In [234]:
```
score_list_dems = pd.Series(score_list_dems)
score_list_dems.value_counts()
```

Out[234]:
```
 0.0    13424
 2.0      698
 1.0      442
-2.0      320
-1.0      293
 3.0      100
-3.0       71
-4.0        7
dtype: int64
```

In [235]:
```
score_list_repubs = pd.Series(score_list_repubs)
score_list_repubs.value_counts()
```

Out[235]:
```
 0.0    18116
 2.0      625
 1.0      557
-2.0      432
-1.0      231
-3.0      144
 3.0      116
 4.0        8
-4.0        3
dtype: int64
```

In [ ]:

The score totals look at each word in the corpus and give them positive or negative values. We see above many words are neutral (given a score of zero) and the total score of each corpus is simply the individual words added up.

Generally, the democrats use more positive words than the republicans, but more significantly perhaps for the score, the republicans use more negative words overall. (-3s and -4s, the strongest negative scores, together total 78 for the democrats and 147 for the republicans).

This makes sense given my understanding of the parties, conservatives by nature will be warning people against assumed threats (immigration, foreign enemies, etc) which involves invoking negative emotions more than liberals will. Liberals tend to depict a positive vision for the future that can be achieved (which would generally involve positive language).

TOPIC MODELS

1. With a general sense of sentiments of the party platforms (i.e., the tones related to how parties talk about their roles in the political future), now explore the topics they are highlighting in their platforms. This will give a sense of the key policy areas they're most interested in. Fit a topic model for each of the major parties (i.e. two topic models) using the latent Dirichlet allocation algorithm, initialized at k = 5 topics as a start. Present the results however you'd like (e.g., visually and/or numerically).

In [323]:
```
#code adopted from https://towardsdatascience.com/end-to-end-topic-model
ing-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0

# considering each paragraph as a separate document
dem_paragraphs = [' '.join(preprocess(d)) for d in dems_raw.split('\n')]

count_data_dems = count_vectorizer.fit_transform(dem_paragraphs)

def print_topics(model, count_vectorizer, n_top_words):
    words = count_vectorizer.get_feature_names()
    for topic_idx, topic in enumerate(model.components_):
        print("\nTopic #%d:" % topic_idx)
        print(" ".join([words[i]
                        for i in topic.argsort()[:-n_top_words - 1:-1
]]))

number_topics = 5
number_words = 10

# LDA model
lda = LDA(n_components=number_topics, n_jobs=-1)
lda.fit(count_data_dems)

print("Topics found via LDA:")
print_topics(lda, count_vectorizer, number_words)
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)

Topics found via LDA:

Topic #0:
laden iran drainag bin restrict depress cybersecur privaci onlin russia

Topic #1:
health care servic american includ democrat hous program fight expand

Topic #2:
financi profit street wall loan law rate consum regul record

Topic #3:
democrat right american peopl protect secur believ countri support glob
al

Topic #4:
democrat support commun educ public american invest nation school make
```

The above is a k=5 model for the democrates. Below is a k=5 model for the republicans

In [357]:
```
rep_paragraphs = [' '.join(preprocess(r)) for r in repubs_raw.split('\n'
)]
count_data_repubs = count_vectorizer.fit_transform(rep_paragraphs)

number_topics = 5
number_words = 10
#LDA model
lda = LDA(n_components=number_topics, n_jobs=-1)
lda.fit(count_data_repubs)

print("Topics found via LDA:")
print_topics(lda, count_vectorizer, number_words)
print("perplexity", lda.perplexity(count_data_repubs))
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)

Topics found via LDA:

Topic #0:
energi tyranni polit america protect freedom nuclear life inalien confr
ont

Topic #1:
state govern nation republican right american law protect support peopl

Topic #2:
feder american tax educ program school student fund privat abort

Topic #3:
opioid amend poverti christian prescript recoveri nearli rent presid co
nstitut

Topic #4:
cyber energi world busi grid nuclear america electr defens russia
perplexity 2042.1385553851233
```

1. Describe the general trends in topics that emerge from this stage. Are the parties focusing on similar or different topics, generally?

Both have a focus on foreign policy and education, although the republicans have a higher foreign policy focus and a linguistic fondness for emphasizing rule of law. The democrats have a more specific policy focus, also speaking about healthcare, and regulating Wall Street. In general the topics associated with republicans are more vague. There is certainly a good deal of overlap in the topics and hard lines are difficult to parse, but even in these naive models we can starts to see the difference between conservative and liberals play out.

1. Fit 6 more topic models at the follow levels of k for each party: 5, 10, 25. Present the results however you'd like (e.g., visually and/or numerically).
2. Calculate the perplexity of each model iteration and describe which technically fits best.

