NLP ASSIGNMENT - DIWAKAR SINHA

Qa) Create the bigram DTM and find the cell containing nonzero values

Assumption : assuming there is a typo in the question and the actual q is "find the total no of cells with non 0 values"

In [1]:
```
# Text preprocessing steps:

# 1) convert everything into small or capital letter

# 2) remove special characters

# 3) remove stopwords
```

In [2]:
```
import pandas as pd
import string
import nltk # natural language (Stop words)
import warnings
from sklearn.feature_extraction.text import CountVectorizer
import numpy as np
warnings.filterwarnings('ignore')
```

In [3]:
```
df= pd.read_csv('narendramodi_tweets.csv')
df.head()
```

Out[3]:

	id	retweets_count	favorite_count	created_at	text	lang	retweeted	follow
0	8.260000e+17	1406	4903	31-01-2017 11:00	The President's address wonderfully encapsulat...	en	False	
1	8.260000e+17	907	2877	31-01-2017 10:59	Rashtrapati Ji's address to both Houses of Par...	en	False	
2	8.260000e+17	694	0	31-01-2017 10:52	RT @PMOIndia: Empowering the marginalised. htt...	en	False	
3	8.260000e+17	666	0	31-01-2017 10:52	RT @PMOIndia: Commitment to welfare of farmers...	en	False	
4	8.260000e+17	716	0	31-01-2017 10:52	RT @PMOIndia: Improving the quality of life fo...	en	False	

In [4]:
```
df_text=df[['text']]
df_text.head()
```

Out[4]:

	text
0	The President's address wonderfully encapsulat...
1	Rashtrapati Ji's address to both Houses of Par...
2	RT @PMOIndia: Empowering the marginalised. htt...
3	RT @PMOIndia: Commitment to welfare of farmers...
4	RT @PMOIndia: Improving the quality of life fo...

In [5]:
```
df_text['text'][0]
```

Out[5]:
```
"The President's address wonderfully encapsulated India's strengths, aspirations,
potential &amp; the efforts towards #TransformingIndia."
```

Converting to lower case

In [6]:
```
df_text['t_l']=df_text['text'].str.lower()
df_text['t_l'][0]
```

Out[6]:
```
"the president's address wonderfully encapsulated india's strengths, aspirations,
potential &amp; the efforts towards #transformingindia."
```

Remove special chars

In [7]:
```
df_text['t_l']=df_text['t_l'].str.replace('[^a-z\']',' ')
df_text.head()
```

Out[7]:

	text	t_l
0	The President's address wonderfully encapsulat...	the president's address wonderfully encapsulat...
1	Rashtrapati Ji's address to both Houses of Par...	rashtrapati ji's address to both houses of par...
2	RT @PMOIndia: Empowering the marginalised. htt...	rt pmoindia empowering the marginalised htt...
3	RT @PMOIndia: Commitment to welfare of farmers...	rt pmoindia commitment to welfare of farmers...
4	RT @PMOIndia: Improving the quality of life fo...	rt pmoindia improving the quality of life fo...

Remove stopwords

In [8]:
```
from nltk.corpus import stopwords
stop = stopwords.words('english')
stop
```

Out[8]:
```
['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"]
```

In [9]:
```
#stop.extend(['https','co','app','amp','rt'])
```

In [10]:
```
len(stop)
```

Out[10]:
```
179
```

In [11]:
```
# Removing the stopwords from clean_text column:

# we need to create a user defined function:

# it will split the entire text into a list of words, and then it will do one-to-or
```

```
## stop (stop word list), and return only those words which are not present in stop

## we are joining them back to form a sentence.

def sw(x):
    x = [word for word in x.split() if word not in stop]
    return ' '.join (x)
```

In [12]:
```
df_text['split_text'] = df_text['t_l'].apply(sw)
df_text
```

Out[12]:

	text	t_l	split_text
0	The President's address wonderfully encapsulat...	the president's address wonderfully encapsulat...	president's address wonderfully encapsulated i...
1	Rashtrapati Ji's address to both Houses of Par...	rashtrapati ji's address to both houses of par...	rashtrapati ji's address houses parliament dep...
2	RT @PMOIndia: Empowering the marginalised. htt...	rt pmoindia empowering the marginalised htt...	rt pmoindia empowering marginalised https co w...
3	RT @PMOIndia: Commitment to welfare of farmers...	rt pmoindia commitment to welfare of farmers...	rt pmoindia commitment welfare farmers https c...
4	RT @PMOIndia: Improving the quality of life fo...	rt pmoindia improving the quality of life fo...	rt pmoindia improving quality life poor https ...
...	...	...	...
2982	Effective regulatory mechanism will lead to or...	effective regulatory mechanism will lead to or...	effective regulatory mechanism lead orderly gr...
2983	Passage of Real Estate Bill is great news for ...	passage of real estate bill is great news for ...	passage real estate bill great news aspiring h...
2984	RT @dpradhanbjp: Highlights of Pradhan Mantri ...	rt dpradhanbjp highlights of pradhan mantri ...	rt dpradhanbjp highlights pradhan mantri ujjwa...
2985	Successful launch of IRNSS-1F is an accomplish...	successful launch of irnss f is an accomplish...	successful launch irnss f accomplishment take ...
2986	On CISF's Raising Day, I salute all CISF perso...	on cisf's raising day i salute all cisf perso...	cisf's raising day salute cisf personnel valou...

2987 rows × 3 columns

In [13]:
```
# Step 4: We will build our Count based DTM

from sklearn.feature_extraction.text import CountVectorizer
```

In [14]:
```
#Bigram analysis:
count_vec = CountVectorizer(ngram_range=(2,2))
```

In [15]:
```
# fit the object on split_text column

count_vec.fit(df_text['split_text'])
```

Out[15]:
```
CountVectorizer(ngram_range=(2, 2))
```

In [16]:
```
# For creating the DTM: fit_transform
```

```
X = count_vec.fit_transform(df_text['split_text'])
X
```

Out[16]:
```
<2987x25895 sparse matrix of type '<class 'numpy.int64'>'
        with 35931 stored elements in Compressed Sparse Row format>
```

Ans a1

In [17]:
```
# DTM
DTM_df = pd.DataFrame(X.toarray(), columns = count_vec.get_feature_names())
DTM_df
```

Out[17]:

	aabhar pm	aadhaar earliest	aadhaar enrolment	aadhaar means	aadhaar related	aadhaar uidai	aadhar amp	aajtak diwali	aamir khan	aanandmay gha
0	0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	
2982	0	0	0	0	0	0	0	0	0	
2983	0	0	0	0	0	0	0	0	0	
2984	0	0	0	0	0	0	0	0	0	
2985	0	0	0	0	0	0	0	0	0	
2986	0	0	0	0	0	0	0	0	0	

2987 rows × 25895 columns

In [18]:
```
Word_freq_count = DTM_df.sum()

Word_freq_count
```

Out[18]:
```
aabhar pm            1
aadhaar earliest     1
aadhaar enrolment    1
aadhaar means        1
aadhaar related      1
                    ..
zvhxd xr             1
zwh ut               1
zx kuwpaa            1
zy jjpg              1
zyrkwkqpxj https      1
Length: 25895, dtype: int64
```

Ans a2

In [19]:
```
# Calculate the total number of cells containing nonzero values
total_nonzero_cells = np.sum(DTM_df.values > 0)
```

```
# Print the grand total number of cells containing nonzero values
print(" Total Number of Cells Containing Nonzero Values:", total_nonzero_cells)
```

```
 Total Number of Cells Containing Nonzero Values: 35931
```

In [20]:
```
word_freq_table = pd.DataFrame(Word_freq_count).reset_index().rename(columns={'inde

word_freq_table.sort_values(by='Freq', ascending = False).head(20)
```

Out[20]:

	Words	Freq
10946	https co	1649
19719	rt pmoindia	127
20058	sandesh soldiers	111
17371	pm narendramodi	81
1775	app https	60
13669	let us	59
11445	india amp	51
16029	nm app	48
7449	diwali let	43
20491	sent sandesh	43
24043	us remember	42
21227	soldiers via	42
19097	remember soldiers	41
21181	soldiers diwali	41
24334	via nm	41
23485	transformingindia https	36
2615	best wishes	36
2808	birth anniversary	31
22893	ties https	31
11558	india https	30

Qb) Find most important 20 words using TF-IDF

In [21]:
```
from sklearn.feature_extraction.text import TfidfVectorizer

# Preprocess the text data
df_text['text'] = df_text['text'].str.lower()
df_text['text'] = df_text['text'].str.replace('[^a-z\']', ' ')
df_text['text'] = df_text['text'].apply(sw)
```

In [22]:
```
# Create a TF-IDF vectorizer
tfidf_vectorizer = TfidfVectorizer()
```

In [23]:
```
# Fit and transform the text data to calculate TF-IDF scores
tfidf_matrix = tfidf_vectorizer.fit_transform(df_text['text'])
```

