Bible NLP Project

March 19, 2023

[61]:
```python
from bs4 import BeautifulSoup
import requests, json
import re

import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

from collections import OrderedDict
import operator
import os
import sys
import random
from collections import Counter, defaultdict
```

[3]:
```python
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from nltk.stem.porter import PorterStemmer

from textblob import TextBlob  #Sentiment Analysis - pip install textblob
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer
from sklearn.decomposition import TruncatedSVD

# from wordcloud import WordCloud  #pip install
import string
import nltk
from nltk.corpus import stopwords
from nltk.stem.wordnet import WordNetLemmatizer
import re
import gensim
from gensim.models import word2vec
from gensim import corpora
from sklearn.manifold import TSNE
```

[4]:
```
#Load the American Standard Version Bible from csv file (has to be in the same␣
↪folder as the notebook)

#Description of columns in the CSV file:

# 'id' is a string of the book number (e.g. Genesis = '1')
# plus the chapter (e.g. chapter 1 = '001')
# plus the verse number (e.g. verse 1 = '001')
# hence, Genesis 1:1 would be "1001001"

# 'b' = book number
# 'c' = chapter number
# 'v' = verse number
# 't' = text

df_Bible = pd.read_csv('/Users/santiagolampon/Desktop/python_work/archive/t_asv.
↪csv')
df_Bible.head(10)
```

[4]:
```
        id  b  c   v                                                  t
0  1001001  1  1   1  In the beginning God created the heavens and t…
1  1001002  1  1   2  And the earth was waste and void; and darkness…
2  1001003  1  1   3  And God said, Let there be light: and there wa…
3  1001004  1  1   4  And God saw the light, that it was good: and G…
4  1001005  1  1   5  And God called the light Day, and the darkness…
5  1001006  1  1   6  And God said, Let there be a firmament in the …
6  1001007  1  1   7  And God made the firmament, and divided the wa…
7  1001008  1  1   8  And God called the firmament Heaven. And there…
8  1001009  1  1   9  And God said, Let the waters under the heavens…
9  1001010  1  1  10  And God called the dry land Earth; and the gat…
```

[5]:
```
#Rename the columns so that when merging the data frames later the column names␣
↪will be unique
df_Bible.columns = ['unique_ID', 'book_ID', 'chapter_ID', 'verse_ID', 'text']
df_Bible.head(10)
```

[5]:
```
   unique_ID  book_ID  chapter_ID  verse_ID  \
0    1001001        1           1         1
1    1001002        1           1         2
2    1001003        1           1         3
3    1001004        1           1         4
4    1001005        1           1         5
5    1001006        1           1         6
6    1001007        1           1         7
7    1001008        1           1         8
```

```
8    1001009        1            1         9
9    1001010        1            1        10

                                             text
0  In the beginning God created the heavens and t…
1  And the earth was waste and void; and darkness…
2  And God said, Let there be light: and there wa…
3  And God saw the light, that it was good: and G…
4  And God called the light Day, and the darkness…
5  And God said, Let there be a firmament in the …
6  And God made the firmament, and divided the wa…
7  And God called the firmament Heaven. And there…
8  And God said, Let the waters under the heavens…
9  And God called the dry land Earth; and the gat…
```

[6]:
```python
#topic modeling cleaning function (from https://www.analyticsvidhya.com/blog/
 ↪2016/08/beginners-guide-to-topic-modeling-in-python/)

#Get standard english stopwords from NLTK
cachedStopWords = stopwords.words("english")
#Extend the list of stopwords to include old English words found in the Bible
cachedStopWords.extend(['thou','thee','thy','thine','ye','er','hast',
                        'hath','art', 'wilt', 'didst', 'thyself', 'shalt'])
#convert the stop words list into a set
stop = set(cachedStopWords)
#convert the punctuation characters into a set
exclude = set(string.punctuation)
#initialize the Lemmatizer to grab only the root of words
lemma = WordNetLemmatizer()
#define a function that can be called to clean each document (verse of the␣
 ↪Bible in this case)
def clean(doc):
    stop_free = " ".join([i for i in doc.lower().split() if i not in stop])
    punc_free = ''.join(ch for ch in stop_free if ch not in exclude)
    normalized = " ".join(lemma.lemmatize(word) for word in punc_free.split())
    return normalized
```

[7]:
```python
#Test the clean(doc) function before using on text of the Bible
test_sentence = 'Hello world! Thy sentence shalt be cleaned immediately.'
test_sentence_clean = clean(test_sentence)
print(test_sentence_clean)
```

```
hello world sentence cleaned immediately
```

[12]:
```python
# change the "t" (text) column in pandas dataframe to an array of sentences for␣
 ↪cleaning
df_text_doc = df_Bible.text.to_numpy()
```

```
print(df_text_doc)
```

```
['In the beginning God created the heavens and the earth.'
 'And the earth was waste and void; and darkness was upon the face of the deep:
and the Spirit of God moved upon the face of the waters.'
 'And God said, Let there be light: and there was light.' …
 'and if any man shall take away from the words of the book of this prophecy,
God shall take away his part from the tree of life, and out of the holy city,
which are written in this book.'
 'He who testifieth these things saith, Yea: I come quickly. Amen: come, Lord
Jesus.'
 'The grace of the Lord Jesus be with the saints. Amen.']
```

[13]:
```
#Convert the text of the Bible into an array of tokenized arrays and check the␣
 ↪first five
df_doc_clean = [clean(doc).split() for doc in df_text_doc]
print(df_doc_clean[0:5])
```

```
[['beginning', 'god', 'created', 'heaven', 'earth'], ['earth', 'waste', 'void',
'darkness', 'upon', 'face', 'deep', 'spirit', 'god', 'moved', 'upon', 'face',
'water'], ['god', 'said', 'let', 'light', 'light'], ['god', 'saw', 'light',
'good', 'god', 'divided', 'light', 'darkness'], ['god', 'called', 'light',
'day', 'darkness', 'called', 'night', 'evening', 'morning', 'one', 'day']]
```

[14]:
```
#create the term dictionary our corpus, where every unique term is assigned an␣
 ↪index.
dictionary = corpora.Dictionary(df_doc_clean)
```

[15]:
```
#check dictionary that we created from the Bible's corpus (11653 unique tokens␣
 ↪from all of
# the 66 books of the Bible combined after cleaning up stopwords and removing␣
 ↪punctuation)
print(dictionary)
```

```
Dictionary(11652 unique tokens: ['beginning', 'created', 'earth', 'god',
'heaven']…)
```

[16]:
```
# Converting list of documents (corpus) into Document Term Matrix using␣
 ↪dictionary prepared above.
doc_term_matrix = [dictionary.doc2bow(doc) for doc in df_doc_clean]
```

[17]:
```
#check first five lines of the document term matrix
doc_term_matrix[0:5]
```

[17]:
```
[[(0, 1), (1, 1), (2, 1), (3, 1), (4, 1)],
 [(2, 1),
  (3, 1),
  (5, 1),
  (6, 1),
  (7, 2),
  (8, 1),
  (9, 1),
  (10, 2),
  (11, 1),
  (12, 1),
  (13, 1)],
 [(3, 1), (14, 1), (15, 2), (16, 1)],
 [(3, 2), (5, 1), (15, 2), (17, 1), (18, 1), (19, 1)],
 [(3, 1),
  (5, 1),
  (15, 1),
  (20, 2),
  (21, 2),
  (22, 1),
  (23, 1),
  (24, 1),
  (25, 1)]]
```

[18]:
```
#data here is 'df_text_doc' derived previously, in a cell above,
#an array of all verses in all 66 books of the Bible
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer

no_features = 5000
```

[19]:
```
# NMF is able to use tf-idf
tfidf_vectorizer = TfidfVectorizer(max_df=0.95, min_df=2,␣
 ↪max_features=no_features, stop_words='english')
tfidf = tfidf_vectorizer.fit_transform(df_text_doc)
tfidf_feature_names = tfidf_vectorizer.get_feature_names()
```

```
/Users/santiagolampon/opt/anaconda3/lib/python3.9/site-
packages/sklearn/utils/deprecation.py:87: FutureWarning: Function
get_feature_names is deprecated; get_feature_names is deprecated in 1.0 and will
be removed in 1.2. Please use get_feature_names_out instead.
  warnings.warn(msg, category=FutureWarning)
```

[20]:
```
# LDA can only use raw term counts for LDA because it is a probabilistic␣
 ↪graphical model
tf_vectorizer = CountVectorizer(max_df=0.95, min_df=2,␣
 ↪max_features=no_features, stop_words='english')
tf = tf_vectorizer.fit_transform(df_text_doc)
tf_feature_names = tf_vectorizer.get_feature_names()
```

```
[21]: #print out the first 10 features from the TF-IDF matrix
      tfidf_feature_names[0:10]
```

```
[21]: ['aaron',
       'abarim',
       'abated',
       'abdon',
       'abed',
       'abel',
       'abhor',
       'abhorred',
       'abhorreth',
       'abiathar']
```

```
[22]: #print out the first 10 features from the TF matrix
      tf_feature_names[0:10]
```

```
[22]: ['aaron',
       'abarim',
       'abated',
       'abdon',
       'abed',
       'abel',
       'abhor',
       'abhorred',
       'abhorreth',
       'abiathar']
```

```
[27]: from sklearn.decomposition import NMF, LatentDirichletAllocation

      no_topics = 20

      # Run NMF
      nmf = NMF(n_components=no_topics, random_state=1, alpha=.1, l1_ratio=.5,␣
       ↪init='nndsvd').fit(tfidf)

      # Run LDA
      lda = LatentDirichletAllocation(n_components=no_topics, max_iter=5,␣
       ↪learning_method='online', learning_offset=50.,random_state=0).fit(tf)
```

```
/Users/santiagolampon/opt/anaconda3/lib/python3.9/site-
packages/sklearn/decomposition/_nmf.py:1422: FutureWarning: `alpha` was
deprecated in version 1.0 and will be removed in 1.2. Use `alpha_W` and
`alpha_H` instead
  warnings.warn(
[24]: #display function for both LDA and Non-negative Matrix Factorization (NMF)
      # from https://medium.com/@aneesha/topic-modeling-with-scikit-learn-e80d33668730
      def display_topics(model, feature_names, no_top_words):
          for topic_idx, topic in enumerate(model.components_):
              print("Topic %d:" % (topic_idx))
              print(" ".join([feature_names[i] for i in topic.argsort()[:
       ↪-no_top_words - 1:-1]]))
```

```
[25]: no_top_words = 10
      print("NMF top 20 topics with top 10 words:\n")
      display_topics(nmf, tfidf_feature_names, no_top_words)
```

```
NMF top 20 topics with top 10 words:

Topic 0:
jehovah saith house israel hosts word hand day evil did
Topic 1:
thou hast art wilt didst thyself thine said knowest know
Topic 2:
shall come day pass days holy eat thereof fall say
Topic 3:
ye know say hear eat things seek love believe brethren
Topic 4:
unto said say answered come called father speak sent brethren
Topic 5:
thy hand heart servant thine father lovingkindness servants face right
Topic 6:
shalt thou make eat surely thereof thyself die bring gold
Topic 7:
god israel spirit glory fear know law kingdom earth heaven
Topic 8:
children israel thousand tribe according did ammon inheritance levites cities
Topic 9:
thee pray make thine behold come thou bring bless deliver
Topic 10:
son sons father david tribe reigned stead bare jonathan saul
Topic 11:
came word pass saying went day jeremiah days month told
Topic 12:
man men woman behold good house wise evil know young
Topic 13:
hath given spoken taken hand away seen father sent thing
Topic 14:
saying spake moses unto jehovah aaron commanded word people words
Topic 15:
land egypt people brought forth pharaoh dwell fathers canaan bring
Topic 16:
```

```
lord jesus christ things saith grace father spirit peace faith
Topic 17:
king house judah israel david jerusalem went babylon men hand
Topic 18:
let come hear pray rejoice shame said make heart say
Topic 19:
offering burnt sin altar offerings meal thereof offer priest bullock
```

[28]:
```
no_top_words = 10
print("LDA top 20 topics with top 10 words:\n")
display_topics(lda, tf_feature_names, no_top_words)
```

```
LDA top 20 topics with top 10 words:

Topic 0:
days love seven years old tree died begat jeremiah food
Topic 1:
thereof midst waters cubits priests ark tongue ear money half
Topic 2:
sea round light temple wine new morning thereof entered oil
Topic 3:
people unto hear brought blood sin voice bread jehovah return
Topic 4:
son father city judah sons brethren dwell wife mother called
Topic 5:
hath great time given god faith dead year seek delivered
Topic 6:
cast chief works child places psalm number gate break north
Topic 7:
shall ye unto jehovah saith man god say lord know
Topic 8:
thou thy unto thee said jehovah things god saying shalt
Topic 9:
day set work rest month book sabbath gates doeth stones
Topic 10:
david joy levites zion image tabernacle fallen daniel lift joab
Topic 11:
israel children christ fathers hosts unto jesus born houses beloved
Topic 12:
jehovah did unto according law god took gave eyes word
Topic 13:
come house holy nations brother righteous behold wicked jehovah iniquity
Topic 14:
mouth disciples written receive salvation second opened counsel joshua eternal
Topic 15:
king came unto said like place jerusalem pass jesus saw
Topic 16:
hand hands right left feet live forth laid deliver fall
```

```
Topic 17:
away way evil sword power good sight turn wrath turned
Topic 18:
earth spirit let sent heaven world high face god man
Topic 19:
men went land thousand egypt cities water gold kings unto
```

[29]:
```
# define a function that accepts text and returns the polarity
def getSentimentPolarity(text):
    return format(TextBlob(text).sentiment.polarity, '.4f')
```

[30]:
```
df_Bible['clean_text'] = df_Bible.text.apply(clean)
```

[31]:
```
df_Bible.head(10)
```

[31]:
```
   unique_ID  book_ID  chapter_ID  verse_ID  \
0    1001001        1           1         1
1    1001002        1           1         2
2    1001003        1           1         3
3    1001004        1           1         4
4    1001005        1           1         5
5    1001006        1           1         6
6    1001007        1           1         7
7    1001008        1           1         8
8    1001009        1           1         9
9    1001010        1           1        10

                                                text  \
0  In the beginning God created the heavens and t…
1  And the earth was waste and void; and darkness…
2  And God said, Let there be light: and there wa…
3  And God saw the light, that it was good: and G…
4  And God called the light Day, and the darkness…
5  And God said, Let there be a firmament in the …
6  And God made the firmament, and divided the wa…
7  And God called the firmament Heaven. And there…
8  And God said, Let the waters under the heavens…
9  And God called the dry land Earth; and the gat…

                                          clean_text
0                    beginning god created heaven earth
1  earth waste void darkness upon face deep spiri…
2                                god said let light light
3      god saw light good god divided light darkness
4  god called light day darkness called night eve…
5  god said let firmament midst water let divide …
6  god made firmament divided water firmament wat…
```

```
7  god called firmament heaven evening morning se…
8  god said let water heaven gathered together un…
9  god called dry land earth gathering together w…
```

[32]:
```
# create a new column for sentiment
df_Bible['sentiment'] = df_Bible.clean_text.apply(getSentimentPolarity)
```

[33]:
```
df_Bible.head(10)
```

[33]:
```
   unique_ID  book_ID  chapter_ID  verse_ID  \
0    1001001        1           1         1
1    1001002        1           1         2
2    1001003        1           1         3
3    1001004        1           1         4
4    1001005        1           1         5
5    1001006        1           1         6
6    1001007        1           1         7
7    1001008        1           1         8
8    1001009        1           1         9
9    1001010        1           1        10

                                                text  \
0  In the beginning God created the heavens and t…
1  And the earth was waste and void; and darkness…
2  And God said, Let there be light: and there wa…
3  And God saw the light, that it was good: and G…
4  And God called the light Day, and the darkness…
5  And God said, Let there be a firmament in the …
6  And God made the firmament, and divided the wa…
7  And God called the firmament Heaven. And there…
8  And God said, Let the waters under the heavens…
9  And God called the dry land Earth; and the gat…

                                          clean_text sentiment
0                   beginning god created heaven earth    0.0000
1  earth waste void darkness upon face deep spiri…   -0.1000
2                             god said let light light    0.4000
3      god saw light good god divided light darkness    0.5000
4  god called light day darkness called night eve…    0.4000
5  god said let firmament midst water let divide …    0.0000
6  god made firmament divided water firmament wat…    0.0000
7  god called firmament heaven evening morning se…    0.0000
8  god said let water heaven gathered together un…   -0.0667
9  god called dry land earth gathering together w…    0.3167
```

[34]:
```
len(df_Bible['sentiment'].values)
```

[34]:
```
31103
```

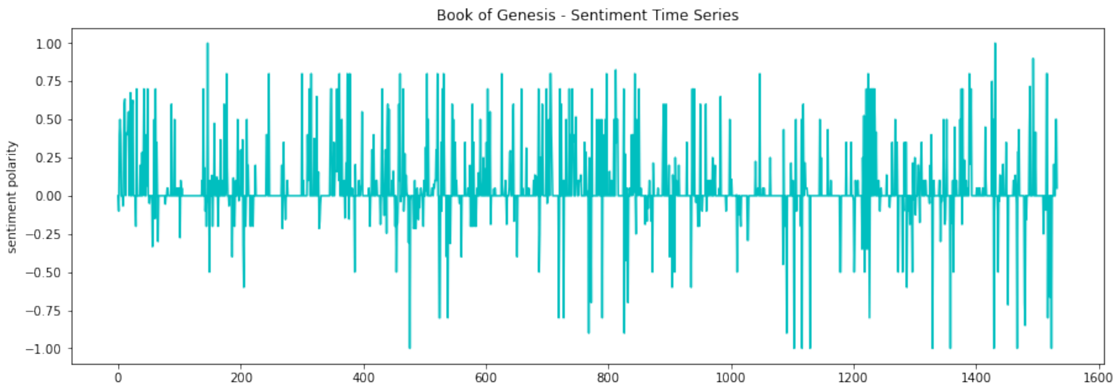
[35]:
```
# Create a new data frame with only the rows (one row per verse) from the book␣
↪of Genesis
df_Genesis = df_Bible[df_Bible['book_ID']== 1]
df_Genesis.head()
```

[35]:
```
   unique_ID  book_ID  chapter_ID  verse_ID  \
0    1001001        1           1         1
1    1001002        1           1         2
2    1001003        1           1         3
3    1001004        1           1         4
4    1001005        1           1         5

                                               text  \
0  In the beginning God created the heavens and t…
1  And the earth was waste and void; and darkness…
2  And God said, Let there be light: and there wa…
3  And God saw the light, that it was good: and G…
4  And God called the light Day, and the darkness…

                                        clean_text sentiment
0                beginning god created heaven earth    0.0000
1  earth waste void darkness upon face deep spiri…   -0.1000
2                          god said let light light    0.4000
3      god saw light good god divided light darkness    0.5000
4  god called light day darkness called night eve…    0.4000
```

[36]:
```
df_Genesis.shape
```

[36]:
```
(1533, 7)
```

[37]:
```
#Sentiment time series for entire Bible from Genesis (1st book) to Revelation␣
↪(66th book)
sentiment_Genesis = pd.Series(data=df_Genesis['sentiment'].values,
                            index=df_Genesis.index).astype(float)

plt.title("Book of Genesis - Sentiment Time Series")
plt.ylabel("sentiment polarity")
sentiment_Genesis.plot(figsize=(15,5), color=random.choice(['b', 'g', 'r', 'c',␣
↪'m', 'y', 'k']));
plt.show()
```



[38]:
```
#Load the genre key from csv file (has to be in the same folder as the notebook)
df_key_1 = pd.read_csv('/Users/santiagolampon/Desktop/python_work/archive/
 ↪key_genre_english.csv')
df_key_1.head(10)
```

