nlptask2-1

March 30, 2023

Text Analysis Using Python – Task1PartB

1. Introduction

Text analysis is the process of classifying, sorting, extracting and analysing text-based information using computer software, in order to understand human-written texts and further relationships and pattern of information. This report conducts text analysis on dataset that is associated with the COVID-19 vaccine expressions and will try to find out the narrative and insights of the dataset. The dataset involves a variety of expressions for 181 common questions about COVID-19 vaccine.

The columns of the dataset are as follow (): 1. Sentence - the expression written by an annotator (or taken from VIRADialogs) 2. label - the question for which the expression was written 3. label_idx - the running class index associated with this label

2. Importing, Loading and Describing Dataset

2.1. Installed required libraries using Python pip command and import the modules from the libraries.

[270]:
```
#Grpahs displayed in the notebook
%matplotlib inline

import pandas as pd
import seaborn as sns
import nltk
import matplotlib.pyplot as plt
from nltk.tokenize import word_tokenize
nltk.download('punkt')
nltk.download('stopwords')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data…
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
```

[270]:
```
True
```

2.2. Then, I used Pandas library to read data from CSV file, which was in my google drive

[271]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

[272]:
```
# After executing the cell above, Drive files will be present in "/content/
  ↪drive/My Drive". The below command lists the contents in the drive:
!ls "/content/drive/My Drive/Colab_Notebooks/ANLP"
```

```
ls: cannot access '/content/drive/My Drive/Colab_Notebooks/ANLP': No such file
or directory
```

[273]:
```
url = '/content/drive/My Drive/train_23.csv'
df = pd.read_csv(url)
df
```

[273]:

	sentence	label	label_idx
0	Do booster shots have side effects worsen than…	Are booster shot side effects worse than those…	175
1	the vaccine has side effects?	Are booster shot side effects worse than those…	175
2	booster vaccine leaves worse side effects than…	Are booster shot side effects worse than those…	175
3	are reinforcements safe?	Are booster shot side effects worse than those…	175
4	because the second dose of the covid-19 vaccin…	Are booster shot side effects worse than those…	175
…	…	…	…
5164	Would you define covid for me?	what is covid?	97
5165	hello, can you help me learn more about covid-19	what is covid?	97
5166	Explain what the Covid virus is.	what is covid?	97
5167	what was the real reason that the covid spread	what is covid?	97
5168	We cannot ignore what COVID means as it is imp…	what is covid?	97

[5169 rows x 3 columns]

2.3. I have first tried to look into summary and brief descriptive statstics of dataset, since dataset is very large.

[274]:
```
# Descriptive statstics that show the average, minimum and maximum of data
df.describe()
```

[274]:
```
          label_idx
count  5169.000000
mean     78.345521
std      54.090741
min       0.000000
25%      29.000000
50%      74.000000
75%     121.000000
max     180.000000
```

[275]:
```
# General information about columns and data type of columns
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5169 entries, 0 to 5168
Data columns (total 3 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   sentence   5169 non-null   object
 1   label      5169 non-null   object
 2   label_idx  5169 non-null   int64 
dtypes: int64(1), object(2)
memory usage: 121.3+ KB
```

[276]:
```
# Showing first 10 rows of dataset
df.head(10)
```

[276]:

	sentence	label	label_idx
0	Do booster shots have side effects worsen than…	Are booster shot side effects worse than those…	175
1	the vaccine has side effects?	Are booster shot side effects worse than those…	175
2	booster vaccine leaves worse side effects than…	Are booster shot side effects worse than those…	175
3	are reinforcements safe?	Are booster shot side effects worse than those…	175
4	because the second dose of the covid-19 vaccin…	Are booster shot side effects worse than those…	175
5	really the side effects are worse after the fi…	Are booster shot side effects worse than those…	175
6	Will the second injection have fewer effects t…	Are booster shot side effects worse than those…	175
7	because the second dose of the vaccine causes …		
8	Are the side effects of the booster worse than…		
9	Does the booster have stronger side effects?		

```
7  Are booster shot side effects worse than those…        175
8  Are booster shot side effects worse than those…        175
9  Are booster shot side effects worse than those…        175
```

[277]:
```
# Showing random 10 rows of dataset
df.sample(10)
```

[277]:
```
                                              sentence  \
2774  should i get the vaccine even if i still have …
4248                                     What is mRNA?
4046  What happens if my child's school has a COVID-…
4978  When my child gets vaccinated, will he or she …
4670                      the vaccine must be mandatory
1418  I don't know if I'm pregnant, should I avoid h…
296   If I've already had covid do I still need to g…
2205     Is the vaccine safe if I have heart problems?
1338   Is the egg one of the components of the vaccine?
3493  after what corona virus did to the world, i ca…

                                                 label  label_idx
2774  I'm still experiencing COVID symptoms even aft…         89
4248                                     What is mRNA?         76
4046  What happens if there is a COVID-19 case at my…        174
4978  Will my child miss school when they get vaccin…        157
4670  Who is required to get vaccinated under the fe…        166
1418                 Does the vaccine impact pregnancy?         50
296   Can I get a second dose even after a COVID exp…         35
2205  I am concerned getting the vaccine because I h…          5
1338                    Does the vaccine contain eggs?        140
3493                         Tell me about the vaccine        121
```

3. Word Count

Word count is useful in text analysis to find out lengths of words and sentences

[278]:
```
s = df['sentence'][4]
print(s)
len(s.split())
```

```
because the second dose of the covid-19 vaccine is higher in side effects
```

[278]:
```
13
```

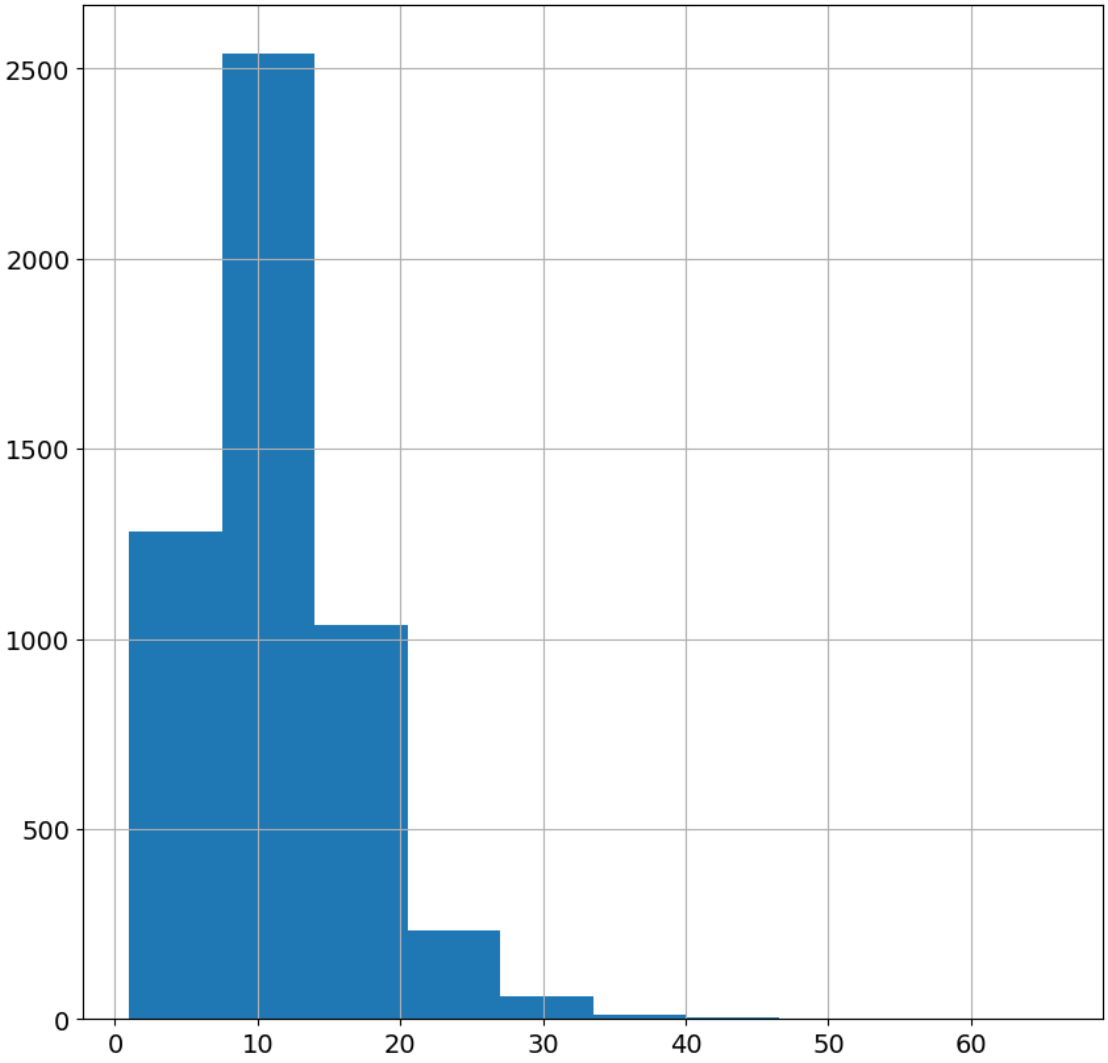
[279]:
```
def word_count(sentence):
    wc = len(sentence.split())
    return wc
```

