Vietnam History discussions on Stack Exchange History

Hoang Long Tran

2024-05-19

Table of contents

Student information

Task: 8HD - Data Cleansing and Text Analysis Challenge

Name: Hoang Long Tran

ID: 223128143

Email: s223128143@deakin.edu.au

Undergraduate: SIT220

Data Preparation

Importing the libraries

```
# Importing all the libraries
import re
import os
import numpy as np
import pandas as pd
from tabulate import tabulate
import seaborn as sns
import matplotlib.pyplot as plt
import nltk
import spacy
nltk.download('stopwords')
nltk.download('punkt')
nltk.download('wordnet')
from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from collections import Counter
from wordcloud import WordCloud
from typing import Dict, List
from IPython.display import Image, display, HTML, clear_output
import folium
import openai
from sentence_transformers import SentenceTransformer
from bertopic import BERTopic
from bertopic.representation import KeyBERTInspired,
↪  MaximalMarginalRelevance, OpenAI, PartOfSpeech
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\tomde\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\tomde\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\tomde\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Convert `XML` to `csv`

StackExchange uses `XML` file, so we would need to convert them to `csv` for easier use.

```
<row>
    <name>John</name>
    <age>30</age>
    <city>New York</city>
</row>
```

- **Elements**: parent element is `<row>`. Child element is `<name>`, `<age>` and `<city>`.Each child element contains text content or attributes of the record.
- **Tag**: a tag has 2 part, opening `<row>` and closing `</row>`. There are 4 tag in this case, `<row> </row>` are tags representing parent element, `<name> </name>`, `<age> </age>`, `<city> </city>` are children tags.
- **Event**: events are points where the parser execute actions like extracting data or processing elements.

```python
def xml_to_df(xml_file, tag):
    """
    Parse XML file iteratively via tag and output dictionary. Each dictionary
 ↪  is a data from XML row.
    """

    context = etree.iterparse(xml_file, events=('end',), tag=tag) # the event
 ↪  starts parsing the end of a tag
    for event, element in context:
        record = {child.tag: child.text for child in element} # loop through
 ↪  the child element
        record.update(element.attrib) # update text content of child element
        yield record # return the dictionary and not holding in memory
```

```
        # Clear the element to free up space
        element.clear()

# # Badges
# xml_file = 'Badges.xml'
# data_tag = 'row'
# Badges = pd.DataFrame(xml_to_df(xml_file, data_tag))
# Badges.to_csv('Badges.csv', index=False)

# # Comments
# xml_file = 'Comments.xml'
# Comments = pd.DataFrame(xml_to_df(xml_file, data_tag))
# Comments.to_csv('Comments.csv', index=False)

# # PostHistory
# xml_file = 'PostHistory.xml'
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('PostHistory.csv', index=False)

# # Postlinks
# xml_file = 'Postlinks.xml'
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('Postlinks.csv', index=False)

# # PostHistory
# xml_file = 'Postlinks.xml'
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('Postlinks.csv', index=False)

# # Posts
# xml_file = 'Posts.xml'
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('Posts.csv', index=False)

# # Tags
# xml_file = 'Tags.xml'
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('Tags.csv', index=False)

# # Users
# xml_file = 'Users.xml'
```

```
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('Users.csv', index=False)

# # Votes
# xml_file = 'Votes.xml'
# PostHistory = pd.DataFrame(xml_to_df(xml_file, data_tag))
# PostHistory.to_csv('Votes.csv', index=False)
```

To export a `xml` file, I use a function that parse XML file iteratively via tag and outputing a dictionary. Each dictionary is a data from XML row.

Data Cleaning

I will only be using the `Post`, `Comment` and `User` csv

```
Comments = pd.read_csv('Comments.csv')
Posts = pd.read_csv('Posts.csv')
Users = pd.read_csv('Users.csv')
```

Group `Post` and `Comments` dataset

```
# Merge the two dataframes
df1 = pd.merge(Posts, Comments, left_on='Id', right_on = 'PostId',
 ↪  how='left')

# Re-arange the columns
df2 = df1.drop(['PostId', 'Score_y', 'Score_x', 'CreationDate_y',
 ↪  'ClosedDate', 'ParentId', 'FavoriteCount', 'LastEditorDisplayName',
 ↪  'CommunityOwnedDate', 'Id_y', 'Score_y', 'LastEditorUserId',
 ↪  'CreationDate_y'], axis=1)
df2.rename({'Id_x':'PostId', 'CreationDate_x':'CreationDate', 'Body':'Post',
 ↪  'Text':'Comment',
            'UserId':'UserCommentId', 'OwnerUserId':'PostUserId'}, axis=1,
             ↪  inplace=True)

df2 = df2[['PostId', 'Post', 'PostUserId', 'Comment', 'UserCommentId',
 ↪  'CreationDate', 'LastActivityDate', 'Tags']] # Choosing specific columns
```

```
# display(df2.head(), df2.shape)
print(df2.head(), df2.shape)
```

```
   PostId                                               Post  PostUserId  \
0       1  <p>What factors related to the Eastern Crisis ...        14.0
1       1  <p>What factors related to the Eastern Crisis ...        14.0
2       1  <p>What factors related to the Eastern Crisis ...        14.0
3       1  <p>What factors related to the Eastern Crisis ...        14.0
4       1  <p>What factors related to the Eastern Crisis ...        14.0

                                          Comment  UserCommentId  \
0  Please elaborate a bit on "Eastern Crisis". I ...           10.0
1  Some elaboration would be handy. This question...           20.0
2  Seconding the request for elaboration on what ...           12.0
3    Congrats on the first question of the site. ha.           16.0
4  Thanks. I thought there was something wrong wi...           14.0

             CreationDate         LastActivityDate  \
0  2011-10-11T19:30:14.017  2013-08-19T18:04:54.217
1  2011-10-11T19:30:14.017  2013-08-19T18:04:54.217
2  2011-10-11T19:30:14.017  2013-08-19T18:04:54.217
3  2011-10-11T19:30:14.017  2013-08-19T18:04:54.217
4  2011-10-11T19:30:14.017  2013-08-19T18:04:54.217

                          Tags
0  |20th-century|world-war-one|
1  |20th-century|world-war-one|
2  |20th-century|world-war-one|
3  |20th-century|world-war-one|
4  |20th-century|world-war-one|   (150384, 8)
```

After grouping the datasets, we can see that the tags column allows us to filter out things related to our topic

Filtered out topics via `Tags` column

I am interested in this column because it allows me to filter out tags related to `vietnam war`.

```
# Fill NaN values in `Tags` column with empty string
df2['Tags'] = df2['Tags'].fillna('')

df2 = df2.loc[df2.Tags.str.contains('(?:vietnam[^|]+)|indochina|vietnam',
 ↪  regex=True)]
print(df2.Tags)
```

```
704            |20th-century|war|united-states|vietnam-war|
3285       |20th-century|military|war|civil-war|vietnam-war|
3286       |20th-century|military|war|civil-war|vietnam-war|
4836       |world-war-two|france|french-empire|vietnam|in...
4837       |world-war-two|france|french-empire|vietnam|in...
                                ...
148728                      |vietnam-war|symbols|protests|
148729                      |vietnam-war|symbols|protests|
148730                      |vietnam-war|symbols|protests|
148731                      |vietnam-war|symbols|protests|
149016                                          |vietnam|
Name: Tags, Length: 300, dtype: object
```

Explanation of regex `(?:vietnam[^,]+)|vietnam`

- `(?:)` - non-capturing group to not capture matched text separately
- `vietnam[^,]+` - matches the string `vietnam` and one or more characters that are not commas
- `|vietnam` - or `vietnam`
- `|indochina` - or `indochina`

In essence I am trying to capture 'vietnam-war', 'vietnam' and 'indochina'

Add the `Users` dataset to our dataframe

```
# Merge the data
df3 = df2.copy()
df3 = df3[['PostId', 'Post', 'PostUserId', 'Comment', 'UserCommentId',
 ↪  'CreationDate', 'LastActivityDate']] # Choosing specific columns
df3 = pd.merge(df3, Users, left_on='UserCommentId', right_on = 'Id',
 ↪  how='left')
```

```
# Rename the columns
df3.rename({'CreationDate_x':'CreationDate',
 ↪  'Location':'UserCommentLocation', 'DisplayName':'UserCommentName'}
 ↪  ,inplace=True, axis=1)

# Fill NaN rows in location. user name column with empty string
df3.loc[:, ['UserCommentLocation', 'UserCommentName', 'Comment']] =
 ↪  df3.loc[:, ['UserCommentLocation', 'UserCommentName',
 ↪  'Comment']].fillna('')

# Choose only a few column from Users dataset
df3 = df3[['PostId', 'Post', 'PostUserId', 'Comment', 'UserCommentId',
 ↪  'CreationDate', 'LastActivityDate', 'UserCommentName',
 ↪  'UserCommentLocation']]
print(df3)
```

```
     PostId                                               Post  PostUserId  \
0       229  <p>What happened in the aftermath of the Tet o...       103.0
1       991  <p>South Vietnam was helped by US. Even when t...       338.0
2       991  <p>South Vietnam was helped by US. Even when t...       338.0
3      1466  <p>After WWII France was in rebuilding mode, a...       579.0
4      1466  <p>After WWII France was in rebuilding mode, a...       579.0
..      ...                                                ...         ...
295   72940  <p>I have <a href="https://designobserver.com/...     63445.0
296   72940  <p>I have <a href="https://designobserver.com/...     63445.0
297   72940  <p>I have <a href="https://designobserver.com/...     63445.0
298   72940  <p>I have <a href="https://designobserver.com/...     63445.0
299   74079  <p>How did Vietnamese era names work before th...     62465.0

                                               Comment  UserCommentId  \
0                                                                  NaN
1    The South was invaded in a regular war by the ...          103.0
2    I'll forego a lengthy analysis and simply reco...        13404.0
3    Did you actually try to find some answer yours...          102.0
4    I did some searching, but as far as I could fi...          579.0
..                                                 ...            ...
295  I was very much alive back then. but I never s...         4225.0
296  Likewise, I was well aware what was going on b...        27140.0
297  There are oodles of pictures of Vietnam war pr...          771.0
298  Maybe this I guess? https://www.gettyimages.co...         3011.0
```

```
299                                                                  NaN

                CreationDate         LastActivityDate  \
0    2011-10-13T08:12:44.507  2012-02-13T12:55:43.963
1    2011-12-17T04:51:31.127  2023-01-17T06:19:46.630
2    2011-12-17T04:51:31.127  2023-01-17T06:19:46.630
3    2012-02-24T05:32:20.690  2021-08-24T16:28:48.823
4    2012-02-24T05:32:20.690  2021-08-24T16:28:48.823
..                       ...                      ...
295  2023-12-09T19:15:28.820  2023-12-09T21:23:49.380
296  2023-12-09T19:15:28.820  2023-12-09T21:23:49.380
297  2023-12-09T19:15:28.820  2023-12-09T21:23:49.380
298  2023-12-09T19:15:28.820  2023-12-09T21:23:49.380
299  2023-12-31T11:19:55.273  2023-12-31T14:28:13.690

                   UserCommentName  \
0
1    Sardathrion - against SE abuse
2                          user3847
3                              o0'.
4                           ihtkwot
..                             ...
295                            fdb
296                            Jos
297                         T.E.D.
298                        Brian Z
299

                                UserCommentLocation
0
1
2
3                                             .:…·•°
4                           Chicago, IL, United States
..                                                ...
295                                         Cambridge
296  Krung Thep Mahanakhon Amon Rattanakosin Mahint...
297                                   Tulsa, Oklahoma
298
299

[300 rows x 9 columns]
```

I might use data from the `Users` dataset later, so I will just grab the `PostUserId` and `UserCommentId` from it.

Convert the `Post` and `Comment` columns to string for investigation

I will convert the rows of these two columns into string to look out for things to filter out using regex. I will be building my regex on `https://regex101.com/`.

