```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import string
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud
from collections import Counter
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from scipy.stats import pearsonr


nltk.download('punkt')
nltk.download('stopwords')
nltk.download('vader_lexicon')
```

```
[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!
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
True
```

```
import pandas as pd

data = [
    {
        'Topic': 'Water',
        'Surah': 'Al-Anbya',
        'Islamic_Data': '"We made every living thing from water, will they not believe?"',
        'Scientific_Data': 'It was only after the discovery of the microscope that it was concluded...'
    },
    {
        'Topic': 'Universe: The Big Bang Theory',
        'Surah': 'Al-Anbya',
        'Islamic_Data': '"Have those who disbelieved not considered that the heavens and the earth were a joined entity, and We separated ther
        'Scientific_Data': 'In 1929, American astronomer Edwin Hubble proposed the Hubbles Law according to which all entities in space are m
    },
    {
        'Topic': 'Universe: The Big Crunch Theory',
        'Surah': 'Al-Anbya',
        'Islamic_Data': '"The Day when We will fold the heaven like the folding of a [written] sheet for the records. As We began the first c
        'Scientific_Data': 'Indeed, the Big Crunch is one of the scenarios predicted by scientists in which the Universe may end...'
    },
    {
        'Topic': 'Embryology',
        'Surah': 'Al-Mu’minun',
        'Islamic_Data': '"We created man from an extract of clay. Then We made him as a drop in a place of settlement, firmly fixed. Then We
        'Scientific_Data': 'Science has only proved this with the help of the latest technology. It is Professor Emeritus Keith L. Moore...'
    },
    {
        'Topic': 'The Sky’s Protection',
        'Surah': 'Al-Anbya',
        'Islamic_Data': '"And We made the sky a protected ceiling, but they, from its signs, are turning away"',
        'Scientific_Data': 'It is a scientific fact that the sky, with all of its gasses, protects the earth and life that is present on it f
    },
    {
        'Topic': 'Iron within Meteorites',
        'Surah': 'Al-Hadid',
        'Islamic_Data': '"We sent down Iron with its great inherent strength and its many benefits for humankind"',
        'Scientific_Data': 'According to M. E. Walrath, iron is not natural to the earth. Scientists state that billions of years ago...'
    },
    {
        'Topic': 'The Meeting of the Seas',
        'Surah': 'Ar-Rahman',
        'Islamic_Data': '"He released the two seas, meeting [side by side], Between them is a barrier [so] neither of them transgresses"',
        'Scientific_Data': 'Science has discovered that in places where two different seas meet, there is a barrier that divides them which h
    },
    {
        'Topic': 'Sun Moving in Orbit',
        'Surah': 'Al-Anbya',
        'Islamic_Data': '"And it is He who created the night and the day and the sun and the moon; all [heavenly bodies] in an orbit are swim
        'Scientific_Data': 'Although it was only a widespread belief in the 20th century amongst the astronomers, today it is a well-establis
    },
```

```
    {
        'Topic': 'Mountains as Stakes',
        'Surah': 'Al-Anbya',
        'Islamic_Data': '"Have We not made the earth a resting place? And the mountains as stakes?"',
        'Scientific_Data': 'In a book by geophysicist Frank Press called Earth (1986), he explains how the mountains are like stakes and are
    },
    {
        'Topic': 'Expansion of the Universe',
        'Surah': 'An-Naba',
        'Islamic_Data': '"And the heaven We constructed with strength, and indeed, We are [its] expander"',
        'Scientific_Data': 'According to the prominent physicist Stephen Hawking in his book A Brief History of Time, “The discovery that the
    },
    {
        'Topic': 'Pain Receptors',
        'Surah': 'Adh-Dhariyat',
        'Islamic_Data': '"For a long time it was thought that the sense of feeling and pain was dependent on the brain. However, it has been
        'Scientific_Data': 'For a long time it was thought that the sense of feeling and pain was dependent on the brain. However, it has bee
    },
    {
        'Topic': 'Internal Waves in the Oceans',
        'Surah': 'An-Nur',
        'Islamic_Data': '"Incredibly, oceanographers have stated that unlike the belief that waves only occur on the surface, there are waves
        'Scientific_Data': 'Incredibly, oceanographers have stated that unlike the belief that waves only occur on the surface, there are wav
    },
    {
        'Topic': 'Frontal Lobe',
        'Surah': 'Al-Alaq',
        'Islamic_Data': '"According to a book titled ‘Essentials of Anatomy and Physiology, it is clearly stated that the forehead or frontal
        'Scientific_Data': 'According to a book titled ‘Essentials of Anatomy and Physiology, it is clearly stated that the forehead or front
    },
    {
        'Topic': 'The Spherical Shape of the Earth',
        'Surah': 'Al-Luqman',
        'Islamic_Data': '"Seest thou not that Allah merges Night into Day And He merges Day into Night?"',
        'Scientific_Data': 'In early times, people believed that the earth is flat. For centuries, men were afraid to venture out too far, le
    },
    {
        'Topic': 'The Water Cycle',
        'Surah': 'Az-Zumar',
        'Islamic_Data': '"Seest thou not that Allah sends down rain from the sky, and leads it through springs in the earth? Then He causes t
        'Scientific_Data': 'In 1580, Bernard Palissy was the first man to describe the present day concept of \'water cycle\'. He described h
    },
     {
        'Topic': 'Frontal Lobe',
        'Surah': 'Al-Alaq',
        'Islamic_Data': '"According to a book titled ‘Essentials of Anatomy and Physiology, it is clearly stated that the forehead or frontal
        'Scientific_Data': 'According to a book titled ‘Essentials of Anatomy and Physiology, it is clearly stated that the forehead or front
    },
     {
        'Topic': 'Barrier between Sweet and Salt Waters',
        'Surah': '',
        'Islamic_Data': '"He has let free the two bodies of flowing water, meeting together: between them is a barrier which they do not tran
        'Scientific_Data': ''
    },
    {
        'Topic': 'Botany',
        'Surah': 'Taha',
        'Islamic_Data': '"And has sent down water from the sky. With it have We produced diverse pairs of plants each separate from the other
        'Scientific_Data': 'Previously humans did not know that plants too have male and female gender distinctions. Botany states that every
    },
    {
        'Topic': "Spider's Web",
        'Surah': 'Al-Ankabut',
        'Islamic_Data': ' "The parable of those who take protectors other than Allah is that of the spider, who builds (to itself) a house; b
        'Scientific_Data': 'Besides giving the physical description of the spiders web as being very flimsy, delicate and weak, the Quran als
    },
     {
        'Topic': 'General Science',
        'Surah': 'Al-Qiyamah',
        'Islamic_Data': '"Does man think that We cannot assemble his bones? Nay, We are able to put together in perfect order the very tips o
        'Scientific_Data': 'In 1880, fingerprinting became the scientific method of identification, after research done by Sir Francis Golt.
    }

    ]

# Create a DataFrame
df = pd.DataFrame(data)
```

```
# Save the DataFrame to a CSV file
csv_file = 'islamic_scientific_data.csv'
df.to_csv(csv_file, index=False)

print(f'Dataset saved to {csv_file}')
```

```
Dataset saved to islamic_scientific_data.csv
```

Got data from :

- https://themuslimvibe.com/faith-islam/13-scientific-facts-in-the-holy-quran
- https://sunnahonline.com/library/the-majestic-quran/430-quran-and-modern-science-compatible-or-incompatible-the

```
data = pd.read_csv("/content/islamic_scientific_data.csv")
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 4 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Topic            20 non-null     object
 1   Surah            19 non-null     object
 2   Islamic_Data     20 non-null     object
 3   Scientific_Data  19 non-null     object
dtypes: object(4)
memory usage: 768.0+ bytes
```

```
data.isna().sum()
```

```
Topic              0
Surah              1
Islamic_Data       0
Scientific_Data    1
dtype: int64
```

```
data = data.dropna()
```

```
data.isna().sum()
```

```
Topic              0
Surah              0
Islamic_Data       0
Scientific_Data    0
dtype: int64
```

```
#check for duplicates
duplicates = data[data.duplicated(keep=False)]
if not duplicates.empty:
  print("Duplicate rows found:")
  print(duplicates)
else:
  print("No duplicates found.")
```

```
Duplicate rows found:
           Topic    Surah                                        Islamic_Data  \
12  Frontal Lobe  Al-Alaq  "According to a book titled ‘Essentials of Ana...
15  Frontal Lobe  Al-Alaq  "According to a book titled ‘Essentials of Ana...

                                    Scientific_Data
12  According to a book titled ‘Essentials of Anat...
15  According to a book titled ‘Essentials of Anat...
```

```
data = data.drop_duplicates()
```

```
# Define the data cleaning function
def clean_text(text):
  text = text.lower()
  text = text.translate(str.maketrans('', '', string.punctuation))
  tokens = word_tokenize(text)
  stop_words = set(stopwords.words('english'))
```

```
  filtered_tokens = [word for word in tokens if word not in stop_words]
  cleaned_text = ' '.join(filtered_tokens)
  return cleaned_text


#cleaning islamic data
data['cleaned_islamic_data'] = data['Islamic_Data'].apply(clean_text)
print(data['cleaned_islamic_data'])
```

```
0                made every living thing water believe
1     disbelieved considered heavens earth joined en...
2     day fold heaven like folding written sheet rec...
3     created man extract clay made drop place settl...
4         made sky protected ceiling signs turning away
5     sent iron great inherent strength many benefit...
6     released two seas meeting side side barrier ne...
7     created night day sun moon heavenly bodies orb...
8             made earth resting place mountains stakes
9           heaven constructed strength indeed expander
10    long time thought sense feeling pain dependent...
11    incredibly oceanographers stated unlike belief...
12    according book titled ‘ essentials anatomy phy...
13    seest thou allah merges night day merges day n...
14    seest thou allah sends rain sky leads springs ...
17    sent water sky produced diverse pairs plants s...
18    parable take protectors allah spider builds ho...
19    man think assemble bones nay able put together...
Name: cleaned_islamic_data, dtype: object
```

```
#cleaning islamic data
data['cleaned_scientific_data'] = data['Scientific_Data'].apply(clean_text)
print(data['cleaned_scientific_data'])
```

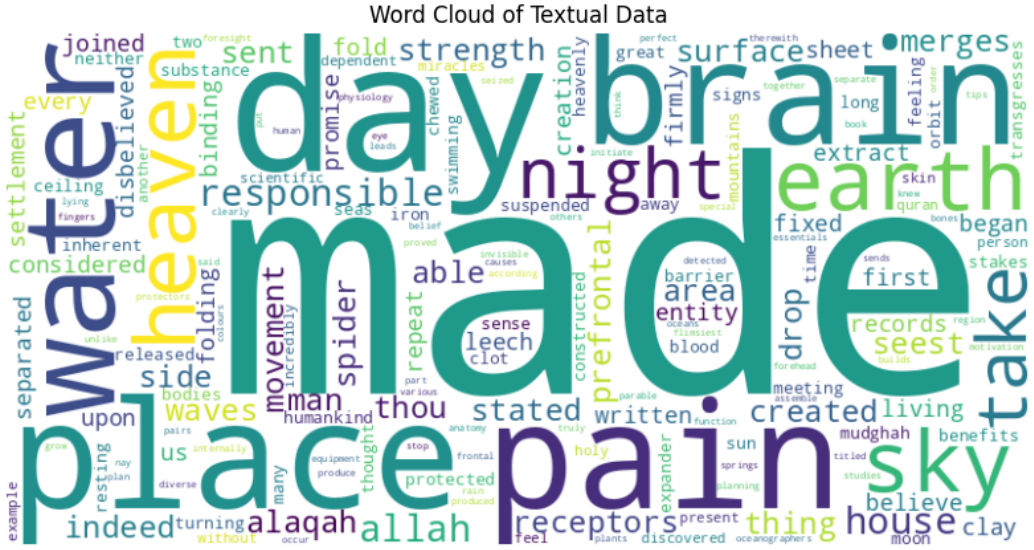
```
0                       discovery microscope concluded
1     1929 american astronomer edwin hubble proposed...
2     indeed big crunch one scenarios predicted scie...
3     science proved help latest technology professo...
4     scientific fact sky gasses protects earth life...
5     according e walrath iron natural earth scienti...
6     science discovered places two different seas m...
7     although widespread belief 20th century amongs...
8     book geophysicist frank press called earth 198...
9     according prominent physicist stephen hawking ...
10    long time thought sense feeling pain dependent...
11    incredibly oceanographers stated unlike belief...
12    according book titled ‘ essentials anatomy phy...
13    early times people believed earth flat centuri...
14    1580 bernard palissy first man describe presen...
17    previously humans know plants male female gend...
18    besides giving physical description spiders we...
19    1880 fingerprinting became scientific method i...
Name: cleaned_scientific_data, dtype: object
```

```
data.info()
```

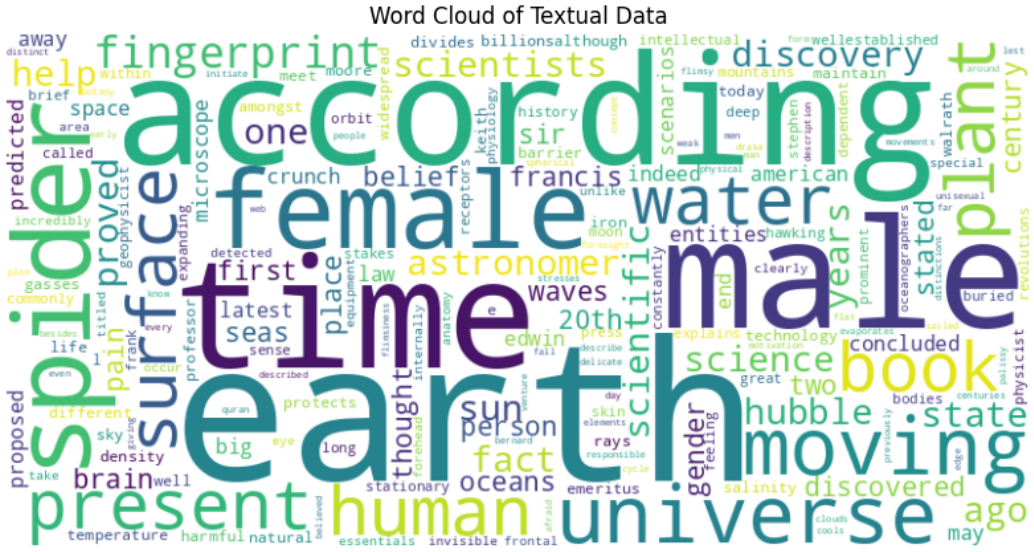
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 18 entries, 0 to 19
Data columns (total 6 columns):
 #   Column                  Non-Null Count  Dtype 
---  ------                  --------------  ----- 
 0   Topic                   18 non-null     object
 1   Surah                   18 non-null     object
 2   Islamic_Data            18 non-null     object
 3   Scientific_Data         18 non-null     object
 4   cleaned_islamic_data    18 non-null     object
 5   cleaned_scientific_data  18 non-null     object
dtypes: object(6)
memory usage: 1008.0+ bytes
```

```
#word cloud of islamic data
text_data = data['cleaned_islamic_data']

# Generate a WordCloud
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(' '.join(text_data))

# Display the WordCloud
plt.figure(figsize=(10, 6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
```

```
plt.title('Word Cloud of Textual Data')
plt.show()
```

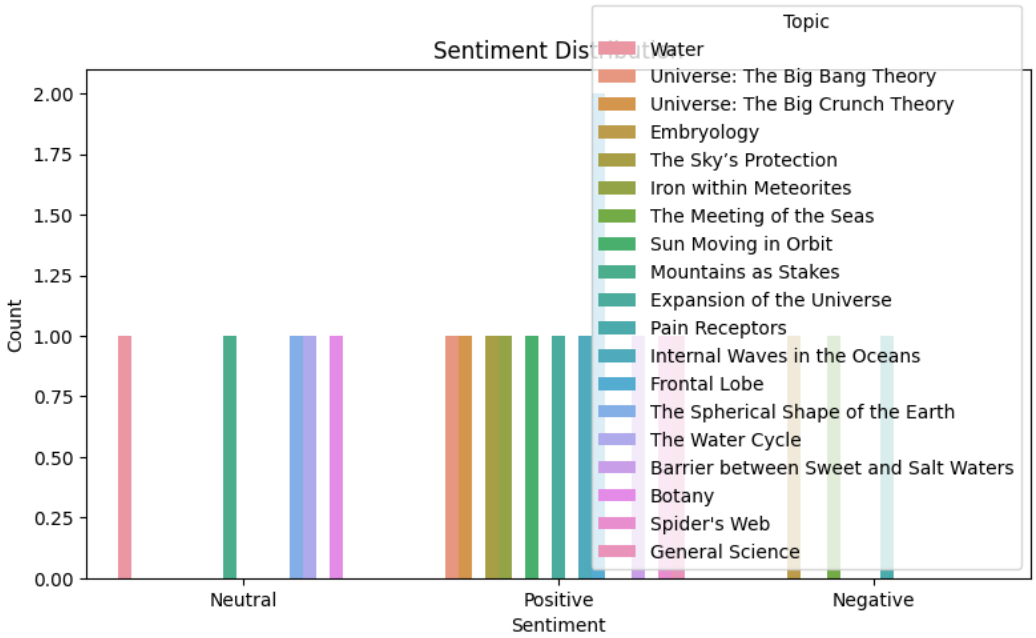



```
#word cloud of scientific data
text_data = data['cleaned_scientific_data']

# Generate a WordCloud
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(' '.join(text_data))

# Display the WordCloud
plt.figure(figsize=(10, 6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Word Cloud of Textual Data')
plt.show()
```

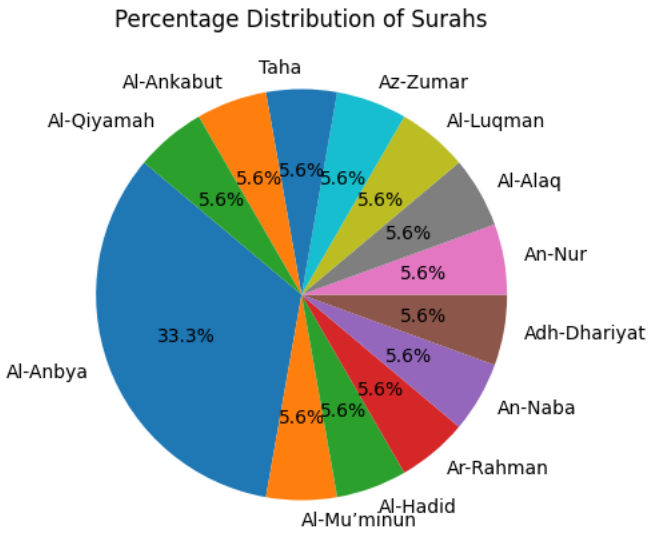



```
# Initialize the sentiment analyzer
sia = SentimentIntensityAnalyzer()

# Calculate sentiment scores for each row
df['Sentiment_Scores'] = df['Islamic_Data'].apply(lambda x: sia.polarity_scores(x)['compound'])

# Classify sentiments based on scores
def classify_sentiment(score):
    if score > 0.05:
        return 'Positive'
    elif score < -0.05:
        return 'Negative'
```

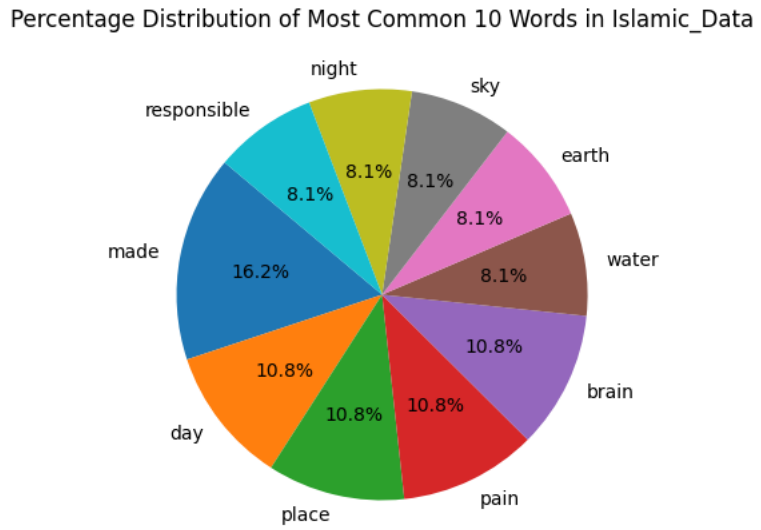
```
    else:
        return 'Neutral'

# Apply sentiment classification
df['Sentiment'] = df['Sentiment_Scores'].apply(classify_sentiment)

# Display sentiment distribution
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x='Sentiment', hue='Topic')
plt.title('Sentiment Distribution')
plt.xlabel('Sentiment')
plt.ylabel('Count')
plt.tight_layout()
plt.show()
```




```
# Calculate the percentage distribution of surahs
surah_counts = data['Surah'].value_counts()
total_surahs = len(surah_counts)
surah_percentages = (surah_counts / total_surahs) * 100

# Create a pie chart
plt.figure(figsize=(5, 5))
plt.pie(surah_percentages, labels=surah_percentages.index, autopct='%1.1f%%', startangle=140)
plt.title('Percentage Distribution of Surahs')
plt.show()
```

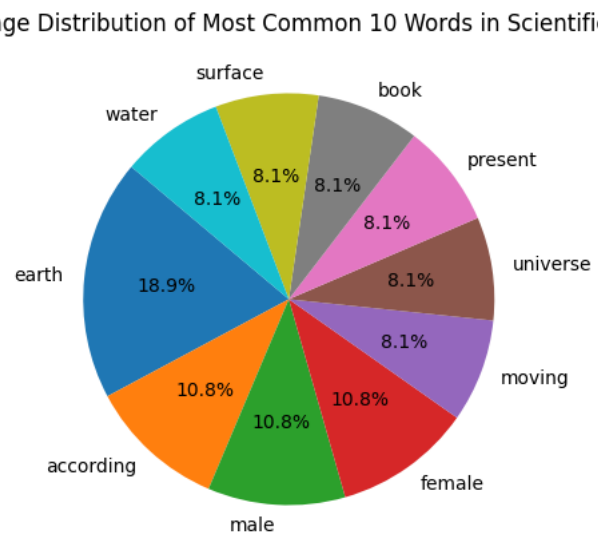


```
#Percentage Distribution of Most Common 10 Words in Islamic_Data
islamic_text = ' '.join(data['cleaned_islamic_data'])
# Tokenize the text
tokens = word_tokenize(islamic_text)

# Remove stopwords
stop_words = set(stopwords.words('english'))
filtered_tokens = [word for word in tokens if word.lower() not in stop_words]

# Count the frequency of each word
word_counts = Counter(filtered_tokens)

# Get the most common 10 words and their counts
most_common_words = word_counts.most_common(10)

# Calculate the total number of words
total_words = sum(word_counts.values())

# Calculate the percentage distribution of the most common words
word_percentages = [(word, (count / total_words) * 100) for word, count in most_common_words]

# Create a pie chart
plt.figure(figsize=(5, 5))
plt.pie([percentage for word, percentage in word_percentages], labels=[word for word, percentage in word_percentages],
        autopct='%1.1f%%', startangle=140)
plt.title('Percentage Distribution of Most Common 10 Words in Islamic_Data')
plt.show()
```




```
#Percentage Distribution of Most Common 10 Words in Scientic_Data
scientific_text = ' '.join(data['cleaned_scientific_data'])
# Tokenize the text
tokens = word_tokenize(scientific_text)

# Remove stopwords
filtered_tokens = [word for word in tokens if word.lower() not in stop_words]
# Remove quotation marks from words
filtered_tokens = [word.replace('"', '') for word in filtered_tokens]

# Count the frequency of each word
word_counts = Counter(filtered_tokens)

# Get the most common 10 words and their counts
most_common_words = word_counts.most_common(10)

# Calculate the total number of words
total_words = sum(word_counts.values())

# Calculate the percentage distribution of the most common words
word_percentages = [(word, (count / total_words) * 100) for word, count in most_common_words]

# Create a pie chart
plt.figure(figsize=(5, 5))
plt.pie([percentage for word, percentage in word_percentages], labels=[word for word, percentage in word_percentages],
```

```
        autopct='%1.1f%%', startangle=140)
plt.title('Percentage Distribution of Most Common 10 Words in Scientific_Data')
plt.show()
```

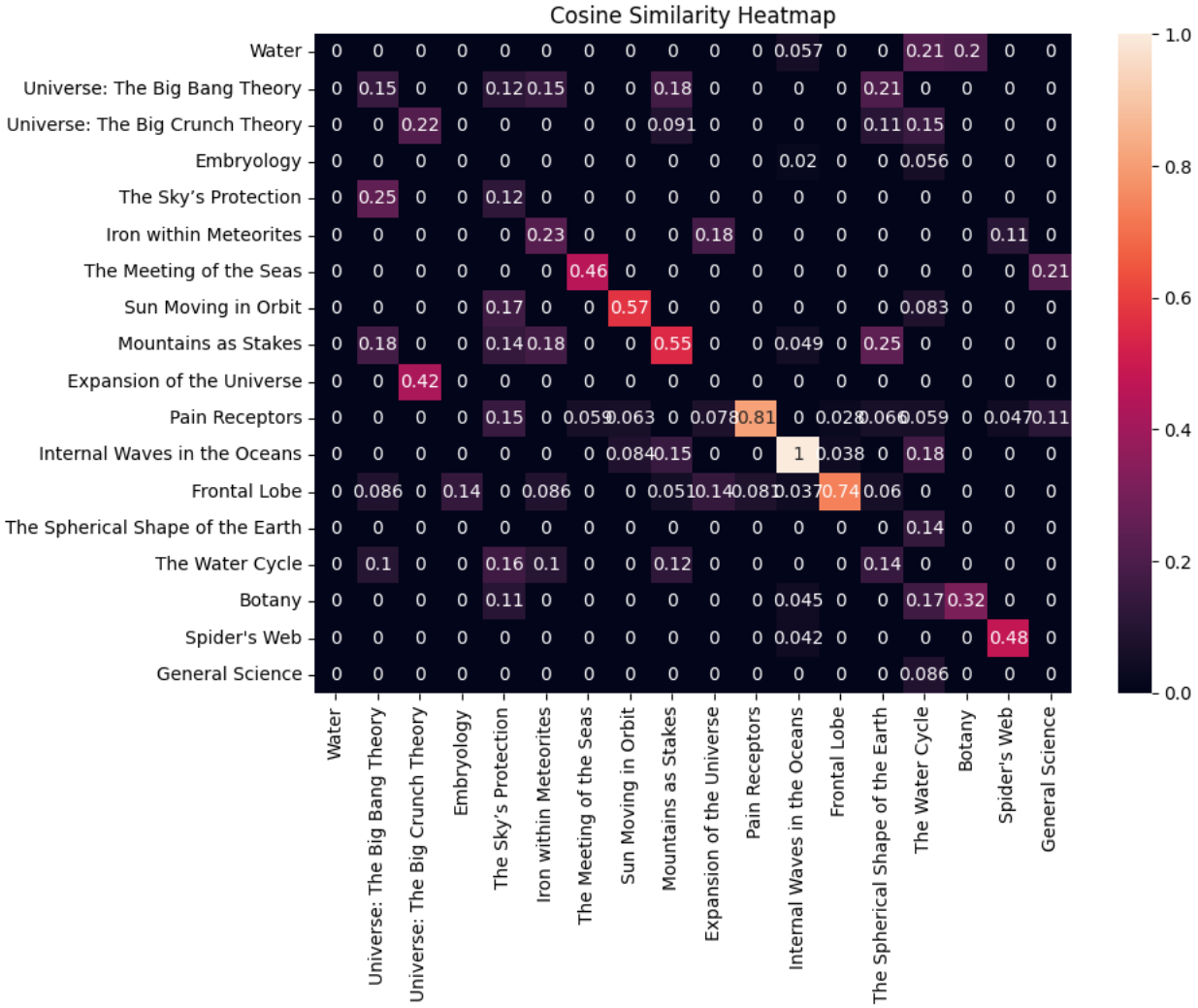



```
# Create a TF-IDF vectorizer
tfidf_vectorizer = TfidfVectorizer()