In [24]:
```
# Get feature names (words)
feature_names = tfidf_vectorizer.get_feature_names_out()
```

In [25]:
```
# Calculate TF-IDF scores for each word
tfidf_scores = tfidf_matrix.sum(axis=0).A1
```

In [26]:
```
# Create a DataFrame to store the TF-IDF scores for each word
tfidf_df = pd.DataFrame({'Word': feature_names, 'TF-IDF Score': tfidf_scores})
```

In [27]:
```
# Sort the words by TF-IDF score in descending order
tfidf_df = tfidf_df.sort_values(by='TF-IDF Score', ascending=False)
```

In [28]:
```
# Select the top 20 words with the highest TF-IDF scores
top_20_words = tfidf_df.head(20)
```

Ans b

In [29]:
```
# Display the most important 20 words using TF-IDF
print(top_20_words)
```

```
             Word  TF-IDF Score
3326         https    132.332322
1262            co    128.892842
291            amp    122.217014
6611            rt     74.211479
3551         india     72.221250
5633        people     41.578122
7131      soldiers     34.087737
5761            pm     31.873015
5769     pmoindia     30.611712
5091  narendramodi     29.748060
7596         thank     29.430090
7697         today     29.261043
8468        wishes     26.609120
8060            us     26.481846
3890            ji     26.472003
6723       sandesh     23.957823
7650          ties     23.534214
5903     president     22.681313
811            bjp     22.383312
3003     greetings     21.994737
```

Qc) Calculate the Sentiment score for each tweet using VADER

In [30]:
```
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
```

In [31]:
```
sentiment_scores = []

for tweet in df_text['text']:
    sentiment = analyzer.polarity_scores(tweet)
    sentiment_scores.append(sentiment)
```

