Peeraya Khantaruangsakul 63070501054

Homework1 : US_gen-dabate

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import re
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
nltk.download('stopwords')
from nltk.util import ngrams
from collections import Counter
from wordcloud import WordCloud
from textacy.extract.kwic import keyword_in_context
import random
#do not show warnings
import warnings
warnings.filterwarnings("ignore")
```

Import Dataset

```
df = pd.read_table('https://github.com/blueprints-for-text-analytics-python/blueprints-text/raw/193c79c7d94973f2398e67da8d20bf7a535f7f40/data/un-general-debates/un-general-debates-blueprint.csv.gz', compression='gzip', sep=',', quotechar='"', error_bad_lines=False)
```

Filter country_name = United States

```
df_US = df[df['country_name'] == 'United States']
```

Clean text

```
# Change lowercase
df_US['text'] = df_US['text'].str.lower()
# remove special Character
def remove_special_characters(text):
```

```
    return re.sub('[^A-Za-z0-9]+', ' ', text)
df_US['text'] = df_US['text'].apply(remove_special_characters).str.strip()
stopwords = set(nltk.corpus.stopwords.words('english'))   # English stopword
# Remove stopwords
def remove_stop(tokens):
    return [w for w in tokens if w not in stopwords]
df_US['tokens'] = df_US['text'].apply(word_tokenize)
df_US['tokens_stopword'] = df_US['tokens'].apply(remove_stop)
# Length after remove stopword
df_US['length_stopword'] = df_US['tokens_stopword'].str.len()
# Count frequency
tokens_stop = df_US['tokens_stopword'].explode().values
counter_stop = Counter(tokens_stop)
counter_stop.most_common(10)
```

Bigram

```
# Find all bigrams and concate 2 word to bigrams
token_words = [x for x in ngrams(tokens_stop, n = 2)]
concatenated_bigrams = [' '.join(x) for x in token_words]
```

Find the top 10 word bigram from US Genral Debates of years (remove if bigram contain stopwords)

Code	Result
counter_bigrams = Counter(concatenated_bigrams) counter_bigrams.most_common(10)	[('united nations', 786), ('united states', 700), ('human rights', 170), ('soviet union', 140), ('security council', 135), ('middle east', 121), ('nuclear weapons', 115), ('general assembly', 108), ('let us', 98), ('years ago', 79)]

Word Cloud

```
wc = WordCloud(background_color="black", max_words=100)
wc.generate_from_frequencies(counter_bigrams)
plt.figure(figsize=(10,6))
```

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

Create a bigram word cloud of the US General Debates dataset (remove if bigram contain stopwprds)

Trend

```
# Find Top10 BIgramost frequency
top_10_bigrams = [bigram[0] for bigram in counter_bigrams.most_common(10)]
```

Create a trend graph showing the bigram and word trend of "sustainable", "poverty", "wars" and 3 others of your choice

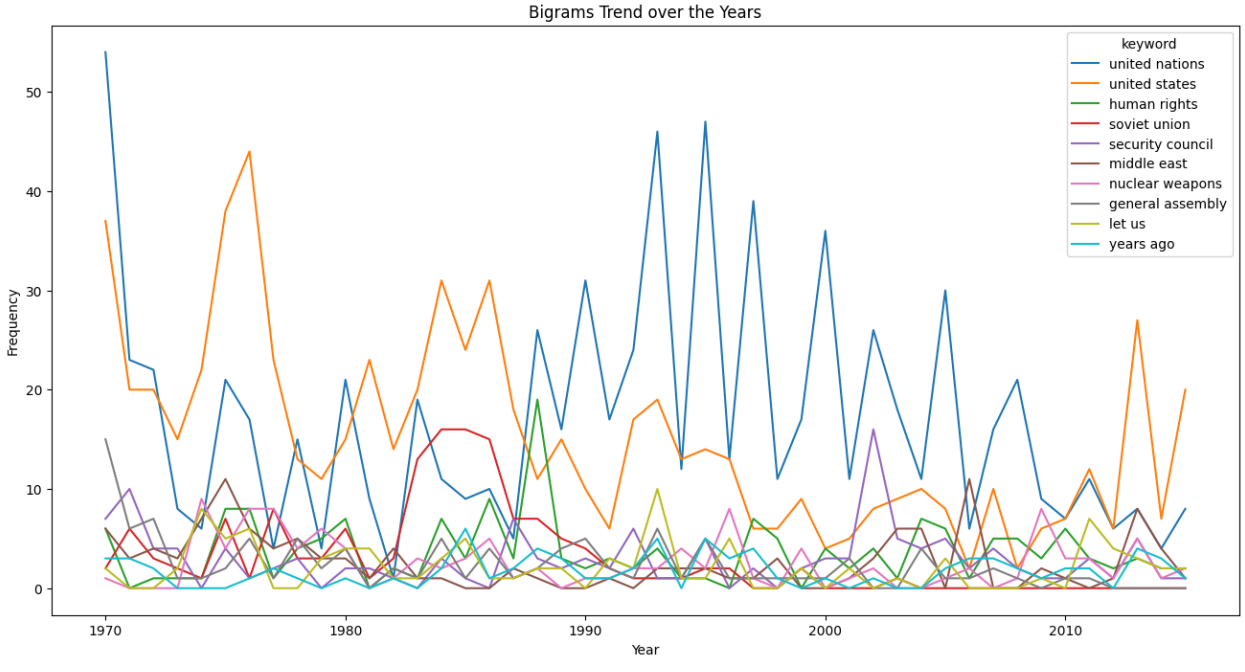
Top10 Bigrams Trend of US