In [358]:
```
#models for republicans
number_words = 10
# Create and fit the LDA model
for i in [5,10,25]:
    lda = LDA(n_components=i, n_jobs=-1)
    lda.fit(count_data_repubs)
    # Print the topics found by the LDA model
    print("Topics found via LDA:")
    print_topics(lda, count_vectorizer, number_words)
    print("perplexity", lda.perplexity(count_data_repubs))
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)

Topics found via LDA:

Topic #0:
right support american state govern nation famili militari human peopl

Topic #1:
energi cyber nuclear defens trade america grid presid busi state

Topic #2:
fund reform feder polit american audit welfar corpor econom great

Topic #3:
law state enforc author court regul presid congress power rule

Topic #4:
govern republican state nation feder american protect right advanc amer
ica
perplexity 2045.3686679947161

/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)
```

```
Topics found via LDA:

Topic #0:
state govern american nation republican right feder protect support peo
pl

Topic #1:
year grid energi million electr democrat presid busi bank speci

Topic #2:
liberti life human captiv endow creator remak appli right truth

Topic #3:
nuclear china law treati presid continu court cyber russia defens

Topic #4:
veteran va israel farm women sacrific best honor agricultur forest

Topic #5:
amend freedom speech protect polit right puls electromagnet christian w
orst

Topic #6:
north preserv korean korea dismantl employe paycheck invit nato broadca
st

Topic #7:
poverti fund academ engin chines approach fifth cumul compar shortfal

Topic #8:
hacker space suicid insecur cybersecur contamin compon information shar
rel

Topic #9:
rebirth rico puerto territori barrasso senat chairman john constitut pr
omis
perplexity 2182.2141236435978
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)
```

```
Topics found via LDA:

Topic #0:
enumer paid argu solut counter divest iraq wrongli champion systemat

Topic #1:
marriag judiciari activist judg court defend obergefel scalia condemn e
lect

Topic #2:
protect abort electromagnet puls firearm infant unborn enforc defi bear

Topic #3:
life inalien declar right equal endow creator liberti remak affirm

Topic #4:
feder guard bank research abort reserv tax fund cell fda

Topic #5:
paycheck stagnant unnecessarili sacrif weak struggl wage suffer becom e
conomi

Topic #6:
govern american republican state nation support law feder peopl right

Topic #7:
peopl want earth respect power countri america expect face fellow

Topic #8:
document believ written endur coven flexibl constitut tax simpl growth

Topic #9:
freedom workforc centuri st workplac kind sidelin chair repres handbook

Topic #10:
energi world busi new cyber presid grid defens agricultur electr

Topic #11:
creat rebuild economi job rel daili chang general relianc flee

Topic #12:
indebted revit violenc etern immigr purs use behavior steer balkan

Topic #13:
poland bike count upset realiti strongest hamilton shot commerci honest

Topic #14:
nearli economi obama person presid recoveri econom poverti compar ameri
can

Topic #15:
judiciari manner starvat treatment sabotag mindset rise federally son w
est

Topic #16:
right constitut amend properti human govern liberti protect combat inte
llectu
```

```
Topic #17:
complic away manageri scandal overpaid funder compos barbar pariah over
du

Topic #18:
region china genocid east middl isi help european vet continu

Topic #19:
dream american restor turnov success potenti vet audit entrench oval

Topic #20:
inconveni shortchang treat equip command chief men necessari number rem
ain

Topic #21:
state unit internet nation nuclear freedom america trade free feder

Topic #22:
sens common approach project start entrepreneurship peonag patron agree
ment nlrb

Topic #23:
chairman john senat barrasso restor dream american socio shut counterfe
it

Topic #24:
titl ix territori excel rico academ puerto display event voluntari
perplexity 2542.5371412176514
```