In [32]:
```
df_text['Sentiment'] = sentiment_scores
```

Ans c

In [33]:
```
sentiment_scores
```

Out[33]:
```
[{'neg': 0.0, 'neu': 0.602, 'pos': 0.398, 'compound': 0.765},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 0.755, 'pos': 0.245, 'compound': 0.3818},
 {'neg': 0.208, 'neu': 0.604, 'pos': 0.188, 'compound': -0.0772},
 {'neg': 0.268, 'neu': 0.732, 'pos': 0.0, 'compound': -0.5106},
 {'neg': 0.082, 'neu': 0.765, 'pos': 0.153, 'compound': 0.3182},
 {'neg': 0.211, 'neu': 0.408, 'pos': 0.381, 'compound': 0.3612},
 {'neg': 0.146, 'neu': 0.703, 'pos': 0.151, 'compound': 0.0258},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.217, 'neu': 0.783, 'pos': 0.0, 'compound': -0.5719},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.18, 'neu': 0.698, 'pos': 0.122, 'compound': -0.25},
 {'neg': 0.207, 'neu': 0.6, 'pos': 0.193, 'compound': -0.0516},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.143, 'neu': 0.659, 'pos': 0.198, 'compound': 0.25},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 0.728, 'pos': 0.272, 'compound': 0.6249},
 {'neg': 0.0, 'neu': 0.76, 'pos': 0.24, 'compound': 0.6249},
 {'neg': 0.188, 'neu': 0.538, 'pos': 0.274, 'compound': 0.4404},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.12, 'neu': 0.573, 'pos': 0.307, 'compound': 0.5574},
 {'neg': 0.169, 'neu': 0.453, 'pos': 0.379, 'compound': 0.6249},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.254, 'neu': 0.593, 'pos': 0.153, 'compound': -0.296},
 {'neg': 0.0, 'neu': 0.578, 'pos': 0.422, 'compound': 0.8074},
 {'neg': 0.173, 'neu': 0.63, 'pos': 0.197, 'compound': 0.0772},
 {'neg': 0.0, 'neu': 0.721, 'pos': 0.279, 'compound': 0.4767},
 {'neg': 0.0, 'neu': 0.657, 'pos': 0.343, 'compound': 0.743},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 0.632, 'pos': 0.368, 'compound': 0.6597},
 {'neg': 0.0, 'neu': 0.629, 'pos': 0.371, 'compound': 0.6486},
 {'neg': 0.0, 'neu': 0.67, 'pos': 0.33, 'compound': 0.7845},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.317, 'neu': 0.683, 'pos': 0.0, 'compound': -0.6249},
 {'neg': 0.0, 'neu': 0.714, 'pos': 0.286, 'compound': 0.5859},
 {'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.2732},
 {'neg': 0.0, 'neu': 0.743, 'pos': 0.257, 'compound': 0.6369},
 {'neg': 0.285, 'neu': 0.362, 'pos': 0.353, 'compound': 0.128},
 {'neg': 0.0, 'neu': 0.719, 'pos': 0.281, 'compound': 0.5994},
 {'neg': 0.393, 'neu': 0.485, 'pos': 0.121, 'compound': -0.6808},
 {'neg': 0.0, 'neu': 0.851, 'pos': 0.149, 'compound': 0.2732},
 {'neg': 0.0, 'neu': 0.764, 'pos': 0.236, 'compound': 0.5267},
 {'neg': 0.0, 'neu': 0.856, 'pos': 0.144, 'compound': 0.4019},
 {'neg': 0.0, 'neu': 0.507, 'pos': 0.493, 'compound': 0.7003},
 {'neg': 0.0, 'neu': 0.631, 'pos': 0.369, 'compound': 0.4767},
 {'neg': 0.0, 'neu': 0.531, 'pos': 0.469, 'compound': 0.6486},
 {'neg': 0.0, 'neu': 0.631, 'pos': 0.369, 'compound': 0.8225},
 {'neg': 0.0, 'neu': 0.519, 'pos': 0.481, 'compound': 0.5719},
 {'neg': 0.0, 'neu': 0.683, 'pos': 0.317, 'compound': 0.7579},
 {'neg': 0.225, 'neu': 0.775, 'pos': 0.0, 'compound': -0.4939},
 {'neg': 0.372, 'neu': 0.472, 'pos': 0.156, 'compound': -0.6779},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 0.865, 'pos': 0.135, 'compound': 0.3612},
 {'neg': 0.0, 'neu': 0.7, 'pos': 0.3, 'compound': 0.4588},
 {'neg': 0.0, 'neu': 0.449, 'pos': 0.551, 'compound': 0.926},
 {'neg': 0.381, 'neu': 0.619, 'pos': 0.0, 'compound': -0.5709},
 {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
 {'neg': 0.0, 'neu': 0.694, 'pos': 0.306, 'compound': 0.6486},
 {'neg': 0.0, 'neu': 0.613, 'pos': 0.387, 'compound': 0.7351},
 {'neg': 0.0, 'neu': 0.476, 'pos': 0.524, 'compound': 0.836},
 {'neg': 0.0, 'neu': 0.838, 'pos': 0.162, 'compound': 0.4767},
 {'neg': 0.0, 'neu': 0.807, 'pos': 0.193, 'compound': 0.4767},
```

```
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.6, 'pos': 0.4, 'compound': 0.6124},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.645, 'pos': 0.355, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.759, 'pos': 0.241, 'compound': 0.2263},
{'neg': 0.0, 'neu': 0.711, 'pos': 0.289, 'compound': 0.6908},
{'neg': 0.0, 'neu': 0.838, 'pos': 0.162, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.443, 'pos': 0.557, 'compound': 0.8316},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.8126},
{'neg': 0.0, 'neu': 0.526, 'pos': 0.474, 'compound': 0.8934},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.553, 'pos': 0.447, 'compound': 0.7845},
{'neg': 0.181, 'neu': 0.47, 'pos': 0.349, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.562, 'pos': 0.438, 'compound': 0.8979},
{'neg': 0.313, 'neu': 0.579, 'pos': 0.108, 'compound': -0.6486},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.099, 'neu': 0.647, 'pos': 0.254, 'compound': 0.5574},
{'neg': 0.11, 'neu': 0.287, 'pos': 0.603, 'compound': 0.8834},
{'neg': 0.0, 'neu': 0.597, 'pos': 0.403, 'compound': 0.7964},
{'neg': 0.099, 'neu': 0.617, 'pos': 0.284, 'compound': 0.6705},
{'neg': 0.159, 'neu': 0.56, 'pos': 0.28, 'compound': 0.4215},
{'neg': 0.2, 'neu': 0.703, 'pos': 0.097, 'compound': -0.4404},
{'neg': 0.461, 'neu': 0.453, 'pos': 0.086, 'compound': -0.8442},
{'neg': 0.14, 'neu': 0.698, 'pos': 0.163, 'compound': 0.1027},
{'neg': 0.133, 'neu': 0.867, 'pos': 0.0, 'compound': -0.3182},
{'neg': 0.0, 'neu': 0.693, 'pos': 0.307, 'compound': 0.7351},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.592, 'pos': 0.408, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.721, 'pos': 0.279, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.647, 'pos': 0.353, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.57, 'pos': 0.43, 'compound': 0.8074},
{'neg': 0.0, 'neu': 0.528, 'pos': 0.472, 'compound': 0.891},
{'neg': 0.127, 'neu': 0.492, 'pos': 0.381, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.588, 'pos': 0.412, 'compound': 0.8442},
{'neg': 0.181, 'neu': 0.549, 'pos': 0.27, 'compound': 0.4767},
{'neg': 0.179, 'neu': 0.655, 'pos': 0.167, 'compound': -0.0516},
{'neg': 0.0, 'neu': 0.694, 'pos': 0.306, 'compound': 0.765},
{'neg': 0.0, 'neu': 0.294, 'pos': 0.706, 'compound': 0.9595},
{'neg': 0.0, 'neu': 0.789, 'pos': 0.211, 'compound': 0.4939},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.07, 'neu': 0.516, 'pos': 0.413, 'compound': 0.8074},
{'neg': 0.0, 'neu': 0.489, 'pos': 0.511, 'compound': 0.8555},
{'neg': 0.0, 'neu': 0.675, 'pos': 0.325, 'compound': 0.6983},
{'neg': 0.262, 'neu': 0.738, 'pos': 0.0, 'compound': -0.5994},
{'neg': 0.0, 'neu': 0.525, 'pos': 0.475, 'compound': 0.8712},
{'neg': 0.0, 'neu': 0.599, 'pos': 0.401, 'compound': 0.7717},
{'neg': 0.0, 'neu': 0.449, 'pos': 0.551, 'compound': 0.8225},
{'neg': 0.0, 'neu': 0.395, 'pos': 0.605, 'compound': 0.8658},
{'neg': 0.0, 'neu': 0.343, 'pos': 0.657, 'compound': 0.9246},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.155, 'neu': 0.276, 'pos': 0.569, 'compound': 0.7579},
{'neg': 0.0, 'neu': 0.45, 'pos': 0.55, 'compound': 0.9042},
{'neg': 0.0, 'neu': 0.65, 'pos': 0.35, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.571, 'pos': 0.429, 'compound': 0.7579},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.704, 'pos': 0.296, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.596, 'pos': 0.404, 'compound': 0.8316},
{'neg': 0.0, 'neu': 0.78, 'pos': 0.22, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.76, 'pos': 0.24, 'compound': 0.6249},
{'neg': 0.148, 'neu': 0.852, 'pos': 0.0, 'compound': -0.3818},
{'neg': 0.0, 'neu': 0.809, 'pos': 0.191, 'compound': 0.3818},
{'neg': 0.523, 'neu': 0.27, 'pos': 0.207, 'compound': -0.5423},
```

```
{'neg': 0.577, 'neu': 0.34, 'pos': 0.083, 'compound': -0.9201},
{'neg': 0.0, 'neu': 0.5, 'pos': 0.5, 'compound': 0.875},
{'neg': 0.0, 'neu': 0.655, 'pos': 0.345, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.857, 'pos': 0.143, 'compound': 0.25},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.844, 'pos': 0.156, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.1, 'neu': 0.429, 'pos': 0.471, 'compound': 0.8316},
{'neg': 0.208, 'neu': 0.52, 'pos': 0.272, 'compound': 0.0258},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.55, 'pos': 0.45, 'compound': 0.8689},
{'neg': 0.239, 'neu': 0.585, 'pos': 0.176, 'compound': -0.0516},
{'neg': 0.32, 'neu': 0.506, 'pos': 0.174, 'compound': -0.3818},
{'neg': 0.148, 'neu': 0.651, 'pos': 0.201, 'compound': 0.2263},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.152, 'neu': 0.652, 'pos': 0.196, 'compound': 0.2023},
{'neg': 0.0, 'neu': 0.534, 'pos': 0.466, 'compound': 0.7269},
{'neg': 0.0, 'neu': 0.319, 'pos': 0.681, 'compound': 0.9694},
{'neg': 0.0, 'neu': 0.517, 'pos': 0.483, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.476, 'pos': 0.524, 'compound': 0.6705},
{'neg': 0.0, 'neu': 0.192, 'pos': 0.808, 'compound': 0.9118},
{'neg': 0.0, 'neu': 0.577, 'pos': 0.423, 'compound': 0.765},
{'neg': 0.0, 'neu': 0.315, 'pos': 0.685, 'compound': 0.9349},
{'neg': 0.0, 'neu': 0.343, 'pos': 0.657, 'compound': 0.9246},
{'neg': 0.0, 'neu': 0.851, 'pos': 0.149, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.345, 'pos': 0.655, 'compound': 0.9349},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.848, 'pos': 0.152, 'compound': 0.3612},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.838, 'pos': 0.162, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.556, 'pos': 0.444, 'compound': 0.7506},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.828, 'pos': 0.172, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.617, 'pos': 0.383, 'compound': 0.8271},
{'neg': 0.0, 'neu': 0.722, 'pos': 0.278, 'compound': 0.6124},
{'neg': 0.0, 'neu': 0.654, 'pos': 0.346, 'compound': 0.7506},
{'neg': 0.144, 'neu': 0.688, 'pos': 0.169, 'compound': 0.1027},
{'neg': 0.0, 'neu': 0.809, 'pos': 0.191, 'compound': 0.5106},
{'neg': 0.618, 'neu': 0.233, 'pos': 0.15, 'compound': -0.9337},
{'neg': 0.0, 'neu': 0.727, 'pos': 0.273, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.597, 'pos': 0.403, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.568, 'pos': 0.432, 'compound': 0.765},
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.7184},
{'neg': 0.087, 'neu': 0.751, 'pos': 0.162, 'compound': 0.3182},
{'neg': 0.0, 'neu': 0.725, 'pos': 0.275, 'compound': 0.5859},
{'neg': 0.0, 'neu': 0.674, 'pos': 0.326, 'compound': 0.7003},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.728, 'pos': 0.272, 'compound': 0.6249},
{'neg': 0.0, 'neu': 0.592, 'pos': 0.408, 'compound': 0.7351},
{'neg': 0.293, 'neu': 0.426, 'pos': 0.281, 'compound': -0.3117},
{'neg': 0.0, 'neu': 0.542, 'pos': 0.458, 'compound': 0.8519},
{'neg': 0.0, 'neu': 0.372, 'pos': 0.628, 'compound': 0.926},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.446, 'pos': 0.554, 'compound': 0.8807},
{'neg': 0.0, 'neu': 0.741, 'pos': 0.259, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.828, 'pos': 0.172, 'compound': 0.4404},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.867, 'pos': 0.133, 'compound': 0.25},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.786, 'pos': 0.214, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.734, 'pos': 0.266, 'compound': 0.4404},
```

```
{'neg': 0.0, 'neu': 0.567, 'pos': 0.433, 'compound': 0.7269},
{'neg': 0.0, 'neu': 0.758, 'pos': 0.242, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.6124},
{'neg': 0.161, 'neu': 0.603, 'pos': 0.236, 'compound': 0.128},
{'neg': 0.0, 'neu': 0.605, 'pos': 0.395, 'compound': 0.8176},
{'neg': 0.0, 'neu': 0.752, 'pos': 0.248, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.757, 'pos': 0.243, 'compound': 0.5423},
{'neg': 0.0, 'neu': 0.822, 'pos': 0.178, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.569, 'pos': 0.431, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.46, 'pos': 0.54, 'compound': 0.8555},
{'neg': 0.0, 'neu': 0.469, 'pos': 0.531, 'compound': 0.9118},
{'neg': 0.0, 'neu': 0.654, 'pos': 0.346, 'compound': 0.6486},
{'neg': 0.174, 'neu': 0.652, 'pos': 0.174, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.5, 'pos': 0.5, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.818, 'pos': 0.182, 'compound': 0.4404},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.737, 'pos': 0.263, 'compound': 0.6124},
{'neg': 0.0, 'neu': 0.787, 'pos': 0.213, 'compound': 0.4215},
{'neg': 0.137, 'neu': 0.637, 'pos': 0.225, 'compound': 0.2023},
{'neg': 0.0, 'neu': 0.874, 'pos': 0.126, 'compound': 0.3182},
{'neg': 0.0, 'neu': 0.652, 'pos': 0.348, 'compound': 0.4939},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.202, 'neu': 0.798, 'pos': 0.0, 'compound': -0.5106},
{'neg': 0.0, 'neu': 0.861, 'pos': 0.139, 'compound': 0.2263},
{'neg': 0.0, 'neu': 0.643, 'pos': 0.357, 'compound': 0.7269},
{'neg': 0.098, 'neu': 0.352, 'pos': 0.551, 'compound': 0.891},
{'neg': 0.0, 'neu': 0.468, 'pos': 0.532, 'compound': 0.9411},
{'neg': 0.0, 'neu': 0.732, 'pos': 0.268, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.723, 'pos': 0.277, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.654, 'pos': 0.346, 'compound': 0.7506},
{'neg': 0.0, 'neu': 0.844, 'pos': 0.156, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.095, 'neu': 0.476, 'pos': 0.429, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.815, 'pos': 0.185, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.685, 'pos': 0.315, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.631, 'pos': 0.369, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.829, 'pos': 0.171, 'compound': 0.5106},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.82, 'pos': 0.18, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.811, 'pos': 0.189, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.826, 'pos': 0.174, 'compound': 0.2732},
{'neg': 0.593, 'neu': 0.267, 'pos': 0.14, 'compound': -0.7783},
{'neg': 0.358, 'neu': 0.405, 'pos': 0.236, 'compound': -0.2023},
{'neg': 0.0, 'neu': 0.462, 'pos': 0.538, 'compound': 0.8519},
{'neg': 0.0, 'neu': 0.515, 'pos': 0.485, 'compound': 0.5719},
{'neg': 0.205, 'neu': 0.511, 'pos': 0.284, 'compound': 0.1027},
{'neg': 0.0, 'neu': 0.556, 'pos': 0.444, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.722, 'pos': 0.278, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.826, 'pos': 0.174, 'compound': 0.2732},
{'neg': 0.171, 'neu': 0.497, 'pos': 0.331, 'compound': 0.4404},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.145, 'neu': 0.701, 'pos': 0.154, 'compound': 0.0516},
{'neg': 0.119, 'neu': 0.622, 'pos': 0.259, 'compound': 0.4019},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.145, 'neu': 0.674, 'pos': 0.181, 'compound': 0.1779},
{'neg': 0.0, 'neu': 0.655, 'pos': 0.345, 'compound': 0.7845},
```