```
def kwic(doc_series, keyword, window=35, print_samples=5):
  def add_kwic(text):
    kwic_list.extend(keyword_in_context(text,keyword, ignore_case=True, window_width=window))
  kwic_list = []
  doc_series.map(add_kwic)
  if print_samples is None or print_samples==0:
    return kwic_list
```

```
  else:
    return len(kwic_list)
```

```
keywords = top_10_bigrams
bigrams_data = []
for year in df['year'].unique():
  for keyword in keywords:
    bigrams_data.append([year, keyword, kwic(df_US[df_US['year']==year]['text'], keyword, print_samples=1)])
bigrams_df = pd.DataFrame(bigrams_data, columns=['year', 'keyword', 'count'])
bigrams_df = bigrams_df.pivot(index='year', columns='keyword', values='count')
bigrams_df = bigrams_df.reset_index()
```

```
# Graph about Top10 bigrams of US
bigrams_df.plot(x='year', y=keywords, kind='line', figsize=(16, 8))
plt.xlabel('Year')
plt.ylabel('Frequency')
plt.title('Bigrams Trend over the Years')
plt.show()
```



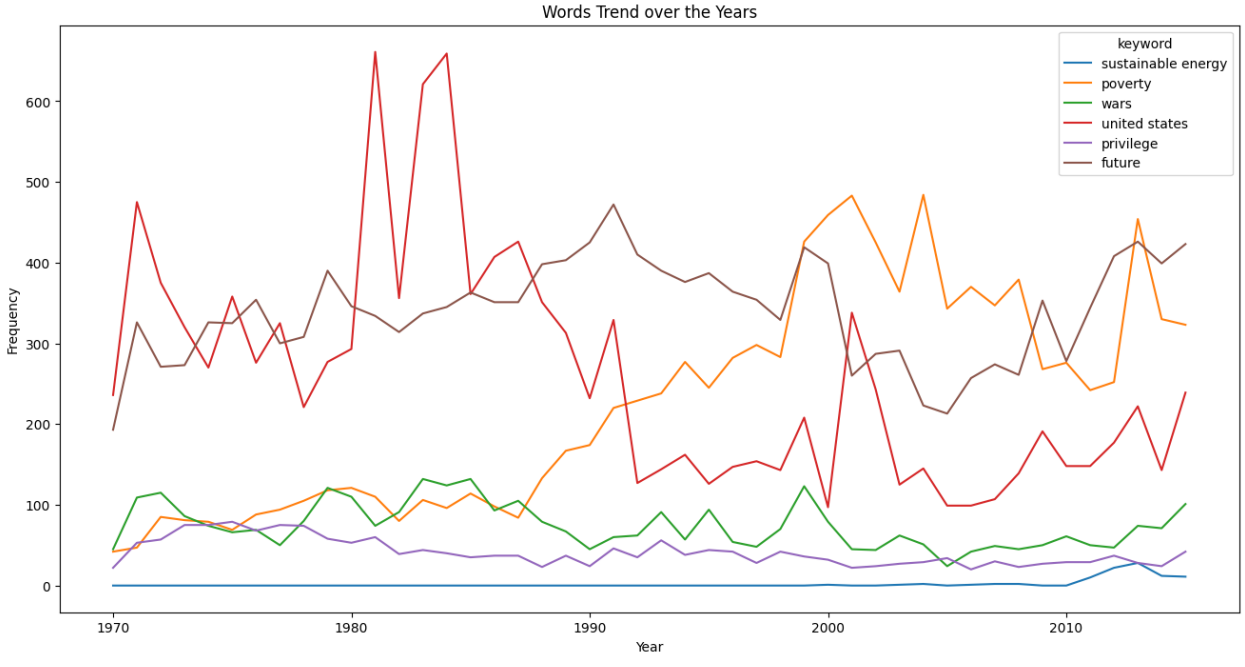
Word Trend Globle

```
key = ['sustainable energy', 'poverty', 'wars', 'united states', 'privilege', 'future']
word_data = []
```

```
for year in df['year'].unique():
  for keyword in key:
    word_data.append([year, keyword, kwic(df[df['year']==year]['text'], keyword, print_samples=1)])

trend_df = pd.DataFrame(word_data, columns=['year', 'keyword', 'count'])
trend_df = trend_df.pivot(index='year', columns='keyword', values='count')
trend_df = trend_df.reset_index()
```

```
trend_df.plot(x='year', y=key, kind='line', figsize=(16, 8))
plt.xlabel('Year')
plt.ylabel('Frequency')
plt.title('Words Trend over the Years')
plt.show()
```



My code in Colab https://colab.research.google.com/drive/1Xd50aXmpkrUh8RTFE3w7xEiHkWK_f2P8?usp=sharing