[38]:
```
   g            n
0  1          Law
1  2      History
2  3       Wisdom
3  4     Prophets
4  5      Gospels
5  6         Acts
6  7     Epistles
7  8  Apocalyptic
```

[39]:
```
#Rename the columns so that when merging the data frames the column names will␣
 ↪be unique
df_key_1.columns = ['genre_ID', 'genre']
df_key_1.head(10)
```

[39]:
```
   genre_ID        genre
0         1          Law
1         2      History
2         3       Wisdom
3         4     Prophets
4         5      Gospels
5         6         Acts
6         7     Epistles
7         8  Apocalyptic
```

[40]:
```
#Load the genre-to-book key from csv file (has to be in the same folder as the␣
 ↪notebook)
```

```
df_key_2 = pd.read_csv('/Users/santiagolampon/Desktop/python_work/archive/
  ↪key_english.csv')
df_key_2.head(10)
```

```
[40]:     b             n   t  g
      0   1       Genesis  OT  1
      1   2        Exodus  OT  1
      2   3     Leviticus  OT  1
      3   4       Numbers  OT  1
      4   5   Deuteronomy  OT  1
      5   6        Joshua  OT  2
      6   7        Judges  OT  2
      7   8          Ruth  OT  2
      8   9      1 Samuel  OT  2
      9  10      2 Samuel  OT  2
```

```
[41]: #Rename the columns so that when merging the data frames the column names will␣
        ↪be unique
      df_key_2.columns = ['book_ID', 'book', 'testament', 'genre_ID']
      df_key_2.head(10)
```

```
[41]:    book_ID         book testament  genre_ID
      0        1      Genesis        OT         1
      1        2       Exodus        OT         1
      2        3    Leviticus        OT         1
      3        4      Numbers        OT         1
      4        5  Deuteronomy        OT         1
      5        6       Joshua        OT         2
      6        7       Judges        OT         2
      7        8         Ruth        OT         2
      8        9     1 Samuel        OT         2
      9       10     2 Samuel        OT         2
```

```
[42]: #inner join merge the two key data frames 'df_key_1' with 'df_key_2',
      df_key_merge = pd.merge(df_key_1, df_key_2, on='genre_ID', how='inner')
      df_key_merge.head(10)
```

```
[42]:    genre_ID    genre  book_ID         book testament
      0         1      Law        1      Genesis        OT
      1         1      Law        2       Exodus        OT
      2         1      Law        3    Leviticus        OT
      3         1      Law        4      Numbers        OT
      4         1      Law        5  Deuteronomy        OT
      5         2  History        6       Joshua        OT
      6         2  History        7       Judges        OT
      7         2  History        8         Ruth        OT
      8         2  History        9     1 Samuel        OT
```

```
9          2  History       10      2 Samuel        OT
```

[63]:
```
# merge the main data frame 'df_key_merge' with 'df_Bible' using an 'inner␣
↪join'
# using the 'book_ID' column
df_Bible_keys = pd.merge(df_key_merge, df_Bible, on='book_ID', how='inner')
df_Bible_keys.head(10)
df_Bible_keys.tail(5)
```

[63]:
```
       genre_ID        genre  book_ID        book testament  unique_ID  \
31098         8  Apocalyptic       66  Revelation        NT   66022017
31099         8  Apocalyptic       66  Revelation        NT   66022018
31100         8  Apocalyptic       66  Revelation        NT   66022019
31101         8  Apocalyptic       66  Revelation        NT   66022020
31102         8  Apocalyptic       66  Revelation        NT   66022021

       chapter_ID  verse_ID  \
31098          22        17
31099          22        18
31100          22        19
31101          22        20
31102          22        21

                                                    text  \
31098  And the Spirit and the bride say, Come. And he…
31099  I testify unto every man that heareth the word…
31100  and if any man shall take away from the words …
31101  He who testifieth these things saith, Yea: I c…
31102  The grace of the Lord Jesus be with the saints…

                                              clean_text sentiment
31098  spirit bride say come heareth let say come ath…    0.4000
31099  testify unto every man heareth word prophecy b…    0.0000
31100  man shall take away word book prophecy god sha…    0.0000
31101  testifieth thing saith yea come quickly amen c…    0.3333
31102                      grace lord jesus saint amen    0.0000
```

[44]:
```
df_Bible_keys.shape
```

[44]:
```
(31103, 11)
```

[45]:
```
# get list of unique genre labels
genre_labels = pd.unique(df_Bible_keys.genre)
print(genre_labels)
```

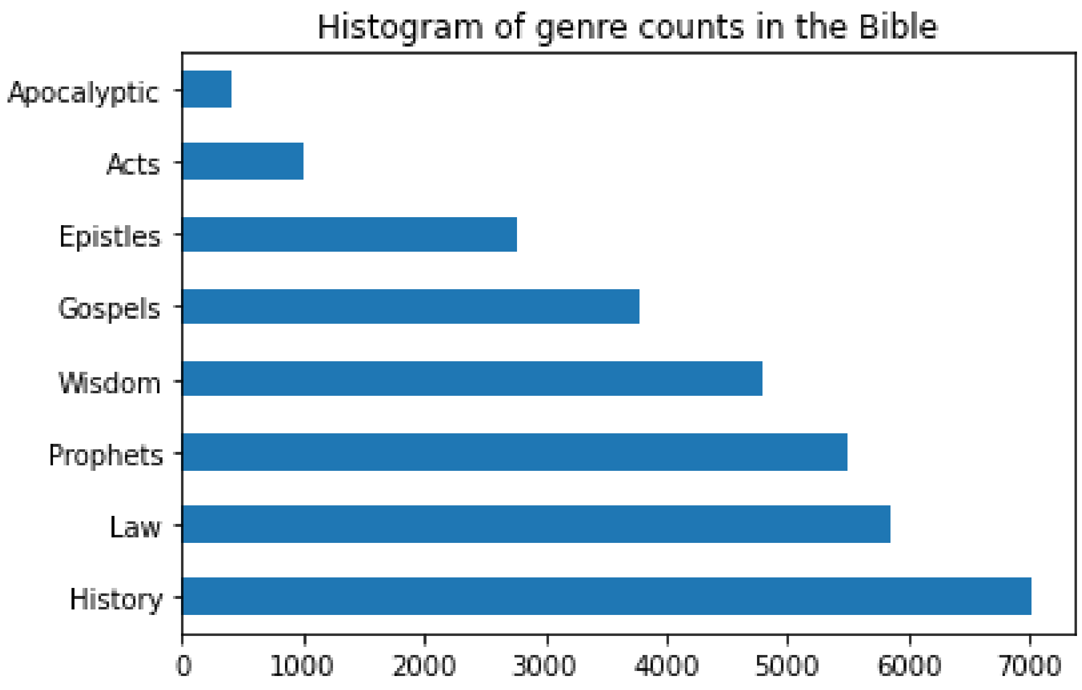
```
['Law' 'History' 'Wisdom' 'Prophets' 'Gospels' 'Acts' 'Epistles'
 'Apocalyptic']
```

```
[46]: #First check the value counts for each genre
      df_Bible_keys.genre.value_counts()
```

```
[46]: History        7018
      Law            5852
      Prophets       5490
      Wisdom         4785
      Gospels        3779
      Epistles       2768
      Acts           1007
      Apocalyptic     404
      Name: genre, dtype: int64
```

```
[47]: genre_counts = pd.value_counts(df_Bible_keys.genre.values, sort=True)
      genre_counts.plot.barh(title="Histogram of genre counts in the Bible")
```

```
[47]: <AxesSubplot:title={'center':'Histogram of genre counts in the Bible'}>
```



```
[48]: # create independent and dependent variables for classification
      X = df_Bible_keys['clean_text']
      y = df_Bible_keys['genre']
```

```
[49]: from sklearn.model_selection import train_test_split
```

```
# split the data into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33,␣
 ↪random_state=101)
```

[50]:
```
# number of train and test samples
print(X_train.shape)
print(X_test.shape)
```

```
(20839,)
(10264,)
```

[51]:
```
# instantiate the vectorizer
vect = CountVectorizer()
```

[52]:
```
# learn training data vocabulary and create document-term matrix
X_train_dtm = vect.fit_transform(X_train)
X_train_dtm
```

[52]:
```
<20839x10268 sparse matrix of type '<class 'numpy.int64'>'
        with 228016 stored elements in Compressed Sparse Row format>
```

[53]:
```
# transform testing data (using fitted vocabulary) into a document-term matrix
X_test_dtm = vect.transform(X_test)
X_test_dtm
```

[53]:
```
<10264x10268 sparse matrix of type '<class 'numpy.int64'>'
        with 110088 stored elements in Compressed Sparse Row format>
```

[54]:
```
# use multinomial NaiveBayes classifier
from sklearn.naive_bayes import MultinomialNB
nb = MultinomialNB()
nb.fit(X_train_dtm, y_train)
```

[54]:
```
MultinomialNB()
```

[55]:
```
y_pred_class = nb.predict(X_test_dtm)
```

[56]:
```
from sklearn import metrics
print(metrics.accuracy_score(y_test, y_pred_class))
```

```
0.7244738893219018
```

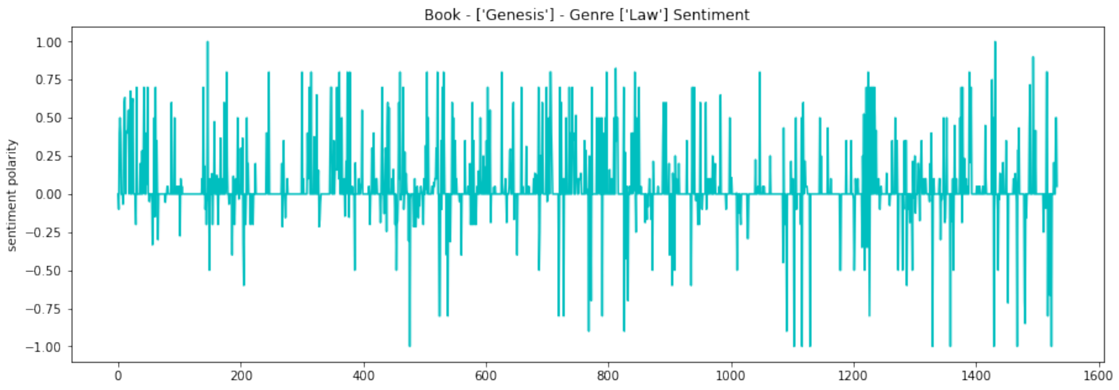
[57]:
```
# confusion matrix
print(metrics.confusion_matrix(y_test, y_pred_class))
```

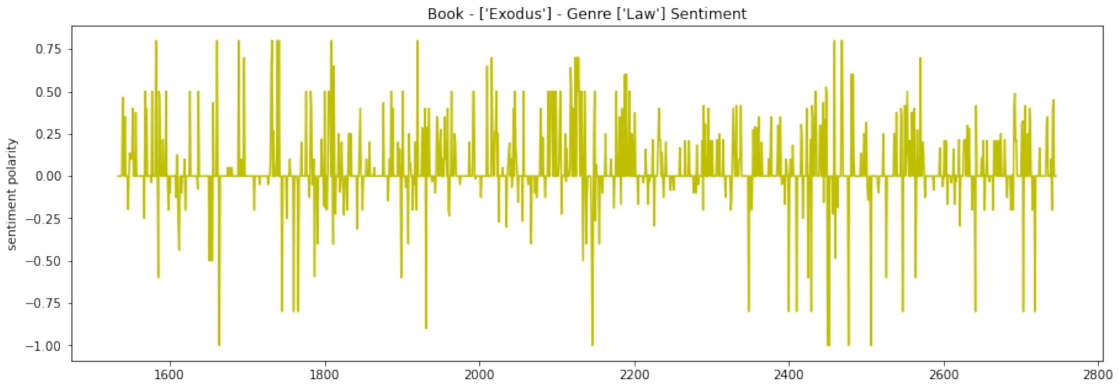
```
[[ 102    0   30  138   27   17   23    4]
 [   1   29   12   24    6   16   30   12]
 [   5    0  695   80   22   36   38   60]
 [   8    1   56  944   59   56   63   39]
 [   3    0    4   32 1850  191  156   56]
 [   0    2    8   41  200 1428  190   67]
 [   0    4    9   34  173   99 1303  179]
 [   2    3   39   26   73   85  289 1085]]
```

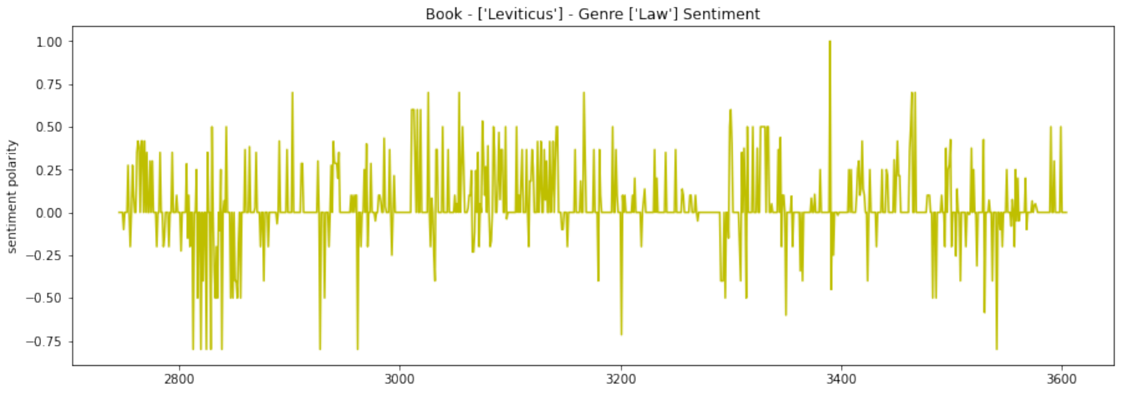
```
[58]: from sklearn.metrics import classification_report
      y_true = y_test
      y_pred = y_pred_class
      target_names = genre_labels
      print(classification_report(y_true, y_pred, target_names=target_names))
```

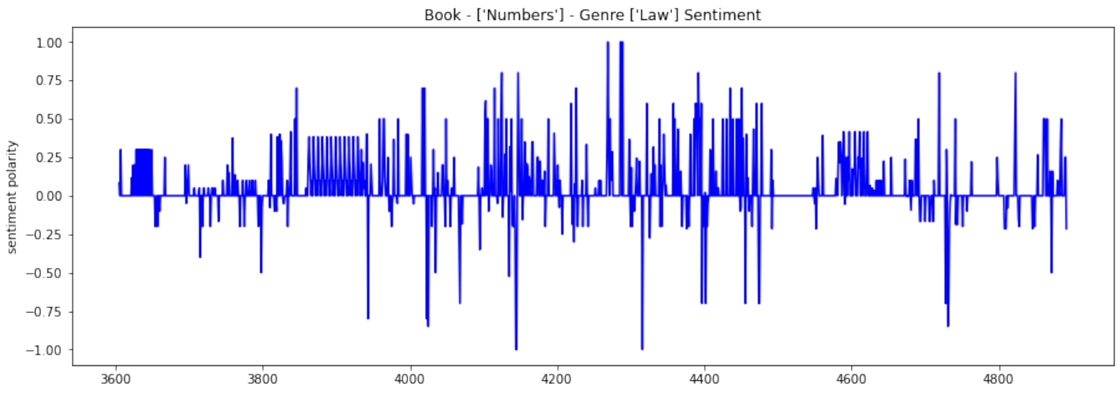
```
              precision    recall  f1-score   support

         Law       0.84      0.30      0.44       341
     History       0.74      0.22      0.34       130
      Wisdom       0.81      0.74      0.78       936
    Prophets       0.72      0.77      0.74      1226
     Gospels       0.77      0.81      0.79      2292
        Acts       0.74      0.74      0.74      1936
    Epistles       0.62      0.72      0.67      1801
 Apocalyptic       0.72      0.68      0.70      1602

    accuracy                           0.72     10264
   macro avg       0.75      0.62      0.65     10264
weighted avg       0.73      0.72      0.72     10264
```

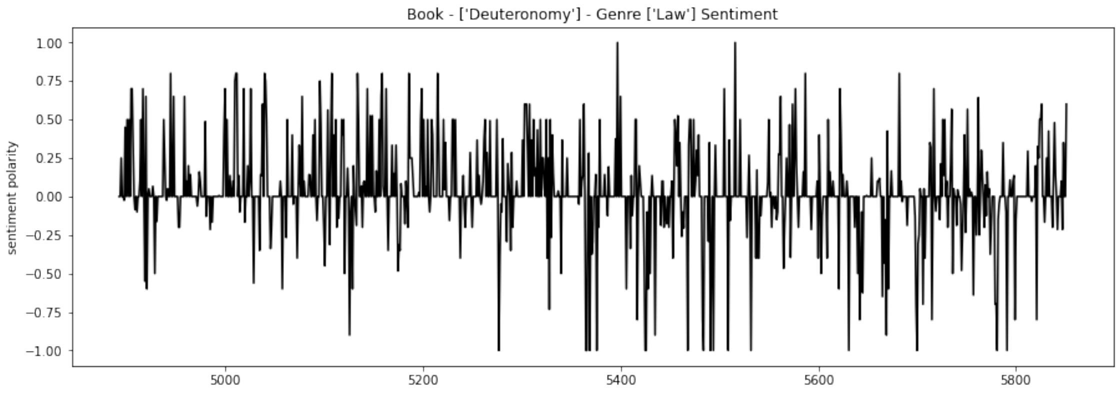
```
[59]: # plot a sentiment time series per book in the Bible
      for book_id in df_Bible_keys['book_ID'].unique():
          df_book = df_Bible_keys[df_Bible_keys['book_ID']== book_id]
          df_book.reset_index(drop=True)
          genre = df_book.genre.unique()
          book_title = df_book.book.unique()
          sentiment_book = pd.Series(data=df_book['sentiment'].values, index=df_book.
       ↪index).astype(float)
          plt.title("Book - %s - Genre %s Sentiment"%(book_title,genre))
          plt.ylabel("sentiment polarity")
          sentiment_book.plot(figsize=(15,5), color=random.choice(['b', 'g', 'r',␣
       ↪'c', 'm', 'y', 'k']));
          plt.show()
```

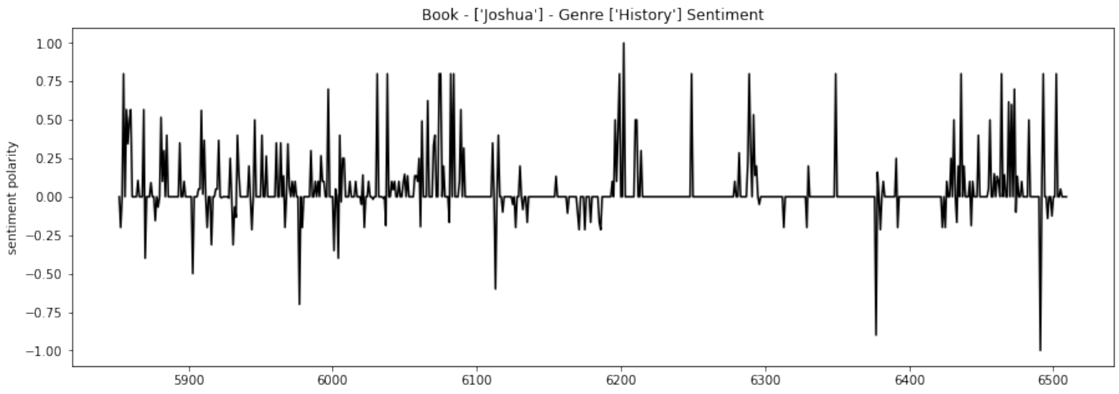

Book - ['Genesis'] - Genre ['Law'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
200
400
600
800
1000
1200
1400
1600

