After Collecting the data :

EDA

```
[8] #understanding the dataset and it's shape

    print('data has ',df.shape[0],'rows and ',df.shape[1],'columns')
```

```
data has  6000 rows and  7 columns
```

```
[9] df.columns
```

```
Index(['Unnamed: 0', 'id', 'title', 'date', 'author', 'story', 'topic'], dtype='object')
```

drop some cloumns that no need for
there is no need for the date as i want by the tittle and the story to cllsify the topic and also no need to the author so that will be unuseful
i didn't remove the id cause i will need it later

```
[10] df.drop(columns = {"Unnamed: 0","date","author"},inplace = True)
```

Understanding data info and checking if there is null values

```
#checking the dataType of each columns
df.info()
```

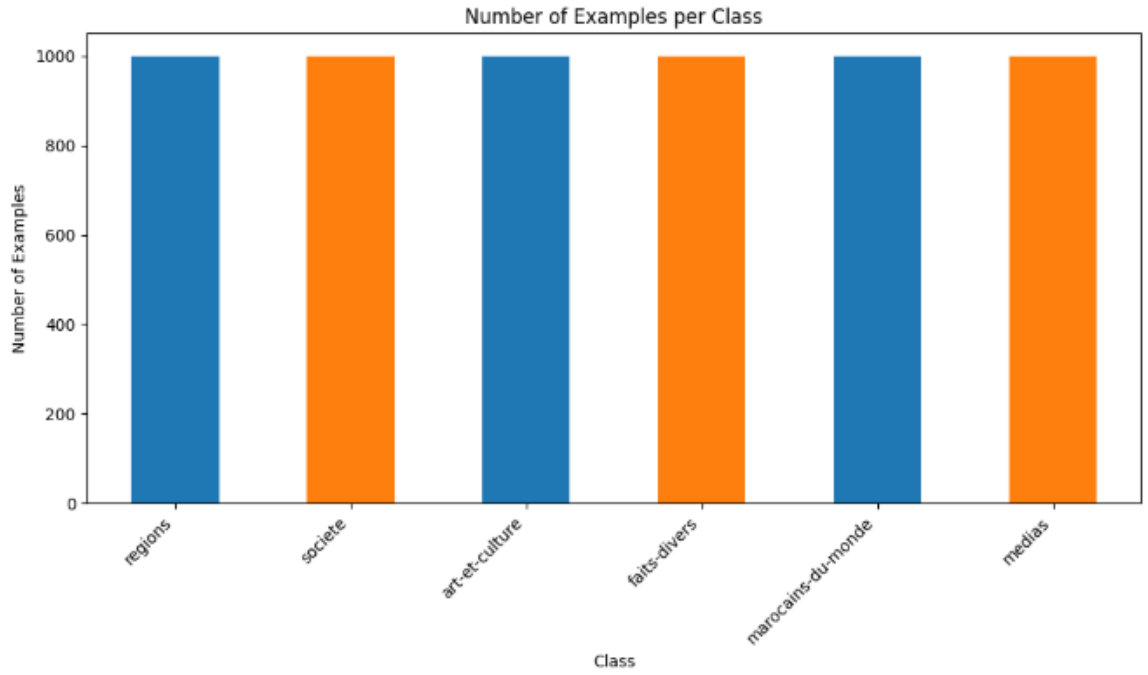
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6000 entries, 0 to 5999
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   id      6000 non-null   object
 1   title   6000 non-null   object
 2   story   6000 non-null   object
 3   topic   6000 non-null   object
dtypes: object(4)
memory usage: 187.6+ KB
```

We Can Note that all the colums are "Categorical" Which must be considered

