Visualizing Word Frequencies with Pandas Library

Abdulaziz M. Alqumayzi

(DS-540) – Intro to python for computer science and data science: Learning to program with AI, Big Data and The Cloud

Colorado State University – Global Campus

Dr. Ernest Bonat

November 11, 2020

Visualizing Word Frequencies with Pandas Library

Introduction

In this critical thinking four activity, using the Pandas Library, we can visualize word frequencies. The word frequencies refer to one of many strategies focused on word frequencies to identify similarity between documents. Which is counting how much in a corpus each word appears. The text that we will explore is retrieved from the bio personal life of the NBA legendary basketball player LeBron James.

Discussion

Import and download necessary libraries.

All libraries were imported, and corpuses were downloaded to do this activity in the following figure 1. *nltk* is a library for natural language processing (NLP). files library was used to upload local PERSONAL_LIFE text file into colab.

Figure 1

import libraries and download corpuses code

```
import nltk
from textblob import TextBlob
from pathlib import Path
from operator import itemgetter
import pandas as pd
from google.colab import files
from nltk.corpus import stopwords
nltk.download("stopwords")
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
nltk.download('brown')
nltk.download('wordnet')
```

Upload the text file from the local device into colab.

In figure 2, after executing the code two buttons will appear. Choose Files and Cancel upload. We must choose the PERSONAL_LIFE file to upload it to colab.

Figure 2

upload text file into colab code

```
#upload the PERSONAL_LIFE file then continue
upload_text = files.upload()
```

Note. In figure 2, file name must be the same "PERSONAL_LIFE"

Loading the data.

Code in figure 3 shows the creation of a TextBlob class holding the text we uploaded in figure 2.

Figure 3

create TextBlob class code

```
blob = TextBlob(Path('PERSONAL_LIFE.txt').read_text())
```

Loading NLTK stop words.

Code in figure 4 shows the loading of English language stop words.

Figure 4

loading English stop words code

```
stop_words = stopwords.words('english')
```

Getting the word frequencies.

Code in figure 5 shows items() method to create a list of tuple that contains the word frequencies.

Figure 5

word_counts() and items() code

```
get_items = blob.word_counts.items()
```

Eliminating the stop words.

Code in figure 6 shows the elimination of stop words using the list comprehension.

Figure 6

eliminate list comprehension code

```
eliminate_items = [item for item in get_items if item[0] not in stop_words]
```

Sorting the words.

Code in figure 7 shows how to sort the words using the sorted() function in reverse order. Words that appear most will be in the first. With the help of itemgetter() function that will allow us the order being by the count of the words, not the words themself.

Figure 7

sorting words code

```
sorted_itmes = sorted(eliminate_items, key=itemgetter(1),reverse=True)
```

Top 30 words.

Code in figure 8 shows the slice of the words to have the top 30 words in the text file.

Figure 8

slice words code

```
top_words = sorted_itmes[0:30]
```

Creating data frame with two columns.

Code in figure 9 shows the creation of a dataframe using Pandas with two columns, “word” and “count” columns.

Figure 9

create a dataframe code

```
word_df = pd.DataFrame(top_words, columns=['word','count'])
```

Plotting the 30 words.

Code in figure 10 shows the creation of the horizontal bar chart.

Figure 10

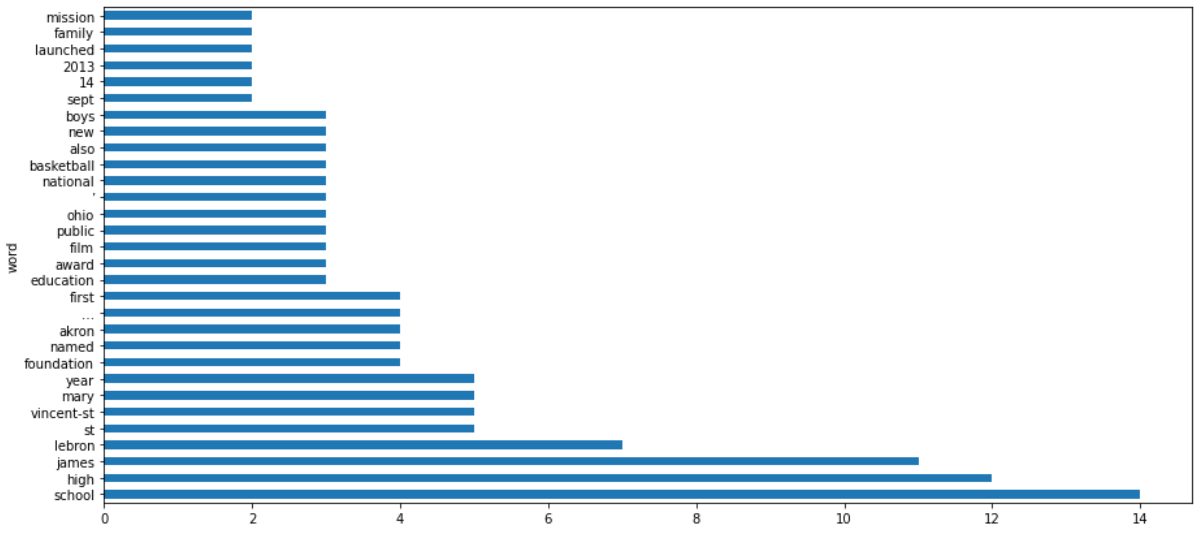
create bar chart code

```
plot_words = word_df.plot.barh(x='word',y='count',legend=False,figsize=(15,7))
```

This code created the following bar chart in figure 11. The result shows that the text needs more cleaning. We can see there are three dots "…" counted four times which is not a word. Also, an apostrophe counted three times. The word "st" counted five times which is abbreviation of something that needs to be read in a sentence to know what it refers to.

Figure 11

horizontal bar chart



References

Deitel, P. J., & Deitel, H. M. (2020). *Intro to Python for computer Science and data science learning to program with AI, big data and the cloud*. Hudson Street, NY: Pearson Education.

Kalb, I. (2016). *Learn to program with Python*. Berkeley, CA: Apress.

Mueller, J., & Emid, A. (2018). *Beginning programming with Python® for dummies®*. Hoboken, NJ: For Dummies, a Wiley brand.

NBA. (2020). *LeBron James*. Retrieved November 06, 2020, from https://www.nba.com/player/2544/lebron-james/bio

Kim, B. (2018). *Importing local files in Google Colab*. Retrieved November 06, 2020, from https://buomsoo-kim.github.io/colab/2018/04/15/Colab-Importing-CSV-and-JSON-files-in-Google-Colab.md/