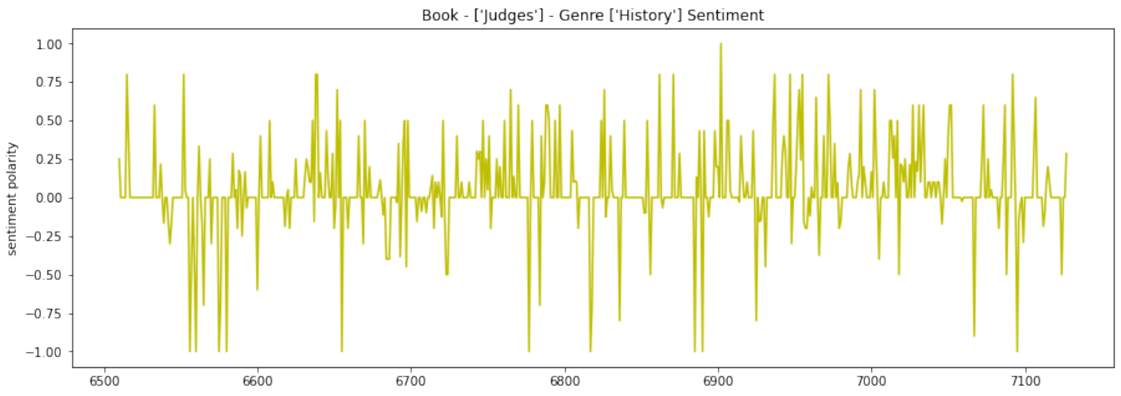

Book - ['Exodus'] - Genre ['Law'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
1600
1800
2000
2200
2400
2600
2800

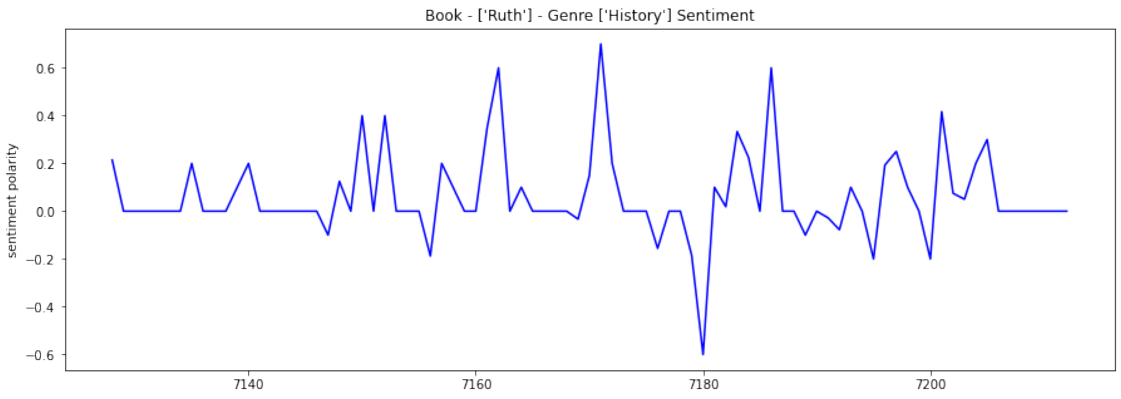

Book - ['Leviticus'] - Genre ['Law'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
2800
3000
3200
3400
3600

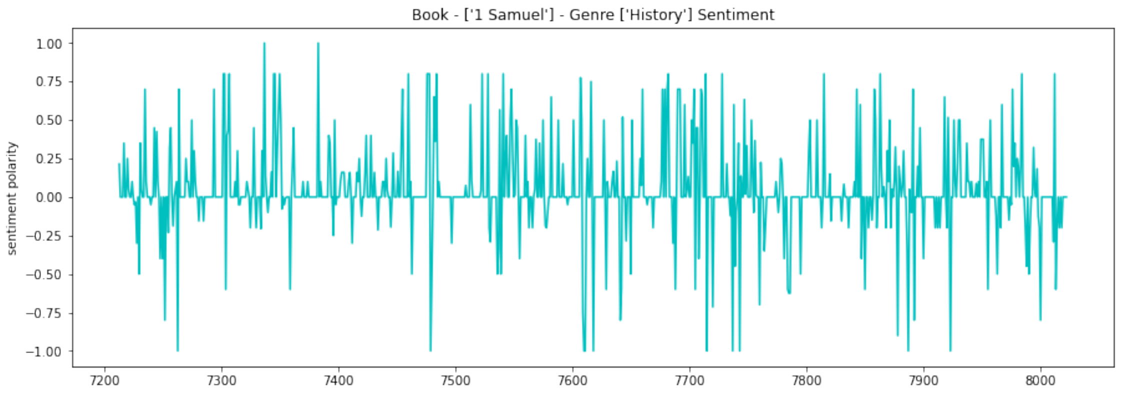

Book - ['Numbers'] - Genre ['Law'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
3600
3800
4000
4200
4400
4600
4800

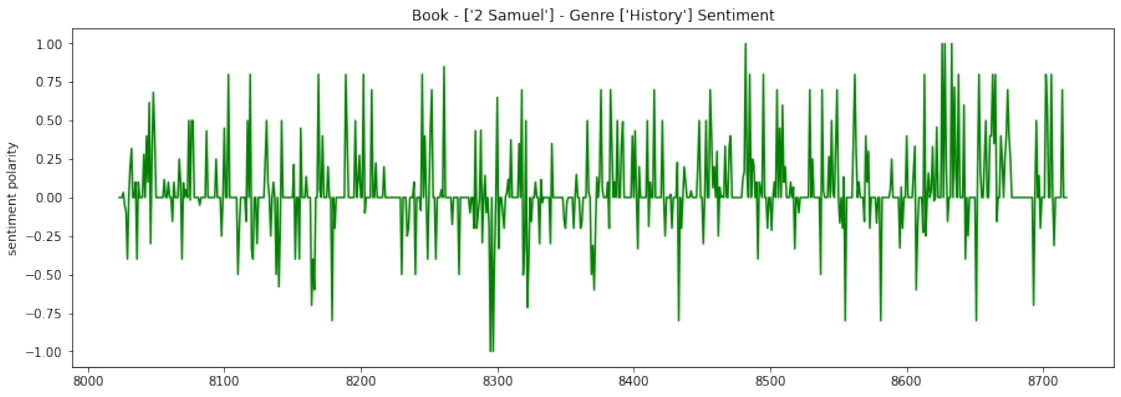

Book - ['Deuteronomy'] - Genre ['Law'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
5000
5200
5400
5600
5800

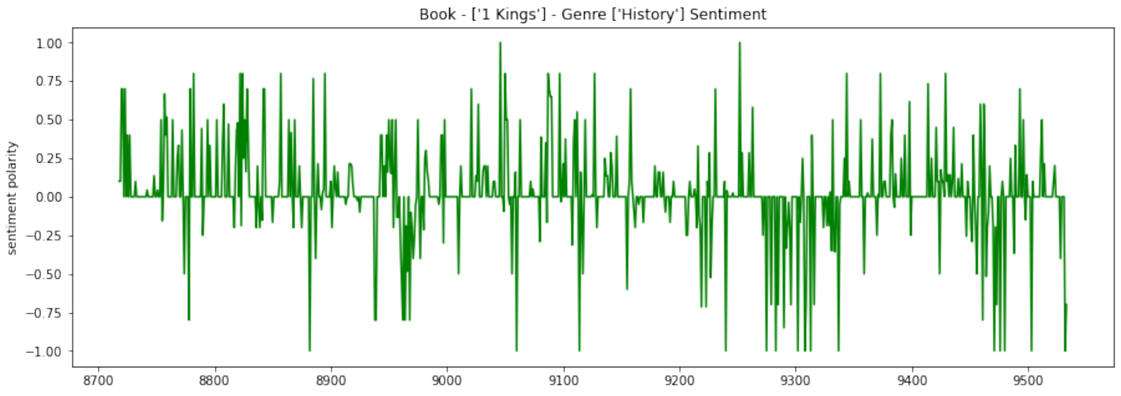

Book - ['Joshua'] - Genre ['History'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
5900
6000
6100
6200
6300
6400
6500

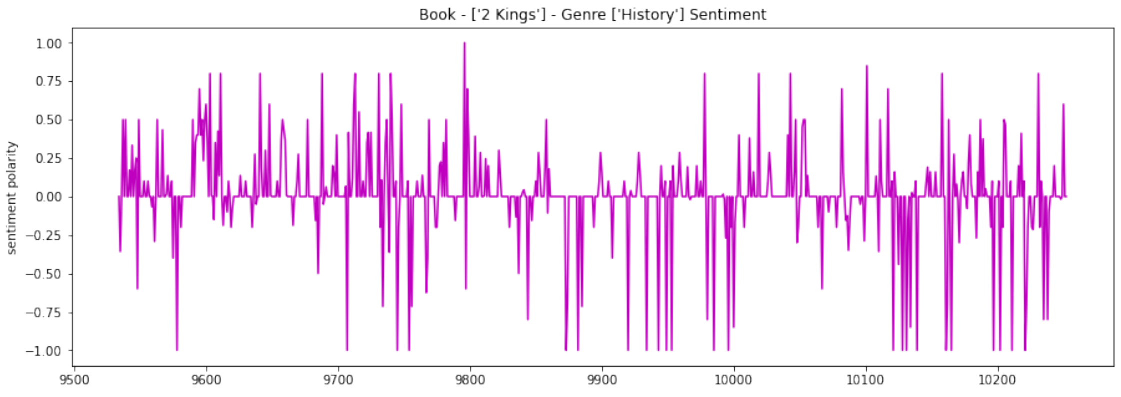

Book - ['Judges'] - Genre ['History'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
6500
6600
6700
6800
6900
7000
7100

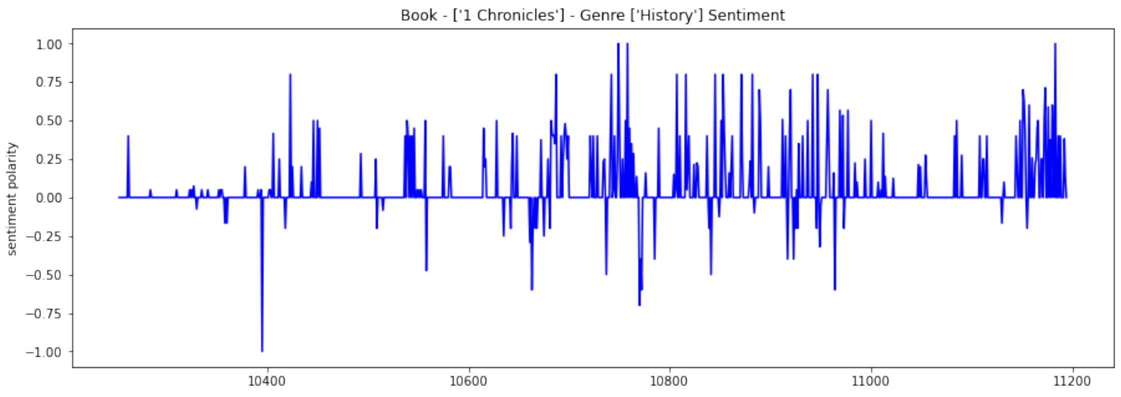

Book - ['Ruth'] - Genre ['History'] Sentiment
sentiment polarity
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
7140
7160
7180
7200

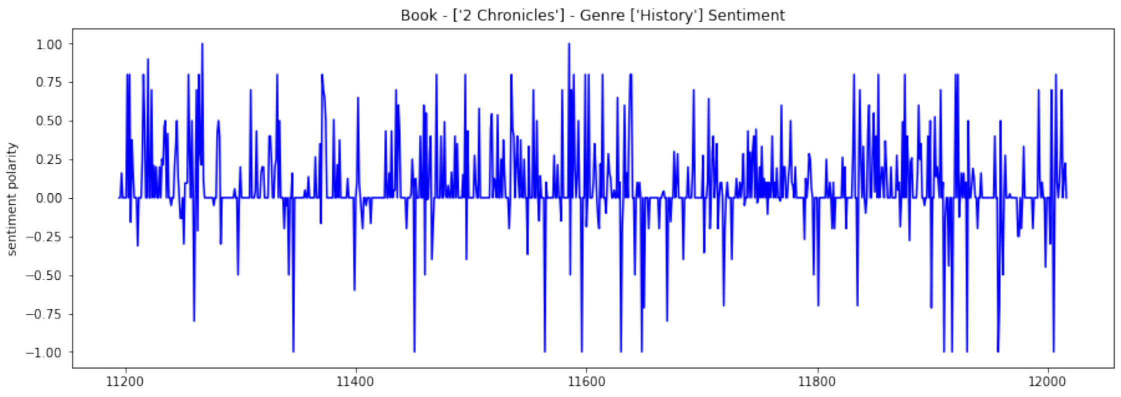

Book - ['1 Samuel'] - Genre ['History'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
7200
7300
7400
7500
7600
7700
7800
7900
8000

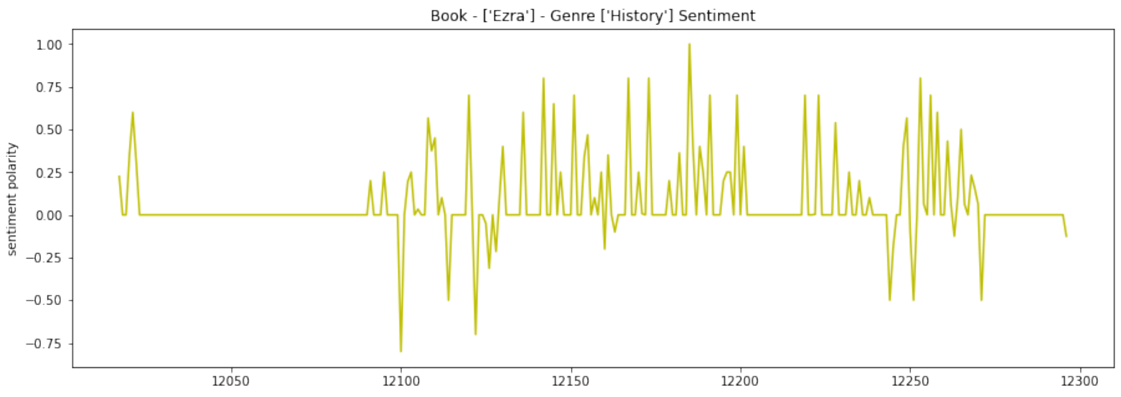

Book - ['2 Samuel'] - Genre ['History'] Sentiment
sentiment polarity

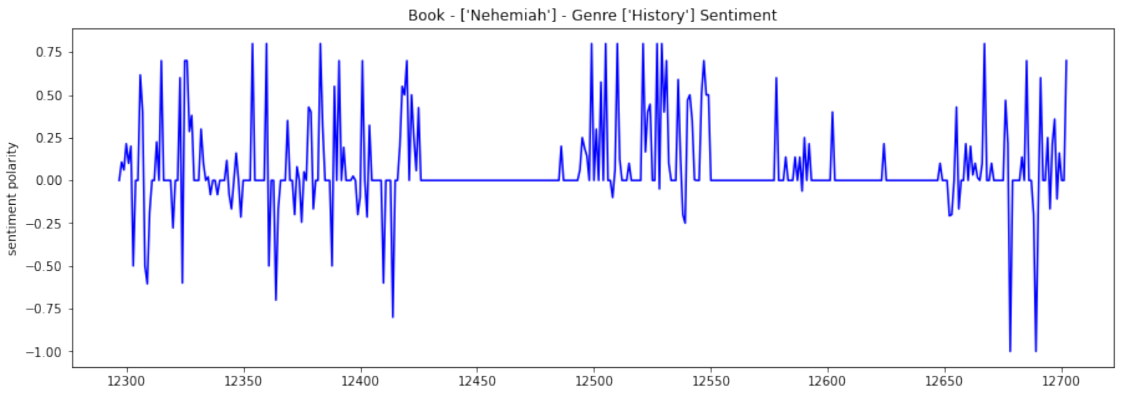

Book - ['1 Kings'] - Genre ['History'] Sentiment
sentiment polarity

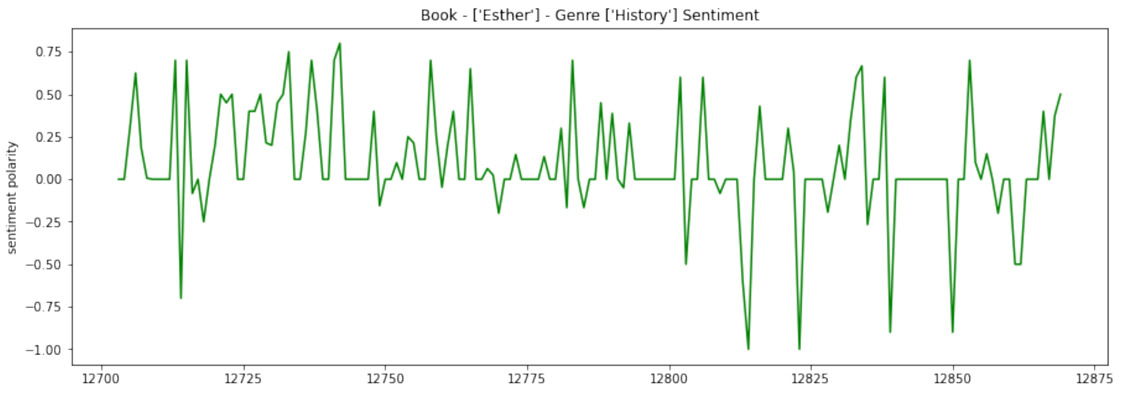

Book - ['2 Kings'] - Genre ['History'] Sentiment
sentiment polarity

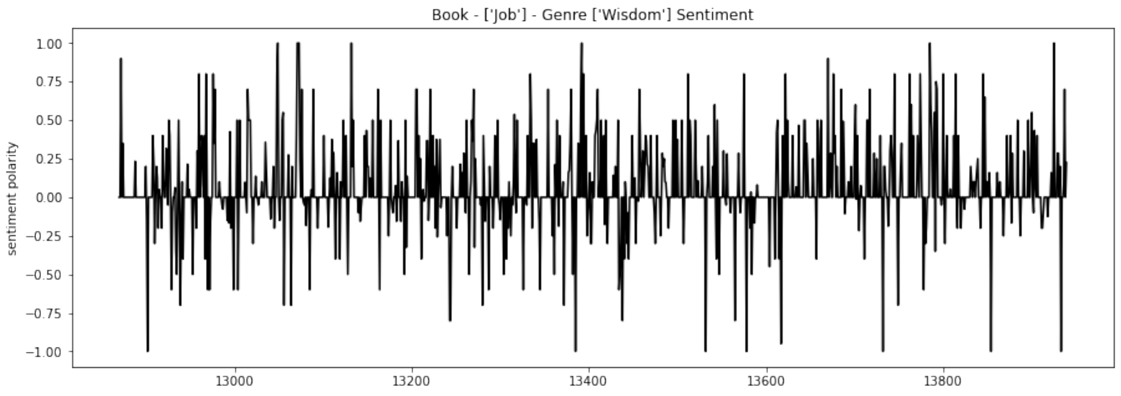

Book - ['1 Chronicles'] - Genre ['History'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
10400
10600
10800
11000
11200