I created word_count function to add word_count column in dataset

[280]:
```
df['word_count'] = df['sentence'].apply(word_count)
df['word_count'].describe()
df['word_count'].head(10)
```

[280]:
```
0    12
1     5
2    11
3     3
4    13
5    10
6    14
7    13
8    13
9     7
Name: word_count, dtype: int64
```

[281]:
```
df['word_count'].hist(bins = 10)
plt.figure(figsize=(2,2))
plt.rcParams['figure.figsize'] = (10,10)
plt.show()
```




```
<Figure size 200x200 with 0 Axes>
```

Histogram of 'word count' column clearly showed the distribution of word lengths.

[282]:
```
#Showing Top 10 shortest word count
df.sort_values(by='word_count').head(10)
```

[282]:

	sentence	label	\
5125	covid-19	what is covid?	
4195	whats?	What is in the vaccine?	
5068	erectile disfunction?	Will the vaccine make me sterile or infertile?	
613	for kids	Can children get the vaccine?	
3540	its safe?	The COVID vaccine is not safe	
1423	I'm prefnant	Does the vaccine impact pregnancy?	
2408	its necessary?	I don't think the vaccine is necessary	

```
5161        emergency covid?                                  what is covid?
3470       secondary effects     Side effects and adverse reactions worry me
172   covid-19 is dangerous?        COVID-19 is not as dangerous as they say

      label_idx  word_count
5125         97           1
4195         28           1
5068         86           2
613          90           2
3540         23           2
1423         50           2
2408         12           2
5161         97           2
3470         22           2
172           0           3
```

[283]:
```
#Showing Top 10 longest word count
df.sort_values(by='word_count', ascending=False).head(10)
```

[283]:
```
                                              sentence  \
2929  Make sure you understand and comply with all t…
3750  The fear that a vaccine will somehow change yo…
2448  These companies could be out to make a profit …
4687  For many, a nationwide return to normalcy from…
3159  The standard FDA approval process is like a si…
3493  after what corona virus did to the world, i ca…
2055  The COVID-19 vaccine will be free for all, reg…
4089  No vaccine is 100% effective so it is still th…
4683  As the US may miss a vaccination goal set by P…
5067  I was told they dont know what kind of effect …

                                               label  label_idx  word_count
2929  Is it okay for me to travel internationally if…        107          66
3750    They will put a chip/microchip to manipulate me         26          62
2448  I don't trust the companies producing the vacc…         13          56
4687  Why are COVID-19 vaccination rates slowing in …        131          55
3159                      Is the vaccine FDA approved?        158          46
3493                         Tell me about the vaccine        121          43
2055       How much will I have to pay for the vaccine         11          40
4089  What if I still get infected even after receiv…         73          37
4683  Why are COVID-19 vaccination rates slowing in …        131          36
5067     Will the vaccine make me sterile or infertile?         86          36
```

4. Word Frequency

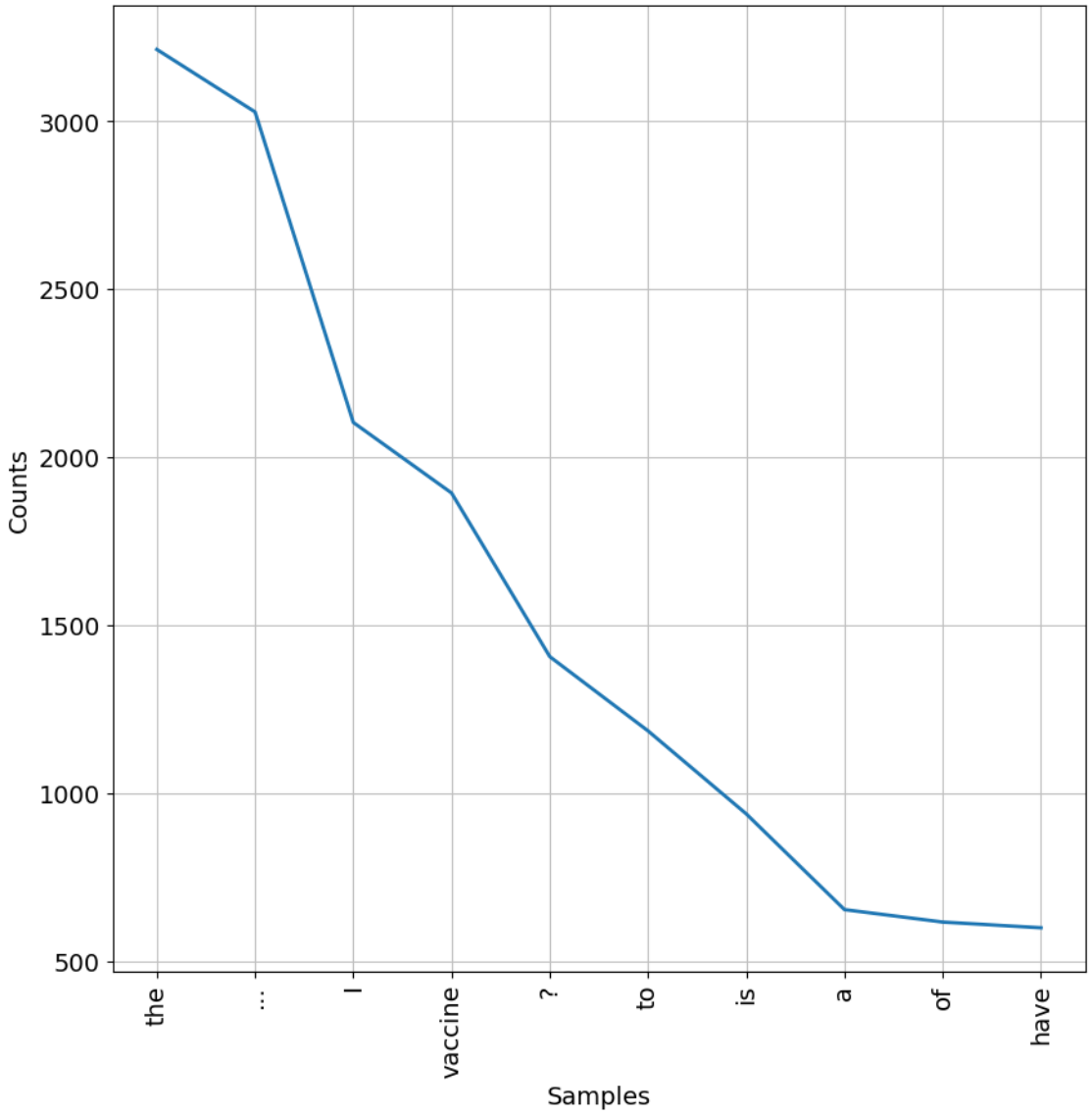
Word frequency is important phase in text analysis to measure how often the specific words appear. The more frequently words appear indicate that words are more important.

[284]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5169 entries, 0 to 5168
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   sentence    5169 non-null   object
 1   label       5169 non-null   object
 2   label_idx   5169 non-null   int64 
 3   word_count  5169 non-null   int64 
dtypes: int64(2), object(2)
memory usage: 161.7+ KB
```

The data type of 'sentence' column is object. It needs to be changed to String data type, in order to tokenize words.

[285]:
```
# Change 'sentence' column data type from object to string
words = df['sentence'].to_string()
tokenized_words = word_tokenize(words)
all_words=nltk.FreqDist(tokenized_words)
all_words.plot(10);
plt.rcParams['figure.figsize'] = (10,10)
print(all_words.most_common(20))
```




```
[('the', 3213), ('…', 3027), ('I', 2103), ('vaccine', 1893), ('?', 1406),
('to', 1185), ('is', 938), ('a', 653), ('of', 616), ('have', 599), ('get', 597),
('it', 498), ('be', 495), ('if', 493), ('do', 487), (',', 486), ('are', 459),
('that', 427), ('What', 400), ('Is', 397)]
```

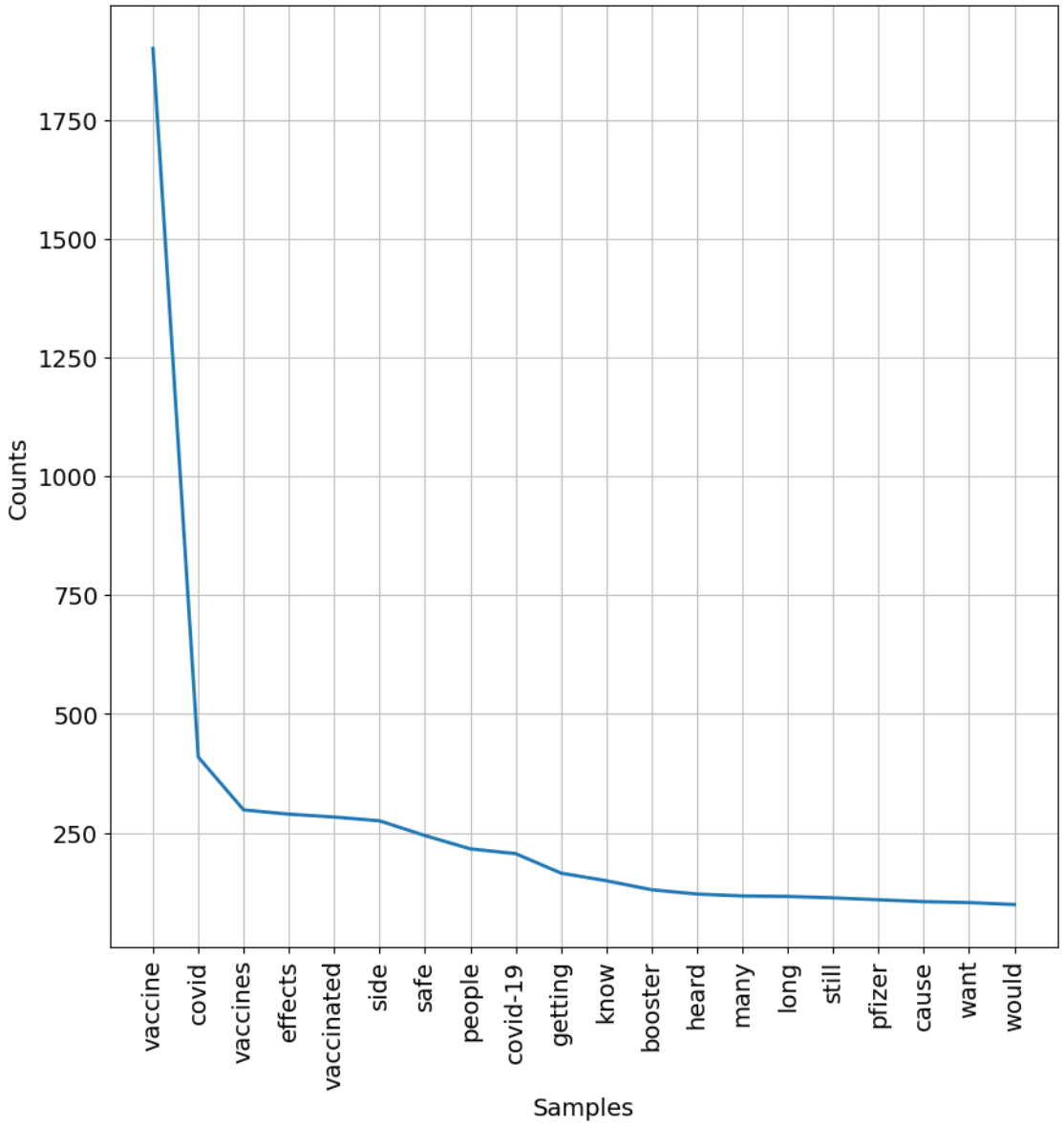
Tokenized words contain stop words (i.e., unnecessary words), such as 'the', 'I' and 'to', disturbing analysis of word frequency. Thus, stop words and other unrelated words need to be removed.

[286]:
```
text_lower = words.lower()
# Only tokenized words that contain lower case letters, as imporatnt words are␣
  ↪more likely to be specific nouns.
tokenized_words=word_tokenize(text_lower)
```

```
import string
from nltk.corpus import stopwords
#stopwords "english" contains many subjective and objective words such as 'i',␣
 ↪'myself' and 'you'.
stop_words=stopwords.words("english")
print(stop_words)
# Adding more unnecessary words into stop_words.
stop_words.extend(["get","need", "s", "a", "the", "I", "What", "Is", "How",␣
 ↪"n't", "Will", "Can", "Does", "Are", "If", "The", "'ve", "'m", "j", "th",␣
 ↪"go",])

filtered_tokens = []
for i in tokenized_words:
  if i not in stop_words:
    filtered_tokens.append(i)

# punctuations, which contain '?', '' and '!'
punctuations=list(string.punctuation)
#Add custom punctuations to the list
punctuations.append("...")
punctuations.append("?")

#Create a variable that include all filtered tokenized words.
filtered_tokens_final=[]
for i in filtered_tokens:
    if i not in punctuations:
        filtered_tokens_final.append(i)

all_words=nltk.FreqDist(filtered_tokens_final)
all_words.plot(20);
print(all_words.most_common(20))
```

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're",
"you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he',
'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's",
'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what',
'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is',
'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having',
'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or',
'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about',
'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above',
'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under',
'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why',
'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some',
'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very',
's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now',
'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn',
```

"couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn',
"hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't",
'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn',
"shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn',
"wouldn't"]



[('vaccine', 1900), ('covid', 409), ('vaccines', 298), ('effects', 289),
('vaccinated', 283), ('side', 275), ('safe', 244), ('people', 216), ('covid-19',
206), ('getting', 165), ('know', 149), ('booster', 130), ('heard', 121),
('many', 117), ('long', 116), ('still', 113), ('pfizer', 109), ('cause', 105),
('want', 103), ('would', 99)]

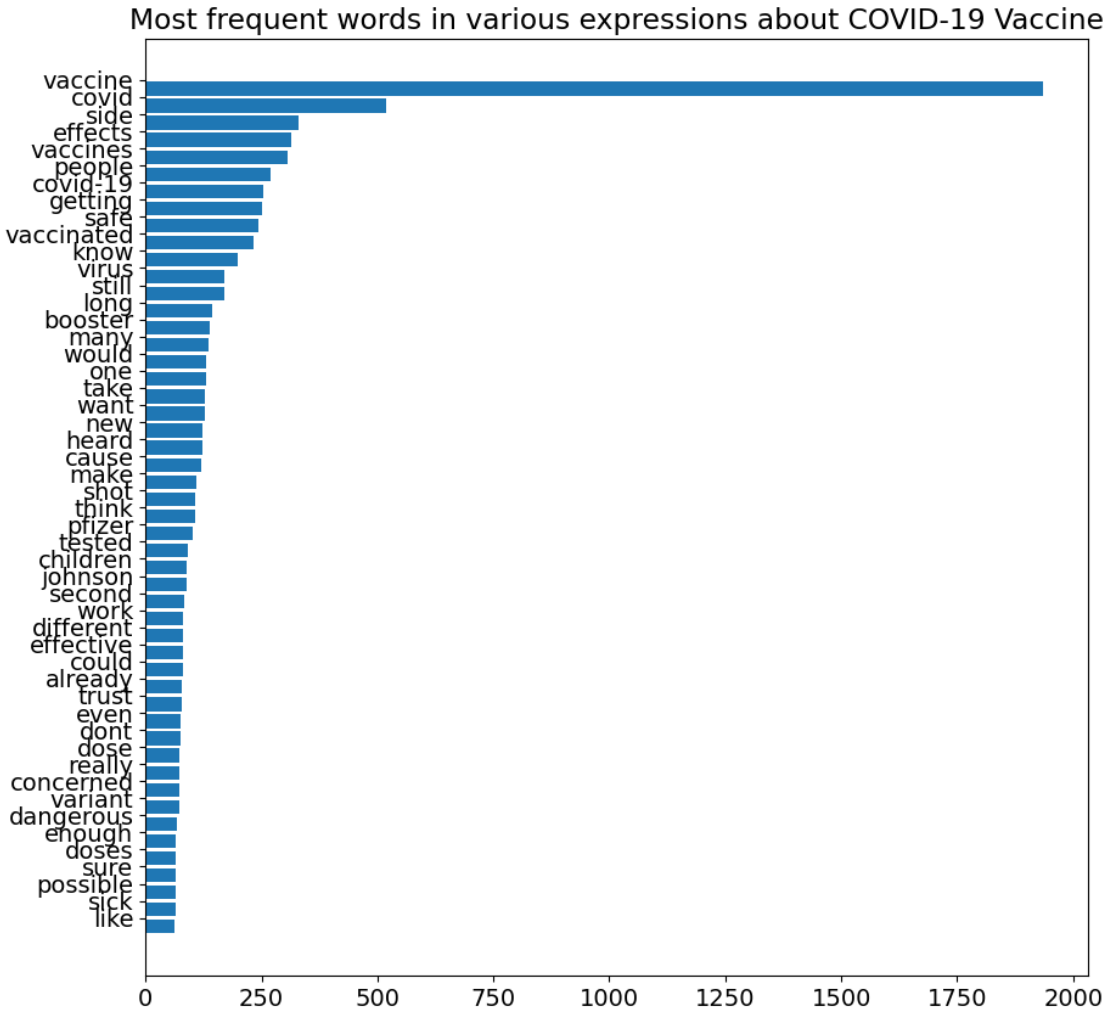
After filtering and cleaning data, the graph clearly shows some of the most important words (i.e., most frequently appeared), such as 'vaccine', 'covid' and 'booster', following with 'side', 'effect' and 'safe'.

[287]:
```
from collections import Counter
from nltk.corpus import stopwords
top_words = dict(Counter(all_words).most_common(100))
print(top_words)
```

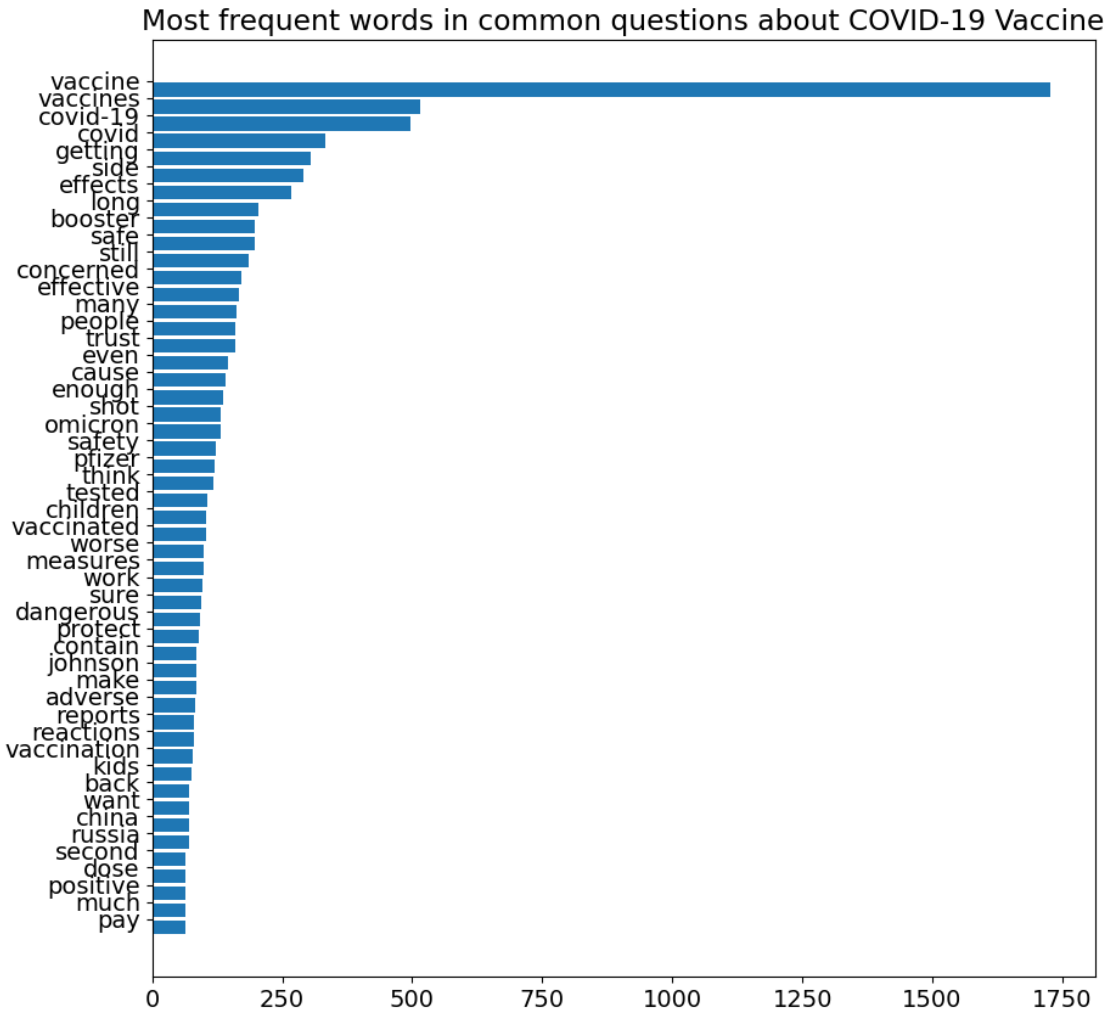
```
{'vaccine': 1900, 'covid': 409, 'vaccines': 298, 'effects': 289, 'vaccinated':
283, 'side': 275, 'safe': 244, 'people': 216, 'covid-19': 206, 'getting': 165,
'know': 149, 'booster': 130, 'heard': 121, 'many': 117, 'long': 116, 'still':
113, 'pfizer': 109, 'cause': 105, 'want': 103, 'would': 99, 'think': 98, 'shot':
96, 'virus': 95, 'children': 91, 'make': 82, 'take': 81, 'johnson': 79, 'new':
78, 'one': 75, 'dangerous': 74, 'tested': 74, 'effective': 71, 'variant': 70,
'work': 68, 'trust': 64, 'dose': 63, 'could': 63, 'concerned': 63, 'dont': 62,
'doses': 62, 'really': 61, 'enough': 57, 'worried': 57, 'kids': 57, 'possible':
56, 'different': 56, 'tell': 56, 'child': 53, 'mrna': 53, 'sick': 52, 'flu': 52,
"'s": 52, 'sure': 50, 'information': 49, 'government': 49, 'vac': 48, 'made':
48, 'necessary': 48, 'test': 48, 'already': 47, 'effect': 47, 'even': 47, 'va':
47, 'find': 47, 'wait': 47, 'worse': 46, 'safety': 46, 'like': 46, 'difference':
46, 'give': 44, 'immune': 43, 'v': 42, 'second': 41, 'c': 41, 'mask': 41,
'vacc': 41, 'vaccination': 41, 'immunity': 39, 'use': 38, 'may': 37, 'wear': 37,
'see': 37, 'protect': 37, 'others': 35, 'omricon': 35, 'companies': 35, 'delta':
34, 'shots': 33, 'true': 33, 'able': 33, 'ivermectin': 33, 'afraid': 32,
'young': 32, 'person': 31, 'school': 31, 'help': 31, 'died': 31, 'positive': 31,
'much': 30, 'stop': 30}
```

[288]:
```
# Create a function to check word frequency in two columns 'sentence' and␣
 ↪'label'
def wordBarGraphFunction(df,column,title):
    topic_words = [ z.lower() for y in
                        [ x.split() for x in df[column] if isinstance(x, str)]
                        for z in y]
    word_count_dict = dict(Counter(topic_words))
    popular_words = sorted(word_count_dict, key = word_count_dict.get, reverse␣
 ↪= True)
    popular_words_nonstop = [w for w in popular_words if w in␣
 ↪filtered_tokens_final]
    plt.barh(range(50), [word_count_dict[w] for w in␣
 ↪reversed(popular_words_nonstop[0:50])])
    plt.yticks([x + 0.5 for x in range(50)], reversed(popular_words_nonstop[0:
 ↪50]))
    plt.title(title)
    plt.show()
#wordCounterFunction(df, 'sentence')
plt.figure(figsize=(10,10))
```

```
wordBarGraphFunction(df,'sentence',"Most frequent words in various expressions␣
↪about COVID-19 Vaccine")
```



[289]:
```
plt.figure(figsize=(10,10))
wordBarGraphFunction(df,'label',"Most frequent words in common questions about␣
↪COVID-19 Vaccine")
```



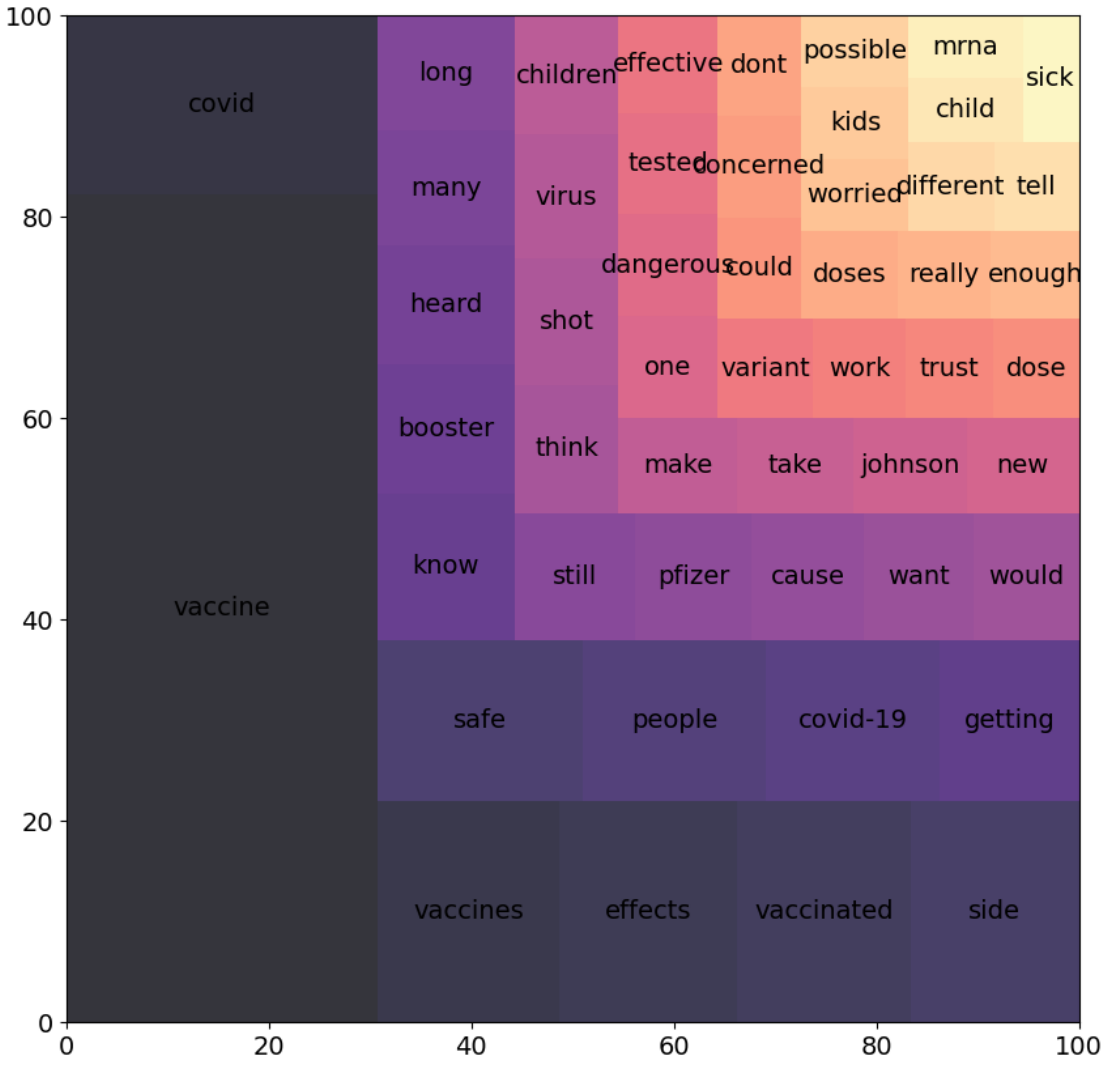
The first bar graph showed Top 50 most frequent words in expressions and the second bar graph showed most frequent words in common questions. As predicted, 'covid', 'vaccine', and 'booster' are the most frequent words. There are two words, 'side' and 'effect' but it is highly assumed that these two words are associated each other. This will be further analysed in word association phase. Also, 'safe' is another frequent word, which is an opposite word to 'side effect', indicating that covid-19 vaccine is a controversial topic.

[124]:
```
pip install squarify
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-
wheels/public/simple/
Requirement already satisfied: squarify in /usr/local/lib/python3.9/dist-
packages (0.4.3)
```

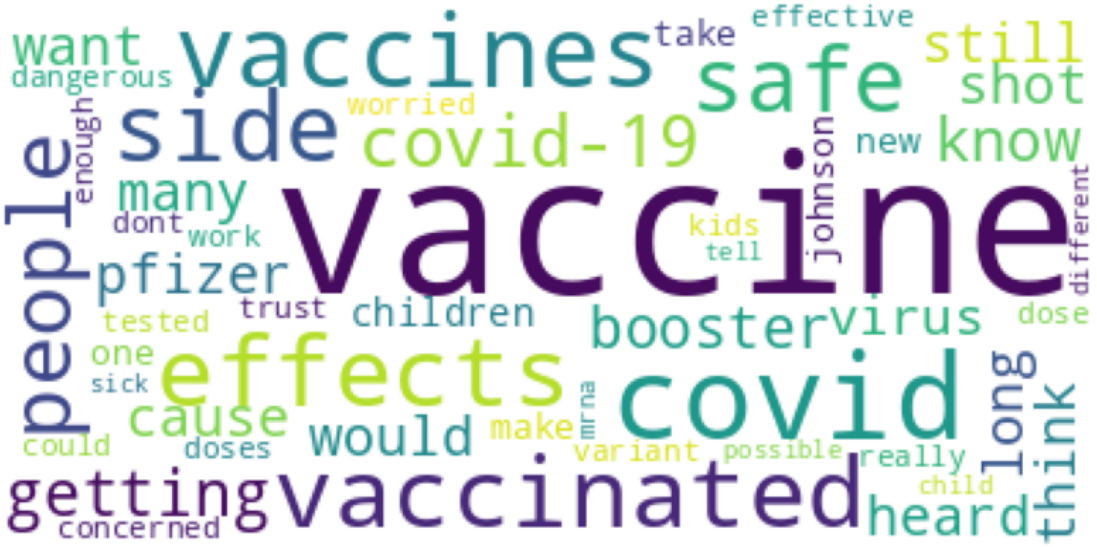
I installed 'squarify' to create a treemap of word frequency.