```
post_string = '\n'.join(df3.drop_duplicates('Post').Post).lower()
comment_string = '\n'.join(df3.Comment).lower()

print(post_string[:100], comment_string[:100])
```

```
<p>what happened in the aftermath of the tet offensive to the viet cong?  was it actually br
the south was invaded in a regular war by the north.
i'll forego a lengthy analysis and simply reco
```

Remove `HTML` tags from `Post` and `Comment` column

```
print(re.findall('<.*?>', df3.Post[109]))
def remove_html(text):
    return re.sub('\<.*?>', '', text) # replace matching substrings with a
    ↪  new string for all occurrences
df3['Post'] = df3['Post'].apply(remove_html)
df3['Comment'] = df3['Comment'].apply(remove_html)
```

```
['<p>', '</p>', '<ol>', '<li>', '</li>', '<li>', '</li>', '</ol>', '<p>', '</p>', '<p>', '</
```

Explanation of regex `<.*?>`

- `<>` - initializer of html paragraph
- `.*?` - lazy quantifer to match any character between zero and unlimited times, as few as possible

Remove `links` and `[]` from `Comments` column

```
# Regex patterns
bracket_pattern = '\[(?:[^\[\]]+)\]'
link_pattern = '(?:http\S+)'

# Example display
print(re.findall(link_pattern, df3['Comment'][298]))
print(re.findall(bracket_pattern, df3['Comment'][50]))

# Remove those patterns
df3['Comment'] = df3['Comment'].replace({bracket_pattern: '', link_pattern:
 ↪  ''}, regex=True)
```

```
['https://www.gettyimages.com/detail/news-photo/printed-in-1968-by-students-at-the-rhode-isla
['[wikipedia]']
```

Explanation of regex `\[(?:[^\[\]]+)\]`'

- Find anything in []
- Since `[]` exist in regex, we need `\` before it.
- `(?:[^\[\]]+)` - Non capturing group of one or more matched character except `\[\]`

Explanation of regex `(?:http\S+)`

- Non capturing group `http` and `\S+` means matches any non-whitespace character

Remove `@` from `Comment` columns

```
re.findall(r'@[^ ]+', df3.Comment[47])
df3['Comment'] = df3['Comment'].str.replace(r'@[^ ]+', '', regex=True)
```

Explanation of regex `@[^ ]+`:

- Remove any characters starting with `@` except white space

Convert date columns to date type

```
df4 = df3.copy()
df4['CreationDate'] = pd.to_datetime(df4['CreationDate'])
df4['LastActivityDate'] = pd.to_datetime(df4['LastActivityDate'])
print(df4.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300 entries, 0 to 299
Data columns (total 9 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   PostId               300 non-null    int64
 1   Post                 300 non-null    object
 2   PostUserId           281 non-null    float64
 3   Comment              300 non-null    object
 4   UserCommentId        277 non-null    float64
 5   CreationDate         300 non-null    datetime64[ns]
 6   LastActivityDate     300 non-null    datetime64[ns]
 7   UserCommentName      300 non-null    object
 8   UserCommentLocation  300 non-null    object
dtypes: datetime64[ns](2), float64(2), int64(1), object(4)
memory usage: 21.2+ KB
None
```

These columns are useful when plotting the time series graph.

Create new columns to remove `stop words` and `punctuation`. Use in finding the most common words.

Remove `punctuation`

```
display(re.findall('[^\w\s]|[0-9]|[_]', df4.Post[0]))

def remove_punctuation(text):
    return re.sub('[^\w\s]|[0-9]|[_]', '', text) # replace matching
    ↪ substrings with a new string for all occurrences

df4['Post2'] = df4['Post'].apply(remove_punctuation)
df4['Comment2'] = df4['Comment'].apply(remove_punctuation)
```

```
['?', '?', ':', '.', '.', '.', '.', '?', '/', '?', '-', '-', ',', '.']
```

Regex explanation `[^\w\s]|[0-9]|[_]`

- `\w` - matches any word character
- `\s` - matches any whitespace character
- `[^\w\s]` - matches any character except for any word character and any whitespace character
- `[0-9]` - matches any digits from 0 to 9
- `[_]` - matches any _
- `|` - or statement

Remove `stop words`

I am using the NLTK libraries to find english stop words and remove them.

```
def remove_stopwords(text):
    # Tokenize text: to break individual words into string
    tokens = word_tokenize(text)

    # Get stopwords
    stop_words = set(stopwords.words('english'))

    # Remove stopwords
    filtered_tokens = [word for word in tokens if word.lower() not in
 ↪  stop_words]

    # Reconstruct the text
    filtered_text = ' '.join(filtered_tokens)

    return filtered_text

df4['Post2'] = df4['Post2'].apply(remove_stopwords)
df4['Comment2'] = df4['Comment2'].apply(remove_stopwords)
```

Create new columns for `Lemmatization`. Use in sentiment analysis, topic modeling, and information retrieval

To lemmatize a word means to reduce the word back to its root form.

```
# Load the English language model in spaCy
nlp = spacy.load("en_core_web_sm")

# Lemmatize text function
def lemmatize_text_spacy(text):
    doc = nlp(text)
    lemmatized_text = " ".join([token.lemma_ for token in doc])
    return lemmatized_text

df4['Post3'] = df4['Post'].apply(lemmatize_text_spacy)
df4['Comment3'] = df4['Comment'].apply(lemmatize_text_spacy)

print(df4.head())
```

```
   PostId                                               Post  PostUserId  \
0     229  What happened in the aftermath of the Tet offe...       103.0
1     991  South Vietnam was helped by US. Even when the ...       338.0
2     991  South Vietnam was helped by US. Even when the ...       338.0
3    1466  After WWII France was in rebuilding mode, and ...       579.0
4    1466  After WWII France was in rebuilding mode, and ...       579.0

                                             Comment  UserCommentId  \
0                                                                NaN
1  The South was invaded in a regular war by the ...          103.0
2  I'll forego a lengthy analysis and simply reco...        13404.0
3  Did you actually try to find some answer yours...          102.0
4  I did some searching, but as far as I could fi...          579.0

             CreationDate        LastActivityDate  \
0 2011-10-13 08:12:44.507 2012-02-13 12:55:43.963
1 2011-12-17 04:51:31.127 2023-01-17 06:19:46.630
2 2011-12-17 04:51:31.127 2023-01-17 06:19:46.630
3 2012-02-24 05:32:20.690 2021-08-24 16:28:48.823
4 2012-02-24 05:32:20.690 2021-08-24 16:28:48.823

                 UserCommentName         UserCommentLocation  \
0
```

```
1  Sardathrion - against SE abuse
2                        user3847
3                            o0'.                        .:…·•°
4                         ihtkwot  Chicago, IL, United States

                                                    Post2  \
0  happened aftermath Tet offensive Viet Cong act...
1  South Vietnam helped US Even US gone million a...
2  South Vietnam helped US Even US gone million a...
3  WWII France rebuilding mode yet insisted tryin...
4  WWII France rebuilding mode yet insisted tryin...

                                                 Comment2  \
0
1                     South invaded regular war North
2  Ill forego lengthy analysis simply recommend b...
3                     actually try find answer asking
4  searching far could find French interested rec...

                                                    Post3  \
0  what happen in the aftermath of the Tet offens...
1  South Vietnam be help by US . even when the US...
2  South Vietnam be help by US . even when the US...
3  after WWII France be in rebuild mode , and yet...
4  after WWII France be in rebuild mode , and yet...

                                                 Comment3
0
1  the South be invade in a regular war by the No...
2  I will forego a lengthy analysis and simply re...
3  do you actually try to find some answer yourse...
4  I do some searching , but as far as I could fi...
```

General Analysis

General analysis is just looking the text on the surface. I will just be looking at the most common words from `Post` and `Comments`. Also, I will look for the trend of the most popular words among the two columns.

Most common words in `Post`

Find the 30 most common words in a list of tuples

```
# Convert `Post2` column into string and lower case
post_string = '\n'.join(df4.drop_duplicates('Post2').Post2).lower()

# Tokenize string into words
tokens = word_tokenize(post_string)

# Count the word frequencies
word_freq = Counter(tokens)

# Find the most common words
top30words = word_freq.most_common(30)
print(top30words)
```

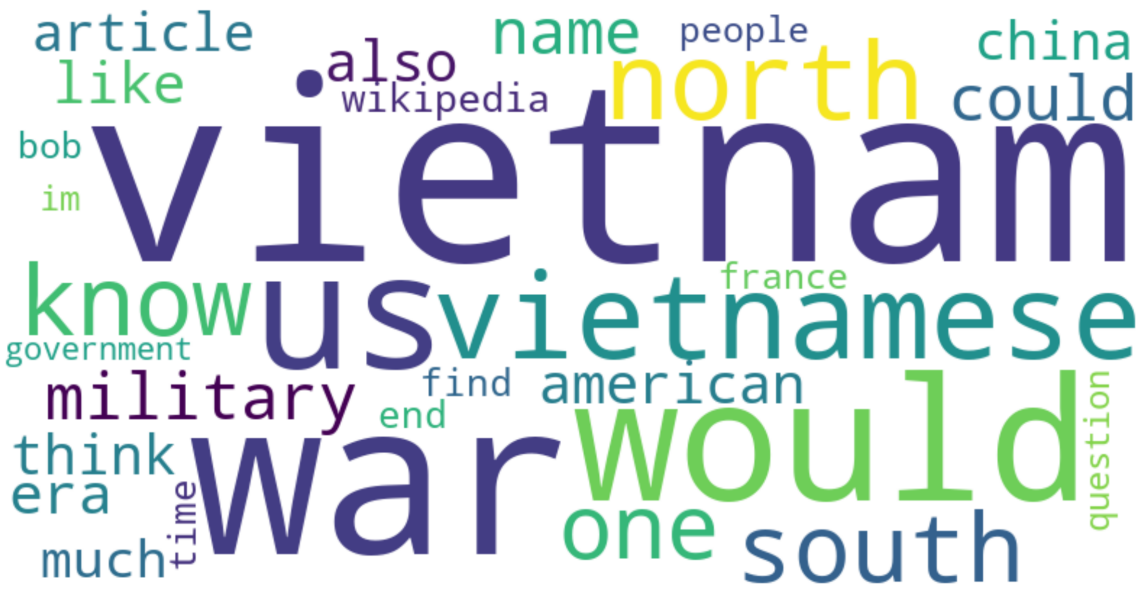
```
[('vietnam', 128), ('war', 91), ('would', 53), ('us', 48), ('vietnamese', 45), ('north', 36)
```

Generate a **word cloud** of the list of tuples

```
# Convert the list of tuples into a dictionary
top30words = dict(top30words)

# Generate a word cloud
wordcloud = WordCloud(width=800, height=400,
 ↪  background_color='white').generate_from_frequencies(top30words)

# Display the word cloud
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```



So the most common words related to Vietnam History in `Post` are: **vietnam, us, war, north, south, american, france**

Most common words in `Comment`

Find the 30 most common words in a list of tuples

```
# Convert `Comment2` column into string and lower case
comment_string = '\n'.join(df4.Comment2).lower()

# Tokenize string of words
tokens = word_tokenize(comment_string)

# Count the word frequencies
word_freq = Counter(tokens)

# Find the most common words
top30words = word_freq.most_common(30)
print(top30words)
```

```
[('question', 75), ('vietnam', 73), ('would', 63), ('war', 61), ('us', 46), ('answer', 30),
```

Generate a **word cloud** of the list of tuples

```
# Convert `Comment2` column into string and lower case
# Convert the list of tuples into a dictionary
top30words = dict(top30words)

# Generate a word cloud
wordcloud = WordCloud(width=800, height=400,
 ↪  background_color='white').generate_from_frequencies(top30words)

# Display the word cloud
plt.figure(figsize=(10, 5))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.show()
```

So the most common words related to Vietnam History in `Comment` are: **vietnam, us, war, french, china, france**

Trend analysis of common words overtime

The following code blocks will be breaking down the steps I used to plot the time series graph for the most popular words from the posts and comments.

Replace duplicated rows of `Post2` column with empty string

```
df5 = df4.copy()
df5.set_index('CreationDate', inplace=True) # Set 'CreationDate' as the index

# Identify duplicate rows except for the first occurrence
duplicates = df5.duplicated(subset=['Post2'], keep='first')

# Replaced duplicated content with empty strings
df5.loc[duplicates, :] = ''

print(df5.head())
```

```
                        PostId  \
CreationDate
2011-10-13 08:12:44.507    229
2011-12-17 04:51:31.127    991
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690   1466
2012-02-24 05:32:20.690

                                                                Post  \
CreationDate
2011-10-13 08:12:44.507  What happened in the aftermath of the Tet offe...
2011-12-17 04:51:31.127  South Vietnam was helped by US. Even when the ...
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690  After WWII France was in rebuilding mode, and ...
2012-02-24 05:32:20.690

                        PostUserId  \
CreationDate
2011-10-13 08:12:44.507      103.0
2011-12-17 04:51:31.127      338.0
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690      579.0
2012-02-24 05:32:20.690

                                                             Comment  \
CreationDate
2011-10-13 08:12:44.507
2011-12-17 04:51:31.127  The South was invaded in a regular war by the ...
2011-12-17 04:51:31.127
```

```
2012-02-24 05:32:20.690  Did you actually try to find some answer yours...
2012-02-24 05:32:20.690

                       UserCommentId        LastActivityDate  \
CreationDate
2011-10-13 08:12:44.507          NaN 2012-02-13 12:55:43.963
2011-12-17 04:51:31.127        103.0 2023-01-17 06:19:46.630
2011-12-17 04:51:31.127                                  NaT
2012-02-24 05:32:20.690        102.0 2021-08-24 16:28:48.823
2012-02-24 05:32:20.690                                  NaT

                                     UserCommentName UserCommentLocation  \
CreationDate
2011-10-13 08:12:44.507
2011-12-17 04:51:31.127  Sardathrion - against SE abuse
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690                           o0'.              .:…·•°
2012-02-24 05:32:20.690

                                                                    Post2  \
CreationDate
2011-10-13 08:12:44.507  happened aftermath Tet offensive Viet Cong act...
2011-12-17 04:51:31.127  South Vietnam helped US Even US gone million a...
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690  WWII France rebuilding mode yet insisted tryin...
2012-02-24 05:32:20.690

                                          Comment2  \
CreationDate
2011-10-13 08:12:44.507
2011-12-17 04:51:31.127  South invaded regular war North
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690  actually try find answer asking
2012-02-24 05:32:20.690

                                                                    Post3  \
CreationDate
2011-10-13 08:12:44.507  what happen in the aftermath of the Tet offens...
2011-12-17 04:51:31.127  South Vietnam be help by US . even when the US...
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690  after WWII France be in rebuild mode , and yet...
2012-02-24 05:32:20.690
```

```
                                                                  Comment3
CreationDate
2011-10-13 08:12:44.507
2011-12-17 04:51:31.127  the South be invade in a regular war by the No...
2011-12-17 04:51:31.127
2012-02-24 05:32:20.690  do you actually try to find some answer yourse...
2012-02-24 05:32:20.690
```

In this code block, I have set `CreationDate` column as the index and replaced duplicate content via the post with empty string while keeping only the first occurence.

