**README**

Contributors: Bryan Houlton, Anish Manne, Shaan Haider, Priyam Kadakia

Summary: This program to pulls news articles from Google News API and performs topic modeling on the content. The following function applies natural language processing on the content strings and outputs the 10 most relevant topics of the articles acquired via nonnegative matrix factorization.

**Note**: This is not a fully built process and will require manual calling of each function in the process.

Functions

* clean\_string
  + Inputs: a string of all the text from an article
  + Output: the natural-language processed string (lowercase and computer simplified)
* make\_matrix
  + Inputs: array of strings that describe the content of each article
  + Outputs: a weighted matrix that shows the term-frequency across all articles
* do\_sql
  + Inputs: array of dicts containing all info pulled from Google News API
    - [{‘source’:\_\_\_\_\_, “author”:\_\_\_\_, “title”:\_\_\_\_\_, “description”:\_\_\_\_, “url”:\_\_\_, “urlToImage”:\_\_\_, “publishedAt”:\_\_\_\_, “content”:\_\_\_\_}, …]
  + Outputs: stores all the info from the dict into a MySQL database
* Convert\_text\_to\_json
  + Inputs: array of dicts in Google News’ API native json format
  + Outputs: takes info from articles and stores in a json file named articles2.json with single layer dicts
* Do\_sql
  + Inputs: array of dicts containing all information
  + Outputs: None, but it adds all data to a previously created table named general\_info\_raw in a MySQL database named group\_project
* Remove\_null\_content\_and\_description
  + Input: all articles
  + Output: all articles that actually have content
* Remove\_escape\_chars
  + Inputs: array of dicts with all information from articles
  + Outputs: file named articles\_final.json with the original dicts, but without escape characters in the content or description
* Make\_cleaned\_text\_json
  + Input: array of dicts with all article information (can be loaded from articles\_final.json)
  + Output: conjoins content and description strings from each article and applies clean\_string function. Then stores all cleaned strings in file named cleaned.json
* Make\_matrix\_and\_write\_to\_json
  + Inputs: None
  + Outputs: performs make\_matrix on cleaned.json and outputs the result to a file called matrix.json
* Do\_sql\_on\_string\_json
  + Inputs: None
  + Outputs: Opens cleaned.json and puts those into a separate MySQL table named nlp\_strings in the database called group\_project
* The program will automatically load matrix.json and perform NMF on said matrix revealing the top 10 topics of the articles

Requirements

* Python 3
* MySQL
* Install newspaper library in Python
  + For pip3, install newspaper3k
* Install psycopg2 library in Python
  + Pip install -U pip and psycopg2
* Import os, sys, shutil, random, numpy, psycopg2, scrape, argparse, logging, json, traceback, re, and pandas
* Import mysql.connector
* Import natural language toolkit
* From the corpus of the nltk, import ’stopwords’
* From the sklearn.feature\_extraction.text library, import ‘TfidfVectorizer’