Final project DS 397

Determining the most Similar Movie to Cars

Hunter Fristick

Below are the Links to the data being used in this project

https://themostinportantwikisincegodwascreated.fandom.com/wiki/THE_ENTIRE_CARS_SCRIPT

https://acerbialberto.com/publication/2018_imsdb/

To run this file on your own you will need to place the movie_scripts folder into your google drive.

Overview

For this project, I wanted to see what movie script in a dataset of 1,093 scripts was the most similar to the Cars movie. I used TF-IDF vectors and scored based on cosine similairty to determine which had the highest similarity. To compute the TF-IDF and cosine similarity quicker, I used sklearn because it took significantly longer to run when coding it without this package. I also used the nltk pachage to import in stopwords that needed to be removed as well as to get the stem of every word.

Formulas Used

TF-IDF = tf(t,d) * log(N/(df+1))

Cosine Similarity = 1-dot(cars_script,script2)/(norm(Lcars_script)*norm(script2))

```
#Importing necessary packages
import pandas as pd
import os
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
extrawords = stopwords.words('english')
from nltk.stem import PorterStemmer
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

```
#Connecting to Google Drive to load in data
from google.colab import drive
drive.mount('/content/gdrive')
%cd '/content/gdrive/My Drive/movie_scripts'
```

```
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mou
/content/gdrive/My Drive/movie_scripts
```

In the below code I first read in each script from the movie_scripts flie in MyDrive. Then, I seperated each script by words. Next, I removed stopwords and switched every word to the stems of them using PorterStemmer(). The scripts are then added to a list of all scripts called scripts.

Disclaimer: This code block takes around 10 minutes to run.

```
#Create lists
scripts = []
filenames = []
stemmer = PorterStemmer()

#Directory loaction
path =r'/content/gdrive/My Drive/movie_scripts'
os.chdir(path)

#Read through the files, split by word, remove stopwords, and append to scripts list
def read_files(file_path):
   with open(file_path) as f:
      lines = f.read()
      line = lines.split()
      filenames.append(file_path)
      removewords(line)
      scripts.append(line)

#Removes stop words from each script
def removewords(word_list):
  for i in range(len(word_list)):
    word_list[i] = word_list[i].lower()
    if i in extrawords:
      word_list.remove(i)
    word_list[i] = stemmer.stem(word_list[i])

#Loop through all files that end in .txt in the directory
for f in os.listdir():
   if f.endswith('.txt'):
      #Create the filepath of each file to import
```

```
      file_path =f"{path}/{f}"
      read_files(file_path)
```

```
#First ten file names
filenames[1:10]
```

```
['/content/gdrive/My Drive/movie_scripts/Script_White Ribbon, The.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Perks of Being a Wallflower, The.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Bodyguard.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Peeping Tom.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Freaked.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Wall Street.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Life of David Gale, The.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Fast Times at Ridgemont High.txt',
 '/content/gdrive/My Drive/movie_scripts/Script_Memento.txt']
```

In this code I created a for-loop to loop through each movie script to compare them to the Cars script. I used TfidfVectorizer from sklearn to get the TF-IDF scores for the words in each script combination. I then calculated the cosine similarity sklearn.metrics and appending each similarity score to a dictionary.

```
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.feature_extraction.text import TfidfVectorizer
#Load in the TF-IF Vectors
vectorizer = TfidfVectorizer()
highestcossim = {}
#Loop through every script in scripts, comapring each to The Cars Script
for i in range(len(scripts)):
  vectors = vectorizer.fit_transform([str(scripts[filenames.index('/content/gdrive/My Drive/m
  feature_names = vectorizer.get_feature_names_out()
  dense = vectors.todense()
  denselist = dense.tolist()
  #Creating a dateframe of the TF-IDF Scores and computing the cosine similairty
  df = pd.DataFrame(denselist, columns=feature_names)
  x = cosine_similarity(df)
  #Add the Cosine Similairy and File name to the highestcosin dict
  highestcossim[filenames[i]] = (x[0][1])
  highestcossim
```

Below I removed the max score since this is were the Cars Script is comparing to itself and then output the true highest cosine similarity score.

```
#Remove the Cars vs itself cosine similairty and return the best match
del highestcossim[(max(highestcossim, key=highestcossim.get))]
maxcos = max(highestcossim, key=highestcossim.get)
```

```
print('Higest Cosine:', highestcossim[maxcos])
print('File Name:', maxcos)
```

```
Higest Cosine: 0.7547541618546647
File Name: /content/gdrive/My Drive/movie_scripts/Script_Cars 2.txt
```

In this codeblock below, I converted the similairty score dictionary to a dateframe and sorted to see top scores. Finally, i exported the findings to a csv file.

```
#Turn Results into dataframe and sort by cosine similarity values
highcosdf = pd.DataFrame.from_dict(highestcossim, orient = 'index')
highcosdf.columns = ['Cosine Similarity']
highcosdf = highcosdf.sort_values('Cosine Similarity', ascending=False)
highcosdf
```

	Cosine Similarity
/content/gdrive/My Drive/movie_scripts/Script_Cars 2.txt	0.754754
/content/gdrive/My Drive/movie_scripts/Script_Big.txt	0.690253
/content/gdrive/My Drive/movie_scripts/Script_Finding Nemo.txt	0.687507
/content/gdrive/My Drive/movie_scripts/Script_Office Space.txt	0.671200
/content/gdrive/My Drive/movie_scripts/Script_True Romance.txt	0.652544
...	...
/content/gdrive/My Drive/movie_scripts/Script_Army of Darkness.txt	0.391806
/content/gdrive/My Drive/movie_scripts/Script_Black Swan.txt	0.388016
/content/gdrive/My Drive/movie_scripts/Script_Evil Dead II_ Dead by Dawn.txt	0.381665
/content/gdrive/My Drive/movie_scripts/Script_Ghostbusters.txt	0.380264
/content/gdrive/My Drive/movie_scripts/Script_Shining, The.txt	0.375125

1092 rows × 1 columns

```
#Download results to a csv file
from google.colab import files
highcosdf.to_csv('highcosdf.csv')
files.download('highcosdf.csv')
```

✓ 0s completed at 9:00 PM