Count the occurrences the common words

```
# List of common words
common_words = ["vietnam", "us", "war", "french", "china", "france", "north",
 ↪  "south", "american"]

# Replace NaN values by empty strings
df5['Post2'] = df5['Post2'].fillna('')
df5['Comment2'] = df5['Comment2'].fillna('')

# Count the occurrences the common words
def count_words(text: str, words: List[str]) -> Dict[str, int]:
    text = text.lower()
    word_count = Counter(re.findall(r"\w+", text)) # Find all the words and
 ↪  count how many time it appears
    return {word: word_count[word] for word in words} # Return each word in
     ↪  `common_words` as key, the value is the count how many time it
     ↪  appears

print(count_words(post_string, common_words))
```

```
{'vietnam': 128, 'us': 48, 'war': 91, 'french': 12, 'china': 17, 'france': 14, 'north': 36,
```

In this code block, I have created a dictionary of the most common words as the key and the number of occurences as value.

Count the occurrences the common words yearly

```
# Dictionary to store the common words counted by year
yearly_counts = {word: Counter() for word in common_words}

for date, row in df5.iterrows(): # iterate over the rows of the dataframe as
 ↪  (index (index), Series (row)) pairs
    year = date.year # extract year from `CreationDate` index
    post2_counts = count_words(row.Post2, common_words) # dictionary contain
 ↪  word counts for `Post2`
    comment2_counts = count_words(row.Comment2, common_words) # dictionary
 ↪  contain word counts for `Comment2`

    for word in common_words:
        yearly_counts[word][year] += post2_counts[word] +
 ↪  comment2_counts[word] # calculate each word occurrence per year

# Convert `yearly_counts` to dataframe
df_counts = pd.DataFrame(yearly_counts).fillna(0).sort_index()
print(df_counts)
```

	vietnam	us	war	french	china	france	north	south	american
2011	5	3	5	0	0	0	4	4	0
2012	4	2	12	4	0	2	0	2	0
2013	7	2	2	0	0	0	0	0	4
2014	3	3	1	0	0	0	0	0	2
2015	13	8	17	2	7	5	1	1	2
2016	15	5	9	2	1	3	2	2	2
2017	22	6	13	0	2	0	2	6	2
2018	15	4	8	0	2	0	4	4	1
2019	39	17	17	4	7	2	22	11	4
2020	3	3	3	0	0	0	2	0	0
2021	2	4	3	5	0	8	0	0	0
2022	8	2	4	0	1	0	0	0	1
2023	3	0	3	0	0	0	0	0	2

The table output of this code block shows the number of occurences per each word yearly.

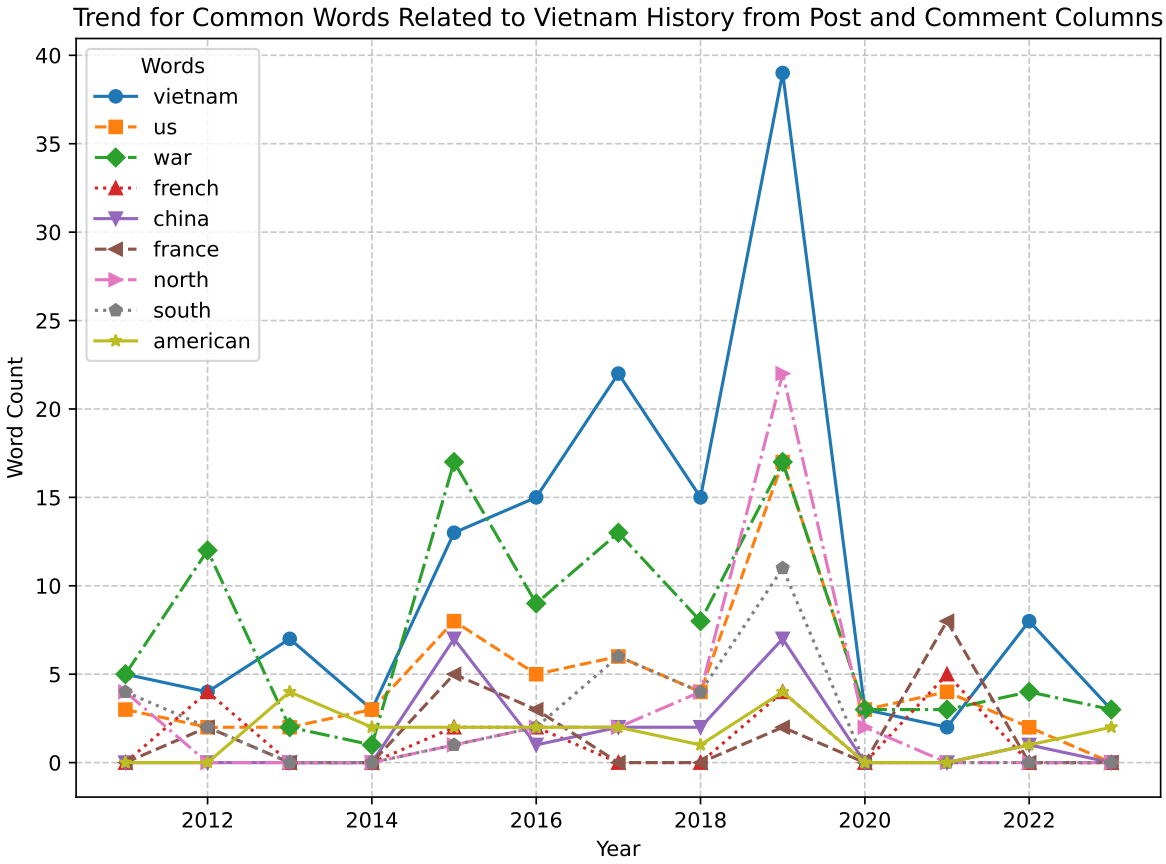
Plot the time series trend

```
plt.figure(figsize=(8, 6))
line_styles = ['-', '--', '-.', ':', '-', '--', '-.', ':', '-']  # Different
 ↪  line styles for each word
```

```
markers = ['o', 's', 'D', '^', 'v', '<', '>', 'p', '*']  # Different markers
 ↪  for each word

for i, word in enumerate(common_words):
    plt.plot(df_counts.index, df_counts[word], label=word,
 ↪  linestyle=line_styles[i % len(line_styles)], marker=markers[i %
 ↪  len(markers)], markersize=6)
    # `df_counts.index` is axis (years), `df_counts[word]` is y-axis(word
     ↪  counts), `label=word` assigns the label for the legend, the rest just
     ↪  ensure each word gets a unique style and marker

plt.xlabel('Year')
plt.ylabel('Word Count')
plt.title('Trend for Common Words Related to Vietnam History from Post and
 ↪  Comment Columns')
plt.grid(True, linestyle='--', alpha=0.7)
plt.legend(title='Words', loc='upper left')
plt.tight_layout()
plt.show()
```



The words vietnam, us, war seem to be discussed yearly. Other words appear here and there. Interestingly, from 2018 to 2020, people are more interested in the Vietnam History compare to other years. Based on the the popular words. I can deduce that the majority of discussions are on the Vietnam War and the events that triggered the war and what happened after it. To analyze that, I will have to use more advance techniques.

Topic modelling for `Post`

I will be applying Bertopic to categorize the post labels. More information on: `https://colab.research.google.com/drive/1BoQ_vakEVtojsd2x_U6-_x52OOuqruj2?usp=sharing#scroll`

OpenAI summarize `Post3`

I will use `openai` api and the model `gpt3.5` to summarize the `Post3`.

```
# client = openai.OpenAI(api_key="...")

# # Function to tell gpt to summarize the text
# def post_summary(text):
#     response = client.chat.completions.create(
#         model="gpt-3.5-turbo",
#         messages=[
#         {"role": "system", "content": "You are a expert in Vietnamese
 ↪  History. Summarize and shorten this given question."},
#         {"role": "user", "content": f"{text}"},
#       ]
#     )
#     return response.choices[0].message.content

# test = df4.head(1)
# # test['Post_summary'] = test['Post3'].dropna().apply(post_summary)
# # test['Post_summary'][0]
# df4['Post_summary'] = df4['Post3'].dropna().apply(post_summary)

# # Saving the post
# df5 = df4.copy()
# df5.to_csv('df5.csv', index=False)

# Load the data
df5 = pd.read_csv('df5.csv')
print(df5.head(5))
```

```
   PostId                                               Post  PostUserId  \
0     229  What happened in the aftermath of the Tet offe...       103.0
1     991  South Vietnam was helped by US. Even when the ...       338.0
2     991  South Vietnam was helped by US. Even when the ...       338.0
3    1466  After WWII France was in rebuilding mode, and ...       579.0
4    1466  After WWII France was in rebuilding mode, and ...       579.0

                                             Comment  UserCommentId  \
0                                                NaN            NaN
1  The South was invaded in a regular war by the ...          103.0
2  I'll forego a lengthy analysis and simply reco...        13404.0
3  Did you actually try to find some answer yours...          102.0
4  I did some searching, but as far as I could fi...          579.0
```

```
              CreationDate         LastActivityDate  \
0  2011-10-13 08:12:44.507  2012-02-13 12:55:43.963
1  2011-12-17 04:51:31.127  2023-01-17 06:19:46.630
2  2011-12-17 04:51:31.127  2023-01-17 06:19:46.630
3  2012-02-24 05:32:20.690  2021-08-24 16:28:48.823
4  2012-02-24 05:32:20.690  2021-08-24 16:28:48.823

                  UserCommentName         UserCommentLocation  \
0                             NaN                         NaN
1  Sardathrion - against SE abuse                         NaN
2                        user3847                         NaN
3                            o0'.                      .:…·•°
4                         ihtkwot  Chicago, IL, United States

                                              Post2  \
0  happened aftermath Tet offensive Viet Cong act...
1  South Vietnam helped US Even US gone million a...
2  South Vietnam helped US Even US gone million a...
3  WWII France rebuilding mode yet insisted tryin...
4  WWII France rebuilding mode yet insisted tryin...

                                            Comment2  \
0                                                NaN
1                    South invaded regular war North
2  Ill forego lengthy analysis simply recommend b...
3                    actually try find answer asking
4  searching far could find French interested rec...

                                              Post3  \
0  what happen in the aftermath of the Tet offens...
1  South Vietnam be help by US . even when the US...
2  South Vietnam be help by US . even when the US...
3  after WWII France be in rebuild mode , and yet...
4  after WWII France be in rebuild mode , and yet...

                                            Comment3  \
0                                                NaN
1  the South be invade in a regular war by the No...
2  I will forego a lengthy analysis and simply re...
3  do you actually try to find some answer yourse...
4  I do some searching , but as far as I could fi...

                                        Post_summary
```

```
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...
2  Despite receiving substantial support from the...
3  Why did post-WWII France prioritize reclaiming...
4  Why did the post-World War II French governmen...
```

I have run the code in a separate jupyter notebook and save the output to a `csv` file.

Row splitter

I will be splitting each row of the `Post_summary` column for training

```
# Drop `Post3` duplicates
df6 = df5.drop_duplicates('Post3')
df6 = df5.copy()

# Split row summary summarized by openai
row_splitter = df6['Post_summary'].tolist()
print(row_splitter[:5])
```

```
['What happened to the Viet Cong after the Tet Offensive? Did they break away or become part
```

Pre-calculate Embeddings

We convert the `row_splitter` into numerical values called embeddings. Then feed that to Bertopic to skip calculating the embeddings each time.

```
# Pre-calculate embeddings
embedding_model = SentenceTransformer("all-MiniLM-L6-v2")
# embeddings = embedding_model.encode(row_splitter, show_progress_bar=True)
```

Apply Bert

```
# # KeyBERT
# keybert_model = KeyBERTInspired()

# # Part-of-Speech
```

```
# pos_model = PartOfSpeech("en_core_web_sm")

# # MMR
# mmr_model = MaximalMarginalRelevance(diversity=0.3)

# # GPT-3.5
# prompt = """
# I have a topic that contains the following documents:
# [DOCUMENTS]
# The topic is described by the following keywords: [KEYWORDS]

# Based on the information above, extract a short but highly descriptive
 ↪  topic label of at most 5 words. Make sure it is in the following format:
# topic: <topic label>
# """
# client =
 ↪  openai.OpenAI(api_key="sk-proj-QFpbTwzLPdgj5h9SbTsLT3BlbkFJrvE2Kz8SXXOuV8VsmqvU")
# openai_model = OpenAI(client, model="gpt-3.5-turbo",
 ↪  exponential_backoff=True, chat=True, prompt=prompt)

# # Fine-tune the topic representations
# representation_model = {
#     "KeyBERT": keybert_model,
#     "OpenAI": openai_model,  # Uncomment if you will use OpenAI
#     "MMR": mmr_model,
#     "POS": pos_model
# }

# # set the model and parameters
# topic_model = BERTopic(embedding_model=embedding_model,
#     min_topic_size=5,  # min topic
#     nr_topics=20, # max topic
#     representation_model=representation_model,
#     calculate_probabilities=True
# )

# topics, probs = topic_model.fit_transform(row_splitter)

# # Saving the model
# topic_model.save("postmodel", serialization="pytorch", save_ctfidf=True,
 ↪  save_embedding_model=embedding_model)
```

```
# Input the saved model
post_model = BERTopic.load("postmodel", embedding_model=embedding_model)
post_model_info = post_model.get_topic_info()
print(post_model_info)
```

```
    Topic  Count                                        Name  \
0      -1     16          -1_language_toilet_translators_used
1       0     89                    0_north_china_war_support
2       1     19             1_served_casualty_wounded_combat
3       2     18                  2_red_symbol_location_photo
4       3     17                  3_racism_theater_pacific_ad
5       4     17                 4_france_indo_did_rebuilding
6       5     14               5_invasion_cambodia_khmer_rouge
7       6     13                6_nations_nigeria_germany_iraq
8       7     12       7_president_industrial_complex_johnson
9       8     12            8_camps_pow_activity_presentation
10      9     10                9_nixon_talks_treason_richard
11     10     10                    10_tons_000_bombs_dropped
12     11      9                   11_blue_term_tag_discharge
13     12      7                    12_draft_ted_nugent_avoid
14     13      7       13_party_members_communist_sympathetic
15     14      6  14_politician_russia_embarrassment_ukraine
16     15      6            15_city_purple_forbidden_imperial
17     16      6                          16_tours_57_uso_bob
18     17      6           17_catholic_diem_buddhist_majority
19     18      6                  18_faction_saigon_trình_thế

                                      Representation  \
0   [language, toilet, translators, used, weapon, ...
1   [north, china, war, support, vietnam, soviet, ...
2   [served, casualty, wounded, combat, high, rate...
3   [red, symbol, location, photo, infantry, divis...
4   [racism, theater, pacific, ad, manipulated, pa...
5   [france, indo, did, rebuilding, french, post, ...
6   [invasion, cambodia, khmer, rouge, cambodian, ...
7   [nations, nigeria, germany, iraq, suffer, liby...
8   [president, industrial, complex, johnson, deci...
9   [camps, pow, activity, presentation, condition...
10  [nixon, talks, treason, richard, 1968, paris, ...
11  [tons, 000, bombs, dropped, 500, napalm, milli...
12  [blue, term, tag, discharge, program, slang, p...
```

```
13  [draft, ted, nugent, avoid, pants, page, wikip...
14  [party, members, communist, sympathetic, draft...
15  [politician, russia, embarrassment, ukraine, r...
16  [city, purple, forbidden, imperial, huế, domai...
17  [tours, 57, uso, bob, hope, record, documentat...
18  [catholic, diem, buddhist, majority, dinh, bud...
19  [faction, saigon, trình, thế, minh, jammes, 19...

                                                KeyBERT  \
0   [vietnamese, vietnam, popeye, drought, minh, t...
1   [vietnam, vietnamese, indochina, troops, viet,...
2   [casualties, wounded, veterans, vietnam, casua...
3   [vietnam, soldiers, marching, regiment, symbol...
4   [vietnam, racism, vietnamese, ethnic, war, ngu...
5   [indochina, france, reclaiming, china, reclaim...
6   [cambodians, vietnam, cambodia, cambodian, vie...
7   [nations, wwii, countries, war, wars, iraq, ec...
8   [lyndon, vietnam, jfk, johnson, lbj, kennedy, ...
9   [camps, prisoners, presentation, torture, hano...
10  [nixon, vietnam, treason, vietnamese, lyndon, ...
11  [bombs, bombing, vietnam, bomb, tons, wwii, so...
12  [vietnam, blue, soldier, military, discharge, ...
13  [vietnam, ted, draft, nugent, defecated, alleg...
14  [vietnam, draftees, draft, vietnamese, drafted...
15  [vietnam, chechnya, russian, russia, embarrass...
16  [vietnam, huế, purple, beijing, emperor, forbi...
17  [tours, tour, 57, record, uso, hope, seeks, or...
18  [buddhists, ngo, buddhist, vietnamese, vietnam...
19  [saigon, vietnam, communists, trình, trinh, gr...

                                             OpenAI  \
0                [Military Translators in Vietnam]
1               [Chinese military support in wars]
2                [Vietnam War casualty statistics]
3              [Vietnam War protest symbol photos]
4                  [Racism in Pacific war theater]
5     [France's Post-WWII Indo-China Reclamation]
6      [Global Opposition to Vietnamese Invasion]
7                  [Post-War Recovery Disparities]
8                [Vietnam War Decision Influences]
9                      [Comparing WWII POW Camps]
10                [Nixon's Manipulation in 1968]
11                 [Bombing Campaign Comparison]
```

```
12  [Unrecognized Term "Blue Tag" Interpretation]
13           [Ted Nugent's Vietnam Draft Dodging]
14              [Treatment of Communist Draftees]
15      [Russian Politician's Vietnam References]
16                [Imperial Purple Forbidden City]
17                   [Bob Hope's USO Tours Record]
18                [Catholic oppression in Vietnam]
19             [US Involvement in Saigon Bombings]
```

	MMR
0	[language, toilet, translators, used, weapon, ...
1	[north, china, war, support, vietnam, soviet, ...
2	[served, casualty, wounded, combat, high, rate...
3	[red, symbol, location, photo, infantry, divis...
4	[racism, theater, pacific, ad, manipulated, pa...
5	[france, indo, did, rebuilding, french, post, ...
6	[invasion, cambodia, khmer, rouge, cambodian, ...
7	[nations, nigeria, germany, iraq, suffer, liby...
8	[president, industrial, complex, johnson, deci...
9	[camps, pow, activity, presentation, condition...
10	[nixon, talks, treason, richard, 1968, paris, ...
11	[tons, 000, bombs, dropped, 500, napalm, milli...
12	[blue, term, tag, discharge, program, slang, p...
13	[draft, ted, nugent, avoid, pants, page, wikip...
14	[party, members, communist, sympathetic, draft...
15	[politician, russia, embarrassment, ukraine, r...
16	[city, purple, forbidden, imperial, huế, domai...
17	[tours, 57, uso, bob, hope, record, documentat...
18	[catholic, diem, buddhist, majority, dinh, bud...
19	[faction, saigon, trình, thế, minh, jammes, 19...

	POS	Representative_Docs
0	[language, toilet, translators, weapon, operat...	NaN
1	[north, war, support, military, states, forces...	NaN
2	[casualty, combat, high, rate, soldiers, days,...	NaN
3	[red, symbol, location, photo, protest, soldie...	NaN
4	[racism, theater, ethnic, perspective, era, ch...	NaN
5	[rebuilding, post, colonies, significant, phas...	NaN
6	[invasion, genocide, opposition, power, lack, ...	NaN
7	[nations, wars, major, able, conflicts, afterm...	NaN
8	[industrial, complex, decision, miscalculation...	NaN
9	[camps, activity, presentation, conditions, di...	NaN
10	[talks, treason, peace, president, manipulatio...	NaN

```
11  [tons, bombs, napalm, pilots, bombing, trail, ...                NaN
12  [blue, term, discharge, tag, program, slang, p...                NaN
13  [draft, pants, page, dodger, deferments, servi...                NaN
14  [members, sympathetic, draftees, military, exe...                NaN
15  [politician, embarrassment, shame, hump, decla...                NaN
16  [purple, imperial, personal, domain, servants,...                NaN
17  [tours, record, documentation, sources, questi...                NaN
18  [majority, persecution, regime, minority, popu...                NaN
19  [faction, caodaiist, novel, actions, parade, c...                NaN
```

I have already saved the Bertopic model and will just import it each time I run this cell.

Save the topics and corresponding posts

So I already have saved the topics and corresponding posts. Now I will just import them.

```
# df7 = pd.DataFrame({'topic': topics, 'Original Post': df6.Post, 'OpenAI
 ↪  Post Summary': df6.Post_summary})
# df7.drop_duplicates('Original Post', inplace=True)
# # Merge OpenAI label on topic column
# df7 = df7.merge(topic_model_info[['Topic', 'OpenAI', 'KeyBERT']],
 ↪  left_on='topic', right_on='Topic', how='left')
# # Rename and rearrange columns
# df7.drop('Topic', axis=1, inplace=True)
# df7.rename({'OpenAI':'OpenAI Post Label','KeyBERT':'KeyBERT Post Label',
 ↪  'topic':'Post Topic'}, inplace=True, axis=1)

# # Export the dataframe
# df7.to_csv('df7.csv', index=False)

# Import df7
df7 = pd.read_csv('df7.csv')
print(df7.head())
```

```
   Post Topic                                      Original Post  \
0           0  What happened in the aftermath of the Tet offe...
1           0  South Vietnam was helped by US. Even when the ...
2           4  After WWII France was in rebuilding mode, and ...
3          17  In 1963, weeks before his own assassination, P...
4           5  The Khmer Rouge killed about 25% of the Cambod...
```

```
                                OpenAI Post Summary  \
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...
2  Why did post-WWII France prioritize reclaiming...
3  In 1963, President Kennedy approved CIA plans ...
4  Were there studies conducted on the potential ...

                               OpenAI Post Label  \
0          ['Chinese military support in wars']
1          ['Chinese military support in wars']
2  ["France's Post-WWII Indo-China Reclamation"]
3            ['Catholic oppression in Vietnam']
4   ['Global Opposition to Vietnamese Invasion']

                                KeyBERT Post Label
0  ['vietnam', 'vietnamese', 'indochina', 'troops...
1  ['vietnam', 'vietnamese', 'indochina', 'troops...
2  ['indochina', 'france', 'reclaiming', 'china',...
3  ['buddhists', 'ngo', 'buddhist', 'vietnamese',...
4  ['cambodians', 'vietnam', 'cambodia', 'cambodi...
```

This dataframe gives me enough information to analyze for the next section.

Vietnam map to show different historical events and places in the `Post`

Since there are not many posts related to Vietnam History. I will have a read through them and look out for different events and locations. I will do that by examining the posts for each `Post Topic`. Here is a sample on how I analyze it.

```
print(df7.loc[df7['Post Topic'] == 5])
```

```
    Post Topic                                      Original Post  \
4            5  The Khmer Rouge killed about 25% of the Cambod...
33           5  This was a major raid by Khmer Rouge forces ov...
50           5  Despite the genocide happening in Cambodia, th...

                                OpenAI Post Summary  \
4   Were there studies conducted on the potential ...
33  Khmer Rouge forces carried out a major raid ov...
50  The question explores the rationale behind the...
```

```
                                OpenAI Post Label  \
4   ['Global Opposition to Vietnamese Invasion']
33  ['Global Opposition to Vietnamese Invasion']
50  ['Global Opposition to Vietnamese Invasion']

                                     KeyBERT Post Label
4   ['cambodians', 'vietnam', 'cambodia', 'cambodi...
33  ['cambodians', 'vietnam', 'cambodia', 'cambodi...
50  ['cambodians', 'vietnam', 'cambodia', 'cambodi...
```

Here is a list of events and locations that happen in Vietnam or close to Vietnam from after reading through each post: Tet Offensive; Fall of Saigon; Operation Popeye; Battle of Dien Bien Phu; Ngo Dinh Diem's assassination; Ba Chúc massacre; My Lai massacre; Operation Wandering Soul; My Son Sanctuary; a bridge destroyed by Big Red One; 1st Infantry Division; Tu Do street or Dong Khoi street; Imperial City of Huel; Ho Chi Minh trail (eastern border with Laos and Cambodia), Vietnam War (Gulf of Tonkin Incident).

I will plot this on the Vietnam map.

```
# coords for Ho Chi Minh Trail
ho_chi_minh_trail = [
    [21.0285, 105.8542],  # Hanoi
    [20.4855, 104.2375],  # Near Lao Bao, Vietnam
    [19.3092, 103.9235],  # Along the trail in Laos
    [18.2048, 103.4167],  # Along the trail in Laos
    [17.9725, 106.2800],  # Near Dong Hoi, Vietnam
    [17.2135, 106.1106],  # Near Khe Sanh, Vietnam
    [16.2610, 107.5807],  # Near A Luoi, Vietnam
    [15.0500, 108.5000],  # Near Dak To, Vietnam
    [13.7563, 109.1996],  # Near Pleiku, Vietnam
    [12.2388, 107.7340],  # Near Buon Ma Thuot, Vietnam
    [11.5449, 104.8922],  # Phnom Penh, Cambodia
    [10.5763, 105.2094],  # Along the trail in Cambodia
    [10.8231, 106.6297]   # Ho Chi Minh City (Saigon)
]

# Locations and descriptions
locations = [
    {
        "coords": (16.47860161558329, 107.57420678169986), # Hue Historic
         ↪ Citadel coords
        "title": "Battle of Hue",
        "content": "On January 31, 1968, The bloodiest battle of the Tet
         ↪ offensive took place at Hue. The PAVN and Viet cong attacked the
         ↪ city on the 31 January of 1968 and took over the city's ancient
         ↪ citadel. The North Vietnam occupation of the city was not long as
         ↪ the American regained control on the February 26, but the damage
         ↪ was done. The statistics show that around 2800 bodies and 3000
         ↪ residents were missing from the South Vietnam, while from North
         ↪ Vietnam 5000 soldiers died from. The battle of Hue and the Tet
         ↪ Offensive in general had halted the peace talk and it is a
         ↪ turning point for the war.\nSource:
         ↪ https://www.history.com/topics/vietnam-war/tet-offensive",
```

```
        "image_url":
         ↪ "https://huedaytour.com/wp-content/uploads/hue-imperial-citadel-2.jpg"
    },
    {
        "coords": (10.7772366108138, 106.69536647283701), # Presidential
         ↪ Palace in Sai Gon coords
        "title": "Ngo Dinh Diem assassination",
        "content": "Ngo Dinh Diem was the South Vietnam president in 1955
         ↪ after a controlled referendum against the Bao Dai, the last
         ↪ emperor of Vietnam. With the support of the US, Diem had
         ↪ resettled hundreds of thousands of refugees escaped from North
         ↪ Vietnam. Despite the alliance with the US and Diem's personal
         ↪ hatred to communism, his presidency shown many characteristics of
         ↪ a dictatorship. From consolidating power within his family,
         ↪ discriminating Buddhists over Catholics in a Buddhist country to
         ↪ failing to implement land reforms and suppression of political
         ↪ opposition. All of these factors led to the US ordering Diem's
         ↪ generals to assassinate him in a coup d'état on 2nd November
         ↪ 1963.\nSource:
         ↪ https://www.britannica.com/biography/Ngo-Dinh-Diem",
        "image_url":
         ↪ "https://lh5.googleusercontent.com/p/AF1QipMSw2-evUD0n0ibQLXkkthzFwOhDct59iVvQLA=
    },
    {
        "coords": (10.816673671427983, 106.64435715018435),
        "title": "The fall of SaiGon in 1975",
        "content": "The Fall of Sai Gon marked the end of a 2-decade war
         ↪ between Vietnam and America. In the final stages of the war, the
         ↪ North Vietnam army, and the Viet Cong capture city after city.
         ↪ This was due to President Nixon policy of Vietnamization where
         ↪ the American would train and equip the South Vietnamese military
         ↪ to take over the conduct of the war, and gradually reduce the
         ↪ American army and support. With the already unstable South
         ↪ Vietnam, couple with the American were leaving, left the South
         ↪ Vietnam to be vulnerable to the attacks of North Vietnam and
         ↪ eventually led to their demise.\nSouce:
         ↪ https://www.history.com/topics/vietnam-war/vietnamization",
        "image_url":
         ↪ "https://api.time.com/wp-content/uploads/2015/04/saigon.jpeg?quality=85&w=1080"
    },
    {
        "coords": (19.986955480290767, 107.75467673613805),
```

```
        "title": "Gulf of Tonkin Incident in 1964",
        "content": "Two American destroyers USS Maddox and USS Turner Jou
         ↪  were conducting intelligence-gathering operations in the Gulf of
         ↪  Tonkin. On the 2nd of August, the Maddox was being approached by
         ↪  3 North Vietnamese torpedo boats. The Maddox fired a warning
         ↪  shot, but the torpedo boats opened fried. This led to the
         ↪  intervention of American air support and the US to believe the
         ↪  North Vietnam was targeting their intelligence gathering mission.
         ↪  2 days after that, on a stormy night both the destroyers were
         ↪  operating in the gulf of Tonkin when they reported detecting
         ↪  multiple unidentified vessels approaching them. They open fired
         ↪  when called in air support. But in actuality, there were no enemy
         ↪  boats to be found. These two incidents had led to the Congress
         ↪  passing the Gulf of Tonkin Resolution granting President Johnson
         ↪  the escalate the US military presence in Vietnam.\nSorce:
         ↪  https://www.britannica.com/event/Gulf-of-Tonkin-incident",
        "image_url":
         ↪  "https://lh5.googleusercontent.com/p/AF1QipPxQMj1GuhSE0UDJ8hEjVlYgjotAqcuWa-toa_
    },
    {
        "coords": (19.3092, 103.9235),
        "title": "Operation Popeye",
        "content": "Operation Popeye or Project Popeye was a weather
         ↪  modification program used by the US in the Vietnam War. The
         ↪  American sent around 2600 cloud seeding (a technique to improve
         ↪  the clouds' ability to produce rain) on a part of the Ho Chi Minh
         ↪  Trail, near the border of Vietnam, Laos, Cambodia. The objective
         ↪  of the operation was to extend the monsoon season fir 30 to 45
         ↪  days in order to disrupt the supply line of the
         ↪  Vietnamese.\nSource. https://polarpedia.eu/en/operation-popeye/",
        "image_url":
         ↪  "https://99percentinvisible.org/app/uploads/2018/03/trail-usage.jpg"
    },
    {
        "coords": (18.2048, 103.4167),
        "title": "Operation Popeye",
        "content": "Operation Popeye or Project Popeye was a weather
         ↪  modification program used by the US in the Vietnam War. The
         ↪  American sent around 2600 cloud seeding (a technique to improve
         ↪  the clouds' ability to produce rain) on a part of the Ho Chi Minh
         ↪  Trail, near the border of Vietnam, Laos, Cambodia. The objective
         ↪  of the operation was to extend the monsoon season fir 30 to 45
         ↪  days in order to disrupt the supply line of the
         ↪  Vietnamese.\nSource. https://polarpedia.eu/en/operation-popeye/",
```

```
        "image_url":
         ↪ "https://99percentinvisible.org/app/uploads/2018/03/trail-usage.jpg"
    },
    {
        "coords": (10.5763, 105.2094),
        "title": "Operation Popeye",
        "content": "Operation Popeye or Project Popeye was a weather
         ↪ modification program used by the US in the Vietnam War. The
         ↪ American sent around 2600 cloud seeding (a technique to improve
         ↪ the clouds' ability to produce rain) on a part of the Ho Chi Minh
         ↪ Trail, near the border of Vietnam, Laos, Cambodia. The objective
         ↪ of the operation was to extend the monsoon season fir 30 to 45
         ↪ days in order to disrupt the supply line of the
         ↪ Vietnamese.\nSource. https://polarpedia.eu/en/operation-popeye/",
        "image_url":
         ↪ "https://99percentinvisible.org/app/uploads/2018/03/trail-usage.jpg"
    },
    {
        "coords": (21.57235955732699, 103.09442549952763),
        "title": "Battle of Dien Bien Phu",
        "content": "Dien Bien Phu Valley was a military stronghold of the
         ↪ French during the First Indochina War between France and Vietnam.
         ↪ The reasons for France occupation of Dien Bien Phu were to
         ↪ destroy the supply lines of Vietnam from Laos and force the Viet
         ↪ Minh to an open attack. Under the leadership of general Vo Nguyen
         ↪ Giap, the Viet Minh army surround the valley. They dug out many
         ↪ artillery positions on steep slopes. On March 13, 1954, Viet Minh
         ↪ started their attacks. For the next two months, the Vietnamese
         ↪ attacked the French using artillery fire and trench warfare to
         ↪ isolate the French garrisons. This caused the French to lose
         ↪ their outposts, airstrips, and many damaged aircrafts. The final
         ↪ victory belongs to the Vietnamese, but the human toll on both
         ↪ sides was tremendous, with the French losing 2200 soldiers in
         ↪ actions and another 3300 prisoners died on captivity. After the
         ↪ battle, the France and Vietnam had a conference named Geneva in
         ↪ July of 1954 for peace talks. The terms of the peace agreement
         ↪ called for temporary partition of North and South Vietnam. The
         ↪ North was the Communist side with the supported of USSR and China
         ↪ while the South had the supported of the US and some of the
         ↪ allies. These events ultimately led to the Vietnam War or Second
         ↪ Indochina War.\nSource:
         ↪ https://www.history.com/topics/european-history/battle-of-dien-bien-phu",
```

```
        "image_url":
         ↪  "https://assets.editorial.aetnd.com/uploads/2009/10/battle-of-dien-bien-phu-getty
    },
    {
        "coords": (10.59194693146919, 104.92222286168393),
        "title": "Ba Chuc massacre",
        "content": "Khmer Rouge, a terror regime led by Pol Pot which killed
         ↪  millions of Cambodians which sought Vietnam was their next
         ↪  target. After their defeats in 1977 where the Khmer Rouge killed
         ↪  1000 Vietnamese, and Vietnam retaliated by launching several
         ↪  successful campaigns, the Khmer Rouge came back in 1978. In April
         ↪  of that year, they crossed the border and attacked Ba Chuc
         ↪  village. Only 12 days, they mass murdered 3000 Vietnamese and
         ↪  there were only 2 known survivors. These caused Vietnam to attack
         ↪  full force and capture Phnom Penh (the capital of Cambodia) in
         ↪  January of 1979, ending the Pol Pot regime. China, who supported
         ↪  the Khmer Rouge financial, feared the expansion of Vietnam,
         ↪  started a Sino-Vietnamese War which lasted for 27 days.\nSource:
         ↪  https://solotravellerontour.com/mekong-delta-border-loop-ba-chuc-tomb/",
        "image_url":
         ↪  "https://todayinhistory.blog/wp-content/uploads/2019/01/killingfields-ft-1.m.jpg
    },
    {
        "coords": (15.266432532765865, 108.87073271815387),
        "title": "My Lai massacre",
        "content": "After the Tet Offensive, the American soldiers' morale
         ↪  was dwindling, and they heard words that the Viet Cong was hiding
         ↪  inside village My Lai in Son My. The Americans then sent soldiers
         ↪  to the village to investigate but found only children, women, old
         ↪  men, and little to no weapons. Nonetheless, the William Calley,
         ↪  the commander of the raid, ordered his men to shoot the
         ↪  villagers. On that day, March 16, 1968, more than 500 people were
         ↪  slaughtered with many young girls and women were raped and then
         ↪  killed. The brutality of My Lai massacre was covered by a year
         ↪  until it was exposed, and this fuelled anti-war sentiment and
         ↪  further divided the US over the Vietnam War.\nSource:
         ↪  https://www.history.com/topics/vietnam-war/my-lai-massacre-1",
        "image_url":
         ↪  "https://lh5.googleusercontent.com/p/AF1QipPg0wtVE4e8-fF4djBNHoO02LjbYLJGWPwwR31
    },
    {
        "coords": (15.792439795074536, 108.11623074979195),
```

```
        "title": "My Son Sanctuary",
        "content": "My Son Sanctuary is a remnant of the religious and
        ↪  political capital of the Champa Kingdom that ruled over central
        ↪  Vietnam between the 4th and 13th centuries. The people of Cham
        ↪  originated from India and the remnant shows impressive tower
        ↪  temples influenced by Hinduism.\nSource:
        ↪  https://whc.unesco.org/en/list/949/",
        "image_url":
        ↪  "https://lh5.googleusercontent.com/p/AF1QipOJB4gopDoyd5npVWRpj150XMZT_v52EHpja9N
    },
]

# Create the map at Da Nang, the central of Vietnam
vietnam_map = folium.Map(location=[15.846337671503933, 108.11359994009183],
 ↪  zoom_start=5.4)

# Add markers to the Vietnam map
for location in locations:
    iframe = folium.IFrame(

        ↪  f"<strong>{location['title']}</strong><br>{location['content']}<br><img
        ↪  src='{location['image_url']}' alt='hist_img' width='270px'>",
        width=300,
        height=300
    )
    popup = folium.Popup(iframe, max_width=400)
    folium.Marker(
        location=location['coords'],
        popup=popup,
        tooltip=location['title']
    ).add_to(vietnam_map)

# Add the Ho Chi Minh trail on the map
folium.PolyLine(ho_chi_minh_trail, color="red", weight=2.5,
 ↪  opacity=1).add_to(vietnam_map)

# Save the map as an HTML file
vietnam_map.save("vietnam_map.html")
```

To view the interactive map, please click here.

The map shows different events and locations from the `Post`. There are many interesting

topics discussed and I have provide a summary for the majority of them on this Vietnam map. When you click on each of the marker, it will show the location and the event happened their. Also, the red line is the Ho Chi Minh Trail, the trail that sent North Vietnamese soilders and supplies into South Vietnam.

Emotion detection for `Post`

Add columns from `df4`

I will add the `PostId` for identification and `Post3` for emotion detection

```
# Merge and rearrange columns
df8 = df7.merge(df4[['PostId', 'Post', 'Post3']],
                left_on='Original Post', right_on='Post', how='left')
df8 = df8.drop_duplicates(['PostId']).reset_index(drop=True)
df8.drop(['Post'], axis=1, inplace=True)
print(df8.head(2))
```

```
   Post Topic                                      Original Post  \
0           0  What happened in the aftermath of the Tet offe...
1           0  South Vietnam was helped by US. Even when the ...

                                OpenAI Post Summary  \
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...

                     OpenAI Post Label  \
0  ['Chinese military support in wars']
1  ['Chinese military support in wars']

                                KeyBERT Post Label  PostId  \
0  ['vietnam', 'vietnamese', 'indochina', 'troops...     229
1  ['vietnam', 'vietnamese', 'indochina', 'troops...     991

                                              Post3
0  what happen in the aftermath of the Tet offens...
1  South Vietnam be help by US . even when the US...
```

Testing out the model

Here is the model I will be using: `https://huggingface.co/cardiffnlp/twitter-roberta-base-emotion-mul`

```
from transformers import pipeline, AutoTokenizer

model_name = "cardiffnlp/twitter-roberta-base-emotion-multilabel-latest"
emotionModel = pipeline("text-classification", model=model_name)
tokenizer = AutoTokenizer.from_pretrained(model_name)

label = emotionModel(df8.Post3[0])
print(df8.Post3[0], label)
```

```
WARNING:tensorflow:From C:\Users\tomde\AppData\Local\Programs\Python\Python311\Lib\site-packa

what happen in the aftermath of the Tet offensive to the Viet Cong ?   be it actually break

 Edit : I be fully aware that the USA lose the Vietnam war .   I be not interested in reitera

 this question focus on the counter insurgency part of the war which could be win but the ov
 [{'label': 'anticipation', 'score': 0.5553023219108582}]
```

Apply Emotion detection to `Post2`

Concepts explain

- **Tokenize**: convert a string of text into individual pieces known as tokens. Tokens can be words, subwords and characters. Example:
  - Text: “Hello, world!”
  - Tokens: [“Hello”, “,”, “world”, “!”] or [7592, 1010, 2088, 999] if the tokens are in pytorch tensors
- **Decode**: convert tokens into human language. Example
  - Tokens: [7592, 1010, 2088, 999]
  - Decoded Text: “Hello, world!”
- **Yield**: return values one at a time from a function, allowing it to be used as an iterator

Example of `yield`

```
def generator_function():
    yield 1
    yield 2
    yield 3
```

```
print(generator_function())
for value in generator_function():
    print(value)
```

```
<generator object generator_function at 0x000001EC227861F0>
1
2
3
```

Split the post into chunk

```
# Function to split text into token-length constrained chunks
def split_text(text, tokenizer, max_length):
    # Tokenize text in form of pytorch tensors, truncation enabled and max
     ↪ length set
    tokens = tokenizer(text, return_tensors='pt', truncation=True,
 ↪ max_length=max_length)
    # Extract the tokenized representation of the text
    input_ids = tokens['input_ids'][0].tolist()

    # Iterate over input IDs in chunks of size `max_length`
    for i in range(0, len(input_ids), max_length):
        # Get a chunk of input IDs of length 'max_length'
        chunk_ids = input_ids[i:i + max_length]
        # Decode the chunk of input IDs back to text
        chunk_text = tokenizer.decode(chunk_ids, skip_special_tokens=True)
        # Yield the chunk of text
        yield chunk_text

# Example use case
max_length = 512
chunks = list(split_text(df8.Post3[0], tokenizer, max_length)) # use
 ↪ `split_text` function to split text into chunks
for index, chunk in enumerate(chunks):
    print(f"Chunk {index + 1}: {chunk}")
```

```
Chunk 1: what happen in the aftermath of the Tet offensive to the Viet Cong?   be it actuall

 Edit : I be fully aware that the USA lose the Vietnam war.   I be not interested in reitera

 this question focus on the counter insurgency part of the war which could be win but the ov
```

The main function of the code block is to split the text from the post column into smaller token-length chunks. This is a necessary process as our model has a limitation on input sequence length.

Get the emotion of each post

```
# Function to get the dominant emotion from the a text
def get_emotion(text):
    max_length = 512 # Max token length of the Roberta model is 512
    emotions = []

    # Use `split_text` function to split text into chunks
    for chunk in split_text(text, tokenizer, max_length):
        # Predict emotion for the current chunk
        result = emotionModel(chunk)
        # Append the predicted emotion label to the `emotions` list
        emotions.append(result[0]['label'])

    # Return the most frequence emotions in the emotion list for that text
    if emotions:
        return max(set(emotions), key=emotions.count)
    return 'neutral' # return neutral of no emotion is found

# Apply to the `Post` column
df8['PostEmotion'] = df8['Post3'].apply(get_emotion)
print(df8['PostEmotion'])
```

```
0     anticipation
1     anticipation
2          disgust
3          disgust
4          disgust
          ...
69    anticipation
70    anticipation
71         disgust
72         sadness
73    anticipation
Name: PostEmotion, Length: 74, dtype: object
```

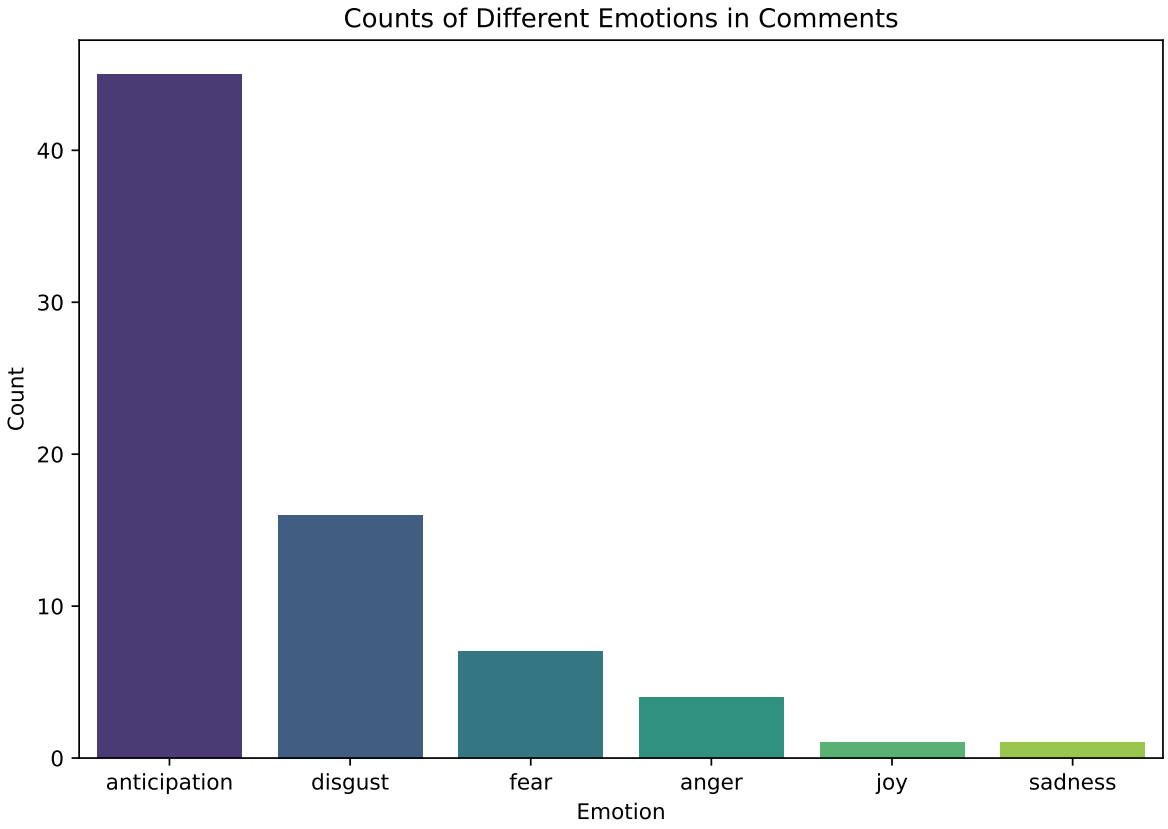
The code block returns a most prominent emotion detected based on the given text from the `split_text` function. Then it find the most frequence emotions of that post and set it as the overall emotion of the post.

Look for the most prominent emotion

```
emotion_counts = df8['PostEmotion'].value_counts()

# Create a bar plot
plt.figure(figsize=(9, 6))
sns.barplot(x=emotion_counts.index, y=emotion_counts.values,
 ↪  palette='viridis')

# Add titles and labels
plt.title('Counts of Different Emotions in Comments')
plt.xlabel('Emotion')
plt.ylabel('Count')
plt.show()
```



Since we are analyzing the `Post3` column, meaning that the texts are of question type. People are curious on the subject of Vietnamese History so the prominent emotion is anticipation. Disgust is the second most prominent emotion, which is understandable because many of the questions are war type questions.

Look for keywords in `Comment`

Merge comments data from `df4`

I will add the comment columns to our dataframe to look for keywords in each comment and for emotion detection.

```
# Merge comments data
df9 = df8.merge(df4[['PostId', 'Comment', 'Comment2', 'Comment3']],
 ↪  on='PostId', how='left')

# Fill NaN rows in comments column with empty strings
```

```
df9['Comment'] = df9['Comment'].fillna('')
df9['Comment2'] = df9['Comment2'].fillna('')
df9['Comment3'] = df9['Comment3'].fillna('')
print(df9.head(2))
```

```
   Post Topic                                      Original Post  \
0           0  What happened in the aftermath of the Tet offe...
1           0  South Vietnam was helped by US. Even when the ...

                               OpenAI Post Summary  \
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...

                     OpenAI Post Label  \
0  ['Chinese military support in wars']
1  ['Chinese military support in wars']

                                KeyBERT Post Label  PostId  \
0  ['vietnam', 'vietnamese', 'indochina', 'troops...     229
1  ['vietnam', 'vietnamese', 'indochina', 'troops...     991

                                             Post3   PostEmotion  \
0  what happen in the aftermath of the Tet offens...  anticipation
1  South Vietnam be help by US . even when the US...  anticipation

                                          Comment  \
0
1  The South was invaded in a regular war by the ...

                        Comment2  \
0
1  South invaded regular war North

                                         Comment3
0
1  the South be invade in a regular war by the No...
```

Use OpenAI to find if there is anything interesting that were discussed in the `comments`

I can not read through all of the comments because there are quite a lot of them, I will use openai to look out for certain keywords.

```
# client = openai.OpenAI(api_key="")

# def comment_analyze(text):
#     response = client.chat.completions.create(
#         model="gpt-3.5-turbo",
#         messages=[
#         {"role": "system", "content": "You are a expert in Vietnamese
 ↪  History. Return the keywords, events, characters related to Vietnamese
 ↪  History from this given text and nothing else. If the text is empty,
 ↪  return nothing"},
#         {"role": "user", "content": f"{text}"},
#       ]
#     )
#     return response.choices[0].message.content

# df9['Comment Keywords'] = df9['Comment'].dropna().apply(comment_analyze)
# df9.to_csv('df9.csv', index=False)

# Load the data
df9 = pd.read_csv('df9.csv')
print(df9.head())
```

```
   Post Topic                                      Original Post  \
0           0  What happened in the aftermath of the Tet offe...
1           0  South Vietnam was helped by US. Even when the ...
2           0  South Vietnam was helped by US. Even when the ...
3           4  After WWII France was in rebuilding mode, and ...
4           4  After WWII France was in rebuilding mode, and ...

                                OpenAI Post Summary  \
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...
2  Despite having strong support from the US and ...
3  Why did post-WWII France prioritize reclaiming...
4  Why did post-WWII France prioritize reclaiming...
```

```
                              OpenAI Post Label  \
0          ['Chinese military support in wars']
1          ['Chinese military support in wars']
2          ['Chinese military support in wars']
3  ["France's Post-WWII Indo-China Reclamation"]
4  ["France's Post-WWII Indo-China Reclamation"]

                                  KeyBERT Post Label  PostId  \
0  ['vietnam', 'vietnamese', 'indochina', 'troops...     229
1  ['vietnam', 'vietnamese', 'indochina', 'troops...     991
2  ['vietnam', 'vietnamese', 'indochina', 'troops...     991
3  ['indochina', 'france', 'reclaiming', 'china',...    1466
4  ['indochina', 'france', 'reclaiming', 'china',...    1466

                                               Post3   PostEmotion  \
0  what happen in the aftermath of the Tet offens...  anticipation
1  South Vietnam be help by US . even when the US...  anticipation
2  South Vietnam be help by US . even when the US...  anticipation
3  after WWII France be in rebuild mode , and yet...       disgust
4  after WWII France be in rebuild mode , and yet...       disgust

                                             Comment  \
0                                                NaN
1  The South was invaded in a regular war by the ...
2  I'll forego a lengthy analysis and simply reco...
3  Did you actually try to find some answer yours...
4  I did some searching, but as far as I could fi...

                                            Comment2  \
0                                                NaN
1                    South invaded regular war North
2  Ill forego lengthy analysis simply recommend b...
3                    actually try find answer asking
4  searching far could find French interested rec...

                                            Comment3  \
0                                                NaN
1  the South be invade in a regular war by the No...
2  I will forego a lengthy analysis and simply re...
3  do you actually try to find some answer yourse...
4  I do some searching , but as far as I could fi...

                                  Comment Keywords
```

```
0                                  Vietnamese History
1  Keywords: South, North, war, invasion\n\nEvent...
2             Keywords: Viet Nam, Nguyen Cao Ky, war
3  No keywords, events, or characters related to ...
4  Keywords: French, political, economic factors,...
```

I have ran the code and saved the output to a `csv` file. I will import the `csv` file for convenient sake.

Analysis on the keywords from openai for each post topic

I will convert the content that gpt returns to string via `Post Topic` and look out for certain keywords relate to Vietnamese History.

```
# Fill NaN values with empty strings in the 'Comment Keywords' column
df9['Comment Keywords'] = df9['Comment Keywords'].fillna('')

# Convert `Comment Keywords` to string based on `Post Topic`
topic_minus1 = df9.loc[df9['Post Topic'] == -1]['Comment Keywords']
topic_minus1 = '\n'.join(topic_minus1)
print(topic_minus1)
```

```
Vietnam, US troops, green flamed M62 tracers, Iraq, NATO, red tracers, Vietnam War, Warsaw Pa
No keywords, events, or characters related to Vietnamese History found in the text.
No keywords, events, or characters related to Vietnamese History were found in the given text
Vietnamese History: None
No keywords, events, or characters related to Vietnamese History were found in the provided t
Keywords: besiegers, catapult, dead animals, sewage, walls
Keywords: trebuchet, urinal
- India
- USA
- Thailand
- Philippines
Keywords: Vietnamese History, information
Operation Grommet
Thiệu, United States, military training, Command and General Staff College, Fort Leavenworth
Vietnamese History
divination purposes
```

I will repeat this for all of the `Post Topic`. Here are they keywords that I found. Thiệu (Former President of the Republic of Vietnam), Invasion of the South by the North (Sai Gon

Fall), Nguyen Cao Ky, Gulf of Tonkin incident (event that started Vietnam War), guerrilla tactics, Indo China, France's defeat and pulling out of Vietnam, Dien Bien Phu, Viet-Minh, Viet Cong, Ho Chi Minh, Indochina, Dien Bien Phu, Khmer Rouge, Pol Pot, Tet offensive, Geneva convention, Nixon, Ho Chi Minh trail, Agent Orange, John F. Kennedy, Chinese invasion of Vietnam, Diem.

These are some very interesting keywords. I will organize these keywords in a chronological order.

```
# Define a list of image and captions
images = ['indochina.png', 'ho chi minh.png', 'first indochina war.png', '17
 ↪  parallel.png',
          'viet cong.png', 'vietnam war.png', 'sai gon fall.png', 'agent
           ↪  orange.png', 'sino vietnam war.png']
titles = ['Indochina', 'Ho Chi Minh', 'First Indochina War', 'The Geneva
 ↪  Accords',
          'Viet Cong', 'Second Indochina War', 'Withdrawal of US troops and
           ↪  the war ends', 'Agent Orange', 'Sino-Vietnamese War']
captions = [
    "Indochina is a term to mention countries between Indian and China. Those
     ↪  countries blend the unique cultures of the two giants into countries
     ↪  we know today as Vietnam, Laos, Cambodia, Thailand, Myanmar, and
     ↪  parts of Malaysia. These countries have a long history with many
     ↪  kingdoms rose and fell. The French popularize the terms after they
     ↪  established colonies in Vietnam, Laos and Cambodia. During WW2, the
     ↪  Japanese took advantage of the French when they were defeated by the
     ↪  NAZI German and occupied the region. However, after WW2, Japan
     ↪  surrendered to the Allied because of nuclear weapons, leading to a
     ↪  power vacuum the SEA (Southeast Asia). Seizing this opportunity, the
     ↪  French tried to regain control of the region, and this led to the
     ↪  First Indochina War. Source:
     ↪  https://kids.britannica.com/students/article/Indochina/275049",
    "The leader of Democratic Republic of Vietnam (North Vietnam) from 1945
     ↪  to 1969. He was one of the most influential communist leaders in the
     ↪  20th century. He founded Viet Minh, an Vietnamese organization that
     ↪  fought the French rule and later the Americans. Source:
     ↪  https://www.britannica.com/biography/Ho-Chi-Minh",
    "In 1946, Ho Chi Minh and the French had some peace negotiations that
     ↪  recognize Vietnam as a free state within the French Union. Despite
     ↪  some initially cooperations, the tensions were still high, and the
     ↪  French intension was still clear that they wanted Vietnam to be a
     ↪  colony. In that same year, the French proclaimed Cochinchina (South
     ↪  Vietnam) as an autonomous republic and launched an attack in Hai
     ↪  Phong, the biggest port city in Northern Vietnam. That resulted in
     ↪  thousands of civilian casualties and sparked the beginning of the
     ↪  First Indochina War. In 1949, France reunited South Vietnam and
     ↪  appointed Bao Dai (former emperor of the Nguyen Dynasty) to the chief
     ↪  of state. However, most nationalists denounced Bao Dai leadership and
     ↪  the political in the South was in a state of struggles. The
     ↪  Americans, who feared for the Communist Domino Effect, supported the
     ↪  French with ammunition and money. But, with the fall of French
     ↪  garrison at Dien Bien Phu, which effectively ended the French Colony
     ↪  in Vietnam and led to the Geneva Conference. Source:
     ↪  https://www.britannica.com/place/Vietnam/World-War-II-and-independence",
```

```
"The Geneva Accords in 1954 were signed by the Viet Minh and French that
 ↪ cease-fire and temporary divide the country into two at latitude 17
 ↪ or 17th parallel. All military forces must withdraw, and the
 ↪ Vietnamese people were allowed to relocate between 2 countries for a
 ↪ limited time frame. The Final Declaration of the Accords states that
 ↪ an election needs to be held to unify the country but that never
 ↪ happened. During the mass migration from the North to South of around
 ↪ 1 million people, the two Vietnams began to reform their countries.
 ↪ The North, with the supports of China and Soviet Union, began to
 ↪ embark the program of socialist industrialization and collected
 ↪ agriculture. Whereas the South, Ngo Dinh Diem was appointed as a new
 ↪ president with the support of John F. Kennedy (president of America).
 ↪ Diem achieved this by removing Bao Dai in a government-controlled
 ↪ referendum. Diem, who was an anti-communist, showed obvious signs of
 ↪ dictatorship in his presidency. From consolidating power within his
 ↪ family, apply totalitarian methods to anyone who did not meet his
 ↪ eyes, show favoritism to Roman Catholics while alienated Buddhist in
 ↪ a Buddhism country. All of these ultimately led to his demise when
 ↪ the Viet Cong launched an insurgency movement which further
 ↪ destabilize the country, and when Diem's general overthrew him,
 ↪ leading to his and his brother's death. Source:
 ↪ https://www.britannica.com/place/Vietnam/World-War-II-and-independence",
"Communist guerrilla force. A force that is separated but an ally of
 ↪ people's army of Vietnam. Viet cong was the name given by President
 ↪ Ngo Dinh Diem to belittle the rebels. In the initial stage, Viet Cong
 ↪ is a collection of groups, but later in 1960, they merged into one
 ↪ with an official name of the National Liberation Front (NLF).
 ↪ Although the name changes, the main objective of the group was to
 ↪ overthrow the South Vietnamese government and reunify Vietnam.
 ↪ Source: https://www.britannica.com/topic/Viet-Cong",
"After the death of Ngo Dinh Diem, South Vietnam politics was in chaos,
 ↪ until in 1955, Nguyen Cao Ky who led the military to seized power.
 ↪ Although Ky faced opposition from Buddhists who overthrown Diem, he
 ↪ suppressed them. This oppressive regime did not change after the
 ↪ election of a new president Nguyen Van Thieu. The regime continues to
 ↪ suppress political freedoms and operate in a repressive framework. On
 ↪ the other hand, the North and the Viet Cong soldiers insurged South
 ↪ Vietnam from 1963 (Diem's death) with 30,000 fighters to 150,000 in
 ↪ 1965. This caused the American to fear for the fall of Sai Gon.
 ↪ Fearing the thread of North Vietnam, the American supported the South
 ↪ regime with the addition of many advisers, and ammunitions, but it
 ↪ could not halt the advance of Northern Vietnam. After the Congress
 ↪ approve the for war in Vietnam after the Gulf of Tonkin incident,
 ↪ President Lyndon B. Johnson appointed 500,000 troops in Vietnam to
 ↪ fight along slide 600,000 South Vietnamese soldiers and the allied
 ↪ from South Korean, Thailand, Australia, and New Zealand. From 1965 to
 ↪ 1968 began the intensive bombing of the Americans to the North
 ↪ Vietnam. Despise that, it only strengthens the resilient of the PVA
 ↪ (People's Army of Vietnam) and the Viet Cong to unify the country.
 ↪ Infiltration of personnel and supplies down the Ho Chi Minh Trail
 ↪ continued to increase with an estimation of more than 100,000 regular
 ↪ troops from the north. The evident of continuing strength became
 ↪ clear during the Tet Offensive where the North Vietnam attacked more
```

```
    "The war still continued under the new US president Richard M. Nixon, but
     ↪  a peace treaty was signed in Paris. As mentioned on the maps,
     ↪  President Nixon had a policy of Vietnamization where he wanted to end
     ↪  the US involvement in the Vietnam War. In the peace treaty signed by
     ↪  the US and three parties in Vietnam (North Vietnam, South Vietnam,
     ↪  and Viet Cong), the US needed to leave Vietnam within 60 days and
     ↪  create a political process to resolute the conflict in the south.
     ↪  However, the peace talks did not mention anything about the 100,000
     ↪  North Vietnamese soldiers stationed in the South, and so the civil
     ↪  war continued. In late 1974, North Vietnam launched a major offensive
     ↪  to the South, which caused the Sai Gon's troops to be in disarray.
     ↪  President Thieu ordered to retreat the troops back the Sai Gon, but
     ↪  it was too late. On April 30, 1975, the South Vietnam emerged
     ↪  victorious and thus united the country. Source:
     ↪  https://www.britannica.com/place/Vietnam/The-two-Vietnams-1954-65",
    "Agent orange was an herbicide used by US during the Vietnam War. This
     ↪  herbicide contains a dangerous chemical called dioxin. The substance
     ↪  dioxin is highly contaminated and exposure to it can cause health
     ↪  issues such as cancers, diabetes, and birth defects. The Red Cross
     ↪  estimated that around 3 million Vietnamese have been affected by this
     ↪  dioxin and at least 150,000 children were born with severe birth
     ↪  defects. Until this day, there are still millions of Vietnamese and
     ↪  Americans are affected by Agent Orange. Additionally, in southern and
     ↪  central Vietnam, millions of acres of forest and farmland are
     ↪  defoliated by Agent Orange. Although Agent Orange was terminated by
     ↪  the US and there is no more Agent Orange chemical today, its impact
     ↪  is still there. Source:
     ↪  https://www.aspeninstitute.org/programs/agent-orange-in-vietnam-program/what-is-agent
    "This text is a copy from the map. Khmer Rouge, a terror regime led by
     ↪  Pol Pot which killed millions of Cambodians which sought Vietnam was
     ↪  their next target. After their defeats in 1977 where the Khmer Rouge
     ↪  killed 1000 Vietnamese, and Vietnam retaliated by launching several
     ↪  successful campaigns, the Khmer Rouge came back in 1978. In April of
     ↪  that year, they crossed the border and attacked Ba Chuc village. Only
     ↪  12 days, they mass murdered 3000 Vietnamese and there were only 2
     ↪  known survivors. These caused Vietnam to attack full force and
     ↪  capture Phnom Penh (the capital of Cambodia) in January of 1979,
     ↪  ending the Pol Pot regime. China, who supported the Khmer Rouge
     ↪  financial, feared the expansion of Vietnam, started a Sino-Vietnamese
     ↪  War which lasted for 27 days. Source:
     ↪  https://solotravellerontour.com/mekong-delta-border-loop-ba-chuc-tomb/"
]
```

```
# Create an HTML page
html_content = f"""
<!DOCTYPE html>
<html>
<head>
    <title>Vietnam War</title>
</head>
<body>
    <!-- Container -->
    <div class="container">

        <!-- Title of the event -->
        <h1 id="title">Indochina</h1>

        <!-- Image -->
        <img id="image" src="indochina.png" alt="Indochina">

        <!-- Caption -->
        <div id="caption" class="caption">Indochina is a term to mention
 ↪  countries between Indian and China. Those countries blend the unique
 ↪  cultures of the two giants into countries we know today as Vietnam, Laos,
 ↪  Cambodia, Thailand, Myanmar, and parts of Malaysia. These countries have
 ↪  a long history with many kingdoms rose and fell. The French popularize
 ↪  the terms after they established colonies in Vietnam, Laos and Cambodia.
 ↪  During WW2, the Japanese took advantage of the French when they were
 ↪  defeated by the NAZI German and occupied the region. However, after WW2,
 ↪  Japan surrendered to the Allied because of nuclear weapons, leading to a
 ↪  power vacuum the SEA (Southeast Asia). Seizing this opportunity, the
 ↪  French tried to regain control of the region, and this led to the First
 ↪  Indochina War. Source:
 ↪  https://kids.britannica.com/students/article/Indochina/275049</div>

        <!-- Navigation buttons -->
        <div class="nav-buttons">
            <button onclick="navigate(-1)">Previous</button>
            <button onclick="navigate(1)">Next</button>
        </div>
    </div>

    <!-- CSS -->
    <style>
```

```
        .container {{
            width: 60%;
            margin: auto;
            text-align: center;
        }}
        img {{
            width: 500px;
            height: 500px;
        }}
        .caption {{
            font-size: 20px;
            text-align: left;
            margin: 18px 0;
        }}
        .nav-buttons {{
            display: flex;
            justify-content: space-between;
        }}
        .nav-buttons button {{
            padding: 10px 20px;
            font-size: 16px;
        }}
    </style>

    <!-- JS -->
    <script>
        var images = {images}; // Array that stores images
        var titles = {titles}; // Array that stores titles
        var captions = {captions}; // Array that stores captions
        var currentIndex = 0; // Initial index

        <!-- Function that updates slide shows based on direction. -1 means
↪   previous, +1 means next -->
        function navigate(direction) {{
            <!-- circular naviation based on current index -->
            currentIndex = (currentIndex + direction + images.length) %
↪   images.length;

            <!-- Update the image -->
            document.getElementById("image").src = images[currentIndex];

            <!-- Update the title -->
```

```
            document.getElementById("title").innerText =
 ↪  titles[currentIndex];

            <!-- Update the caption -->
            document.getElementById("caption").innerText =
 ↪  captions[currentIndex];
        }}
    </script>
</body>
</html>
"""

# Save the HTML page
output_file = "vietnam_war.html"
with open(output_file, "w") as file:
    file.write(html_content)
```

To view the interactive widget, please click here.

In this section, I have used the keywords from the comments and provided a summary based on each of the keywords on the Vietnamese History, more specifically on the events leading up to the Vietnam War, the Vietnam War, and what happened after it.