{'neg': 0.337, 'neu': 0.663, 'pos': 0.0, 'compound': -0.7269},
{'neg': 0.324, 'neu': 0.294, 'pos': 0.382, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.566, 'pos': 0.434, 'compound': 0.8481},
{'neg': 0.153, 'neu': 0.591, 'pos': 0.256, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.929, 'pos': 0.071, 'compound': 0.0772},
{'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.655, 'pos': 0.345, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.535, 'pos': 0.465, 'compound': 0.7717},
{'neg': 0.0, 'neu': 0.469, 'pos': 0.531, 'compound': 0.8834},
{'neg': 0.476, 'neu': 0.524, 'pos': 0.0, 'compound': -0.8442},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.238, 'neu': 0.631, 'pos': 0.131, 'compound': -0.296},
{'neg': 0.0, 'neu': 0.659, 'pos': 0.341, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.781, 'pos': 0.219, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.545, 'pos': 0.455, 'compound': 0.7579},
{'neg': 0.0, 'neu': 0.723, 'pos': 0.277, 'compound': 0.5574},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.575, 'pos': 0.425, 'compound': 0.8625},
{'neg': 0.146, 'neu': 0.601, 'pos': 0.253, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.791, 'pos': 0.209, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.696, 'pos': 0.304, 'compound': 0.5859},
{'neg': 0.0, 'neu': 0.755, 'pos': 0.245, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.825, 'pos': 0.175, 'compound': 0.5267},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.702, 'pos': 0.298, 'compound': 0.6249},
{'neg': 0.0, 'neu': 0.597, 'pos': 0.403, 'compound': 0.7964},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.071, 'neu': 0.305, 'pos': 0.624, 'compound': 0.8979},
{'neg': 0.068, 'neu': 0.631, 'pos': 0.301, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.503, 'pos': 0.497, 'compound': 0.7845},
{'neg': 0.0, 'neu': 0.459, 'pos': 0.541, 'compound': 0.8957},
{'neg': 0.124, 'neu': 0.515, 'pos': 0.361, 'compound': 0.6808},
{'neg': 0.102, 'neu': 0.384, 'pos': 0.514, 'compound': 0.8643},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.746, 'pos': 0.254, 'compound': 0.6249},
{'neg': 0.0, 'neu': 0.58, 'pos': 0.42, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.498, 'pos': 0.502, 'compound': 0.8442},
{'neg': 0.0, 'neu': 0.388, 'pos': 0.612, 'compound': 0.9413},
{'neg': 0.0, 'neu': 0.705, 'pos': 0.295, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.579, 'pos': 0.421, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.556, 'pos': 0.444, 'compound': 0.8807},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.4215},
{'neg': 0.124, 'neu': 0.52, 'pos': 0.356, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.563, 'pos': 0.437, 'compound': 0.8519},
{'neg': 0.577, 'neu': 0.423, 'pos': 0.0, 'compound': -0.872},
{'neg': 0.0, 'neu': 0.706, 'pos': 0.294, 'compound': 0.6124},
{'neg': 0.0, 'neu': 0.78, 'pos': 0.22, 'compound': 0.4767},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.133, 'neu': 0.612, 'pos': 0.255, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.777, 'pos': 0.223, 'compound': 0.3182},
{'neg': 0.0, 'neu': 0.804, 'pos': 0.196, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.548, 'pos': 0.452, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.61, 'pos': 0.39, 'compound': 0.8074},
{'neg': 0.0, 'neu': 0.516, 'pos': 0.484, 'compound': 0.8834},
{'neg': 0.0, 'neu': 0.275, 'pos': 0.725, 'compound': 0.9643},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.42, 'pos': 0.58, 'compound': 0.9485},
{'neg': 0.0, 'neu': 0.859, 'pos': 0.141, 'compound': 0.3182},
{'neg': 0.0, 'neu': 0.694, 'pos': 0.306, 'compound': 0.6486},
{'neg': 0.0, 'neu': 0.44, 'pos': 0.56, 'compound': 0.802},

{'neg': 0.0, 'neu': 0.706, 'pos': 0.294, 'compound': 0.6124},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.776, 'pos': 0.224, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.514, 'pos': 0.486, 'compound': 0.8176},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.789, 'pos': 0.211, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.764, 'pos': 0.236, 'compound': 0.5719},
{'neg': 0.142, 'neu': 0.679, 'pos': 0.179, 'compound': 0.1531},
{'neg': 0.0, 'neu': 0.816, 'pos': 0.184, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.5859},
{'neg': 0.0, 'neu': 0.71, 'pos': 0.29, 'compound': 0.6486},
{'neg': 0.0, 'neu': 0.732, 'pos': 0.268, 'compound': 0.6705},
{'neg': 0.0, 'neu': 0.612, 'pos': 0.388, 'compound': 0.765},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.493, 'pos': 0.507, 'compound': 0.8834},
{'neg': 0.0, 'neu': 0.542, 'pos': 0.458, 'compound': 0.8519},
{'neg': 0.0, 'neu': 0.4, 'pos': 0.6, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.3818},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.629, 'pos': 0.371, 'compound': 0.6486},
{'neg': 0.0, 'neu': 0.719, 'pos': 0.281, 'compound': 0.5994},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.899, 'pos': 0.101, 'compound': 0.2023},
{'neg': 0.0, 'neu': 0.659, 'pos': 0.341, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.722, 'pos': 0.278, 'compound': 0.6124},
{'neg': 0.195, 'neu': 0.629, 'pos': 0.176, 'compound': -0.0772},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.649, 'pos': 0.351, 'compound': 0.4019},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.8016},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.207, 'neu': 0.61, 'pos': 0.183, 'compound': -0.128},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.861, 'pos': 0.139, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.645, 'pos': 0.355, 'compound': 0.765},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.806, 'pos': 0.194, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.519, 'pos': 0.481, 'compound': 0.8807},
{'neg': 0.0, 'neu': 0.505, 'pos': 0.495, 'compound': 0.8689},
{'neg': 0.0, 'neu': 0.843, 'pos': 0.157, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.302, 'pos': 0.698, 'compound': 0.9313},
{'neg': 0.0, 'neu': 0.558, 'pos': 0.442, 'compound': 0.8271},
{'neg': 0.0, 'neu': 0.354, 'pos': 0.646, 'compound': 0.8991},
{'neg': 0.0, 'neu': 0.824, 'pos': 0.176, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.872, 'pos': 0.128, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.34},
{'neg': 0.442, 'neu': 0.558, 'pos': 0.0, 'compound': -0.8658},
{'neg': 0.0, 'neu': 0.728, 'pos': 0.272, 'compound': 0.6808},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.764, 'pos': 0.236, 'compound': 0.5719},
{'neg': 0.0, 'neu': 0.726, 'pos': 0.274, 'compound': 0.5994},
{'neg': 0.335, 'neu': 0.539, 'pos': 0.126, 'compound': -0.5423},
{'neg': 0.0, 'neu': 0.63, 'pos': 0.37, 'compound': 0.5719},
{'neg': 0.0, 'neu': 0.715, 'pos': 0.285, 'compound': 0.5849},
{'neg': 0.354, 'neu': 0.457, 'pos': 0.189, 'compound': -0.4404},
{'neg': 0.0, 'neu': 0.226, 'pos': 0.774, 'compound': 0.9524},
{'neg': 0.0, 'neu': 0.838, 'pos': 0.162, 'compound': 0.4404},