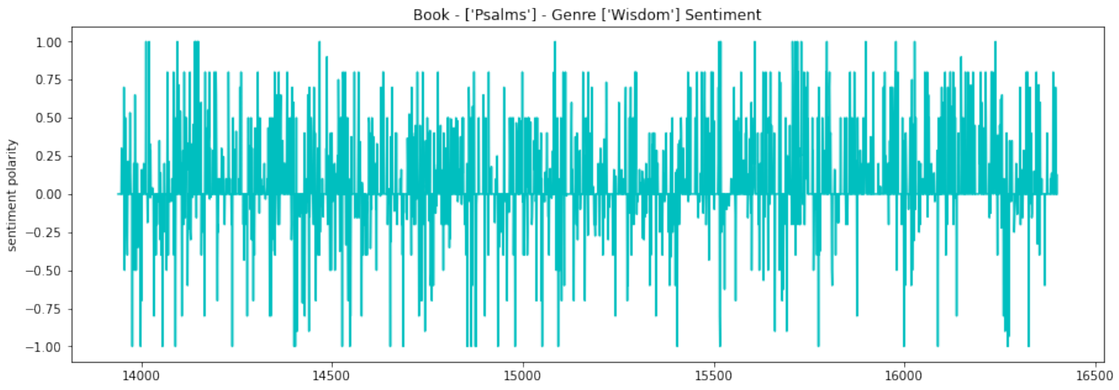

Book - ['2 Chronicles'] - Genre ['History'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
11200
11400
11600
11800
12000

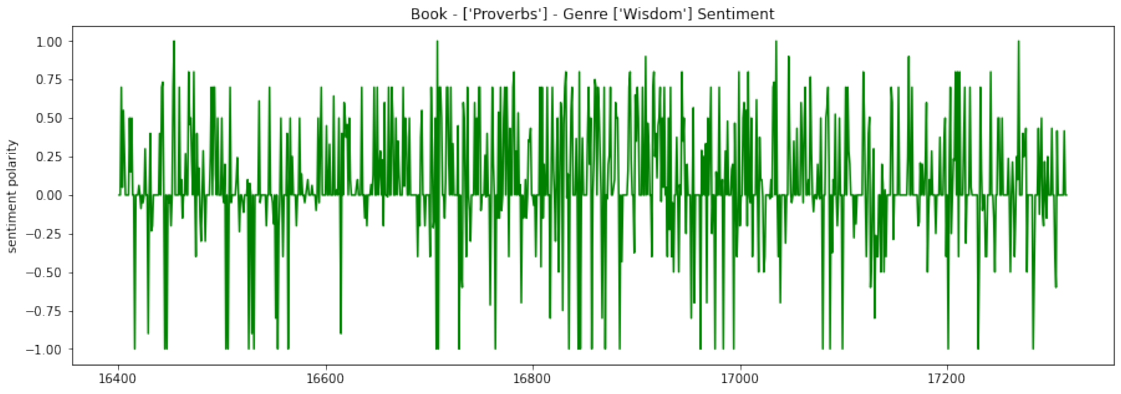

Book - ['Ezra'] - Genre ['History'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
12050
12100
12150
12200
12250
12300

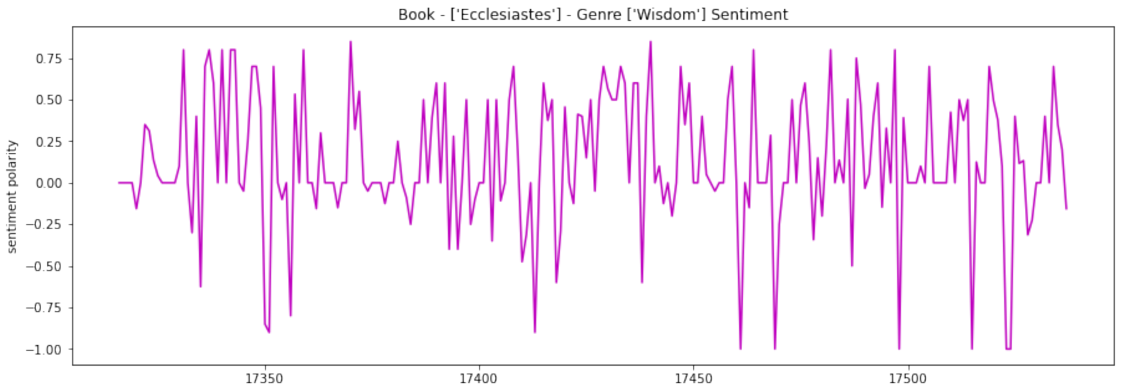

Book - ['Nehemiah'] - Genre ['History'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
12300
12350
12400
12450
12500
12550
12600
12650
12700

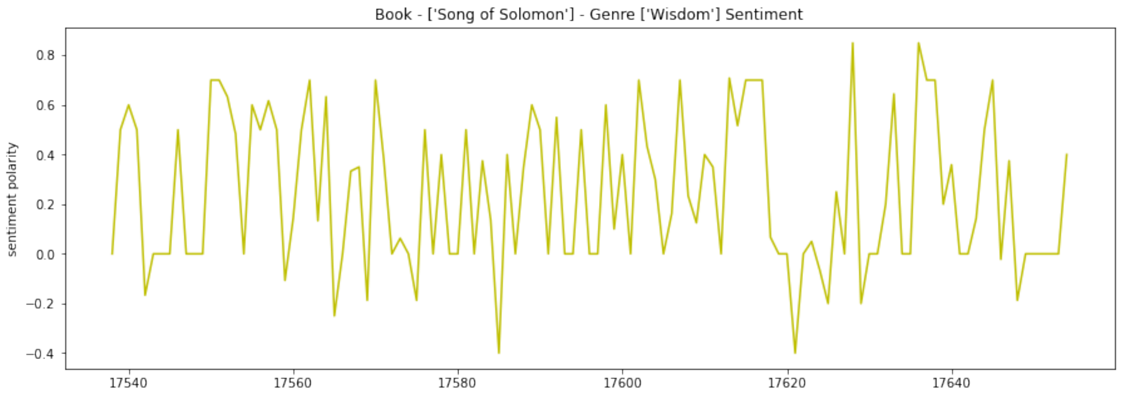

Book - ['Esther'] - Genre ['History'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
12700
12725
12750
12775
12800
12825
12850
12875

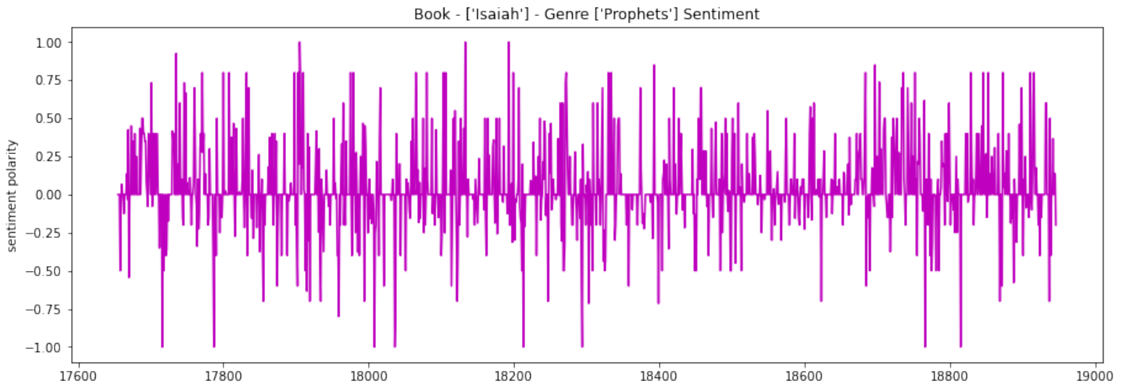

Book - ['Job'] - Genre ['Wisdom'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
13000
13200
13400
13600
13800

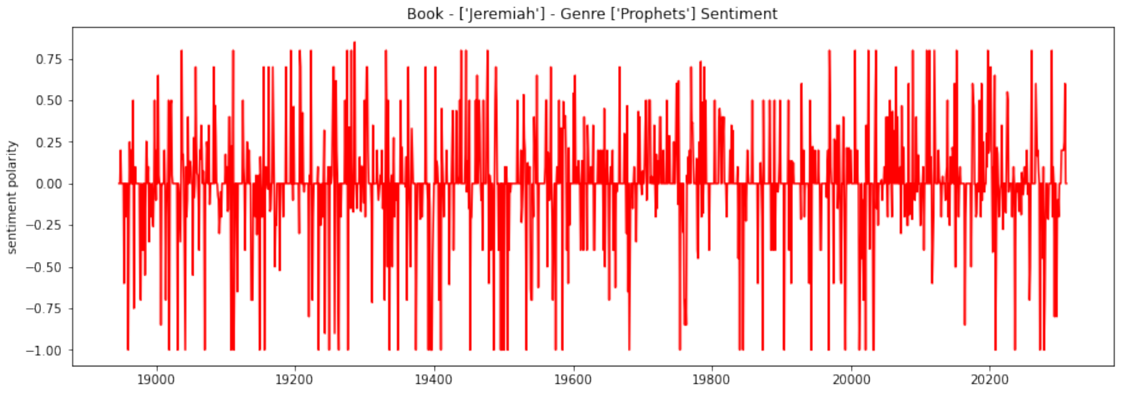

Book - ['Psalms'] - Genre ['Wisdom'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
14000
14500
15000
15500
16000
16500

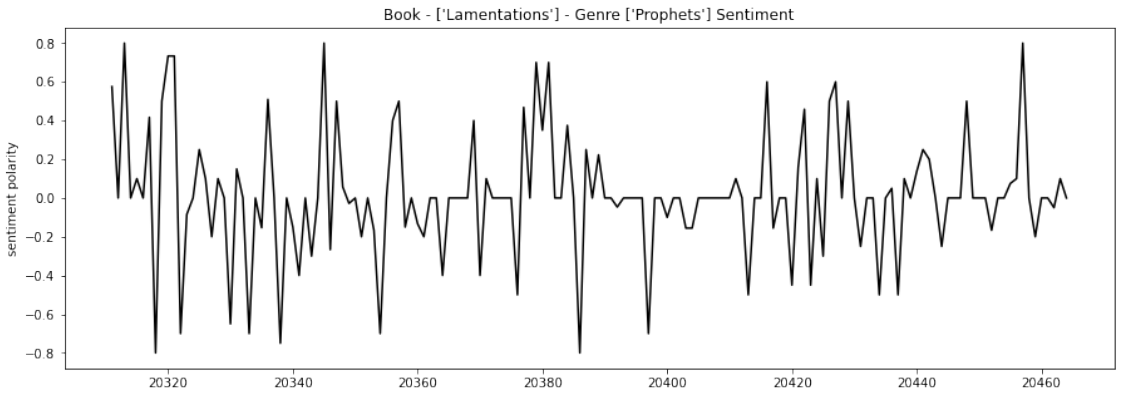

Book - ['Proverbs'] - Genre ['Wisdom'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
16400
16600
16800
17000
17200

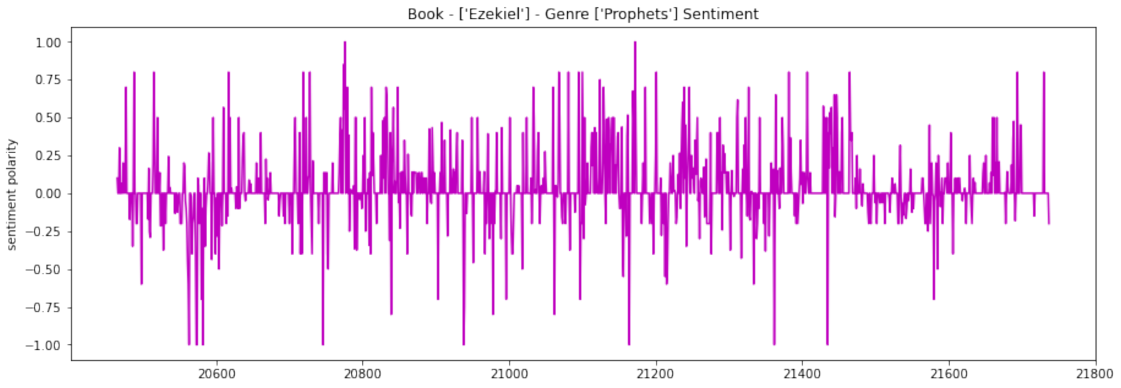

Book - ['Ecclesiastes'] - Genre ['Wisdom'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
17350
17400
17450
17500

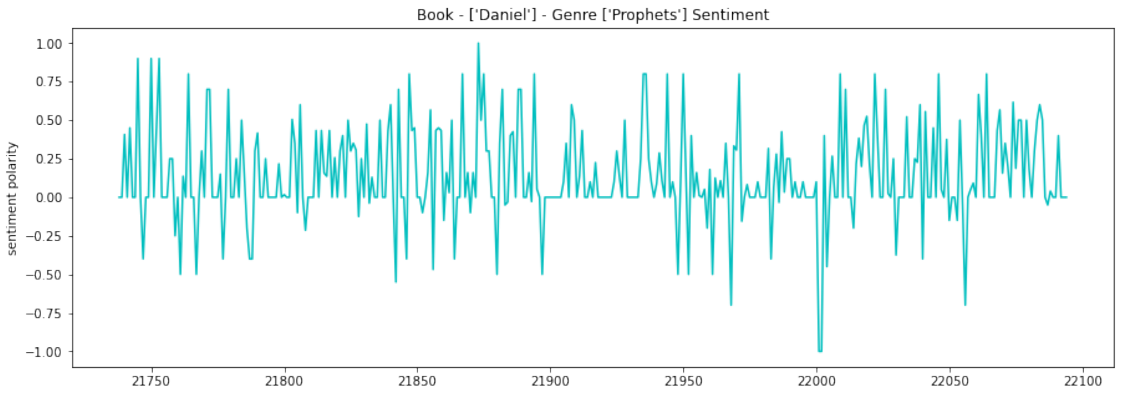

Book - ['Song of Solomon'] - Genre ['Wisdom'] Sentiment
sentiment polarity
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
17540
17560
17580
17600
17620
17640

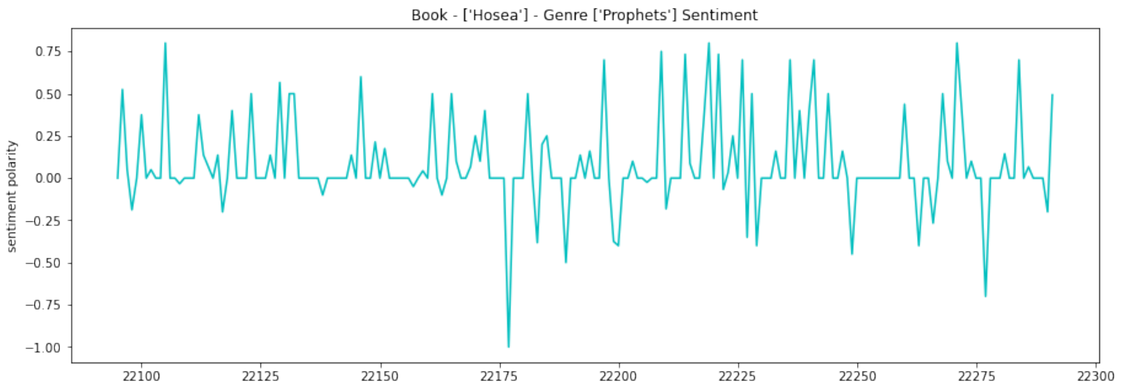

Book - ['Isaiah'] - Genre ['Prophets'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
17600
17800
18000
18200
18400
18600
18800
19000

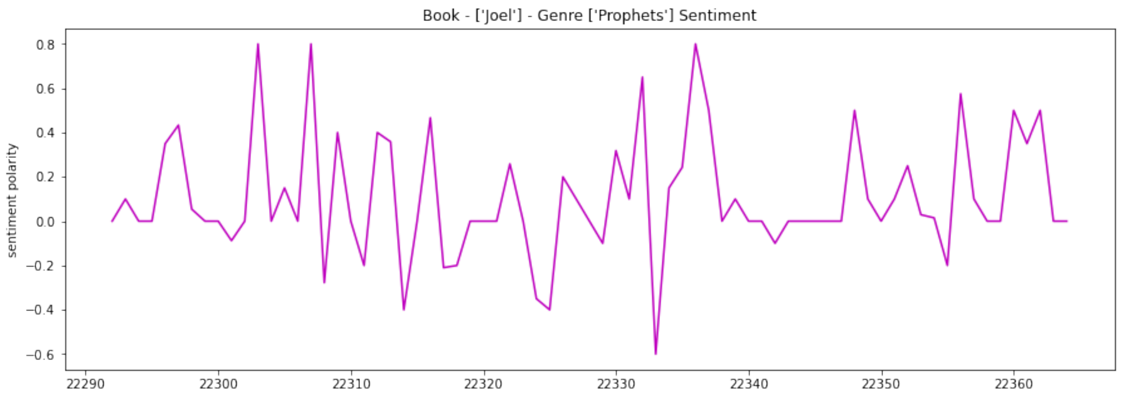

Book - ['Jeremiah'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
19000
19200
19400
19600
19800
20000
20200

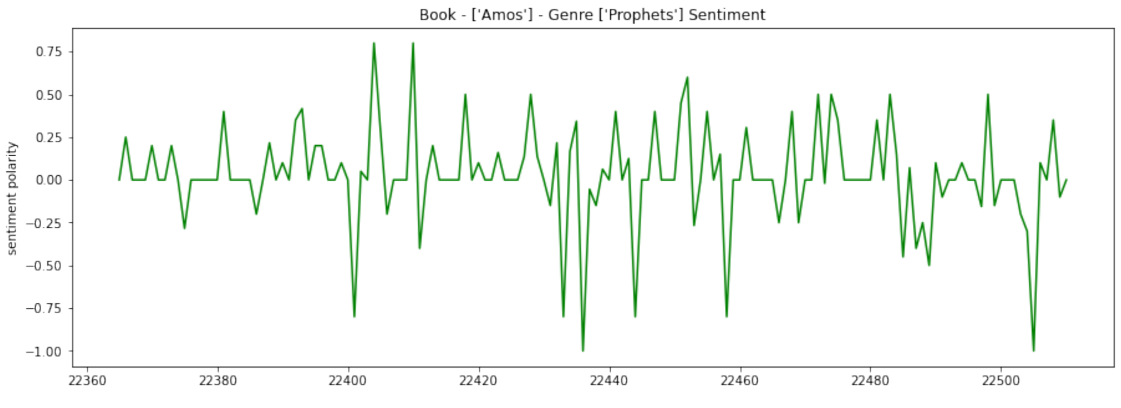

Book - ['Lamentations'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
20320
20340
20360
20380
20400
20420
20440
20460

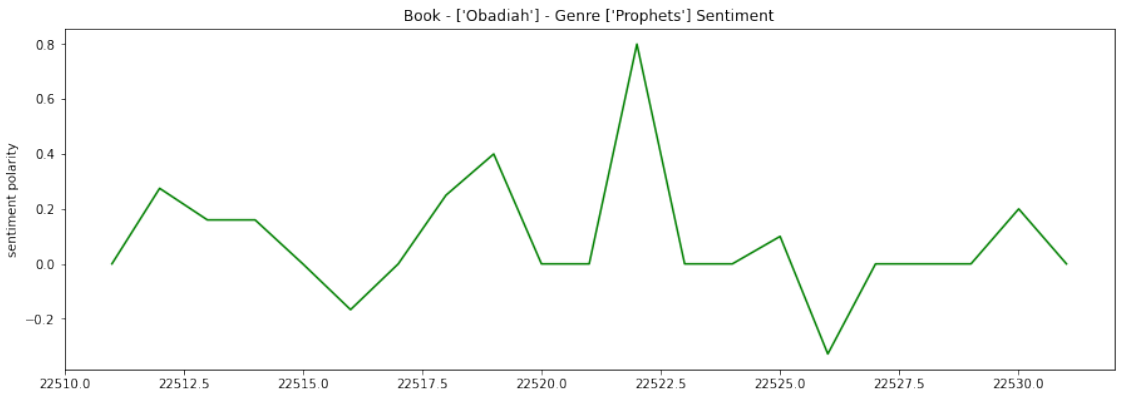

Book - ['Ezekiel'] - Genre ['Prophets'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
20600
20800
21000
21200
21400
21600
21800

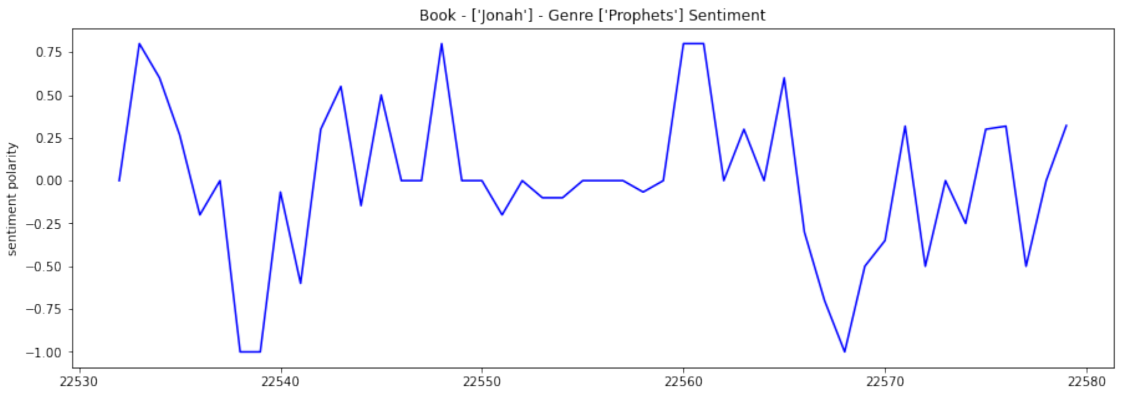

Book - ['Daniel'] - Genre ['Prophets'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
21750
21800
21850
21900
21950
22000
22050
22100

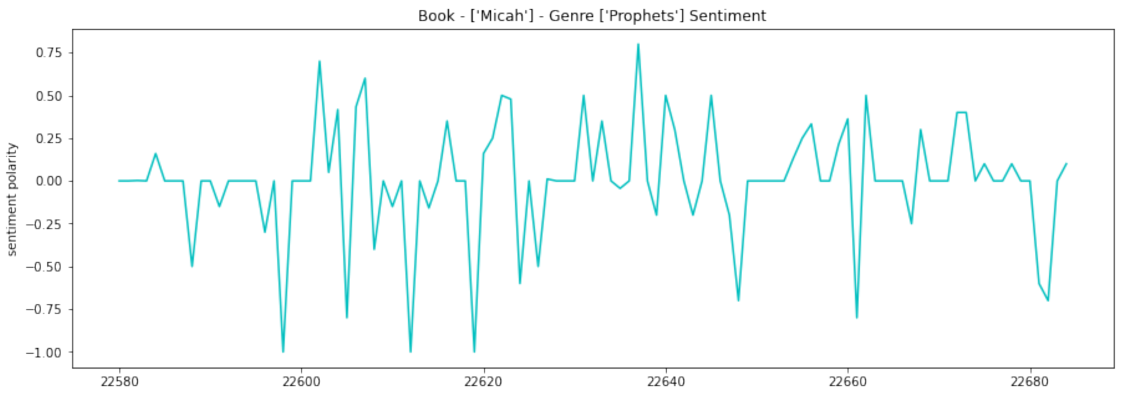

Book - ['Hosea'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22100
22125
22150
22175
22200
22225
22250
22275
22300

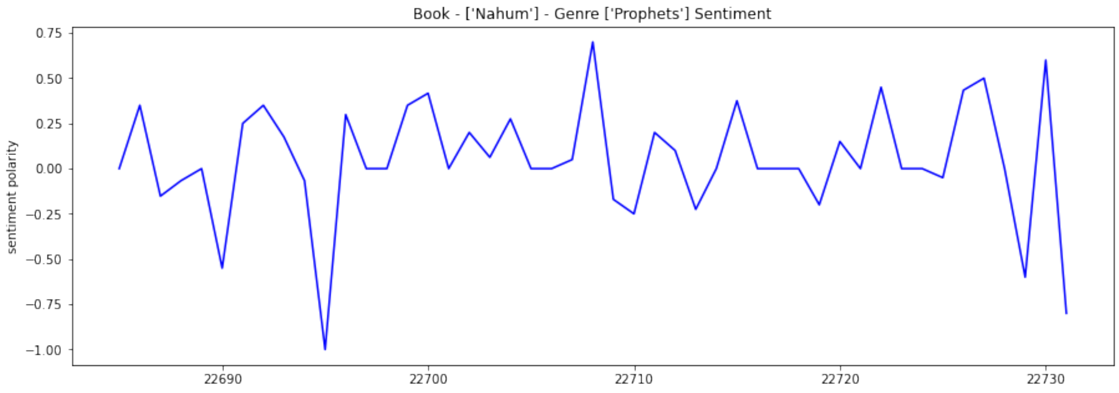

Book - ['Joel'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
22290
22300
22310
22320
22330
22340
22350
22360

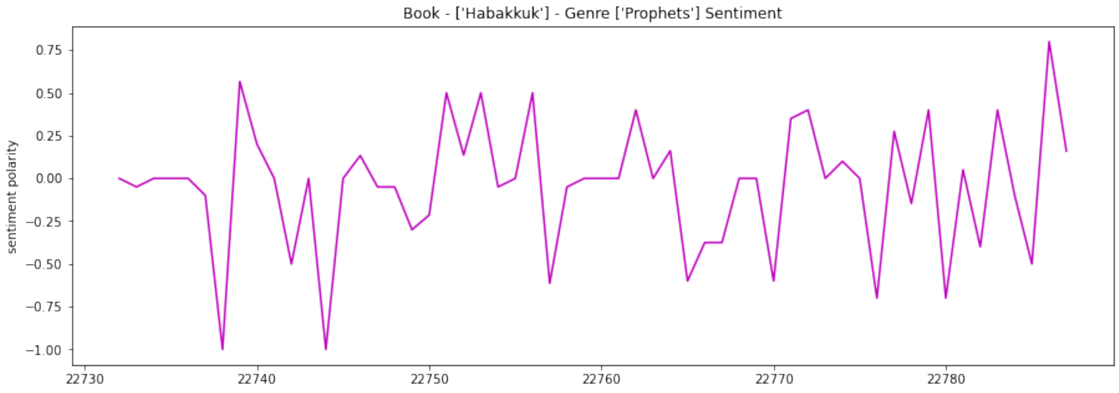

Book - ['Amos'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22360
22380
22400
22420
22440
22460
22480
22500

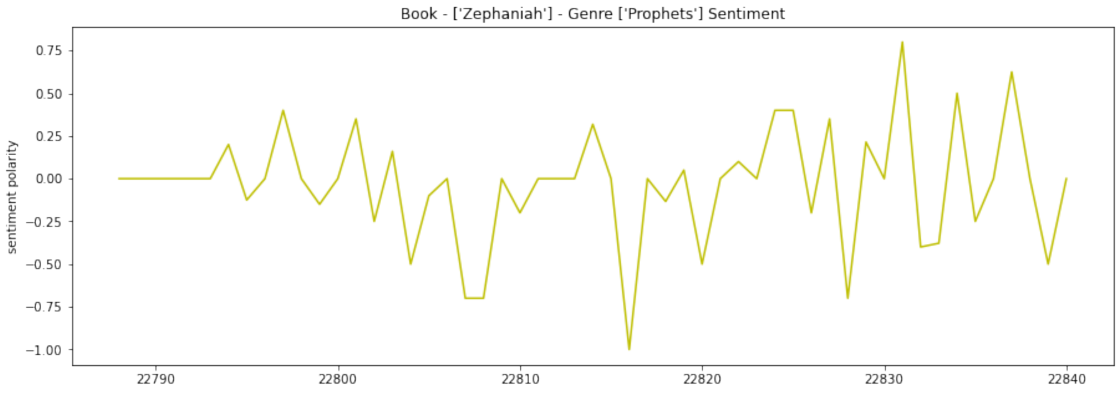

Book - ['Obadiah'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.8
0.6
0.4
0.2
0.0
−0.2
22510.0
22512.5
22515.0
22517.5
22520.0
22522.5
22525.0
22527.5
22530.0

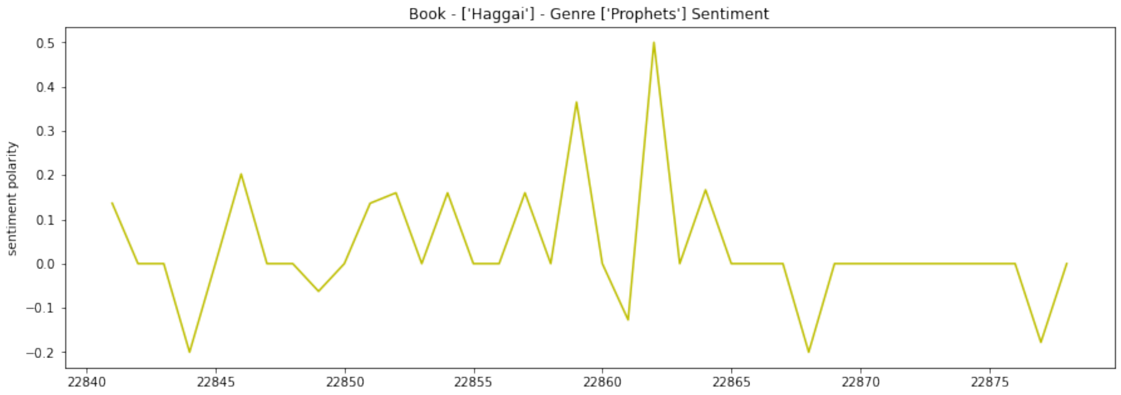

Book - ['Jonah'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22530
22540
22550
22560
22570
22580

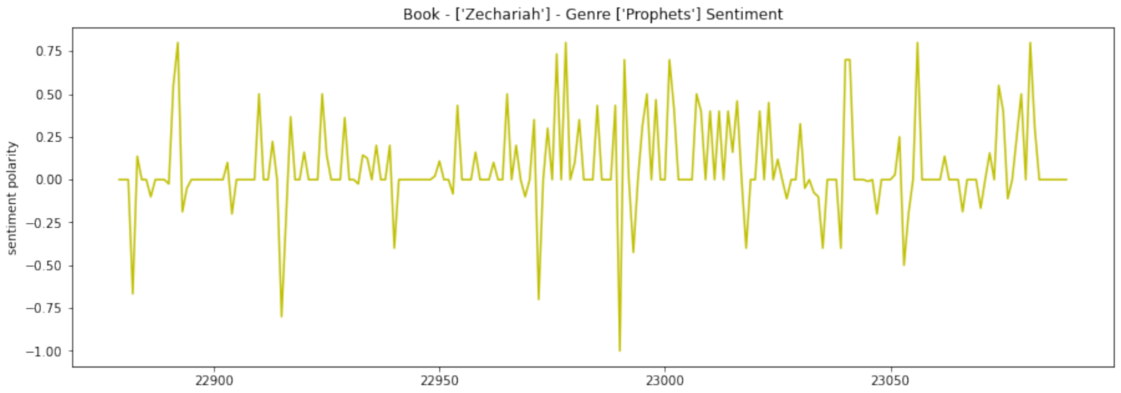

Book - ['Micah'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22580
22600
22620
22640
22660
22680

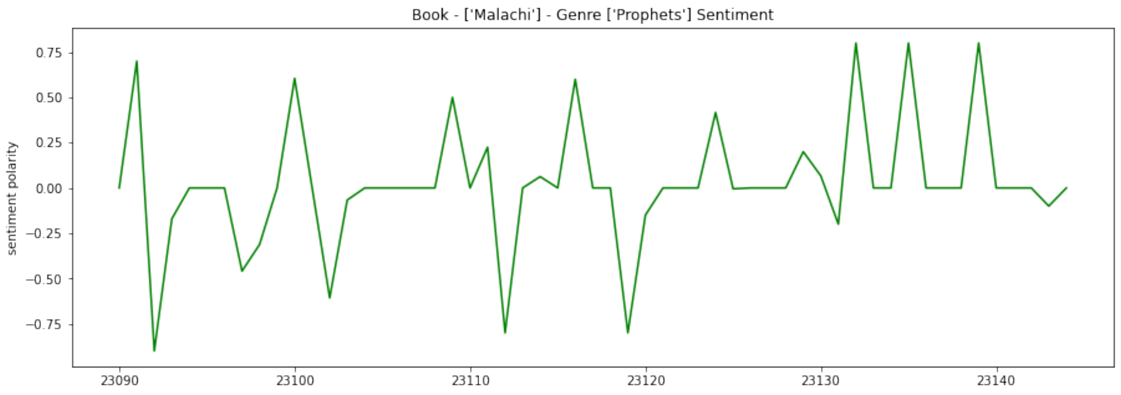

Book - ['Nahum'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22690
22700
22710
22720
22730

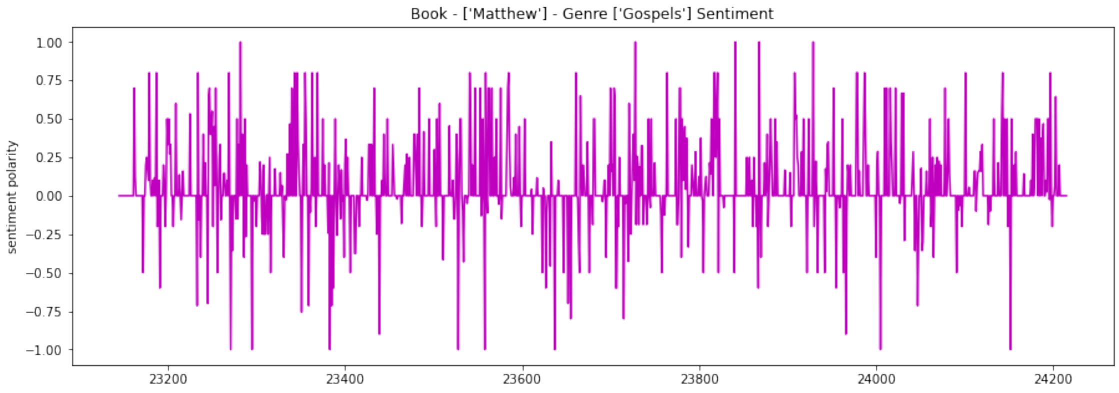

Book - ['Habakkuk'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22730
22740
22750
22760
22770
22780

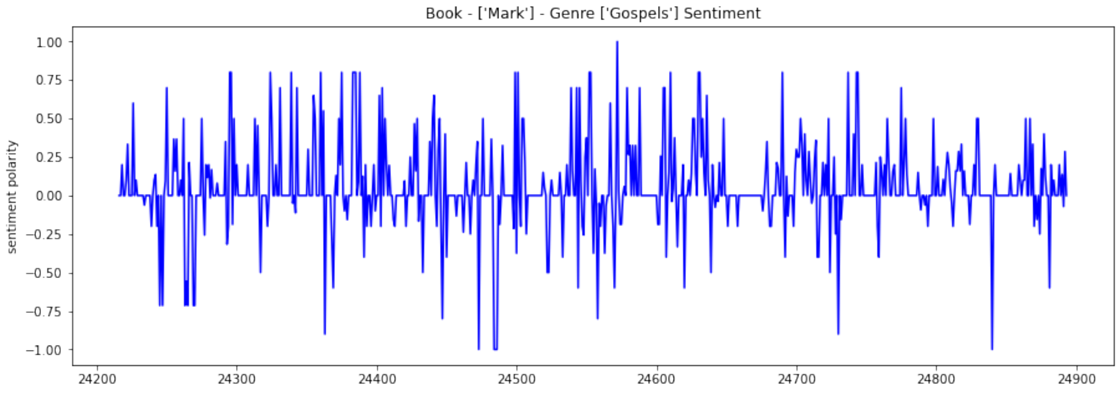

Book - ['Zephaniah'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22790
22800
22810
22820
22830
22840

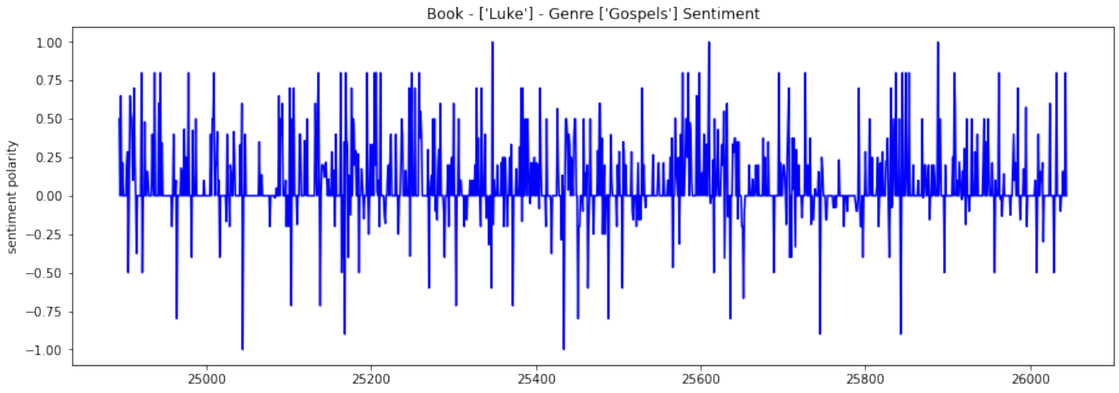

Book - ['Haggai'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
22840
22845
22850
22855
22860
22865
22870
22875

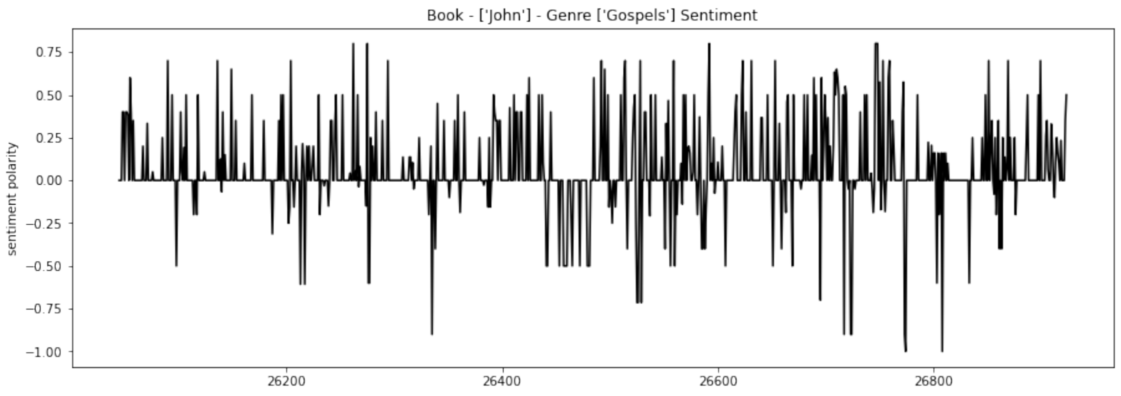

Book - ['Zechariah'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
22900
22950
23000
23050

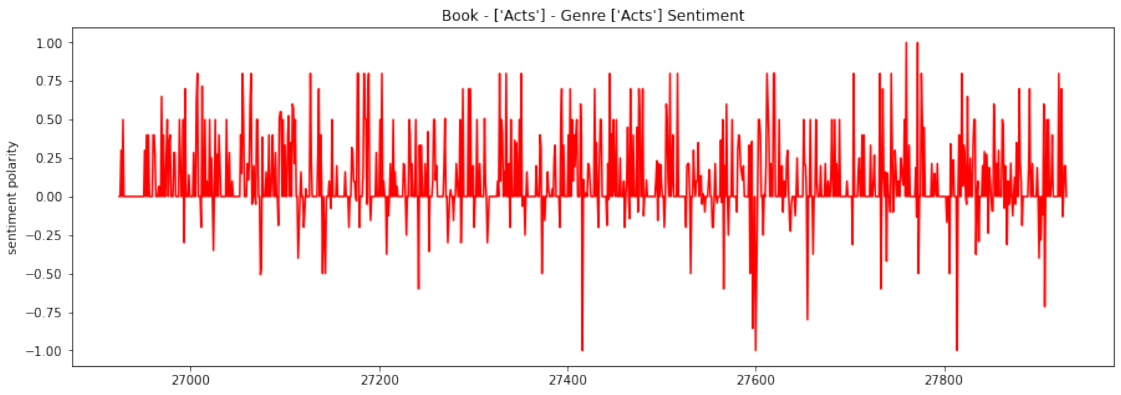

Book - ['Malachi'] - Genre ['Prophets'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
23090
23100
23110
23120
23130
23140

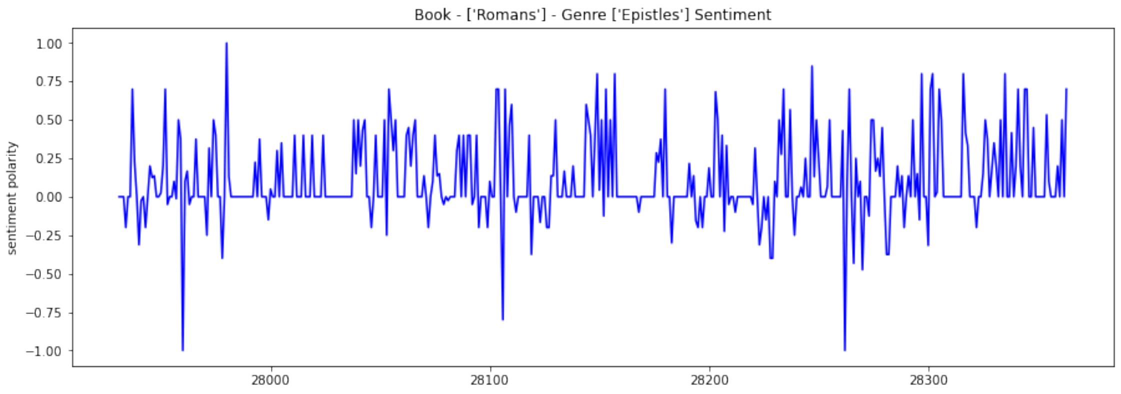

Book - ['Matthew'] - Genre ['Gospels'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
23200
23400
23600
23800
24000
24200

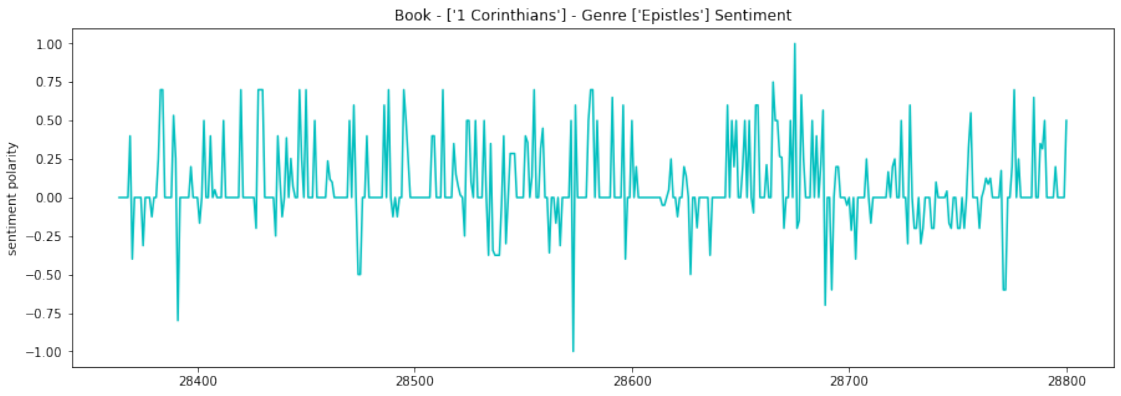

Book - ['Mark'] - Genre ['Gospels'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
24200
24300
24400
24500
24600
24700
24800
24900

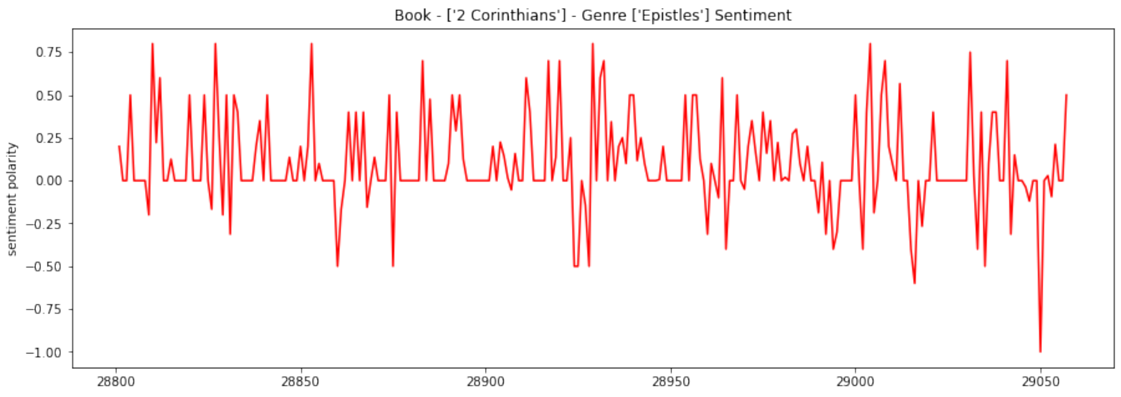

Book - ['Luke'] - Genre ['Gospels'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
25000
25200
25400
25600
25800
26000

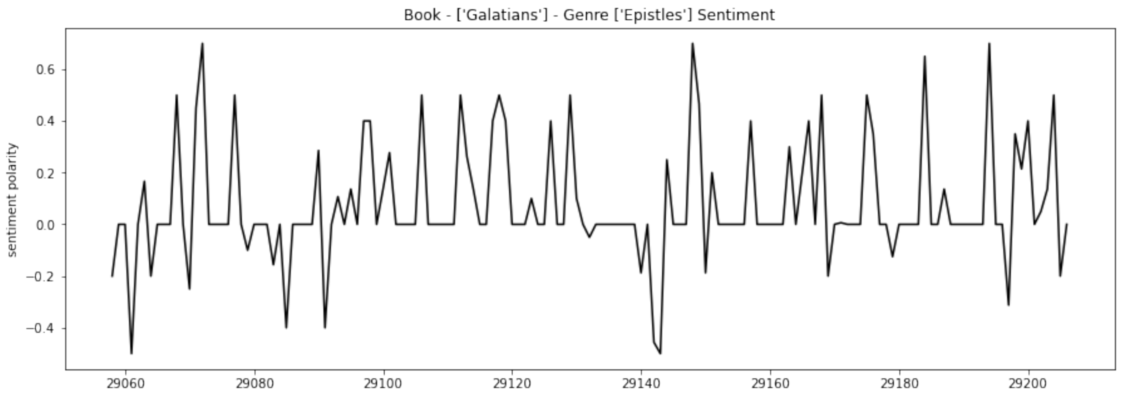

Book - ['John'] - Genre ['Gospels'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
26200
26400
26600
26800

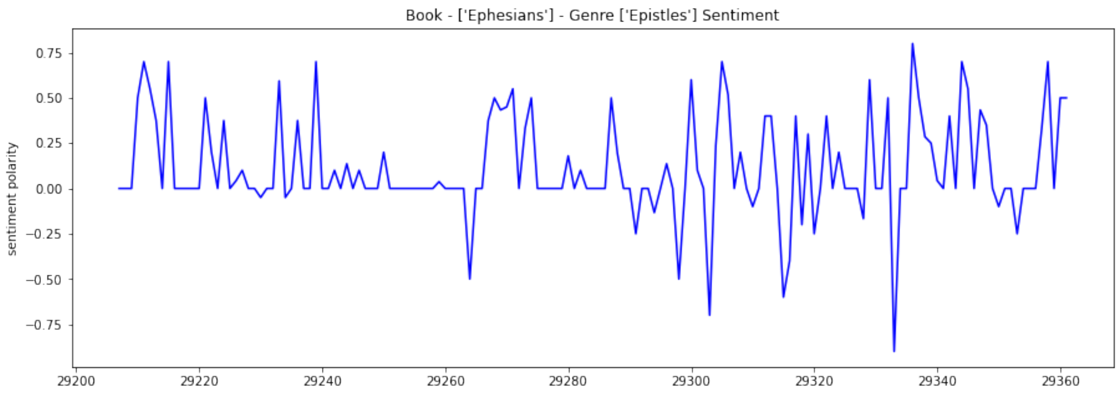

Book - ['Acts'] - Genre ['Acts'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
27000
27200
27400
27600
27800

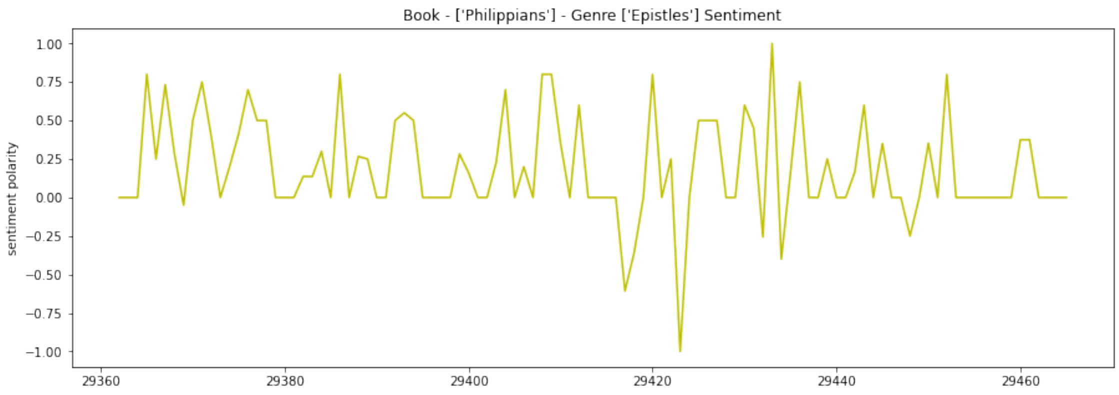

Book - ['Romans'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
28000
28100
28200
28300

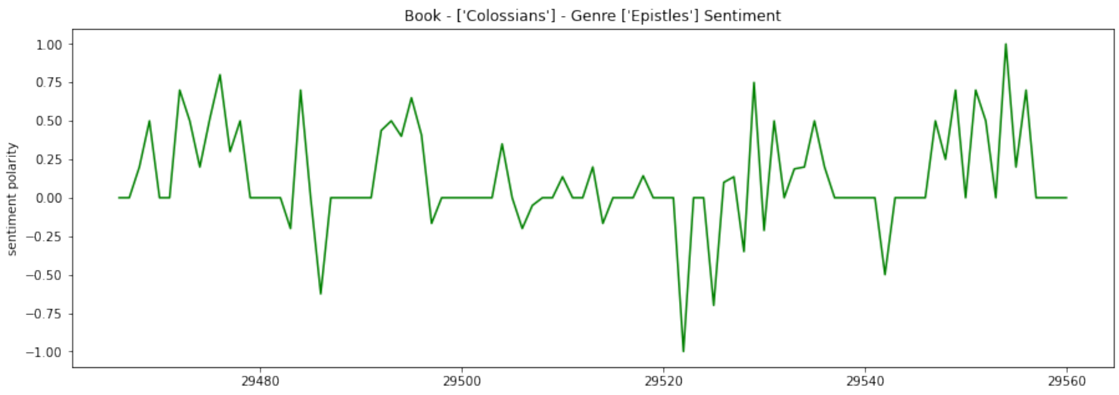

Book - ['1 Corinthians'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
28400
28500
28600
28700
28800

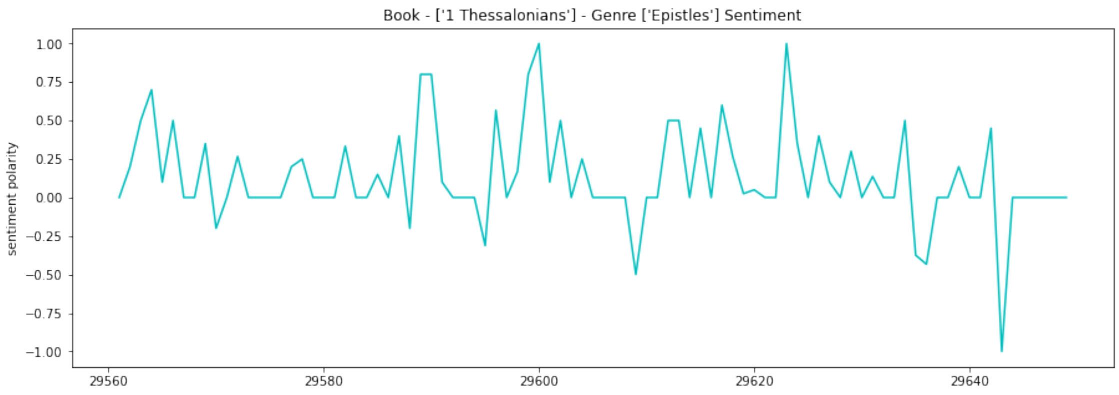

Book - ['2 Corinthians'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
28800
28850
28900
28950
29000
29050

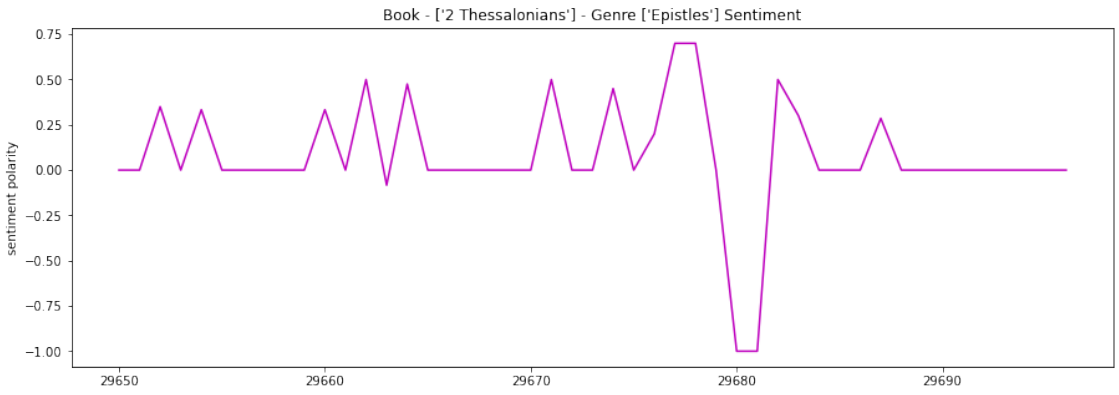

Book - ['Galatians'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.6
0.4
0.2
0.0
−0.2
−0.4
29060
29080
29100
29120
29140
29160
29180
29200

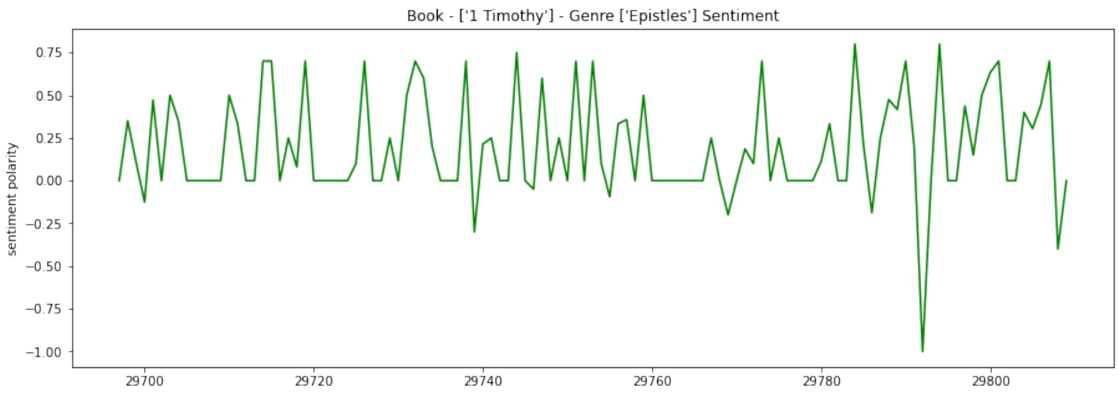

Book - ['Ephesians'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
29200
29220
29240
29260
29280
29300
29320
29340
29360

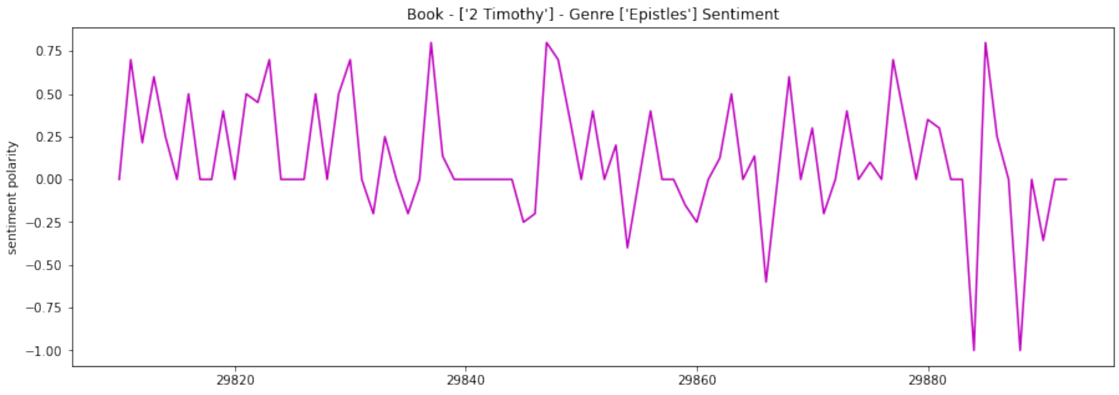

Book - ['Philippians'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29360
29380
29400
29420
29440
29460

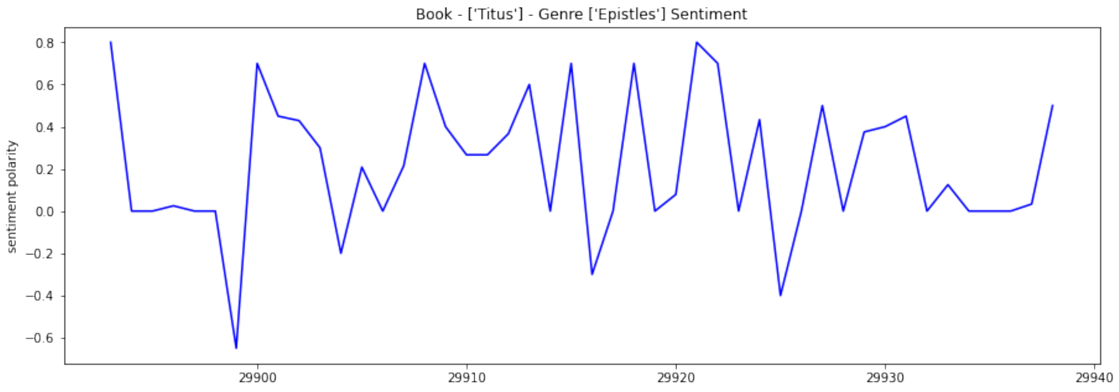

Book - ['Colossians'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29480
29500
29520
29540
29560

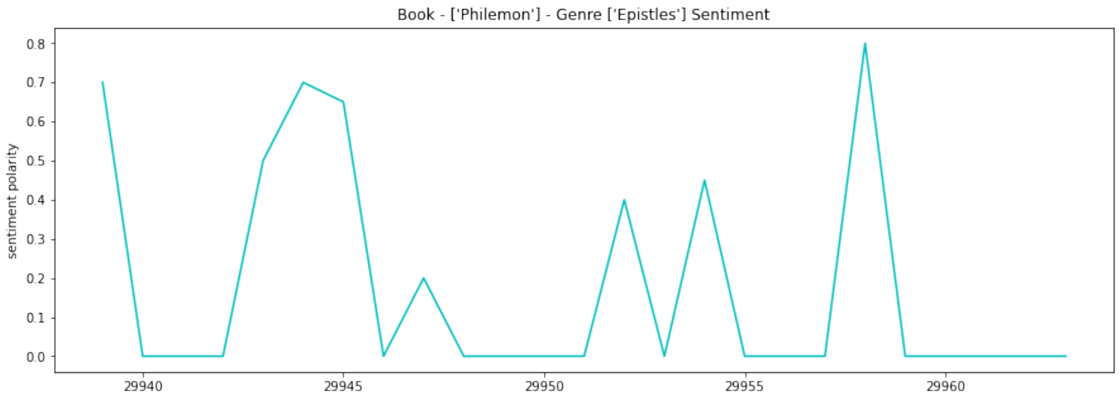

Book - ['1 Thessalonians'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29560
29580
29600
29620
29640

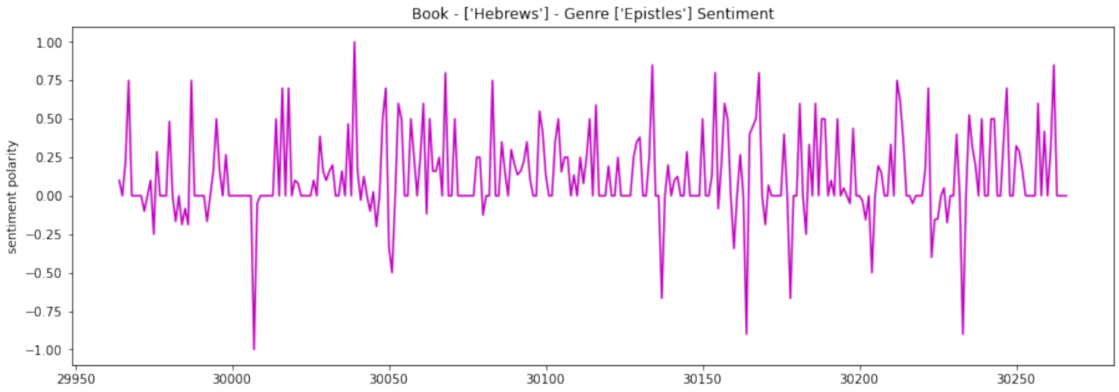

Book - ['2 Thessalonians'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29650
29660
29670
29680
29690

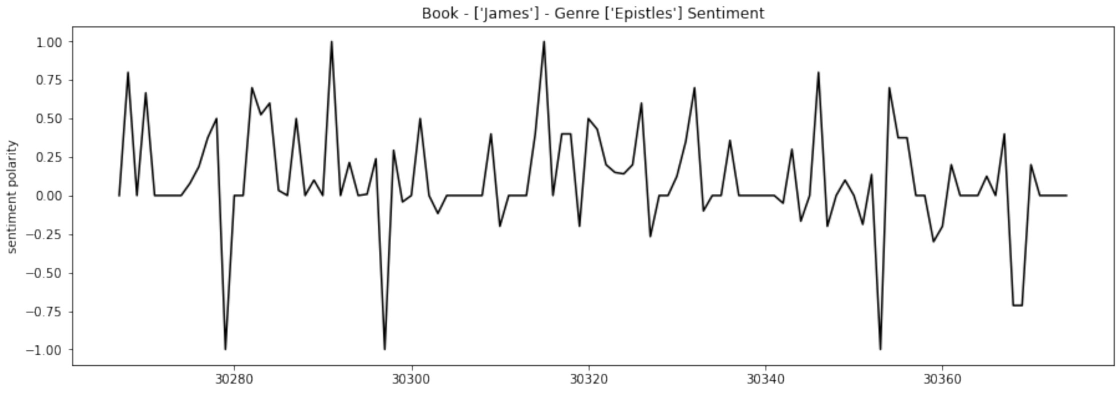

Book - ['1 Timothy'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29700
29720
29740
29760
29780
29800

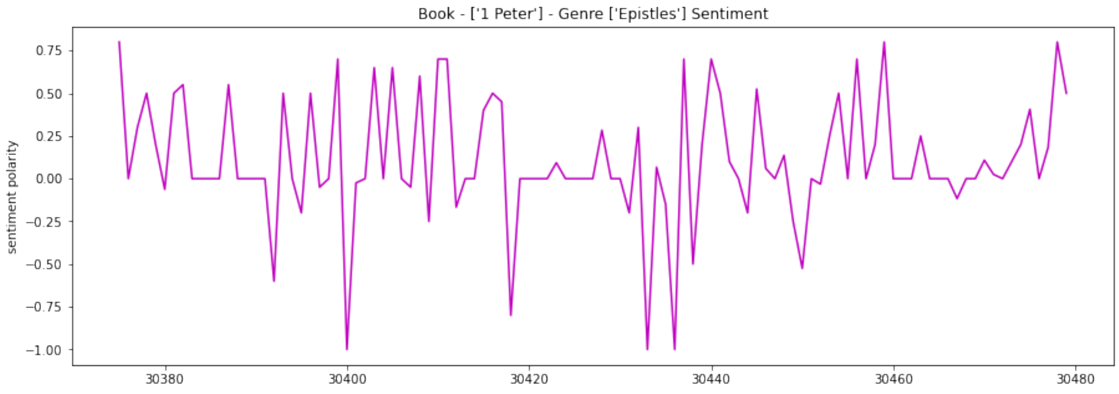

Book - ['2 Timothy'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29820
29840
29860
29880

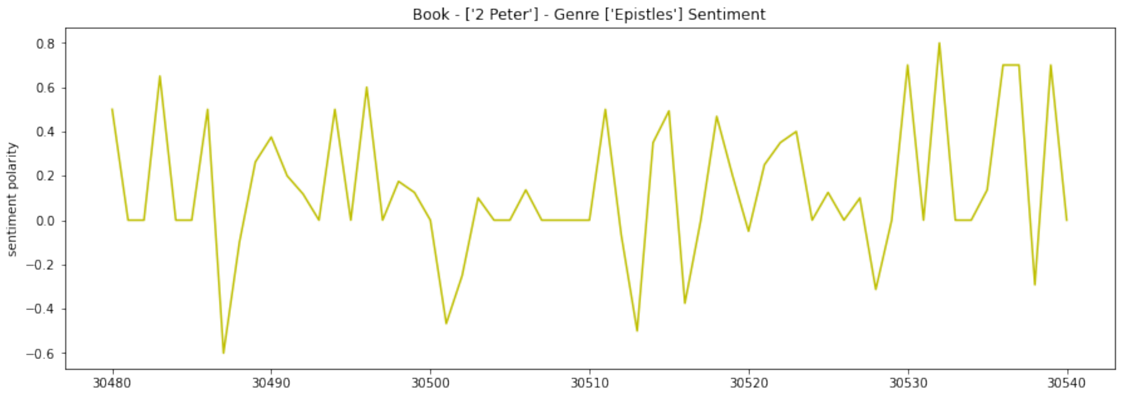

Book - ['Titus'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
29900
29910
29920
29930
29940

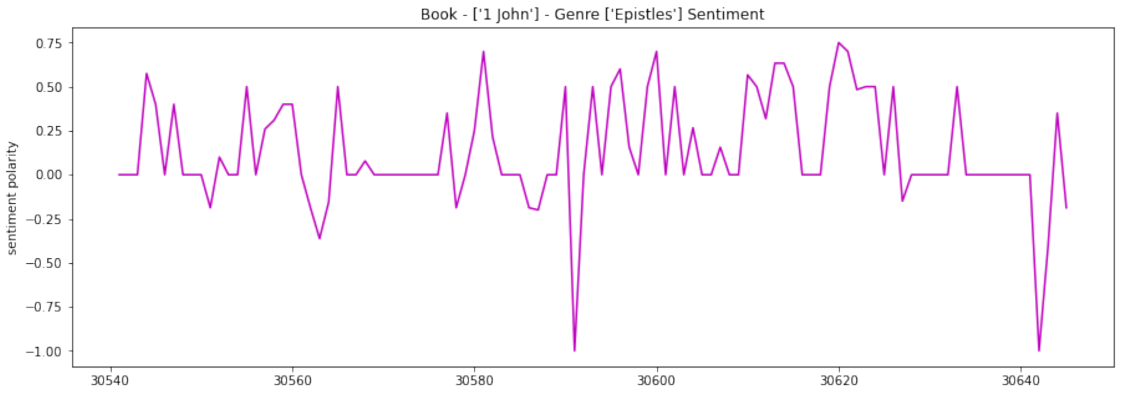

Book - ['Philemon'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
29940
29945
29950
29955
29960

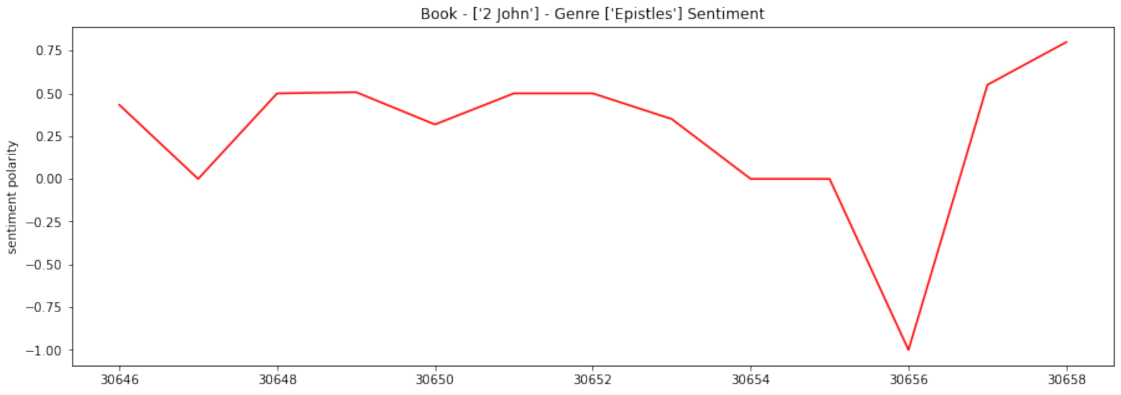

Book - ['Hebrews'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
29950
30000
30050
30100
30150
30200
30250

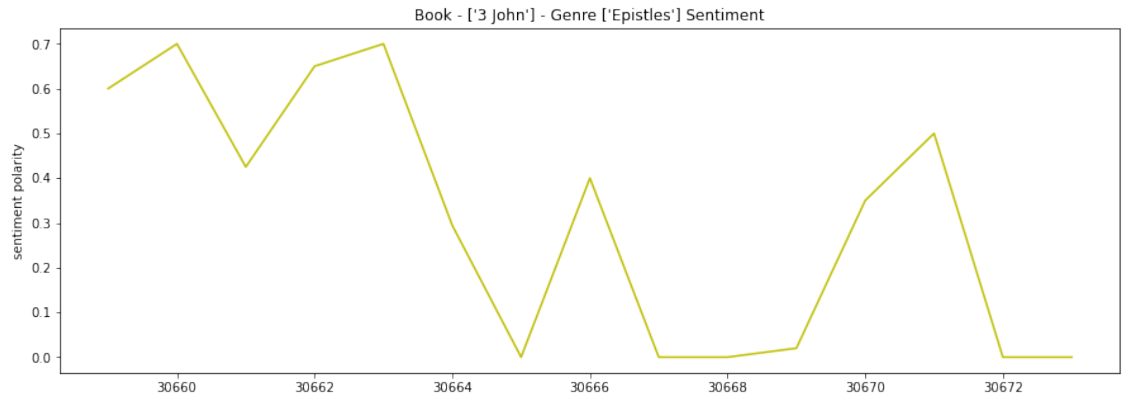

Book - ['James'] - Genre ['Epistles'] Sentiment
sentiment polarity
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
30280
30300
30320
30340
30360

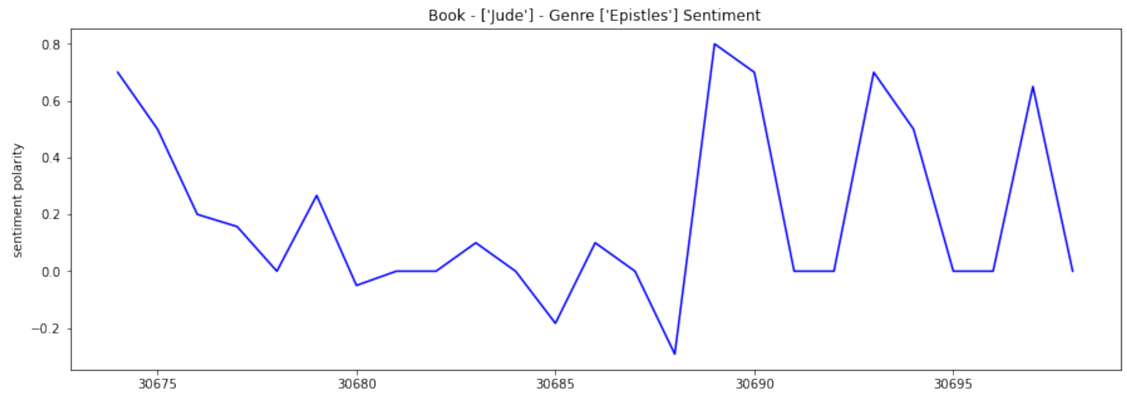

Book - ['1 Peter'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
30380
30400
30420
30440
30460
30480

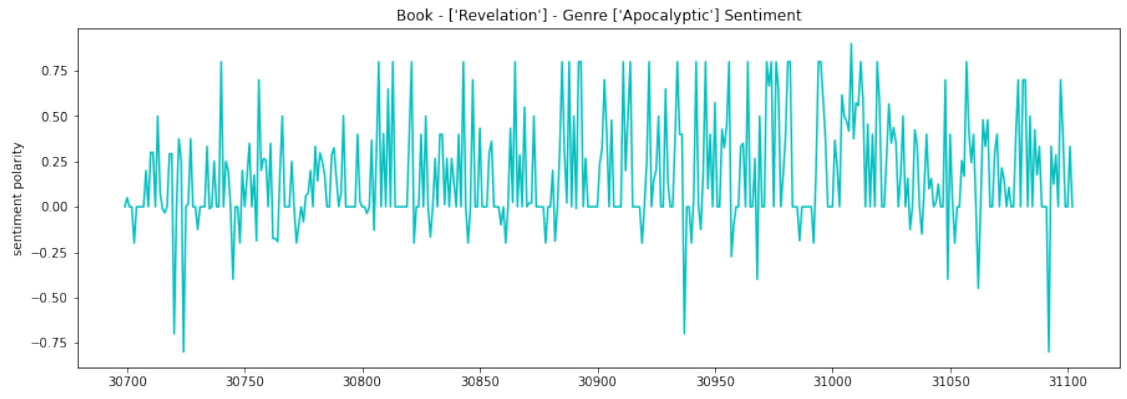

Book - ['2 Peter'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
30480
30490
30500
30510
30520
30530
30540


Book - ['1 John'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
30540
30560
30580
30600
30620
30640


Book - ['2 John'] - Genre ['Epistles'] Sentiment
sentiment polarity
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
30646
30648
30650
30652
30654
30656
30658



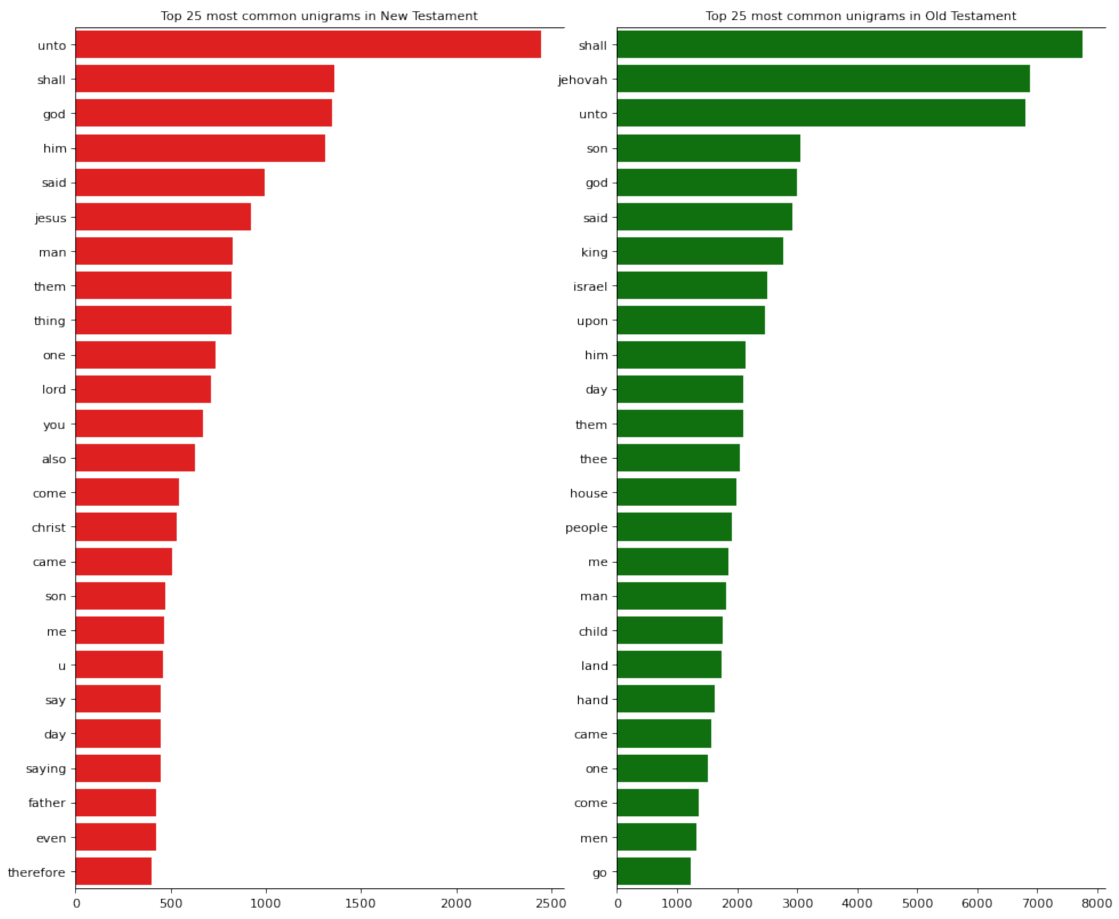




[71]:
```
OldTestament = df_Bible_keys[df_Bible_keys.book_ID < 40]
NewTestament = df_Bible_keys[df_Bible_keys.book_ID >= 40]
```