```
[12] #Checking for missing values
     df.isnull().sum()
```

```
id       0
title    0
story    0
topic    0
dtype: int64
```

number of examples per class

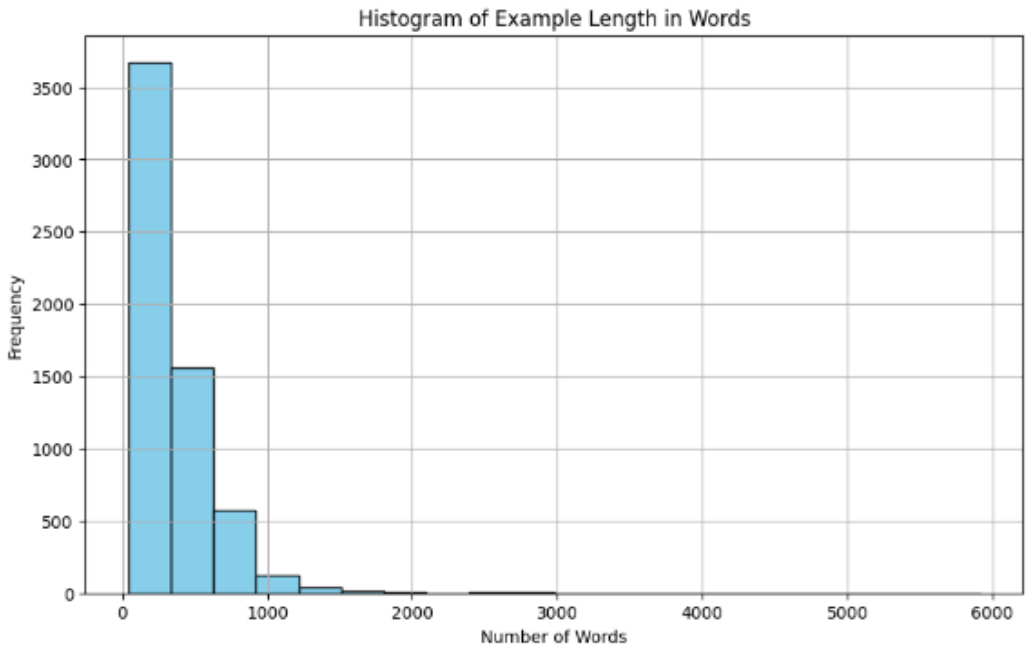


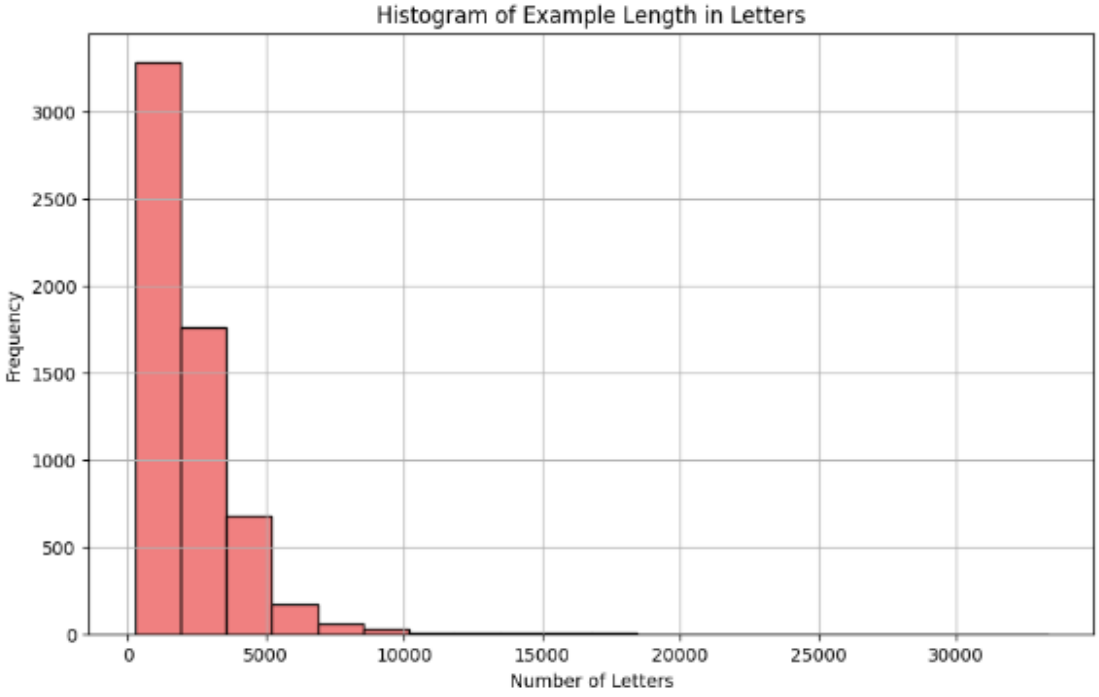
Understand the Length of Examples(word , letters)

```
[15] #Length of Examples
     example_word_length =df['story'].apply(lambda x: len(x.split()))
     example_letter_length = df['story'].apply(lambda x: len(x))
```

```
[16] # Combine the lengths into a new DataFrame
     length_df = pd.DataFrame({'Example_ID': df.id, 'Words_Length': example_word_length, 'Letters_Length': example_letter_length})

     # Print the first few rows of the length DataFrame
     length_df.head()
```

	Example_ID	Words_Length	Letters_Length
0	2390721404e111eb8234646e69d991ea	354	2086
1	252ae82804e111eba71c646e69d991ea	79	488
2	2658ba0a04e111eb8e5f646e69d991ea	261	1629
3	2768a33a04e111eb9c88646e69d991ea	152	946
4	2882027604e111eb8b80646e69d991ea	547	3382


Histogram of Example Length in Words
Frequency
Number of Words
0
500
1000
1500
2000
2500
3000
3500
0
1000
2000
3000
4000
5000
6000


Histogram of Example Length in Letters
Frequency
Number of Letters
0
500
1000
1500
2000
2500
3000
0
5000
10000
15000
20000
25000
30000

top frequent n-grams generally and per class

```
[23] df['title'] = df['title'].apply(stemming)
     df['story'] = df['story'].apply(stemming)
```

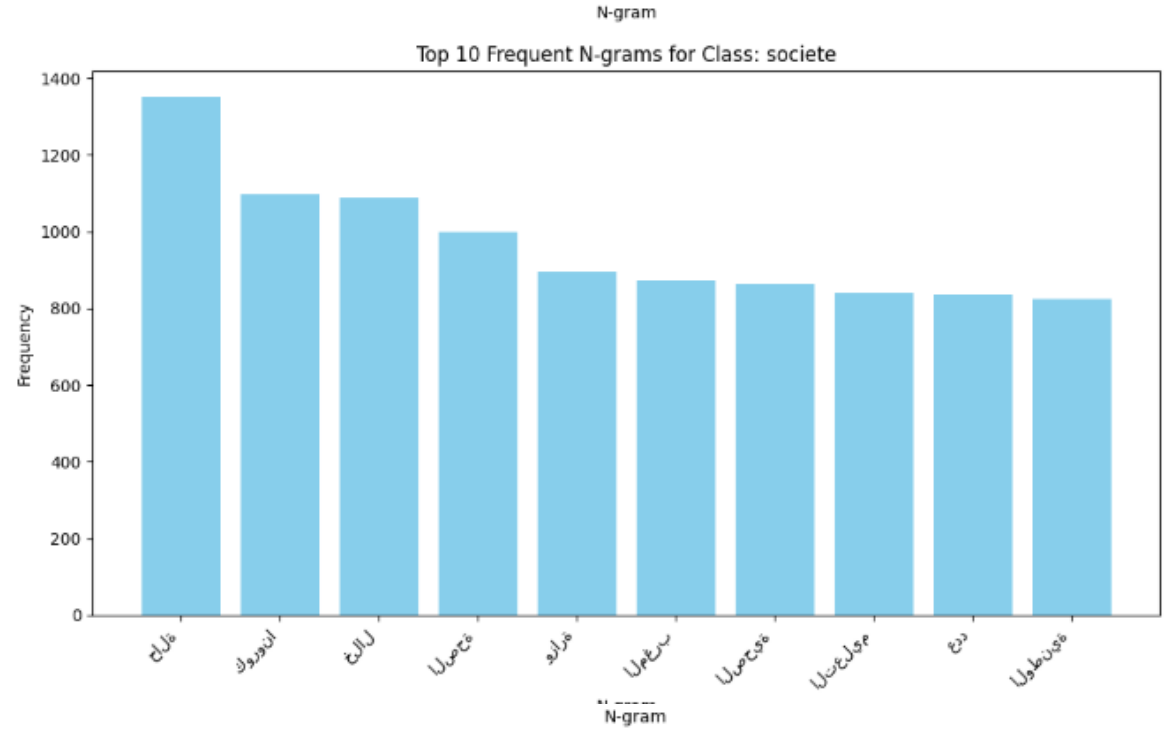
```
ngram_range = (1, 1)
vectorizer = CountVectorizer(ngram_range=ngram_range, stop_words=None)

# Fit and transform the preprocessed text data to get the n-gram frequencies
ngrams_matrix = vectorizer.fit_transform(df['story'])

# Create a DataFrame to store the n-gram frequencies
ngrams_df = pd.DataFrame(ngrams_matrix.toarray(), columns=vectorizer.get_feature_names_out())

# Calculate the sum of each n-gram frequency across all examples
general_ngram_frequencies = ngrams_df.sum().sort_values(ascending=False)
```

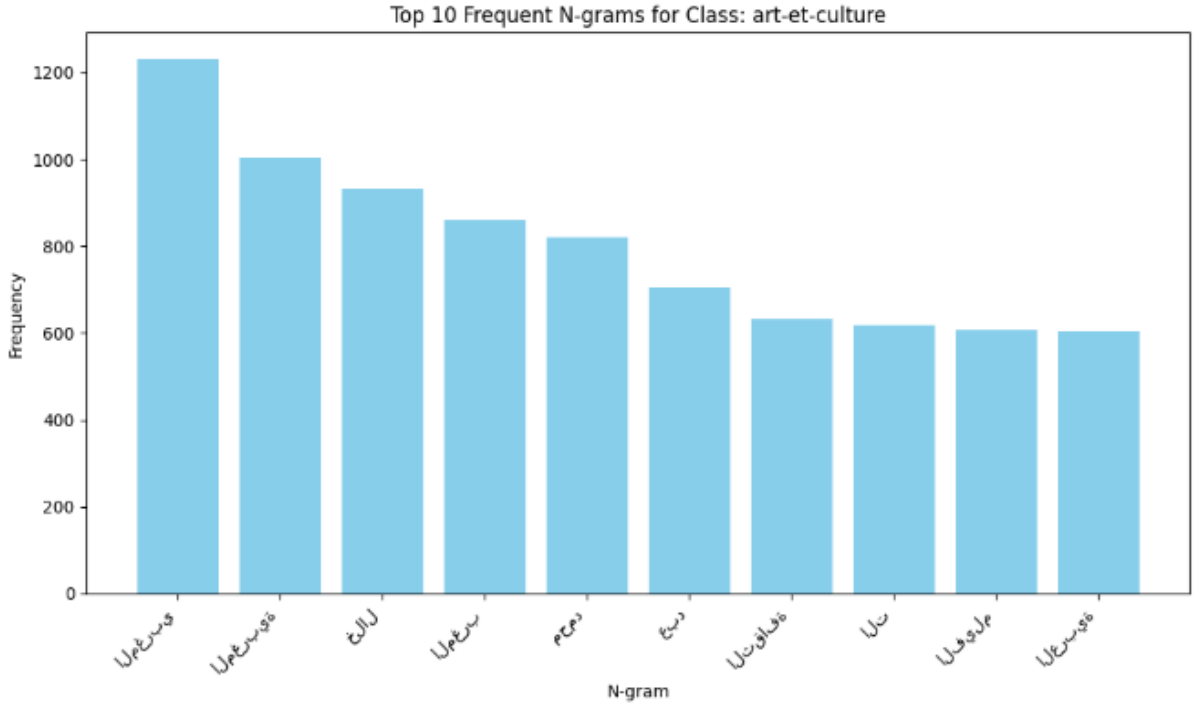
```
[25] ngrams_df_with_class = pd.concat([df['topic'], ngrams_df], axis=1)
     ngrams_per_class = ngrams_df_with_class.groupby('topic').sum()
```

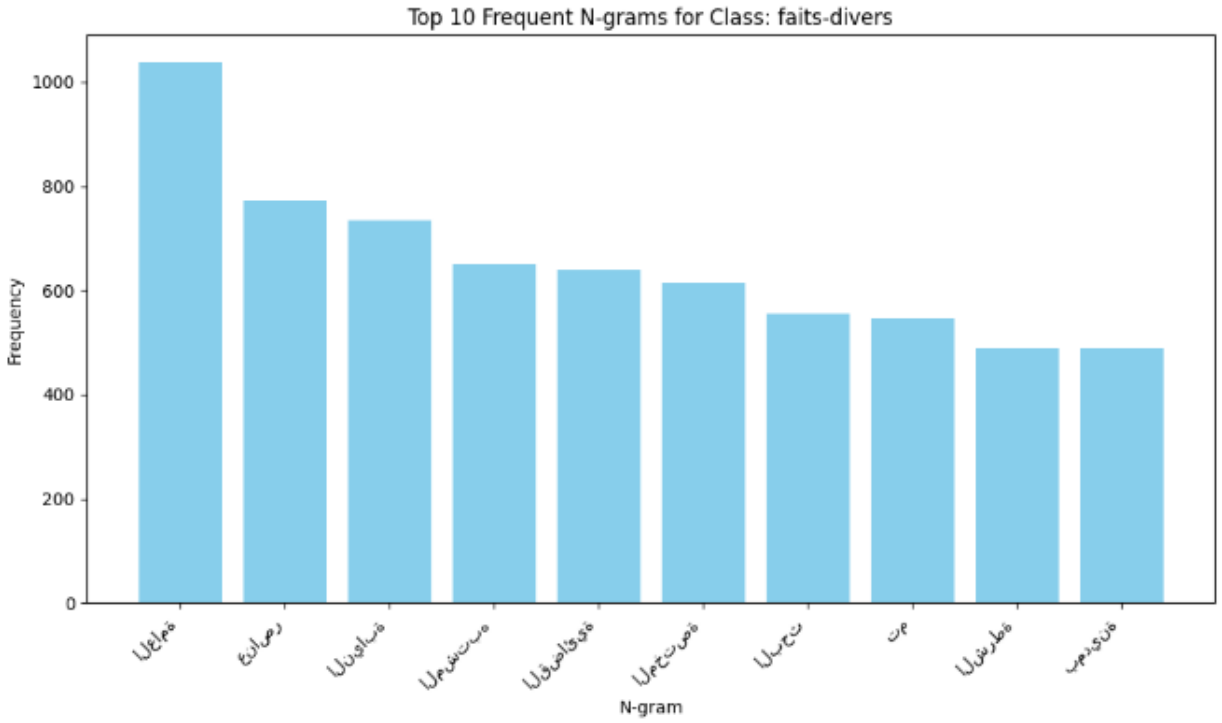
```
[26] ngrams_per_class
```

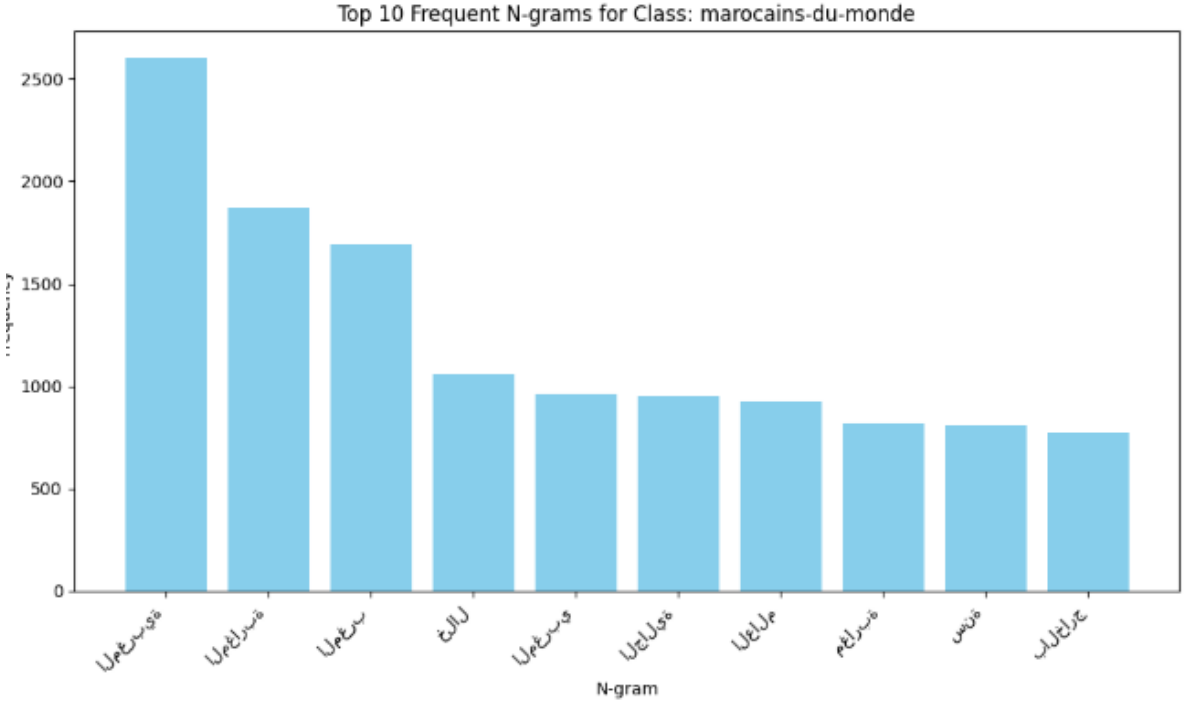
topic	ءا	آباء	آباءهم	آباؤهم	آبازهن	آبائنا	آبائهم	آبائهن	آباه	آبار	...	بيرو	بيس	بيسر	بيسي	بيف	بيكسيانغ	بيمو	بين	بينا	بينوا
art-et-culture	1	1	0	1	0	2	0	0	0	0	...	1	1	0	0	1	1	1	2	2	1
faits-divers	0	2	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
marocains-du-monde	1	6	0	1	0	0	3	0	1	0	...	0	0	0	1	0	0	0	1	0	0
medias	0	16	1	0	0	0	1	1	0	4	...	0	0	1	0	0	0	0	0	0	0
regions	0	18	0	0	0	0	0	0	0	6	...	0	0	0	0	0	0	0	1	0	0
societe	0	69	2	1	1	0	5	0	0	1	...	0	0	0	0	0	0	0	2	0	0

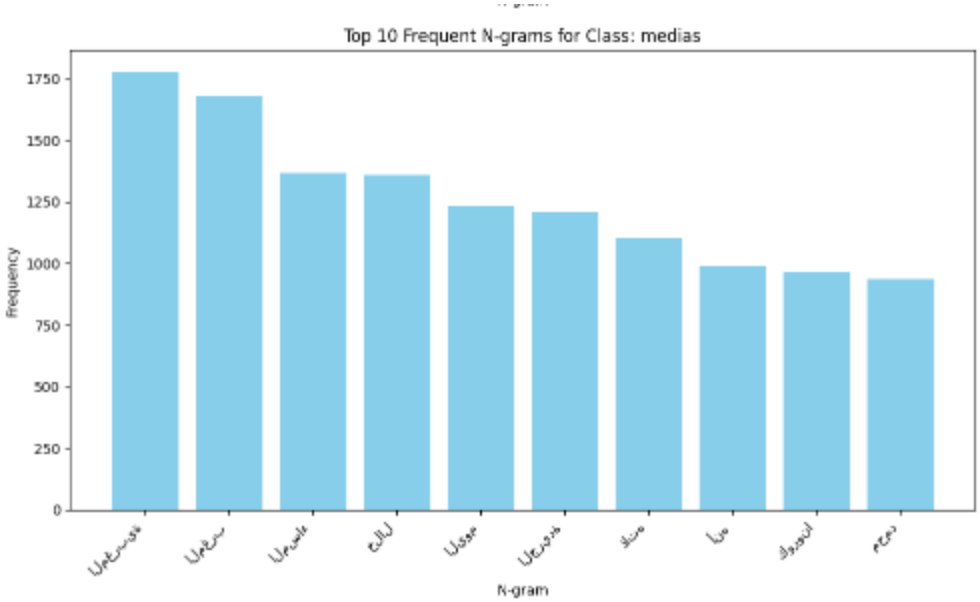
6 rows × 126756 columns


N-gram
Top 10 Frequent N-grams for Class: societe
Frequency
0
200
400
600
800
1000
1200
1400
N-gram


Top 10 Frequent N-grams for Class: art-et-culture
Frequency
0
200
400
600
800
1000
1200
N-gram


Top 10 Frequent N-grams for Class: faits-divers
Frequency
N-gram
0
200
400
600
800
1000


Top 10 Frequent N-grams for Class: marocains-du-monde
N-gram
0
500
1000
1500
2000
2500


Top 10 Frequent N-grams for Class: medias
Frequency
1750
1500
1250
1000
750
500
250
0
N-gram