# Fit and transform the Islamic data
tfidf_matrix_islamic = tfidf_vectorizer.fit_transform(data['cleaned_islamic_data'])

# Fit and transform the scientific data
tfidf_matrix_scientific = tfidf_vectorizer.transform(data['cleaned_scientific_data'])

# Calculate cosine similarity between the TF-IDF matrices
cosine_sim = cosine_similarity(tfidf_matrix_islamic, tfidf_matrix_scientific)

# Print the cosine similarity matrix
print("Cosine Similarity Matrix:")
print(cosine_sim)
```

```
Cosine Similarity Matrix:
[[0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.05748098
  0.         0.         0.21350995 0.20201031 0.         0.        ]
 [0.         0.14868477 0.         0.         0.11688329 0.14868477
  0.         0.         0.17600594 0.         0.         0.
  0.         0.20564937 0.         0.         0.         0.        ]
 [0.         0.         0.21779518 0.         0.         0.
  0.         0.         0.09100248 0.         0.         0.
  0.         0.10632938 0.15379851 0.         0.         0.        ]
 [0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.02023343
  0.         0.         0.05581826 0.         0.         0.        ]
 [0.         0.24947373 0.         0.         0.12140849 0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.         0.22514573
  0.         0.         0.         0.18218268 0.         0.
  0.         0.         0.         0.         0.10986029 0.        ]
 [0.         0.         0.         0.         0.         0.
  0.45584231 0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.21320072]
 [0.         0.         0.         0.         0.17013676 0.
  0.         0.57196565 0.         0.         0.         0.
  0.         0.         0.08277804 0.         0.         0.        ]
 [0.         0.17738192 0.         0.         0.13944255 0.17738192
  0.         0.         0.54915757 0.         0.         0.04916251
  0.         0.24534106 0.         0.         0.         0.        ]
 [0.         0.         0.42218122 0.         0.         0.
  0.         0.         0.         0.         0.         0.
  0.         0.         0.         0.         0.         0.        ]
 [0.         0.         0.         0.         0.15074737 0.
  0.05864867 0.0633478  0.         0.07758489 0.81152185 0.
  0.02839181 0.06630781 0.05923774 0.         0.04678545 0.10972161]
 [0.         0.         0.         0.         0.         0.
  0.         0.08391734 0.15035394 0.         0.         1.
```

```
 0.03761086 0.         0.17563258 0.         0.         0.        ]
[0.         0.08637513 0.         0.13978548 0.         0.08637513
 0.         0.         0.05112338 0.13978548 0.08091438 0.03683389
 0.7366104  0.05973372 0.         0.         0.         0.        ]
[0.         0.         0.         0.         0.         0.
 0.         0.         0.         0.         0.         0.
 0.         0.         0.13823323 0.         0.         0.        ]
[0.         0.10454016 0.         0.         0.16436113 0.10454016
 0.         0.         0.12374966 0.         0.         0.
 0.         0.14459193 0.         0.         0.         0.        ]
[0.         0.         0.         0.         0.10628941 0.
 0.         0.         0.         0.         0.         0.04497848
 0.         0.         0.16707009 0.3161434  0.         0.        ]
[0.         0.         0.         0.         0.         0.
 0.         0.         0.         0.         0.         0.04152163
 0.         0.         0.         0.         0.47697643 0.        ]
[0.         0.         0.         0.         0.         0.
 0.         0.         0.         0.         0.         0.
 0.         0.         0.08612692 0.         0.         0.        ]]
```

```
# Create a heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(cosine_sim, annot=True, xticklabels=data['Topic'], yticklabels=data['Topic'])
plt.title('Cosine Similarity Heatmap')
plt.xticks(rotation=90)
plt.yticks(rotation=0)
plt.tight_layout()
plt.show()
```



Findings: There are some entries that show relatively high similarity between their Islamic and scientific data, telling that certain concepts are discussed in a similar manner in both contexts. There are entries that have lower similarity scores, indicating that they focus on different concepts in Islamic and scientific contexts. Diagonal values tell you how much overlap or similarity exists within the data of each entry itself. Moreover mostly Topics are from surrah Al-Anbya