{'neg': 0.193, 'neu': 0.807, 'pos': 0.0, 'compound': -0.4767},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.51, 'pos': 0.49, 'compound': 0.8625},
{'neg': 0.0, 'neu': 0.84, 'pos': 0.16, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.651, 'pos': 0.349, 'compound': 0.7579},
{'neg': 0.107, 'neu': 0.467, 'pos': 0.425, 'compound': 0.7783},
{'neg': 0.0, 'neu': 0.76, 'pos': 0.24, 'compound': 0.6249},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.811, 'pos': 0.189, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.708, 'pos': 0.292, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.34},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.7579},
{'neg': 0.19, 'neu': 0.613, 'pos': 0.196, 'compound': 0.0258},
{'neg': 0.0, 'neu': 0.781, 'pos': 0.219, 'compound': 0.4215},
{'neg': 0.273, 'neu': 0.533, 'pos': 0.193, 'compound': -0.0516},
{'neg': 0.0, 'neu': 0.714, 'pos': 0.286, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.711, 'pos': 0.289, 'compound': 0.6908},
{'neg': 0.0, 'neu': 0.414, 'pos': 0.586, 'compound': 0.8176},
{'neg': 0.0, 'neu': 0.578, 'pos': 0.422, 'compound': 0.743},
{'neg': 0.0, 'neu': 0.567, 'pos': 0.433, 'compound': 0.8658},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.393, 'neu': 0.429, 'pos': 0.179, 'compound': -0.4588},
{'neg': 0.438, 'neu': 0.441, 'pos': 0.121, 'compound': -0.8225},
{'neg': 0.201, 'neu': 0.584, 'pos': 0.214, 'compound': -0.2023},
{'neg': 0.0, 'neu': 0.573, 'pos': 0.427, 'compound': 0.802},
{'neg': 0.0, 'neu': 0.453, 'pos': 0.547, 'compound': 0.9053},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.851, 'pos': 0.149, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.653, 'pos': 0.347, 'compound': 0.8176},
{'neg': 0.0, 'neu': 0.923, 'pos': 0.077, 'compound': 0.128},
{'neg': 0.0, 'neu': 0.784, 'pos': 0.216, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.698, 'pos': 0.302, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.578, 'pos': 0.422, 'compound': 0.743},
{'neg': 0.0, 'neu': 0.699, 'pos': 0.301, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.729, 'pos': 0.271, 'compound': 0.3818},
{'neg': 0.096, 'neu': 0.662, 'pos': 0.243, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.739, 'pos': 0.261, 'compound': 0.5574},
{'neg': 0.15, 'neu': 0.58, 'pos': 0.271, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.591, 'pos': 0.409, 'compound': 0.8402},
{'neg': 0.0, 'neu': 0.581, 'pos': 0.419, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.4215},
{'neg': 0.287, 'neu': 0.594, 'pos': 0.119, 'compound': -0.5267},
{'neg': 0.0, 'neu': 0.789, 'pos': 0.211, 'compound': 0.34},
{'neg': 0.295, 'neu': 0.705, 'pos': 0.0, 'compound': -0.6808},
{'neg': 0.148, 'neu': 0.622, 'pos': 0.23, 'compound': 0.1779},
{'neg': 0.0, 'neu': 0.815, 'pos': 0.185, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.327, 'pos': 0.673, 'compound': 0.9371},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.791, 'pos': 0.209, 'compound': 0.4404},
{'neg': 0.231, 'neu': 0.769, 'pos': 0.0, 'compound': -0.5994},
{'neg': 0.151, 'neu': 0.645, 'pos': 0.204, 'compound': 0.25},
{'neg': 0.0, 'neu': 0.608, 'pos': 0.392, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.324, 'pos': 0.676, 'compound': 0.9493},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.601, 'pos': 0.399, 'compound': 0.8074},
{'neg': 0.0, 'neu': 0.503, 'pos': 0.497, 'compound': 0.836},
{'neg': 0.142, 'neu': 0.309, 'pos': 0.549, 'compound': 0.765},

```
{'neg': 0.348, 'neu': 0.652, 'pos': 0.0, 'compound': -0.4939},
{'neg': 0.177, 'neu': 0.629, 'pos': 0.194, 'compound': 0.0772},
{'neg': 0.231, 'neu': 0.625, 'pos': 0.144, 'compound': -0.3384},
{'neg': 0.0, 'neu': 0.588, 'pos': 0.412, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.476, 'pos': 0.524, 'compound': 0.836},
{'neg': 0.0, 'neu': 0.256, 'pos': 0.744, 'compound': 0.9246},
{'neg': 0.0, 'neu': 0.902, 'pos': 0.098, 'compound': 0.0772},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.811, 'pos': 0.189, 'compound': 0.4215},
{'neg': 0.119, 'neu': 0.383, 'pos': 0.498, 'compound': 0.7845},
{'neg': 0.0, 'neu': 0.538, 'pos': 0.462, 'compound': 0.8834},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.526, 'pos': 0.474, 'compound': 0.8689},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.166, 'neu': 0.518, 'pos': 0.316, 'compound': 0.4404},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.727, 'pos': 0.273, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.776, 'pos': 0.224, 'compound': 0.6369},
{'neg': 0.241, 'neu': 0.359, 'pos': 0.4, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.473, 'pos': 0.527, 'compound': 0.836},
{'neg': 0.0, 'neu': 0.584, 'pos': 0.416, 'compound': 0.7506},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.733, 'pos': 0.267, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.564, 'pos': 0.436, 'compound': 0.8176},
{'neg': 0.0, 'neu': 0.811, 'pos': 0.189, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.828, 'pos': 0.172, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.561, 'pos': 0.439, 'compound': 0.8225},
{'neg': 0.0, 'neu': 0.519, 'pos': 0.481, 'compound': 0.8779},
{'neg': 0.0, 'neu': 0.511, 'pos': 0.489, 'compound': 0.8885},
{'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.677, 'pos': 0.323, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.584, 'pos': 0.416, 'compound': 0.6908},
{'neg': 0.0, 'neu': 0.684, 'pos': 0.316, 'compound': 0.7184},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.839, 'pos': 0.161, 'compound': 0.3612},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.745, 'pos': 0.255, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.685, 'pos': 0.315, 'compound': 0.5574},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.755, 'pos': 0.245, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.41, 'pos': 0.59, 'compound': 0.9538},
{'neg': 0.0, 'neu': 0.811, 'pos': 0.189, 'compound': 0.2732},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.708, 'pos': 0.292, 'compound': 0.7351},
{'neg': 0.128, 'neu': 0.496, 'pos': 0.376, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.4019},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'compound': 0.802},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.823, 'pos': 0.177, 'compound': 0.4215},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.716, 'pos': 0.284, 'compound': 0.7133},
{'neg': 0.0, 'neu': 0.784, 'pos': 0.216, 'compound': 0.296},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.897, 'pos': 0.103, 'compound': 0.128},
{'neg': 0.0, 'neu': 0.682, 'pos': 0.318, 'compound': 0.6808},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
```

```
{'neg': 0.271, 'neu': 0.381, 'pos': 0.348, 'compound': 0.3818},
{'neg': 0.28, 'neu': 0.467, 'pos': 0.252, 'compound': -0.0772},
{'neg': 0.153, 'neu': 0.591, 'pos': 0.256, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.85, 'pos': 0.15, 'compound': 0.4588},
{'neg': 0.108, 'neu': 0.657, 'pos': 0.235, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.45, 'pos': 0.55, 'compound': 0.875},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.709, 'pos': 0.291, 'compound': 0.5719},
{'neg': 0.0, 'neu': 0.332, 'pos': 0.668, 'compound': 0.9201},
{'neg': 0.0, 'neu': 0.556, 'pos': 0.444, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.647, 'pos': 0.353, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.865, 'pos': 0.135, 'compound': 0.1027},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.3612},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.845, 'pos': 0.155, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.718, 'pos': 0.282, 'compound': 0.6705},
{'neg': 0.0, 'neu': 0.828, 'pos': 0.172, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.812, 'pos': 0.188, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.523, 'pos': 0.477, 'compound': 0.802},
{'neg': 0.0, 'neu': 0.474, 'pos': 0.526, 'compound': 0.8777},
{'neg': 0.0, 'neu': 0.51, 'pos': 0.49, 'compound': 0.8625},
{'neg': 0.0, 'neu': 0.649, 'pos': 0.351, 'compound': 0.7579},
{'neg': 0.0, 'neu': 0.78, 'pos': 0.22, 'compound': 0.5423},
{'neg': 0.199, 'neu': 0.243, 'pos': 0.558, 'compound': 0.7506},
{'neg': 0.706, 'neu': 0.294, 'pos': 0.0, 'compound': -0.802},
{'neg': 0.504, 'neu': 0.317, 'pos': 0.179, 'compound': -0.8481},
{'neg': 0.0, 'neu': 0.433, 'pos': 0.567, 'compound': 0.9022},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.747, 'pos': 0.253, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.714, 'pos': 0.286, 'compound': 0.4939},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.147, 'neu': 0.711, 'pos': 0.142, 'compound': -0.0258},
{'neg': 0.509, 'neu': 0.491, 'pos': 0.0, 'compound': -0.8519},
{'neg': 0.0, 'neu': 0.903, 'pos': 0.097, 'compound': 0.128},
{'neg': 0.0, 'neu': 0.745, 'pos': 0.255, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.289, 'pos': 0.711, 'compound': 0.9062},
{'neg': 0.494, 'neu': 0.506, 'pos': 0.0, 'compound': -0.8316},
{'neg': 0.451, 'neu': 0.549, 'pos': 0.0, 'compound': -0.802},
{'neg': 0.0, 'neu': 0.617, 'pos': 0.383, 'compound': 0.8271},
{'neg': 0.0, 'neu': 0.647, 'pos': 0.353, 'compound': 0.7184},
{'neg': 0.144, 'neu': 0.593, 'pos': 0.263, 'compound': 0.4215},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.6705},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.823, 'pos': 0.177, 'compound': 0.4215},
{'neg': 0.198, 'neu': 0.802, 'pos': 0.0, 'compound': -0.5709},
{'neg': 0.0, 'neu': 0.684, 'pos': 0.316, 'compound': 0.5719},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.519, 'pos': 0.481, 'compound': 0.9186},
{'neg': 0.0, 'neu': 0.602, 'pos': 0.398, 'compound': 0.6486},
{'neg': 0.163, 'neu': 0.481, 'pos': 0.356, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.585, 'pos': 0.415, 'compound': 0.7269},
{'neg': 0.0, 'neu': 0.809, 'pos': 0.191, 'compound': 0.5106},
{'neg': 0.08, 'neu': 0.739, 'pos': 0.182, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.531, 'pos': 0.469, 'compound': 0.8271},
{'neg': 0.068, 'neu': 0.932, 'pos': 0.0, 'compound': -0.0258},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.14, 'neu': 0.86, 'pos': 0.0, 'compound': -0.1531},
{'neg': 0.0, 'neu': 0.856, 'pos': 0.144, 'compound': 0.4019},
```

```
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.441, 'pos': 0.559, 'compound': 0.8934},
{'neg': 0.109, 'neu': 0.693, 'pos': 0.198, 'compound': 0.2023},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.759, 'pos': 0.241, 'compound': 0.4215},
{'neg': 0.305, 'neu': 0.481, 'pos': 0.214, 'compound': -0.4019},
{'neg': 0.0, 'neu': 0.733, 'pos': 0.267, 'compound': 0.6249},
{'neg': 0.0, 'neu': 0.567, 'pos': 0.433, 'compound': 0.8126},
{'neg': 0.387, 'neu': 0.613, 'pos': 0.0, 'compound': -0.743},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.492, 'pos': 0.508, 'compound': 0.8519},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.684, 'pos': 0.316, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.3, 'pos': 0.7, 'compound': 0.936},
{'neg': 0.0, 'neu': 0.735, 'pos': 0.265, 'compound': 0.6597},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.682, 'pos': 0.318, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.778, 'pos': 0.222, 'compound': 0.5719},
{'neg': 0.182, 'neu': 0.529, 'pos': 0.288, 'compound': 0.2023},
{'neg': 0.112, 'neu': 0.683, 'pos': 0.205, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.773, 'pos': 0.227, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.585, 'pos': 0.415, 'compound': 0.7783},
{'neg': 0.213, 'neu': 0.242, 'pos': 0.546, 'compound': 0.7845},
{'neg': 0.0, 'neu': 0.706, 'pos': 0.294, 'compound': 0.6124},
{'neg': 0.0, 'neu': 0.632, 'pos': 0.368, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.815, 'pos': 0.185, 'compound': 0.3612},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.855, 'pos': 0.145, 'compound': 0.296},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.205, 'neu': 0.594, 'pos': 0.201, 'compound': -0.0258},
{'neg': 0.0, 'neu': 0.758, 'pos': 0.242, 'compound': 0.6249},
{'neg': 0.0, 'neu': 0.838, 'pos': 0.162, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.872, 'pos': 0.128, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.796, 'pos': 0.204, 'compound': 0.3182},
{'neg': 0.082, 'neu': 0.702, 'pos': 0.216, 'compound': 0.3182},
{'neg': 0.0, 'neu': 0.721, 'pos': 0.279, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.629, 'pos': 0.371, 'compound': 0.7579},
{'neg': 0.0, 'neu': 0.847, 'pos': 0.153, 'compound': 0.4019},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.14, 'neu': 0.86, 'pos': 0.0, 'compound': -0.3818},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.472, 'pos': 0.528, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.562, 'pos': 0.438, 'compound': 0.5994},
{'neg': 0.444, 'neu': 0.556, 'pos': 0.0, 'compound': -0.802},
{'neg': 0.0, 'neu': 0.476, 'pos': 0.524, 'compound': 0.875},
{'neg': 0.126, 'neu': 0.503, 'pos': 0.372, 'compound': 0.5994},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.789, 'pos': 0.211, 'compound': 0.34},
{'neg': 0.0, 'neu': 0.707, 'pos': 0.293, 'compound': 0.6597},
{'neg': 0.0, 'neu': 0.602, 'pos': 0.398, 'compound': 0.6486},
{'neg': 0.144, 'neu': 0.605, 'pos': 0.251, 'compound': 0.3182},
{'neg': 0.0, 'neu': 0.495, 'pos': 0.505, 'compound': 0.8481},
{'neg': 0.0, 'neu': 0.674, 'pos': 0.326, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.546, 'pos': 0.454, 'compound': 0.8074},
{'neg': 0.0, 'neu': 0.828, 'pos': 0.172, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.805, 'pos': 0.195, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.519, 'pos': 0.481, 'compound': 0.9022},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.207, 'neu': 0.469, 'pos': 0.324, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.513, 'pos': 0.487, 'compound': 0.8591},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.169, 'neu': 0.594, 'pos': 0.237, 'compound': 0.128},
```