In [329]:
```
#models for democrates
number_words = 10
# Create and fit the LDA model
count_data_dems = count_vectorizer.fit_transform(dem_paragraphs)

for i in [5, 10, 25]:
    lda = LDA(n_components=i, n_jobs=-1)
    lda.fit(count_data_dems)
    # Print the topics found by the LDA model
    print("Topics found via LDA:")
    print_topics(lda, count_vectorizer, number_words)
    print("perplexity", lda.perplexity(count_data_dems))
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)

Topics found via LDA:

Topic #0:
secur alli right countri democrat nuclear state nato trump donald

Topic #1:
american democrat health nation educ indian support women peopl student

Topic #2:
wage hour caregiv connect number wealth worker minimum close rais

Topic #3:
democrat support american believ health protect right work peopl public

Topic #4:
busi small credit immigr job commun entrepreneurship tax support citi
perplexity 1475.1326230314382

/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)
```

```
Topics found via LDA:

Topic #0:
street wall fix half centuri race financi rein césar park

Topic #1:
loan borrow student space repay nasa bankruptci income discharg default

Topic #2:
cfpb lend consum modern defend predatori bureau traffick slaveri clarif
i

Topic #3:
democrat support american protect right public nation commun invest pro
gram

Topic #4:
bay drainag privaci cybersecur russia onlin close guantánamo bristol pe
bbl

Topic #5:
health democrat american peopl believ state care secur countri econom

Topic #6:
vote elect immigr financ campaign peopl corpor voter broken citizen

Topic #7:
small right busi guarante entrepreneurship civil lesbian gay bisexu lib
erti

Topic #8:
nuclear nato alli continu relationship north weapon push commit attack

Topic #9:
palestinian negoti europ independ capit africa incit isra viabl jerusal
em
perplexity 1599.344186122788

/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)
```

```
Topics found via LDA:

Topic #0:
aids root consent provis promulg firmli businesses vile support hold

Topic #1:
nuclear weapon china iran test north world servic korea includ

Topic #2:
wealth american organ middl believ worker democrat class play econom

Topic #3:
tortur piraci incarcer billion holder hyde nato sow properti renew

Topic #4:
busi small america street credit entrepreneurship rural market wall tax

Topic #5:
nato vote alli attack right elect afghanistan protect articl voter

Topic #6:
servic offer incentiv valid expans metropolitan dioxid unparallel usp e
verglad

Topic #7:
candid excess canada recoveri short waiver promot homeown enhanc raid

Topic #8:
homeless hous million neighborhood weaker wher poorer inequality mak fo
reclosur

Topic #9:
democrat peopl right support health state commun secur care continu

Topic #10:
restrict bay guantánamo drainag effort perfect wrong employe law teache
r

Topic #11:
hous afford fund rental number block immigr wastewat bridg road

Topic #12:
democrat american pay america trade believ make econom global tax

Topic #13:
technolog innov scientist sector rest america continu compet econom sci
enc

Topic #14:
russia restor econom incom rais secur class middl propos variou

Topic #15:
art innov promot technolog cultur nation research heritag artist contri
but

Topic #16:
sharp bear appeal man basest principl candid nomin contrast stand
```

```
Topic #17:
support north syria korea work busi famili america small homeown

Topic #18:
right guarante civil better bisexu lesbian gay transgend peopl societi

Topic #19:
shar retir dignifi security stakehold ponzi starkli scheme refer target

Topic #20:
immigr leader onlin privaci cybersecur pta lawyer soldier heritag know

Topic #21:
union join worker tabl organ war valid simpl bind elig

Topic #22:
punish penalti death basic pipelin lack taxpay exoner reliabl unjust

Topic #23:
democrat public support student american educ school nation health ener
gi

Topic #24:
caregiv reunit wait immigr childcar workforc medic cutting edg enabl
perplexity 1869.9507211312332
```

In both the republican and democratic case, the lowest perplexity is when k=5. This is the best model fit, although the score is relatively close in both cases to that of k=10, suggesting that perhaps the best fit lies in between the two (but closer to 5). 25 topics has a notably terrible score, continuing to provide evidence that these corpi are best described by a relatively low number of topics.

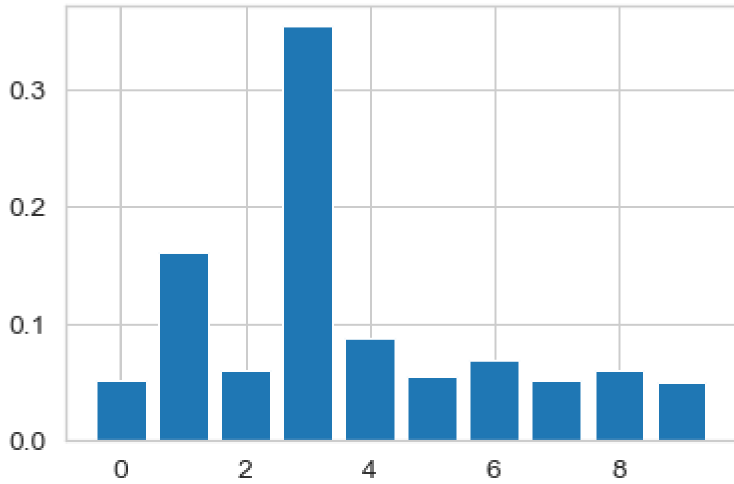
1. Building on the previous question, display a barplot of the k = 10 model for each party, and offer some general inferences as to the main trends that emerge. Are there similar themes between the parties? Do you think k = 10 likely picks up differences more efficiently? Why or why not?

In [340]:
```
lda = LDA(n_components=10, n_jobs=-1)
lda.fit(count_data_dems)
print_topics(lda, count_vectorizer, 10)
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)

Topic #0:
punish penalti death immeasur marin view sacrif brave navi admir

Topic #1:
american democrat pay make job famili tax support work worker

Topic #2:
trade infrastructur law protect requir unfair restrict employe bay numb
er

Topic #3:
democrat support health american peopl nation right believ commun count
ri

Topic #4:
nuclear alli secur nato israel continu push trump donald state

Topic #5:
servic manufactur postal effici energi internet offer bank deliveri con
nect

Topic #6:
right vote elect voter peopl protect financ guarante better campaign

Topic #7:
half centuri race onlin cybersecur privaci lewi dolor rosa sat

Topic #8:
immigr leav paid famili fix member commun contribut societi medic

Topic #9:
senior furthermor neglect older roof stay greatest choos elder buy
```

In [343]:
```
plt.bar(x=np.arange(10), height=lda.transform(count_data_dems).mean(axis
=0))
```

Out[343]: <BarContainer object of 10 artists>



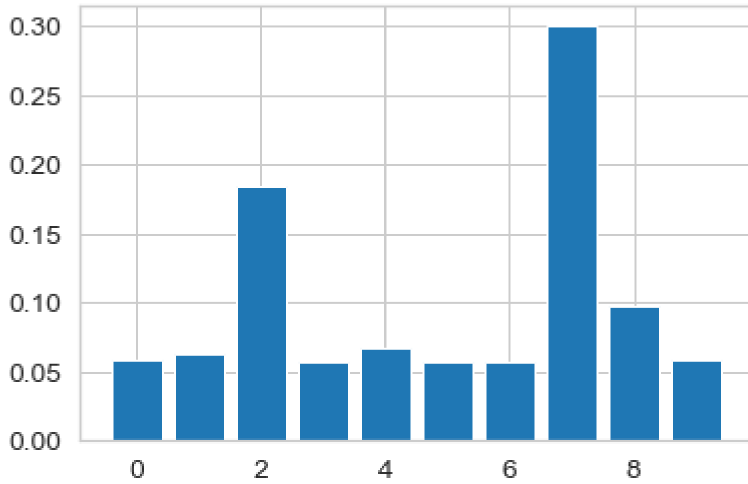
The most important democratic topic involves healthcare and generally positive words. The second most important is related to workers adn familys economic devleopment.

In [365]:
```
count_data_repubs = count_vectorizer.fit_transform(rep_paragraphs)
lda = LDA(n_components=10, n_jobs=-1)
lda.fit(count_data_repubs)
print_topics(lda, count_vectorizer, 10)
```

```
/Users/quinnunderriner/anaconda3/lib/python3.7/site-packages/sklearn/de
composition/online_lda.py:536: DeprecationWarning: The default value fo
r 'learning_method' will be changed from 'online' to 'batch' in the rel
ease 0.20. This warning was introduced in 0.18.
  DeprecationWarning)

Topic #0:
europ nato middl east eastern puls electromagnet ukrain pakistan genoci
d

Topic #1:
taiwan speci agreement esa trade list democraci cemeteri fair missil

Topic #2:
republican govern right nation secur econom year presid protect privat

Topic #3:
spe high rebirth corpor regard railroad interc nowher northeast califor
nia

Topic #4:
constitut execut treati unit senat tyranni presid agreement allianc bin
d

Topic #5:
histor judiciari defend liberty katrina bled sentinel commensur hurrica
n town

Topic #6:
world insecur cybersecur cuba evil wish friendship cuban solidar muslim

Topic #7:
state american support nation govern feder law protect peopl right

Topic #8:
abort healthcar feder fund territori care american tax mental research

Topic #9:
space arab spring isi hope seen capabl fear field murder
```

In [366]:
```
plt.bar(x=np.arange(10), height=lda.transform(count_data_repubs).mean(axis=0))
```

Out[366]: <BarContainer object of 10 artists>



The most important republican topics involve nationalism and economic development, and the second most important topics evokes america, the rule of law and security.

CONCLUSION: Per the opening question, based on your analyses (including exploring party brands, general tones/sentiments, political outlook, and policy priorities), which party would you support in the 2020 election (again, this is hypothetical)?

The more positive tone, and more specific policy priorities (espeically relating to healthcare, an issue very important to me as a voter) would cause me to support the democratic party in the 2020 elections. The focus on language of force rather than language of productivity and solidarity instead of the language of force makes the platform of the democrats clearly more appleaing to me.