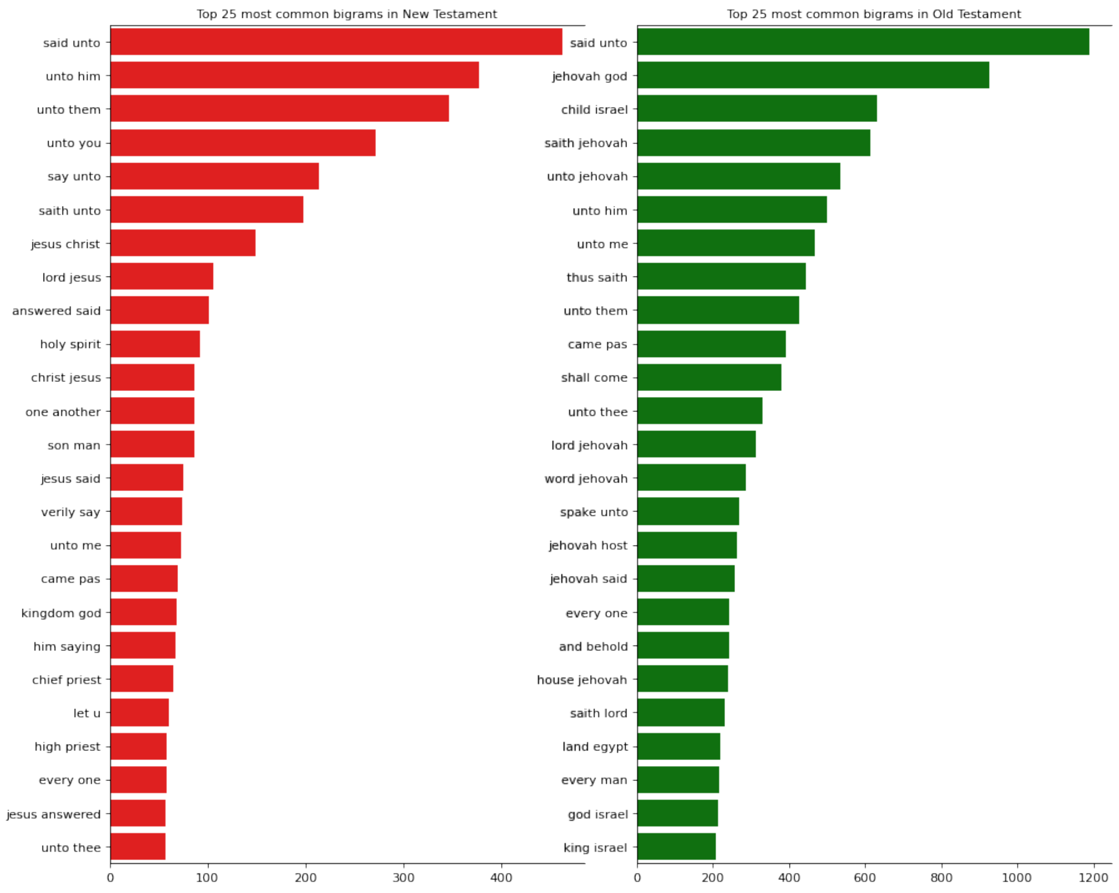
[79]:
```
def generate_ngrams(clean_text, n_gram=1):
    token = [token for token in clean_text.lower().split(' ') if token != '']
    ngrams = zip(*[token[i:] for i in range(n_gram)])
    return [' '.join(ngram) for ngram in ngrams]
```

[80]:
```
# Unigrams
NT_unigrams = defaultdict(int) #New Testament
OT_unigrams = defaultdict(int) #Old Testament

for line in NewTestament['clean_text']:
    for word in generate_ngrams(line):
        NT_unigrams[word] += 1

for line in OldTestament['clean_text']:
    for word in generate_ngrams(line):
        OT_unigrams[word] += 1

df_NT_unigrams = pd.DataFrame(sorted(NT_unigrams.items(), key=lambda x: x[1])[::
 ↪-1])
df_OT_unigrams = pd.DataFrame(sorted(OT_unigrams.items(), key=lambda x: x[1])[::
 ↪-1])

fig, axes = plt.subplots(ncols=2, figsize=(12, 10), dpi=80)
plt.tight_layout()

N = 25

sns.barplot(y=df_NT_unigrams[0].values[:N], x=df_NT_unigrams[1].values[:N],␣
 ↪ax=axes[0], color='red')
sns.barplot(y=df_OT_unigrams[0].values[:N], x=df_OT_unigrams[1].values[:N],␣
 ↪ax=axes[1], color='green')

for i in range(2):
    axes[i].spines['right'].set_visible(False)
    axes[i].set_xlabel('')
    axes[i].set_ylabel('')
    axes[i].tick_params(axis='x', labelsize=10)
    axes[i].tick_params(axis='y', labelsize=10)

axes[0].set_title(f'Top {N} most common unigrams in New Testament', fontsize=10)
axes[1].set_title(f'Top {N} most common unigrams in Old Testament', fontsize=10)

plt.show()
```

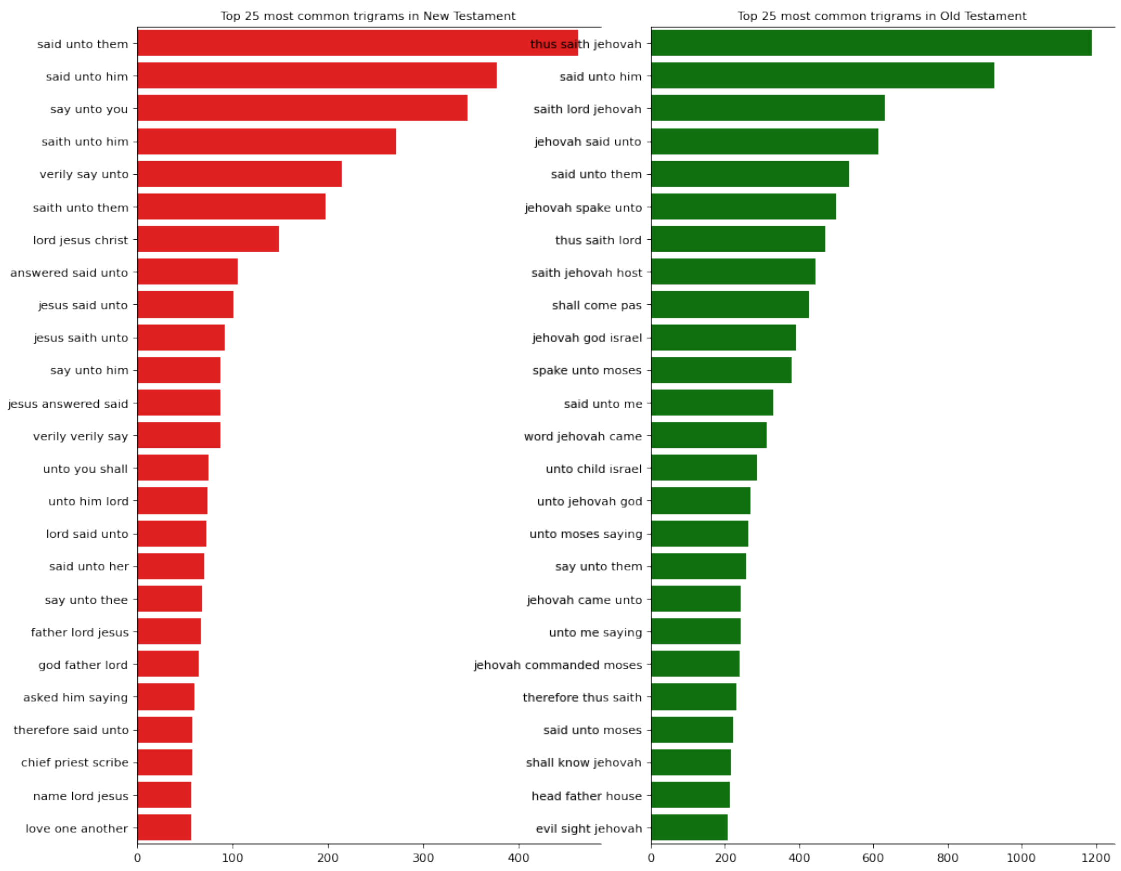


[83]:
```
# Bigrams
NT_bigrams = defaultdict(int)
OT_bigrams = defaultdict(int)

for line in NewTestament['clean_text']:
    for word in generate_ngrams(line, n_gram=2):
        NT_bigrams[word] += 1

for line in OldTestament['clean_text']:
    for word in generate_ngrams(line, n_gram=2):
        OT_bigrams[word] += 1

df_NT_bigrams = pd.DataFrame(sorted(NT_bigrams.items(), key=lambda x: x[1])[::
 ↪-1])
df_OT_bigrams = pd.DataFrame(sorted(OT_bigrams.items(), key=lambda x: x[1])[::
 ↪-1])

fig, axes = plt.subplots(ncols=2, figsize=(12, 10), dpi=80)
```

```
plt.tight_layout()

N = 25

sns.barplot(y=df_NT_bigrams[0].values[:N], x=df_NT_bigrams[1].values[:N],␣
  ↪ax=axes[0], color='red')
sns.barplot(y=df_OT_bigrams[0].values[:N], x=df_OT_bigrams[1].values[:N],␣
  ↪ax=axes[1], color='green')

for i in range(2):
    axes[i].spines['right'].set_visible(False)
    axes[i].set_xlabel('')
    axes[i].set_ylabel('')
    axes[i].tick_params(axis='x', labelsize=10)
    axes[i].tick_params(axis='y', labelsize=10)

axes[0].set_title(f'Top {N} most common bigrams in New Testament', fontsize=10)
axes[1].set_title(f'Top {N} most common bigrams in Old Testament', fontsize=10)

plt.show()
```



```
[84]: # Trigrams
NT_trigrams = defaultdict(int)
OT_trigrams = defaultdict(int)

for line in NewTestament['clean_text']:
    for word in generate_ngrams(line, n_gram=3):
        NT_trigrams[word] += 1

for line in OldTestament['clean_text']:
    for word in generate_ngrams(line, n_gram=3):
        OT_trigrams[word] += 1

df_NT_trigrams = pd.DataFrame(sorted(NT_trigrams.items(), key=lambda x: x[1])[::
 ↪-1])
df_OT_trigrams = pd.DataFrame(sorted(OT_trigrams.items(), key=lambda x: x[1])[::
 ↪-1])

fig, axes = plt.subplots(ncols=2, figsize=(12, 10), dpi=80)
plt.tight_layout()

N = 25

sns.barplot(y=df_NT_trigrams[0].values[:N], x=df_NT_bigrams[1].values[:N],␣
 ↪ax=axes[0], color='red')
sns.barplot(y=df_OT_trigrams[0].values[:N], x=df_OT_bigrams[1].values[:N],␣
 ↪ax=axes[1], color='green')

for i in range(2):
    axes[i].spines['right'].set_visible(False)
    axes[i].set_xlabel('')
    axes[i].set_ylabel('')
    axes[i].tick_params(axis='x', labelsize=10)
    axes[i].tick_params(axis='y', labelsize=10)

axes[0].set_title(f'Top {N} most common trigrams in New Testament', fontsize=10)
axes[1].set_title(f'Top {N} most common trigrams in Old Testament', fontsize=10)

plt.show()
```



```
[91]: vectorizer = TfidfVectorizer(max_df = 0.5, max_features = 1000)
      X = vectorizer.fit_transform(df_Bible_keys.text)

      svd = TruncatedSVD(n_components=2, n_iter=7, random_state=42)
      X = svd.fit_transform(X)

      cluster_data = pd.DataFrame({'Comp1': X[:,0], 'Comp2': X[:,1], 'Testament':␣
       ↪df_Bible_keys.testament, 'Book': df_Bible_keys.book, 'Genre': df_Bible_keys.
       ↪genre, 'Sentiment': df_Bible_keys.sentiment})
      cluster_data.head()
```

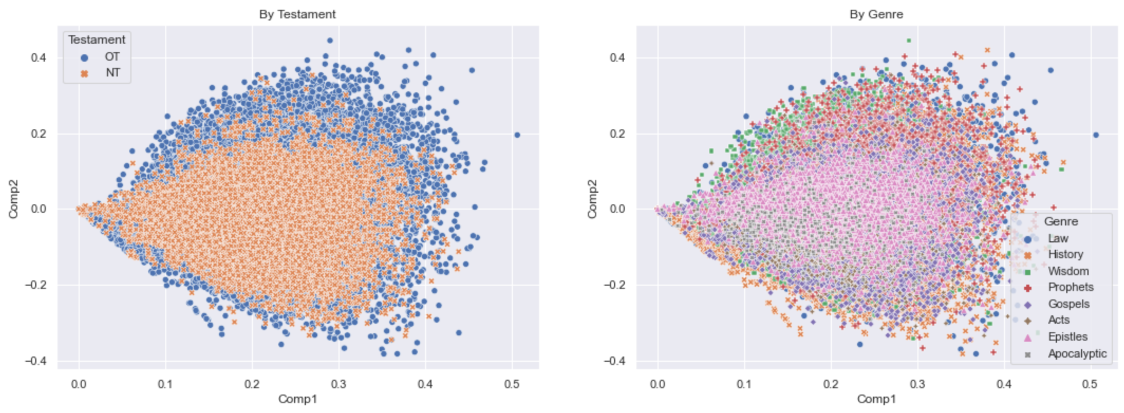
```
[91]:       Comp1     Comp2 Testament     Book Genre Sentiment
      0  0.093172  0.004119        OT  Genesis   Law    0.0000
      1  0.092076 -0.053957        OT  Genesis   Law   -0.1000
      2  0.137198 -0.006470        OT  Genesis   Law    0.4000
      3  0.147214 -0.034785        OT  Genesis   Law    0.5000
      4  0.144770 -0.116760        OT  Genesis   Law    0.4000
```

```
[92]: f, axes = plt.subplots(1, 2, figsize=(18, 6))
```

```
sns.scatterplot('Comp1', 'Comp2', data=cluster_data, hue='Testament',␣
 ↪ax=axes[0], style="Testament").set_title('By Testament')
sns.scatterplot('Comp1', 'Comp2', data=cluster_data, hue='Genre', ax=axes[1],␣
 ↪style="Genre").set_title('By Genre')
```

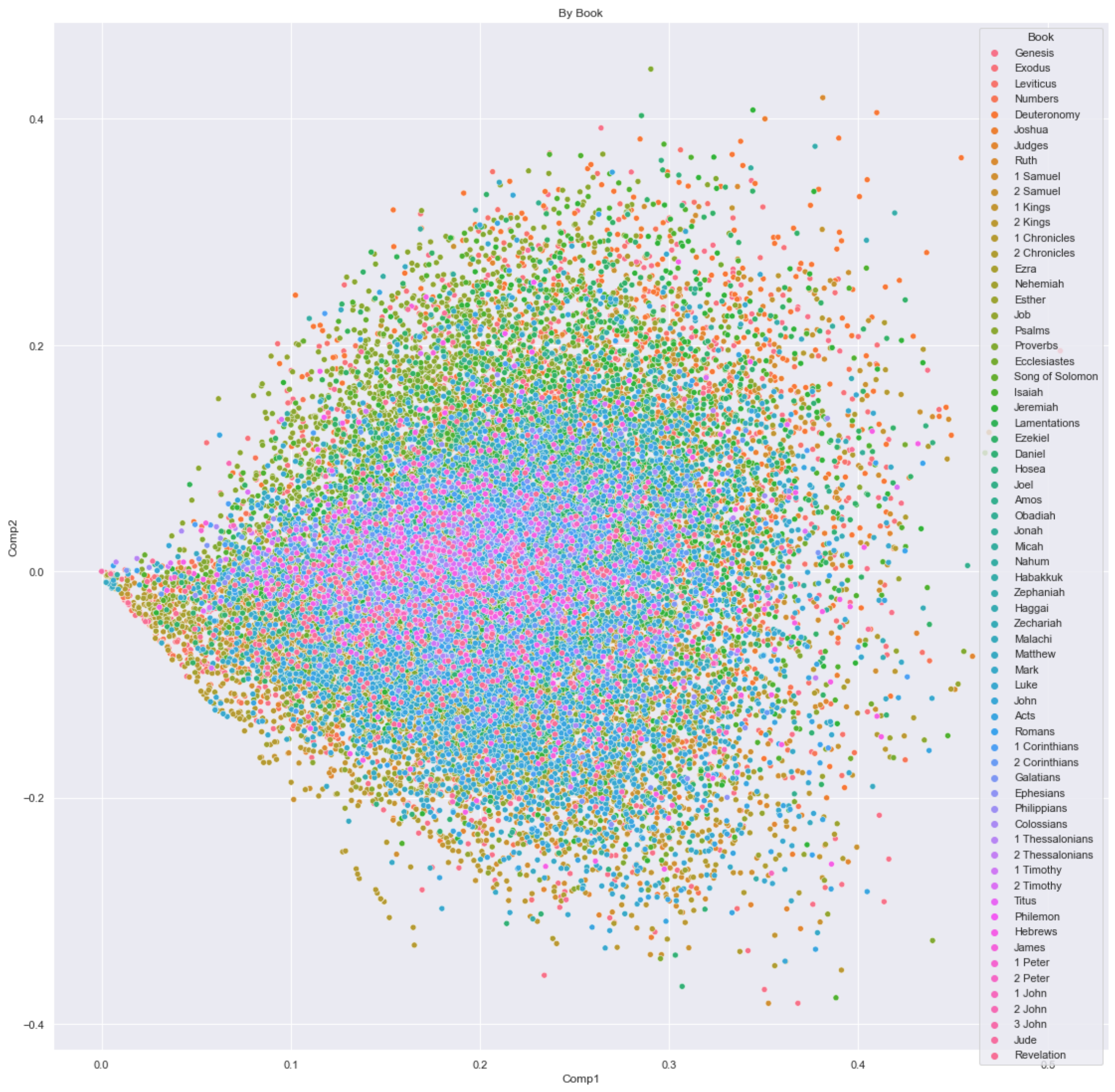
```
/Users/santiagolampon/opt/anaconda3/lib/python3.9/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/santiagolampon/opt/anaconda3/lib/python3.9/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
```

[92]: Text(0.5, 1.0, 'By Genre')



```
[93]: sns.set(rc={'figure.figsize':(19, 19)})
sns.scatterplot('Comp1', 'Comp2', data=cluster_data, hue='Book').set_title('By␣
 ↪Book')
plt.show()
```

```
/Users/santiagolampon/opt/anaconda3/lib/python3.9/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
```


By Book
Comp1
Comp2
0.0
0.1
0.2
0.3
0.4
0.5
0.4
0.2
0.0
−0.2
−0.4
Book
Genesis
Exodus
Leviticus
Numbers
Deuteronomy
Joshua
Judges
Ruth
1 Samuel
2 Samuel
1 Kings
2 Kings
1 Chronicles
2 Chronicles
Ezra
Nehemiah
Esther
Job
Psalms
Proverbs
Ecclesiastes
Song of Solomon
Isaiah
Jeremiah
Lamentations
Ezekiel
Daniel
Hosea
Joel
Amos
Obadiah
Jonah
Micah
Nahum
Habakkuk
Zephaniah
Haggai
Zechariah
Malachi
Matthew
Mark
Luke
John
Acts
Romans
1 Corinthians
2 Corinthians
Galatians
Ephesians
Philippians
Colossians
1 Thessalonians
2 Thessalonians
1 Timothy
2 Timothy
Titus
Philemon
Hebrews
James
1 Peter
2 Peter
1 John
2 John
3 John
Jude
Revelation

```
[ ]:
```