{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.811, 'pos': 0.189, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.588, 'pos': 0.412, 'compound': 0.7717},
{'neg': 0.159, 'neu': 0.652, 'pos': 0.188, 'compound': 0.1027},
{'neg': 0.101, 'neu': 0.456, 'pos': 0.443, 'compound': 0.8591},
{'neg': 0.233, 'neu': 0.545, 'pos': 0.222, 'compound': -0.0772},
{'neg': 0.0, 'neu': 0.691, 'pos': 0.309, 'compound': 0.7003},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.741, 'pos': 0.259, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.267, 'pos': 0.733, 'compound': 0.875},
{'neg': 0.0, 'neu': 0.725, 'pos': 0.275, 'compound': 0.6486},
{'neg': 0.153, 'neu': 0.656, 'pos': 0.191, 'compound': 0.1779},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.653, 'pos': 0.347, 'compound': 0.7845},
{'neg': 0.0, 'neu': 0.824, 'pos': 0.176, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.212, 'neu': 0.562, 'pos': 0.225, 'compound': 0.0516},
{'neg': 0.0, 'neu': 0.856, 'pos': 0.144, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.783, 'pos': 0.217, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.741, 'pos': 0.259, 'compound': 0.6369},
{'neg': 0.0, 'neu': 0.752, 'pos': 0.248, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.787, 'pos': 0.213, 'compound': 0.4019},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.718, 'pos': 0.282, 'compound': 0.6249},
{'neg': 0.231, 'neu': 0.4, 'pos': 0.369, 'compound': 0.4767},
{'neg': 0.0, 'neu': 0.703, 'pos': 0.297, 'compound': 0.6705},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.386, 'pos': 0.614, 'compound': 0.936},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.34, 'neu': 0.524, 'pos': 0.136, 'compound': -0.5994},
{'neg': 0.341, 'neu': 0.495, 'pos': 0.165, 'compound': -0.4939},
{'neg': 0.203, 'neu': 0.495, 'pos': 0.302, 'compound': 0.25},
{'neg': 0.237, 'neu': 0.382, 'pos': 0.382, 'compound': 0.4215},
{'neg': 0.328, 'neu': 0.529, 'pos': 0.143, 'compound': -0.5423},
{'neg': 0.0, 'neu': 0.758, 'pos': 0.242, 'compound': 0.4939},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.53, 'pos': 0.47, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.185, 'pos': 0.815, 'compound': 0.9559},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.783, 'pos': 0.217, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.8, 'pos': 0.2, 'compound': 0.4588},
{'neg': 0.311, 'neu': 0.583, 'pos': 0.107, 'compound': -0.6369},
{'neg': 0.0, 'neu': 0.837, 'pos': 0.163, 'compound': 0.5106},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.552, 'pos': 0.448, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.875, 'pos': 0.125, 'compound': 0.25},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.239, 'neu': 0.621, 'pos': 0.14, 'compound': -0.1511},
{'neg': 0.0, 'neu': 0.691, 'pos': 0.309, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.824, 'pos': 0.176, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.46, 'pos': 0.54, 'compound': 0.8555},
{'neg': 0.0, 'neu': 0.584, 'pos': 0.416, 'compound': 0.6597},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},

