Data Science Fundamentals

(CS 890ES)

Instructor: Dr. Alireza Manashty
Department of Computer Science
University of Regina
Winter 2020

Project Setup Guide
on

"Games Description Analysis & Keyword Suggestions"

Submitted by:

Name	ID	Email	Role
Sai Keerthi Mettu	200416252	smj102@uregina.ca	BI Analyst Data Engineer Data Scientist Communicator
Somi Deepthi Nalamalpu	200412879	sny899@uregina.ca	

Setup Introduction:
This data science project was done by creating a dataset through web scrapping. In order to run the project and see the output predictions out of it, we need to do the following steps. These are all the requirements that were gathered to develop this project from scratch:

Software Requirements:
Anaconda IDE or Google Collaboratory
Microsoft Excel
Local storage or Google Drive
Browser

If a local IDE like Anaconda is installed in the working system, then there is a need to download all the following libraries to run the project code.

Libraries for web scrapping to create a dataset:
Google play scrapper
Play scrapper
Beautiful Soup

Libraries for ETL processing on dataset:
Numpy
Pandas
Counter
Beautiful Soup
Re
String
NLTK
word_tokenize
PorterStemmer

Libraries for visualization:
Matplot
Seaborn
PCA
TNSE

Libraries for modelling:
MiniBatchKMeans from sklearn.cluster
KMeans from sklearn.cluster
CountVectorizer
TfidfVectorizer
TfidfTransformer
pickle

Libraries for webpage development:
Pickle
Flask-ngrok

Pyspark

If Anaconda is not used, and instead Google Colab is used, then there is a need only to import all the above libraries along with the drive library that actually helps to mount Google Drive on to the Google Collaboratory.

Steps to create a dataset:

- After the installation of libraries required for web scrapping, load the PS_display_apps.py notebook file into the working environment.
- There is a need to provide the web link of a play store game page from the browser in the code cell that accepts it.
- On giving the rightful link, then we have to run the cell scripts from that code cell till the bottom.
- At the end, this would result in successful scrapping of the game details that are available on that page into a separate downloadable csv file.
- This process is repeated till all the different game genre links are captured and download them as multiple csv files.
- Once all different genre pages are scrapped, there is a need to pull out some of the archived games from the play store.
- In order to do that, club all these csv records into a single csv file and load the similar_apps_from_ps.py file
- In the cell that takes a batch count value provide a batch range of 50-75 (should never be more this value as this would result in greater amount of processing time) game app Id's from this clubbed records file.
- Run all the code cells from that point till the end which would take 5 plus minutes. Each time, this derives a list of similar games that are close enough to that type of given App Id's and are downloadable as a CSV files.
- Thus, this process could be repeated until all the batches are completed. (It could approximately be 31 batches if no errors were found in the given App Id's)
- All the downloaded batch files can be merged into a single csv file manually and call it as batchfiles.csv.

Steps to see project output on Google Colab:

Load the Keyword_suggestions.py notebook file into the python working environment.

Make sure the dataset containing csv file is placed in google drive.

Mount the google drive onto the colab with respected to the user that is logged in.

Give the path correctly of the csv file that is placed in drive in the notebook file that takes the file path.

Once, this cell successfully executes, move on running all the remaining cells further it.

This would show all the processes of ETL, visualization, text extraction of description content, vectorization and modelling.

The program would stop at a place where it takes a user description for testing/ predicting the model output.

On changing this user given test description, it would show the resultant cluster suggestions as well as the TF-IDF suggestions according to the trained model.