Emotion detection in `Comments`

Theses part is similar to the emotion detection in `Post`

Split the post into chunk

```
# Fill NaN values with empty strings
df9['Comment3'] = df9['Comment3'].fillna('')
df9['Comment'] = df9['Comment'].fillna('')

# Function to split text into token-length constrained chunks
def split_text(text, tokenizer, max_length):
    # Tokenize text in form of pytorch tensors, truncation enabled and max
     ↪  length set
    tokens = tokenizer(text, return_tensors='pt', truncation=True,
 ↪  max_length=max_length)
    # Extract the tokenized representation of the text
```

```
    input_ids = tokens['input_ids'][0].tolist()

    # Iterate over input IDs in chunks of size `max_length`
    for i in range(0, len(input_ids), max_length):
        # Get a chunk of input IDs of length 'max_length'
        chunk_ids = input_ids[i:i + max_length]
        # Decode the chunk of input IDs back to text
        chunk_text = tokenizer.decode(chunk_ids, skip_special_tokens=True)
        # Yield the chunk of text
        yield chunk_text

# Example use case
max_length = 512
chunks = list(split_text(df9.Comment3[1], tokenizer, max_length)) # use
 ↪  `split_text` function to split text into chunks
for index, chunk in enumerate(chunks):
    print(f"Chunk {index + 1}: {chunk}")
```

```
Chunk 1: the South be invade in a regular war by the North.
```

The main function of the code block is to split the text from the comment column into smaller token-length chunks. This is a necessary process as our model has a limitation on input sequence length.

Get the emotion of each comment

```
# Function to get the dominant emotion from the a text
def get_emotion(text):
    max_length = 512 # Max token length of the Roberta model
    emotions = []

    # Use `split_text` function to split text into chunks
    for chunk in split_text(text, tokenizer, max_length):
        # Predict emotion for the current chunk
        result = emotionModel(chunk)
        # Append the predicted emotion label to the `emotions` list
        emotions.append(result[0]['label'])

    # Return the most frequence emotions in the emotion list for that text
    if emotions:
```

```
        return max(set(emotions), key=emotions.count)
    return 'neutral' # return neutral of no emotion is found

# Apply to the `Post` column
df9['CommentEmotion'] = df9['Comment3'].apply(get_emotion)
# Remove comment emotion if the comment is empty
df9.loc[df9.Comment3 == '', 'CommentEmotion'] = 'neutral'
print(df9.head())
```

```
   Post Topic                                        Original Post  \
0           0  What happened in the aftermath of the Tet offe...
1           0  South Vietnam was helped by US. Even when the ...
2           0  South Vietnam was helped by US. Even when the ...
3           4  After WWII France was in rebuilding mode, and ...
4           4  After WWII France was in rebuilding mode, and ...

                                OpenAI Post Summary  \
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...
2  Despite having strong support from the US and ...
3  Why did post-WWII France prioritize reclaiming...
4  Why did post-WWII France prioritize reclaiming...

                               OpenAI Post Label  \
0          ['Chinese military support in wars']
1          ['Chinese military support in wars']
2          ['Chinese military support in wars']
3  ["France's Post-WWII Indo-China Reclamation"]
4  ["France's Post-WWII Indo-China Reclamation"]

                                KeyBERT Post Label  PostId  \
0  ['vietnam', 'vietnamese', 'indochina', 'troops...     229
1  ['vietnam', 'vietnamese', 'indochina', 'troops...     991
2  ['vietnam', 'vietnamese', 'indochina', 'troops...     991
3  ['indochina', 'france', 'reclaiming', 'china',...    1466
4  ['indochina', 'france', 'reclaiming', 'china',...    1466

                                               Post3   PostEmotion  \
0  what happen in the aftermath of the Tet offens...  anticipation
1  South Vietnam be help by US . even when the US...  anticipation
2  South Vietnam be help by US . even when the US...  anticipation
```

```
3  after WWII France be in rebuild mode , and yet...       disgust
4  after WWII France be in rebuild mode , and yet...       disgust

                                                 Comment  \
0
1  The South was invaded in a regular war by the ...
2  I'll forego a lengthy analysis and simply reco...
3  Did you actually try to find some answer yours...
4  I did some searching, but as far as I could fi...

                                                Comment2  \
0                                                    NaN
1                    South invaded regular war North
2  Ill forego lengthy analysis simply recommend b...
3                    actually try find answer asking
4  searching far could find French interested rec...

                                                Comment3  \
0
1  the South be invade in a regular war by the No...
2  I will forego a lengthy analysis and simply re...
3  do you actually try to find some answer yourse...
4  I do some searching , but as far as I could fi...

                                  Comment Keywords CommentEmotion
0                                Vietnamese History        neutral
1  Keywords: South, North, war, invasion\n\nEvent...           fear
2            Keywords: Viet Nam, Nguyen Cao Ky, war   anticipation
3  No keywords, events, or characters related to ...   anticipation
4  Keywords: French, political, economic factors,...   anticipation
```

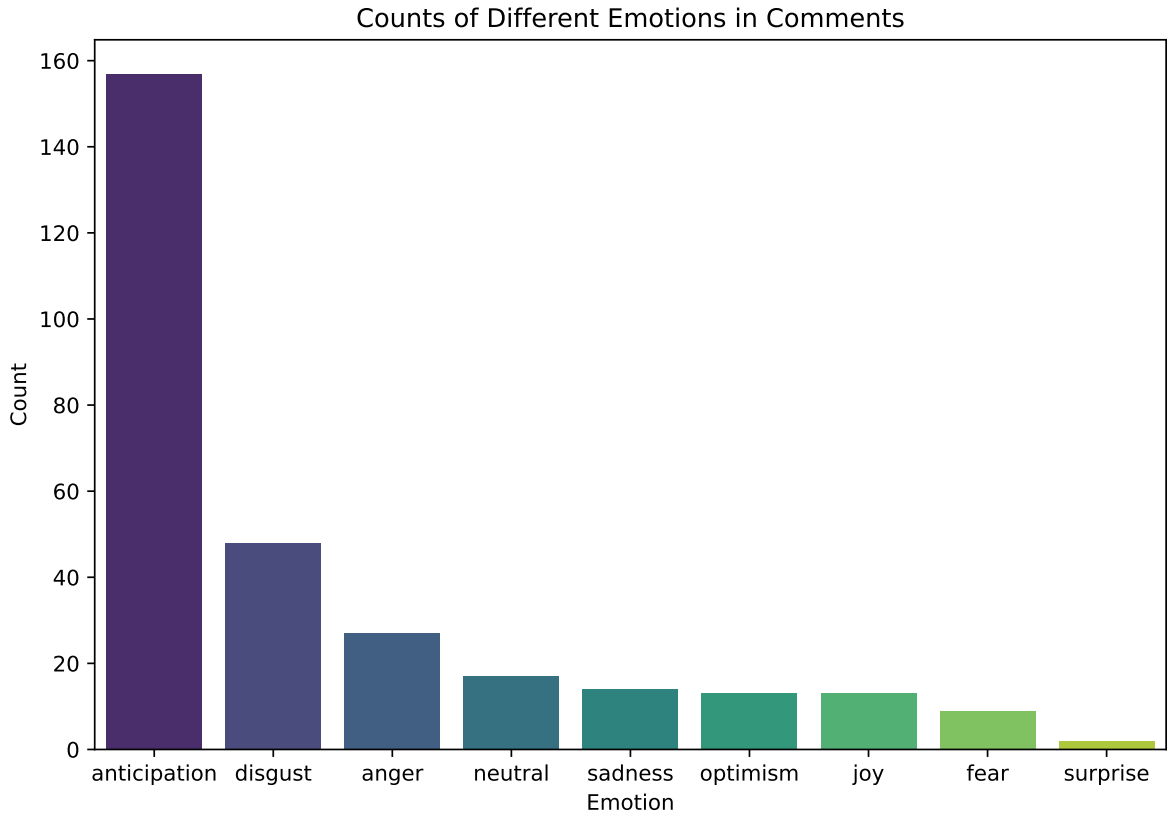
The code block returns a most prominent emotion detected based on the given text from the `split_text` function. Then it find the most frequence emotions of that comment and set it as the overall emotion of the comment.

Look for the most prominent emotion

```
# Count the occurrences of each emotions
emotion_counts = df9['CommentEmotion'].value_counts()
```

```
# Create a bar plot
plt.figure(figsize=(9, 6))
sns.barplot(x=emotion_counts.index, y=emotion_counts.values,
 ↪  palette='viridis')

# Add titles and labels
plt.title('Counts of Different Emotions in Comments')
plt.xlabel('Emotion')
plt.ylabel('Count')
plt.show()
```



The comment emotions distribution is quite similar to the post. As most of the comments are about war topic, and there are many unanswered questions leading to the feeling of anticipation. Negative emotions such as disgust, sadness and anger are also prominent. Like already mentioned, most of the topic about Vietnam History is about war, and war triggered many negative feelings.

Final table of post and comment deep analysis

Rearrange columns

```
df10 = df9.drop(['Comment2', 'Comment3', 'Post3'], axis=1)
df10 = df10.reindex(columns=['PostId', 'Post Topic', 'Original Post', 'OpenAI
 ↪  Post Summary',
                             'OpenAI Post Label', 'KeyBERT Post Label',
                             'PostEmotion', 'Comment', 'Comment Keywords',
                              ↪  'CommentEmotion'])
df10.rename({'PostEmotion': 'Post Emotion', 'CommentEmotion': 'Comment
 ↪  Emotion'}, axis=1, inplace=True)
print(df10.head())
```

```
   PostId  Post Topic                                      Original Post  \
0     229           0  What happened in the aftermath of the Tet offe...
1     991           0  South Vietnam was helped by US. Even when the ...
2     991           0  South Vietnam was helped by US. Even when the ...
3    1466           4  After WWII France was in rebuilding mode, and ...
4    1466           4  After WWII France was in rebuilding mode, and ...

                              OpenAI Post Summary  \
0  What happened to the Viet Cong after the Tet O...
1  Despite having strong support from the US and ...
2  Despite having strong support from the US and ...
3  Why did post-WWII France prioritize reclaiming...
4  Why did post-WWII France prioritize reclaiming...

                             OpenAI Post Label  \
0          ['Chinese military support in wars']
1          ['Chinese military support in wars']
2          ['Chinese military support in wars']
3  ["France's Post-WWII Indo-China Reclamation"]
4  ["France's Post-WWII Indo-China Reclamation"]

                               KeyBERT Post Label  Post Emotion  \
0  ['vietnam', 'vietnamese', 'indochina', 'troops...  anticipation
1  ['vietnam', 'vietnamese', 'indochina', 'troops...  anticipation
2  ['vietnam', 'vietnamese', 'indochina', 'troops...  anticipation
3  ['indochina', 'france', 'reclaiming', 'china',...       disgust
4  ['indochina', 'france', 'reclaiming', 'china',...       disgust
```

```
                                                Comment  \
0
1  The South was invaded in a regular war by the ...
2  I'll forego a lengthy analysis and simply reco...
3  Did you actually try to find some answer yours...
4  I did some searching, but as far as I could fi...

                                  Comment Keywords Comment Emotion
0                                Vietnamese History         neutral
1  Keywords: South, North, war, invasion\n\nEvent...            fear
2            Keywords: Viet Nam, Nguyen Cao Ky, war    anticipation
3  No keywords, events, or characters related to ...    anticipation
4  Keywords: French, political, economic factors,...    anticipation
```

This dataframe shows our deep analysis on the `Post` and `Comments`. It shows the `Post Topic` which is the Bertopic model categorizing the posts. `OpenAI Post Summary` is the summary generated to OpenAI. That couple with the `Original Post` column helped me to understand the overall theme of the posts. `OpenAI Post Label` and `KeyBERT Post Label` are the labelling of each topic from openai and keybert. The `Post Emotion` and `Comment Emotion` show the emotions for each post and comment. `Comment Keywords` is the column show casing the keywords of each comments, which help me to build an visualization on the chronological order of events happen prior, at and after the Vietnam War.

Conclusion

Here are the conclusion for the overall analysis and potential data privacy and ethics issues that arise during the my data analyzing process. First is the data preparation process where I convert the `xml` files from Stack Exchange to `csv` files. Then comes the data cleaning part. In the section, I will merge the different `csv` files together and use regexes along with other libraries to filter and clean my text columns. Next is the general analysis part where I look for the most common words in `Post` and `Comment` column. I also plot a time series graph on those most common words through out the years. After that, I need to perform a deeper analysis on the post and its corresponding comments. I first analyze the posts by using `OpenAI` and `Bertopic` to understand the overall theme and look out for different events and locations. These is crucial for plotting a Vietnam Map on different events happen during the Vietnamese History, more specifically the modern Vietnamese History and the location of the said event. I also apply `Emotion Detection` to identify the main emotions surrounding the posts. Second, I look at the comments. Since there are quite a lot of comments, I have gone for a different approach. I use `OpenAI` to return specific keywords and from those keywords, I have organized a chronological visualization on events surrouding those keywords. I also look for emotions surrounding each comments.

For the data privacy and ethics issues, I have some assumptions. Firstly, I was planning to use the `Users` dataset to plot a map of different users and their names to see who are interested in the Vietnam History and at what time. But, I feel like that is invading that person privacy and interest because I do not have their consent to use their information. I will just keep it as an anonymous QnA data. Also, I am a Vietnamese and analyzing different topics surrounding my country history. For that, I might be biased on my view towards certain topics. And Since this analysis is analyzing about war, meaning the context of its are sensitive, so different people of different origins may react differently while reading my analysis.