{'neg': 0.0, 'neu': 0.745, 'pos': 0.255, 'compound': 0.5859},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.808, 'pos': 0.192, 'compound': 0.5859},
{'neg': 0.0, 'neu': 0.709, 'pos': 0.291, 'compound': 0.6249},
{'neg': 0.438, 'neu': 0.562, 'pos': 0.0, 'compound': -0.5994},
{'neg': 0.0, 'neu': 0.714, 'pos': 0.286, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.266, 'pos': 0.734, 'compound': 0.93},
{'neg': 0.0, 'neu': 0.529, 'pos': 0.471, 'compound': 0.8689},
{'neg': 0.0, 'neu': 0.516, 'pos': 0.484, 'compound': 0.7579},
{'neg': 0.198, 'neu': 0.676, 'pos': 0.126, 'compound': -0.3818},
{'neg': 0.163, 'neu': 0.526, 'pos': 0.311, 'compound': 0.2023},
{'neg': 0.0, 'neu': 0.802, 'pos': 0.198, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.505, 'pos': 0.495, 'compound': 0.8957},
{'neg': 0.0, 'neu': 0.605, 'pos': 0.395, 'compound': 0.8176},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.718, 'pos': 0.282, 'compound': 0.6705},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.738, 'pos': 0.262, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.822, 'pos': 0.178, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.568, 'pos': 0.432, 'compound': 0.872},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.116, 'neu': 0.741, 'pos': 0.143, 'compound': 0.128},
{'neg': 0.0, 'neu': 0.841, 'pos': 0.159, 'compound': 0.5267},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.692, 'pos': 0.308, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.709, 'pos': 0.291, 'compound': 0.5719},
{'neg': 0.0, 'neu': 0.578, 'pos': 0.422, 'compound': 0.8591},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.823, 'pos': 0.177, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.783, 'pos': 0.217, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.476, 'pos': 0.524, 'compound': 0.8316},
{'neg': 0.0, 'neu': 0.723, 'pos': 0.277, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.537, 'pos': 0.463, 'compound': 0.8591},
{'neg': 0.0, 'neu': 0.705, 'pos': 0.295, 'compound': 0.5574},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.872, 'pos': 0.128, 'compound': 0.2263},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.542, 'pos': 0.458, 'compound': 0.8519},
{'neg': 0.09, 'neu': 0.738, 'pos': 0.172, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.697, 'pos': 0.303, 'compound': 0.7269},
{'neg': 0.0, 'neu': 0.551, 'pos': 0.449, 'compound': 0.8625},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.691, 'pos': 0.309, 'compound': 0.7717},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.58, 'pos': 0.42, 'compound': 0.8271},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.743, 'pos': 0.257, 'compound': 0.5423},
{'neg': 0.0, 'neu': 0.303, 'pos': 0.697, 'compound': 0.9442},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.85, 'pos': 0.15, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.553, 'pos': 0.447, 'compound': 0.8658},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.809, 'pos': 0.191, 'compound': 0.5106},

```
{'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.824, 'pos': 0.176, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.566, 'pos': 0.434, 'compound': 0.8481},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.737, 'pos': 0.263, 'compound': 0.6908},
{'neg': 0.0, 'neu': 0.85, 'pos': 0.15, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.552, 'pos': 0.448, 'compound': 0.7579},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.92, 'pos': 0.08, 'compound': 0.0772},
{'neg': 0.0, 'neu': 0.849, 'pos': 0.151, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.756, 'pos': 0.244, 'compound': 0.6705},
{'neg': 0.0, 'neu': 0.854, 'pos': 0.146, 'compound': 0.34},
{'neg': 0.0, 'neu': 0.598, 'pos': 0.402, 'compound': 0.8555},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.345, 'pos': 0.655, 'compound': 0.9524},
{'neg': 0.0, 'neu': 0.8, 'pos': 0.2, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.876, 'pos': 0.124, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.762, 'pos': 0.238, 'compound': 0.6486},
{'neg': 0.0, 'neu': 0.708, 'pos': 0.292, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.634, 'pos': 0.366, 'compound': 0.7579},
{'neg': 0.0, 'neu': 0.747, 'pos': 0.253, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.48, 'pos': 0.52, 'compound': 0.9186},
{'neg': 0.0, 'neu': 0.718, 'pos': 0.282, 'compound': 0.6249},
{'neg': 0.0, 'neu': 0.563, 'pos': 0.437, 'compound': 0.8519},
{'neg': 0.0, 'neu': 0.847, 'pos': 0.153, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.719, 'pos': 0.281, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.794, 'pos': 0.206, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.729, 'pos': 0.271, 'compound': 0.6369},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.142, 'neu': 0.683, 'pos': 0.174, 'compound': 0.144},
{'neg': 0.157, 'neu': 0.692, 'pos': 0.151, 'compound': -0.0258},
{'neg': 0.171, 'neu': 0.829, 'pos': 0.0, 'compound': -0.4767},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.579, 'pos': 0.421, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.857, 'pos': 0.143, 'compound': 0.3612},
{'neg': 0.0, 'neu': 0.606, 'pos': 0.394, 'compound': 0.7783},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.1, 'neu': 0.9, 'pos': 0.0, 'compound': -0.2732},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
```

```
{'neg': 0.074, 'neu': 0.645, 'pos': 0.281, 'compound': 0.5423},
{'neg': 0.163, 'neu': 0.318, 'pos': 0.519, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.824, 'pos': 0.176, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.66, 'pos': 0.34, 'compound': 0.7717},
{'neg': 0.0, 'neu': 0.498, 'pos': 0.502, 'compound': 0.8777},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.847, 'pos': 0.153, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.856, 'pos': 0.144, 'compound': 0.4019},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.862, 'pos': 0.138, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.847, 'pos': 0.153, 'compound': 0.4404},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.806, 'pos': 0.194, 'compound': 0.4767},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.857, 'pos': 0.143, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.861, 'pos': 0.139, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.82, 'pos': 0.18, 'compound': 0.296},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.663, 'pos': 0.337, 'compound': 0.6808},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.734, 'pos': 0.266, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.611, 'pos': 0.389, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.899, 'pos': 0.101, 'compound': 0.2023},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.6697},
{'neg': 0.0, 'neu': 0.543, 'pos': 0.457, 'compound': 0.8126},
{'neg': 0.0, 'neu': 0.753, 'pos': 0.247, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.498, 'pos': 0.502, 'compound': 0.9022},
{'neg': 0.0, 'neu': 0.565, 'pos': 0.435, 'compound': 0.7717},
{'neg': 0.0, 'neu': 0.7, 'pos': 0.3, 'compound': 0.7184},
{'neg': 0.0, 'neu': 0.749, 'pos': 0.251, 'compound': 0.5719},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.3818},
{'neg': 0.0, 'neu': 0.856, 'pos': 0.144, 'compound': 0.4019},
{'neg': 0.0, 'neu': 0.698, 'pos': 0.302, 'compound': 0.6369},
{'neg': 0.0, 'neu': 0.531, 'pos': 0.469, 'compound': 0.8658},
{'neg': 0.0, 'neu': 0.402, 'pos': 0.598, 'compound': 0.9081},
{'neg': 0.0, 'neu': 0.611, 'pos': 0.389, 'compound': 0.7906},
{'neg': 0.0, 'neu': 0.843, 'pos': 0.157, 'compound': 0.3818},
{'neg': 0.483, 'neu': 0.348, 'pos': 0.17, 'compound': -0.7351},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.739, 'pos': 0.261, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.699, 'pos': 0.301, 'compound': 0.7845},
{'neg': 0.0, 'neu': 0.743, 'pos': 0.257, 'compound': 0.5423},
```

```
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.327, 'neu': 0.396, 'pos': 0.277, 'compound': -0.25},
{'neg': 0.0, 'neu': 0.876, 'pos': 0.124, 'compound': 0.1779},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.438, 'neu': 0.323, 'pos': 0.24, 'compound': -0.6486},
{'neg': 0.276, 'neu': 0.385, 'pos': 0.34, 'compound': 0.25},
{'neg': 0.295, 'neu': 0.546, 'pos': 0.158, 'compound': -0.3612},
{'neg': 0.716, 'neu': 0.284, 'pos': 0.0, 'compound': -0.9491},
{'neg': 0.0, 'neu': 0.519, 'pos': 0.481, 'compound': 0.9169},
{'neg': 0.0, 'neu': 0.893, 'pos': 0.107, 'compound': 0.2023},
{'neg': 0.0, 'neu': 0.637, 'pos': 0.363, 'compound': 0.6908},
{'neg': 0.0, 'neu': 0.761, 'pos': 0.239, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.852, 'pos': 0.148, 'compound': 0.3818},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.741, 'pos': 0.259, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.741, 'pos': 0.259, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.846, 'pos': 0.154, 'compound': 0.25},
{'neg': 0.0, 'neu': 0.503, 'pos': 0.497, 'compound': 0.836},
{'neg': 0.0, 'neu': 0.82, 'pos': 0.18, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.757, 'pos': 0.243, 'compound': 0.5423},
{'neg': 0.0, 'neu': 0.524, 'pos': 0.476, 'compound': 0.872},
{'neg': 0.0, 'neu': 0.524, 'pos': 0.476, 'compound': 0.872},
{'neg': 0.0, 'neu': 0.843, 'pos': 0.157, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.795, 'pos': 0.205, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.674, 'pos': 0.326, 'compound': 0.743},
{'neg': 0.0, 'neu': 0.854, 'pos': 0.146, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.521, 'pos': 0.479, 'compound': 0.8779},
{'neg': 0.0, 'neu': 0.458, 'pos': 0.542, 'compound': 0.9217},
{'neg': 0.0, 'neu': 0.685, 'pos': 0.315, 'compound': 0.7845},
{'neg': 0.0, 'neu': 0.553, 'pos': 0.447, 'compound': 0.8658},
{'neg': 0.0, 'neu': 0.595, 'pos': 0.405, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.676, 'pos': 0.324, 'compound': 0.5859},
{'neg': 0.0, 'neu': 0.546, 'pos': 0.454, 'compound': 0.8074},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.184, 'neu': 0.56, 'pos': 0.256, 'compound': 0.2263},
{'neg': 0.209, 'neu': 0.633, 'pos': 0.158, 'compound': 0.0516},
{'neg': 0.0, 'neu': 0.284, 'pos': 0.716, 'compound': 0.9201},
{'neg': 0.0, 'neu': 0.814, 'pos': 0.186, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.662, 'pos': 0.338, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.867, 'pos': 0.133, 'compound': 0.25},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.263, 'neu': 0.419, 'pos': 0.317, 'compound': 0.2263},
{'neg': 0.713, 'neu': 0.287, 'pos': 0.0, 'compound': -0.9309},
{'neg': 0.0, 'neu': 0.532, 'pos': 0.468, 'compound': 0.8316},
{'neg': 0.0, 'neu': 0.547, 'pos': 0.453, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.92, 'pos': 0.08, 'compound': 0.1027},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.84, 'pos': 0.16, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.694, 'pos': 0.306, 'compound': 0.6486},
{'neg': 0.4, 'neu': 0.367, 'pos': 0.232, 'compound': -0.6258},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.682, 'pos': 0.318, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'compound': 0.4588},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.62, 'pos': 0.38, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.635, 'pos': 0.365, 'compound': 0.5574},
{'neg': 0.0, 'neu': 0.53, 'pos': 0.47, 'compound': 0.7269},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.676, 'pos': 0.324, 'compound': 0.7717},
{'neg': 0.0, 'neu': 0.435, 'pos': 0.565, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.674, 'pos': 0.326, 'compound': 0.7003},
{'neg': 0.0, 'neu': 0.827, 'pos': 0.173, 'compound': 0.3182},
```

```
{'neg': 0.0, 'neu': 0.786, 'pos': 0.214, 'compound': 0.4588},
{'neg': 0.0, 'neu': 0.677, 'pos': 0.323, 'compound': 0.7351},
{'neg': 0.0, 'neu': 0.699, 'pos': 0.301, 'compound': 0.6808},
{'neg': 0.0, 'neu': 0.769, 'pos': 0.231, 'compound': 0.4939},
{'neg': 0.0, 'neu': 0.896, 'pos': 0.104, 'compound': 0.2732},
{'neg': 0.0, 'neu': 0.671, 'pos': 0.329, 'compound': 0.6597},
{'neg': 0.298, 'neu': 0.585, 'pos': 0.117, 'compound': -0.5267},
{'neg': 0.0, 'neu': 0.821, 'pos': 0.179, 'compound': 0.34},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.784, 'pos': 0.216, 'compound': 0.296},
{'neg': 0.0, 'neu': 0.809, 'pos': 0.191, 'compound': 0.3818},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.482, 'pos': 0.518, 'compound': 0.8225},
{'neg': 0.0, 'neu': 0.359, 'pos': 0.641, 'compound': 0.9337},
{'neg': 0.0, 'neu': 0.719, 'pos': 0.281, 'compound': 0.5994},
{'neg': 0.0, 'neu': 0.87, 'pos': 0.13, 'compound': 0.2023},
{'neg': 0.0, 'neu': 0.417, 'pos': 0.583, 'compound': 0.4215},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0},
{'neg': 0.0, 'neu': 0.679, 'pos': 0.321, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.783, 'pos': 0.217, 'compound': 0.5574},
{'neg': 0.148, 'neu': 0.766, 'pos': 0.086, 'compound': -0.3182},
{'neg': 0.0, 'neu': 0.517, 'pos': 0.483, 'compound': 0.4215},
{'neg': 0.0, 'neu': 0.711, 'pos': 0.289, 'compound': 0.6908},
{'neg': 0.0, 'neu': 0.4, 'pos': 0.6, 'compound': 0.9001},
{'neg': 0.0, 'neu': 0.443, 'pos': 0.557, 'compound': 0.8834},
{'neg': 0.0, 'neu': 0.833, 'pos': 0.167, 'compound': 0.5267},
{'neg': 0.0, 'neu': 0.559, 'pos': 0.441, 'compound': 0.7964},
{'neg': 0.0, 'neu': 0.837, 'pos': 0.163, 'compound': 0.5106},
{'neg': 0.0, 'neu': 0.731, 'pos': 0.269, 'compound': 0.7096},
{'neg': 0.0, 'neu': 0.657, 'pos': 0.343, 'compound': 0.7003},
{'neg': 0.167, 'neu': 0.833, 'pos': 0.0, 'compound': -0.5574},
{'neg': 0.0, 'neu': 0.423, 'pos': 0.577, 'compound': 0.872},
{'neg': 0.101, 'neu': 0.485, 'pos': 0.414, 'compound': 0.7964},
{'neg': 0.116, 'neu': 0.455, 'pos': 0.429, 'compound': 0.802},
{'neg': 0.0, 'neu': 0.812, 'pos': 0.188, 'compound': 0.4588},
{'neg': 0.16, 'neu': 0.711, 'pos': 0.129, 'compound': -0.1779},
{'neg': 0.0, 'neu': 0.861, 'pos': 0.139, 'compound': 0.4404},
{'neg': 0.0, 'neu': 0.739, 'pos': 0.261, 'compound': 0.6486},
{'neg': 0.0, 'neu': 0.561, 'pos': 0.439, 'compound': 0.8555},
...]
```