[290]:
```
import squarify
import matplotlib.pyplot as plt
import seaborn as sb

# Tree map visualization of most frequent words
top_words = dict(Counter(all_words).most_common(50))
plt.rcParams['figure.figsize'] = (10,10)
plt.rcParams['font.size'] = 14

squarify.plot(label= top_words.keys()
              ,sizes= top_words.values(),
              color = sb.color_palette("magma",
                                       len(top_words)),
              alpha= 0.8);
```



A tree map is another simple and clear way to show the most frequent and important words.

[291]:
```
# WordCloud Visualization of most frequent words
from wordcloud import WordCloud
wc = WordCloud(background_color = 'white')
wc.generate_from_frequencies(top_words)

import matplotlib.pyplot as plt
plt.imshow(wc)
plt.axis("off")
plt.show()
```



Word cloud emphasize the popularity of words by making words larger and bolder.

5. Word Association

Word association is the process of finding a natural combination of words that are closely associated each other. Word association is an essential phase in text analysis to discover a combination of two or more important words.

[292]:
```
from nltk.collocations import *
from nltk import BigramAssocMeasures
# Two words association
bigram_measures = BigramAssocMeasures()
finder = BigramCollocationFinder.from_words(filtered_tokens_final)

finder.nbest(bigram_measures.likelihood_ratio, 20)
```

[292]:
```
[('side', 'effects'),
 ('booster', 'shot'),
```

```
 ('johnson', 'johnson'),
 ('delta', 'variant'),
 ('herd', 'immunity'),
 ('many', 'doses'),
 ('booster', 'shots'),
 ('would', 'like'),
 ('wear', 'mask'),
 ('long', 'term'),
 ('covid', '19'),
 ('south', 'african'),
 ('blood', 'clots'),
 ('many', 'people'),
 ('priority', 'groups'),
 ('pfizer', 'vaccine'),
 ('lymph', 'nodes'),
 ('convalescent', 'plasma'),
 ('fetal', 'tissue'),
 ('united', 'states')]
```

As mentioned above, 'side' and 'effect' were two associated words, indicating the meaning of 'side effect'. The word 'side effect' may indicate that people were concerned with side effects of COVID-19 vaccine. We can also see that 'booster' and 'shot' meaning 'booster shot'.

[293]:
```
# Three words association
trigram_measures = nltk.collocations.TrigramAssocMeasures()

vaccine_filter = lambda *w: 'vaccine' not in w
finder = TrigramCollocationFinder.from_words(filtered_tokens_final)

# only trigrams that appear 3+ times
finder.apply_freq_filter(3)
# trigrams that contain 'vaccine'
finder.apply_ngram_filter(vaccine_filter)

for i in finder.score_ngrams(trigram_measures.likelihood_ratio):
    print (i)
```

```
(('vaccine', 'side', 'effects'), 2072.0306482029687)
(('side', 'effects', 'vaccine'), 2068.4985382179825)
(('johnson', 'johnson', 'vaccine'), 439.47087237032963)
(('pfizer', 'vaccine', 'cause'), 317.22534158539236)
(('many', 'doses', 'vaccine'), 270.47018475400563)
(('covid-19', 'vaccine', 'cause'), 260.2127653304978)
(('covid', 'vaccine', 'cause'), 222.83034602512043)
(('pfizer', 'vaccine', 'safe'), 220.8445740476576)
(('vaccine', 'help', 'new'), 211.79780886890657)
(('covid', '19', 'vaccine'), 206.60623900552721)
(('vaccine', 'cause', 'autism'), 199.30668218907647)
```

(('johnson', 'vaccine', 'cause'), 196.75941059618395)
(('doses', 'pfizer', 'vaccine'), 193.72516883678418)
(('vaccine', 'cause', 'harm'), 185.62753534398172)
(('know', 'pfizer', 'vaccine'), 184.51944847265207)
(('vaccine', 'tested', 'community'), 181.27722758546304)
(('know', 'vaccine', 'tested'), 171.75361423057865)
(('sure', 'vaccine', 'tested'), 163.90272428448282)
(('vaccine', 'cause', 'myocarditis'), 162.26752786635092)
(('ive', 'heard', 'vaccine'), 162.22887961054334)
(('vaccine', 'cause', 'death'), 157.54912941503676)
(('covid', 'vaccine', 'make'), 155.4138672651649)
(('think', 'vaccine', 'tested'), 153.78622623355568)
(('vaccine', 'cause', 'long'), 146.03102948826256)
(('vaccine', 'counter', 'new'), 141.8084100912625)
(('vaccine', 'make', 'sick'), 140.70709186131265)
(('vaccine', 'work', 'new'), 140.40248721823968)
(('vaccine', 'change', 'dna'), 139.73002222442722)
(('vaccine', 'serve', 'new'), 136.97740098949228)
(('vaccine', 'effective', 'new'), 135.27769205893009)
(('vaccine', 'update', 'new'), 128.77248381824887)
(('covid', 'vaccine', 'last'), 125.00513194911034)
(('vaccine', 'make', 'sterile'), 123.95980525549086)
(('vaccine', 'could', 'worse'), 119.2490014021281)
(('covid', 'vaccine', 'safe'), 119.15340568890072)
(('link', 'covid', 'vaccine'), 111.00083779093369)
(('worth', 'getting', 'vaccine'), 110.35771747680198)
(('vaccine', 'contain', 'eggs'), 106.19680199141)
(('one', 'dose', 'vaccine'), 104.78243452617988)
(('worried', 'vaccine', 'make'), 103.4606567238358)
(('heard', 'vaccine', 'causes'), 102.39049769415732)
(('long', 'covid', 'vaccine'), 98.23418175465696)
(('vaccine', 'make', 'test'), 97.96202071913036)
(('point', 'getting', 'vaccine'), 97.35181485044708)
(('vaccine', 'make', 'immune'), 96.3536988405672)
(('heard', 'vaccine', 'contains'), 94.82049380658049)
(('heard', 'vaccine', 'recommended'), 90.54803404199231)
(('booster', 'original', 'vaccine'), 89.83902818514478)
(('side', 'effect', 'vaccine'), 89.25638548183646)
(('dont', 'think', 'vaccine'), 89.00836538562714)
(('wo', 'take', 'vaccine'), 87.53694913368349)
(('dont', 'trust', 'vaccine'), 86.51580030188988)
(('covid', 'vaccine', 'still'), 85.6243926965611)
(('animal', 'products', 'vaccine'), 84.52107411548498)
(('know', 'vaccine', 'safe'), 81.63131723234713)
(('take', 'vaccine', 'even'), 80.54583001025844)
(('vaccine', 'works', 'agains'), 80.35827874932784)
(('1', 'dose', 'vaccine'), 79.89304900914553)
(('vaccine', 'fda', 'approved'), 78.0399512585712)

```
(('current', 'vaccine', 'works'), 74.78152521059414)
(('vaccine', 'contains', 'animal'), 74.72034466524529)
(('died', 'receiving', 'vaccine'), 74.1859051624861)
(('vaccine', 'contains', 'live'), 73.96125619956733)
(('covid-19', 'vaccine', 'different'), 72.56344402067248)
(('make', 'sure', 'vaccine'), 71.13480812250441)
(('true', 'vaccine', 'contains'), 70.11520746112217)
(('want', 'know', 'vaccine'), 67.79526870763354)
(('vaccine', 'stop', 'getting'), 65.88890352587735)
(('implement', 'vaccine', 'helps'), 65.77193538997253)
(('immunity', 'vaccine', 'last'), 65.1786647125998)
(('insurance', 'cover', 'vaccine'), 64.96265106512925)
(('concerned', 'vaccine', 'may'), 64.78464926175948)
(('vaccine', '100', 'effective'), 64.31356759744202)
(('multiple', 'jabs', 'vaccine'), 62.11120334449919)
(('kids', 'vaccine', "'re"), 61.86533711476636)
(('die', 'receiving', 'vaccine'), 61.01376207432569)
(('vaccine', 'helps', 'wit'), 60.36390732636839)
(('jabs', 'vaccine', 'n'), 59.57220188428447)
(('vaccine', 'safe', 'child'), 58.73908879270432)
(('two', 'doses', 'vaccine'), 56.89722005535213)
(('relatives', 'use', 'vaccine'), 47.748110158763325)
(('long', 'vaccine', 'protect'), 43.23820997681974)
(('vaccine', 'may', 'cause'), 42.49967532413291)
(('vaccine', 'even', 'ha'), 39.257721704001405)
(('think', 'current', 'vaccine'), 38.22268149351895)
(('would', 'vaccine', 'fda'), 36.33683207187823)
(('know', 'long', 'vaccine'), 27.33016813726335)
(('vaccine', 'really', 'safe'), 24.542004159345126)
```

Again, in assocation of three words, we can see 'vaccine', 'side', 'effect' most frequently occuring.

[294]:
```
# Four words association
fourgram_measures = nltk.collocations.QuadgramAssocMeasures()

finder = QuadgramCollocationFinder.from_words(filtered_tokens_final)
vaccine_filter = lambda *w: 'vaccine' not in w


finder.apply_freq_filter(3)

finder.apply_ngram_filter(vaccine_filter)

for i in finder.score_ngrams(fourgram_measures.likelihood_ratio):
    print (i)
```

```
(('vaccine', 'side', 'effects', 'common'), 2114.522345841054)
(('johnson', 'johnson', 'vaccine', 'cause'), 586.1258081035212)
```

```
(('heard', 'johnson', 'johnson', 'vaccine'), 470.6180336803278)
(('many', 'doses', 'pfizer', 'vaccine'), 461.82848827523947)
(('pfizer', 'vaccine', 'cause', 'myocarditis'), 357.7344718114797)
(('vaccine', 'help', 'new', 'mutations'), 272.1526843008809)
(('current', 'vaccine', 'works', 'agains'), 130.2440189533217)
(('implement', 'vaccine', 'helps', 'wit'), 115.79741343657933)
(('would', 'vaccine', 'fda', 'approved'), 111.0369329073071)
(('think', 'current', 'vaccine', 'works'), 107.94894030199669)
(('multiple', 'jabs', 'vaccine', 'n'), 106.63242261125944)
(('concerned', 'vaccine', 'may', 'cause'), 96.64645057468698)
```

6. K-means Clustering

K-means clustering is the process of grouping similar types or categories of data and discovering certain patterns of data. Clustering may be useful in sorting average opinions and expressions on COVID-19 vaccines.

[295]:
```
import nltk
nltk.download('punkt')
nltk.download('wordnet')
import string
from nltk import word_tokenize
from nltk.stem import WordNetLemmatizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

words = df['sentence'].to_string()

#tokenized_words=word_tokenize(text_lower)

# lemmatizing words based on contexts and usage of words in sentences
lemmar = WordNetLemmatizer()
def LemTokens(tokens):
    return [lemmar.lemmatize(token) for token in tokens]


# Removing punctuations, which contain '?', '' and '!'
punctuations= dict((ord(punctuation), None) for punctuation in string.
 ↪punctuation)


#filtered_tokens_final=[]
#for i in filtered_tokens:
    #if i not in punctuations:
        #filtered_tokens_final.append(i)

def filtered_token_final(text):
```

```
  text_lower = text.lower().translate(punctuations) # converting text into␣
 ↪lower case and removing punctuations
  tokenized_words=word_tokenize(text_lower) # tokenizing words
  return LemTokens(tokenized_words) # Lemmatizing tokenzied words

tfidf_vectorizer = TfidfVectorizer(stop_words='english' , ngram_range=(1,2),
                            tokenizer = filtered_token_final, min_df=0.05,␣
 ↪max_df=0.85)

# TfidfVectorizer
feature_vect = tfidf_vectorizer.fit_transform(df['sentence'])
```

```
[nltk_data] Downloading package punkt to /root/nltk_data…
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data…
[nltk_data]   Package wordnet is already up-to-date!
/usr/local/lib/python3.9/dist-packages/sklearn/feature_extraction/text.py:528:
UserWarning: The parameter 'token_pattern' will not be used since 'tokenizer' is
not None'
  warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/feature_extraction/text.py:409:
UserWarning: Your stop_words may be inconsistent with your preprocessing.
Tokenizing the stop words generated tokens ['ha', 'le', 'u', 'wa'] not in
stop_words.
  warnings.warn(
```

Before clustering, there is a process of cleaning unrelated words (i.e., punctuations and stopwords), tokenizing and lemmatizing words. Finally, I used TfidfVectorizer to convert a colletion of sorted and filtered words into a matrix of TF-IDF features. TF-IDF is useful method to order and arrange the importance of words.

[296]:
```
from sklearn.cluster import KMeans

# will split data into three clusters.
km_cluster = KMeans(n_clusters=3, max_iter=10000, random_state=0)
km_cluster.fit(feature_vect)

# cluster label and centers
cluster_label = km_cluster.labels_
cluster_centers = km_cluster.cluster_centers_

# Add 'cluster_label' column
df['cluster_label'] = cluster_label
df.sample(5)
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870:
FutureWarning: The default value of `n_init` will change from 10 to 'auto' in
1.4. Set the value of `n_init` explicitly to suppress the warning
```

```
  warnings.warn(
```

[296]:
```
                                                sentence  \
1773  How long will immunity last, after having Covi…
3838  are there reports stating that the vaccine can…
4721  Should I try to get a certain vaccine?  What a…
4734  I thought only weak people who needed a flu va…
2726  I don't think this covid vaccine is very effec…

                                                 label  label_idx  \
1773  How long am I immune from COVID-19 if I had th…        128
3838  What about reports of abnormal periods due to …        119
4721                  Why are there different vaccines?         79
4734  Why do I need the COVID vaccine if I don't get…         80
2726                I'm not sure it is effective enough         16

      word_count  cluster_label
1773           8              0
3838          11              1
4721          17              1
4734          15              2
2726           9              2
```

I split a collection of filtered words into three clusters using k-means.

[297]:
```
df[df['cluster_label']==0].sort_values(by='sentence')
```

[297]:
```
                                                sentence  \
5109  A new study finds that the Pfizer COVID-19 vac…
5001           A permit is needed to vaccinate a child?
812   A volunteer in AstraZeneca's Covid-19 vaccine …
3790  According to no side effects have been recorde…
3026  According to what I heard, Johnson & Johnson's…
…                                                   …
2683  yes i concern that because it have very second…
1137                          yes mutations of the virus
3550                  yes they have very second effects
2494                  yes, vaccines showed side effects
5103  you may get side effects when you get the vaccine

                                                 label  label_idx  \
5109  Will vaccination lead to more dangerous variants?        123
5001  Will my child need my permission to get vaccin…        155
812            Did a volunteer in the Oxford trial die?        125
3790     Vaccine side effects are not getting reported         71
3026  Is the Johnson & Johnson vaccine less effectiv…         62
…                                                   …        …
2683       I'm concerned the vaccine will make me sick.          7
```

```
1137  Do vaccines work against the mutated strains o…         25
3550                      The COVID vaccine is not safe         23
2494  I don't trust vaccines if they're from China o…         33
5103  Will vaccination lead to more dangerous variants?        123

      word_count  cluster_label
5109          15              0
5001           8              0
812           19              0
3790          19              0
3026          17              0
…            …              …
2683          10              0
1137           5              0
3550           6              0
2494           5              0
5103          10              0

[2767 rows x 5 columns]
```

The first cluster is associated with expressions of concerns about side-effects after getting vaccinated. There are 2767 expressions that are associated with side effects of vaccine.

[299]:
```
df[df['cluster_label']==1].sort_values(by='sentence')
```

[299]:
```
                                                sentence  \
4407  A mRNA and viral vector vaccines completely di…
3781  A microchip is small and can be injected in me…
723   A virus is a virus one vaccine should cover al…
479      After I had the vaccine, can I meet in groups?
862   After I've had the vaccine do I need to still …
…                                                    …
341   would I have swollen lymph nodes from the vacc…
3168                  would the vaccine FDA be approved
3157            would the vaccine FDA be approved soon
3162  would the vaccine FDA be approved soon this month
585              younger children can get the vaccine?

                                                  label  label_idx  \
4407  What is the difference between mRNA and viral …          77
3781    They will put a chip/microchip to manipulate me          26
723        Can other vaccines protect me from COVID-19?          31
479          Can I meet in groups after I'm vaccinated?         101
862   Do I need to continue safety measures after ge…           1
…                                                    …        …
341     Can I get swollen lymph nodes from the vaccine?         135
3168                      Is the vaccine FDA approved?         158
```

```
3157                      Is the vaccine FDA approved?        158
3162                      Is the vaccine FDA approved?        158
585                      Can children get the vaccine?         90

      word_count  cluster_label
4407          10              1
3781          14              1
723           12              1
479           10              1
862           21              1
…            …              …
341            9              1
3168           6              1
3157           7              1
3162           9              1
585            6              1

[1755 rows x 5 columns]
```

The second cluster may be associated with important questions and expressions prior to getting vaccinated.

[300]:
```
df[df['cluster_label']==2].sort_values(by='sentence')
```

[300]:
```
                                                 sentence  \
2417    A vaccine surely isn't necessary against Covid?
4961  After being vaccinated, is it possible to test…
845   After getting a vaccine, should I change the w…
490   After getting the Covid vaccination, is it saf…
523   After getting vaccinated, could I go out witho…
…                                                   …
221   why should I get the vaccine if there is no mo…
681   will a newborn be immune to covid if i have be…
4884  will the first dose of the vaccine mean i am s…
525   will the vaccine protect me from the covid virus?
4981  would my son miss classes after being vaccinat…

                                                 label  label_idx  \
2417            I don't think the vaccine is necessary         12
4961   Will I test positive after getting the vaccine?         30
845   Do I need to change my masking and social dist…        113
490         Can I meet in groups after I'm vaccinated?        101
523   Can I still get COVID even after being vaccina…         41
…                                                   …        …
221    COVID-19 is over, why should I get the vaccine?        137
681   Can my newborn become immune to COVID-19 if I'…        136
4884                Will 1 dose of vaccine protect me?         83
```

```
525   Can I still get COVID even after being vaccina…         41
4981  Will my child miss school when they get vaccin…        157

      word_count  cluster_label
2417           7              2
4961          11              2
845           21              2
490           15              2
523           15              2
…            …              …
221           13              2
681           12              2
4884          13              2
525            9              2
4981          10              2

[647 rows x 5 columns]
```

The third cluster can be associated with widespread impacts of COVID-19, along with vaccine on individuals and soicety.

[301]:
```
cluster_centers = km_cluster.cluster_centers_

print('cluster_centers shape :', cluster_centers.shape)
print(cluster_centers)
```

```
cluster_centers shape : (3, 9)
[[8.06899176e-04 1.05717131e-01 8.05435531e-02 1.31825037e-01
  6.96071669e-02 7.78060043e-02 9.60990019e-02 1.35324937e-01
  1.24473589e-01]
 [0.00000000e+00 2.63901512e-03 2.21910887e-03 8.54133454e-04
  1.35095987e-03 1.33631322e-03 8.87352662e-04 0.00000000e+00
  9.95577154e-01]
 [8.18699043e-01 1.19116113e-03 5.74021952e-02 3.05754610e-02
  5.00527056e-02 5.06764991e-02 2.24475382e-02 7.36074682e-02
  2.24755400e-01]]
```

7. Sentiment Analysis

Sentiment analysis is the process of determining whether words are positive, negatie or neutral. By doing so, it may help to understand various expressions and perspectives on COVID-19 vaccine.

[302]:
```
import nltk
from textblob import TextBlob

blob = ' '.join(str(e) for e in filtered_tokens_final)
blob = TextBlob(blob)
blob.sentiment
```

[302]:
```
Sentiment(polarity=0.10892172634584407, subjectivity=0.5135549219716558)
```

Filtered_tokens_final is a collection of filtered and cleaned words, already conducted in previous stage (3. Word Frequency). I first tried to use TextBlob module, which shows sentiment polarity (range [-1.0, 1.0]) and subjectivity of words (range [0.0, 1.0]). The average sentiment score of words seems neutral. I subsequently conduct one more sentiment analysis technique.

[303]:
```
from afinn import Afinn
import numpy as np
import pandas as pd
afn = Afinn()
all_words=nltk.FreqDist(filtered_tokens_final)

scores = [afn.score(i) for i in filtered_tokens_final]
sentiment = ['positive' if score > 0 else 'negative' if score < 0
                            else 'neutral'
                                for score in scores]

mean = np.array(scores)
np.mean(mean)



df1 = pd.DataFrame()
df1['words'] = filtered_tokens_final
df1['scores'] = scores
df1['sentiment'] = sentiment
print("Average of sentiment scores : ", np.mean(mean))
df1.sample(30)
```

```
Average of sentiment scores :  -0.01891637421895061
```

[303]:

	words	scores	sentiment
794	effects	0.0	neutral
8709	b	0.0	neutral
1103	225	0.0	neutral
5412	b	0.0	neutral
6958	cant	0.0	neutral
12628	2763	0.0	neutral
16263	yes	1.0	positive
5066	limit	0.0	neutral
13753	delta	0.0	neutral
8303	told	0.0	neutral
21642	earth	0.0	neutral
12663	vaccine	0.0	neutral
9389	cover	0.0	neutral
11741	want	1.0	positive
8756	1900	0.0	neutral

5754	vaccine	0.0	neutral
12666	effective	2.0	positive
2675	vaccine	0.0	neutral
23248	come	0.0	neutral
21618	vaccine	0.0	neutral
6521	1405	0.0	neutral
3857	827	0.0	neutral
1263	covid-19	0.0	neutral
20363	better	2.0	positive
175	32	0.0	neutral
21005	dont	0.0	neutral
16541	contains	0.0	neutral
21901	approved	2.0	positive
17242	implants	0.0	neutral
13569	2962	0.0	neutral

I used afinn module to double check sentiment of words. The average sentiment score using afinn (range of [-5 and 5]) was -0.01891637421895061, which is slightly negative but almost neutral.

Overall, it is difficult to determine sentiment of primary words used in this expression. Perhaps, many of words, such as covid-19 and vaccine, were difficult to be determined as positive or negative.

8. Discussion and Conclusion

This report conducted text analysis using various techniques, including word count, word frequency, word association, k-mean clustering and sentiment analysis. In the phase of word frequency, 'covid-19', 'vaccine', 'booster', 'side effect' and 'safe' were most frequent words. 'side effect' may reveal social concerns and negative perspective on covid-19 vaccine. On the other hand, 'safe' indicates a positive perspective on vaccine.

Furthermore, word association confirmed that 'side' and 'effect' are one word and 'side effect' were frequently associated with the word 'vaccine' in more than two words assoication. Also, K-mean clustering also confirmed that at least 2767 expressions were related to concerns and side effects of vaccine.

Lastly, it was difficult to measure sentiment of primary words since many of primary words were mostly neutral. However, if advanced sentiment analysis, using deep learning, is performed, there will be more likely that primary words are negative.

Overall, 'side effect' is one of the main words, indicating negative perspectives of COVID-19 vaccine.

9. Reference

Benchmark Data and Evaluation Framework for Intent Discovery Around COVID-19 Vaccine Hesitancy. Shai Gretz, Assaf Toledo, Roni Friedman, Dan Lahav, Rose Weeks, Naor Bar-Zeev, João Sedoc, Pooja Sangha, Yoav Katz, Noam Slonim arXiv, 2022

Chaudhary, M. (2021, December 14). TF-IDF Vectorizer scikit-learn - Mukesh Chaudhary - Medium. Medium. https://medium.com/@cmukesh8688/tf-idf-vectorizer-scikit-learn-dbc0244a911a#:~:text=In%20TfidfVectorizer%20we%20consider%20overall,they%20appear%20in%20the%20docu

GeeksforGeeks. (2020, November 26). Change plot size in Matplotlib Python.

https://www.geeksforgeeks.org/change-plot-size-in-matplotlib-python/

GeeksforGeeks. (2023, February 17). Python Sentiment Analysis using Affin. https://www.geeksforgeeks.org/python-sentiment-analysis-using-affin/