In [34]:
```
# Display the "Sentiment" scores
df_text[['text', 'Sentiment']]
```

Out[34]:

	text	Sentiment
0	president's address wonderfully encapsulated i...	{'neg': 0.0, 'neu': 0.602, 'pos': 0.398, 'comp...
1	rashtrapati ji's address houses parliament dep...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
2	rt pmoindia empowering marginalised https co w...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
3	rt pmoindia commitment welfare farmers https c...	{'neg': 0.0, 'neu': 0.755, 'pos': 0.245, 'comp...
4	rt pmoindia improving quality life poor https ...	{'neg': 0.208, 'neu': 0.604, 'pos': 0.188, 'co...
...	...	...
2982	effective regulatory mechanism lead orderly gr...	{'neg': 0.0, 'neu': 0.476, 'pos': 0.524, 'comp...
2983	passage real estate bill great news aspiring h...	{'neg': 0.0, 'neu': 0.785, 'pos': 0.215, 'comp...
2984	rt dpradhanbjp highlights pradhan mantri ujjwa...	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...
2985	successful launch irnss f accomplishment take ...	{'neg': 0.0, 'neu': 0.64, 'pos': 0.36, 'compou...
2986	cisf's raising day salute cisf personnel valou...	{'neg': 0.0, 'neu': 0.625, 'pos': 0.375, 'comp...

2987 rows × 2 columns

d) Classify the tweets into positive, Negative or Neutral as below and find the total number of tweets in each category:

a. Positive (score > 0.05) b. Negative (score < -0.05) c. Neutral (-0.05 <= score <= 0.05)

In [35]:
```
# Classify tweets into positive, negative, or neutral categories
positive_tweets = df_text[df_text['Sentiment'].apply(lambda x: x['compound'] > 0.05
negative_tweets = df_text[df_text['Sentiment'].apply(lambda x: x['compound'] < -0.0
neutral_tweets = df_text[df_text['Sentiment'].apply(lambda x: -0.05 <= x['compound'
```

Ans d

In [36]:
```
# Find the total number of tweets in each category
total_positive_tweets = len(positive_tweets)
total_negative_tweets = len(negative_tweets)
total_neutral_tweets = len(neutral_tweets)

# Print the results
print("Total Positive Tweets:", total_positive_tweets)
print("Total Negative Tweets:", total_negative_tweets)
print("Total Neutral Tweets:", total_neutral_tweets)
```

```
Total Positive Tweets: 2056
Total Negative Tweets: 204
Total Neutral Tweets: 727
```

In [ ]: