COMP472 – Project 4 Report

Rhina Kim – 40130779

December 12, 2022

Overview

The purpose of this project is to experiment with crawling and scraping Concordia website https://www.concordia.ca. After the clustering of websites using k-means as clustering algorithm the project then analyzes each cluster's sentiment score using Afinn sentiment lexicon. This report will discuss the followings:

1) Choice of crawling tools and any other dependencies used & how it is implemented
2) Assess two results of the clusters brought by k-3 and k-6
3) Analyze behaviors of two result cluster collections with k-3 and k-6
4) Assess sentiment values for each cluster and derive own formula
5) Analyze the usefulness of sentiment values

1) Technologies and Implementations

Language

- Python 3.8

Dependencies

- *beautifulsoup4* 4.11.1
- *scipy* 1.9.3
- *afinn* 0.1
- *scikit-learn* 1.2.0
    - *TfidfVectorizer*
    - *KMeans*
- *reppy* 0.4.14
- *urllib3* 1.26.13

Python 3.8 is used as programming language for this project since it supports any machine learning features from *scikit-learn* package and all the newest version of packages necessary for this project. This report only listed necessary packages for this project, yet all the other packages used are found in ***requirements.txt*** inside */P4* folder.

First of all, *reppy* package was used to look up and fetch robots.txt file for given URL. *Urllib* package is used to parse the URL as an input in order to find robots.txt file. *Spidy* was used as a crawling tool for this project since it comes incorporated with various set of user input and detailed loggings and is extensively configurable. Moreover, it has support for numerous useful features such as multi-threading or robot exclusion. Robot exclusion is also done by part of *Spidy* when RESPECT_ROBOTS is set to True (see more details in *DEMO*.pdf file). However, part of Robot exclusion step was badly implemented (failure to look up robots.txt for most of the URL given during the execution), which I made another class RobotsFetcher() inside P4/*crawl.py* that replaces the given URL path with robots.txt to create path for Robots file in order to look up appropriate robots.txt file for given URL, prior to analyzing the content of Robots text. *BeautifulSoup4* was used to extract the text of the crawled and downloaded html pages, which is implemented in get_text_from_pages() from *main.py* (see more details in *DEMO*.pdf file). After collecting the text from the web pages, this project then vectorizes every crawled article in order to represent them as a numeric vector, which was done via *TfidfVectorizer* method inside *scikit-learn* packages to have words ratio. The *TfidfVectorizer* method comes with preprocessing of the text before forming an inverted index, such as lowercasing, removing stop words, and other preprocessing. *KMeans* was used to clusterize the tfidf numeric vectors, which is implemented in

cluster_collection() from *main.py* (see more details in *DEMO*.pdf file). Finally, afinn was used to determine sentiment values for the each two set of clusters experiment (k=3 and k=6), which is implemented in compute_afinn_score() in *main.py* (see more details in *DEMO*.pdf file).

Although Spidy was used to crawl Concordia website, I have modified some codes inside the existing Spidy crawling script to adapt to this project's purpose. Refer to *DEMO*.pdf for further details.

Additionally, to crawl Concordia website, some configurations have to be applied (e.g., maximum html files to download, number of threads to crawl with, etc.). The crawling config file came with *Spidy* package which resides under *P4/config/* folder, and I have made additional config file that is suitable for this project's purpose, which is in *P4/config/**concordia.cfg*** file. Refer to *P4/config/**blank.cfg*** for the description of every config input variable.

2) Clusters Naming

One of the examples of interesting output with maximum document to download = 144 was:

```
CLUSTERING THE EXTRACTED TEXT WITH K=3
Number of elements assigned to each cluster: [124  19   3]

Top 20 terms for cluster 0 with k = 3: school concordia science services academic calendar
gina cody computer engineering research news graduate student university arts events colleges
schools class
AFINN score for cluster 0 with k = 3 and top 20 terms: 0.0

Top 20 terms for cluster 1 with k = 3: concordia school calendar event science services campus
graduate academic student arts schools class university colleges students news research cody
gina
AFINN score for cluster 1 with k = 3 and top 20 terms: 0.0

Top 20 terms for cluster 2 with k = 3: concordiaâ concordia sustainable research says national
glatard school director institute cyber university science calendar consortium smart graduate
development future campus
AFINN score for cluster 2 with k = 3 and top 20 terms: 3.0

CLUSTERING THE EXTRACTED TEXT WITH K=6
Number of elements asigned to each cluster: [124  16   1   3   1   1]

Top 20 terms for cluster 0 with k = 6: school concordia science services academic calendar
gina cody computer engineering research news graduate student university arts events colleges
schools class
AFINN score for cluster 0 with k = 6 and top 20 terms: 0.0

Top 20 terms for cluster 1 with k = 6: concordia school calendar event science services campus
graduate student academic arts university schools students class colleges news research gina
cody
AFINN score for cluster 1 with k = 6 and top 20 terms: 0.0

Top 20 terms for cluster 2 with k = 6: test testing isolation execution production method risk
cloud propose case methods activities software interferences environment fault event solution
school providers
AFINN score for cluster 2 with k = 6 and top 20 terms: -1.0
```

```
Top 20 terms for cluster 3 with k = 6: concordiaâ concordia sustainable research says national glatard school director institute cyber university science calendar consortium smart graduate development future campus
AFINN score for cluster 3 with k = 6 and top 20 terms: 3.0

Top 20 terms for cluster 4 with k = 6: noma comp access multiple ris networks user fd techniques ceus bss cell integration power communication performance compared investigate literature spectral
AFINN score for cluster 4 with k = 6 and top 20 terms: 0.0

Top 20 terms for cluster 5 with k = 6: founded ba boutique based manette perfect retail 99 mary coffee concordia school suitablee fit pet holiday jordan comics seafood world
AFINN score for cluster 5 with k = 6 and top 20 terms: 4.0
```

Before assessing the result clusters, there is one important point to be noted: Here, the project used top 20 terms for each cluster as a display and **APPLIED** that top 20 terms to compute Afinn score for the purpose of easily highlighting terms that affected the Afinn score, instead of applying Afinn score to every term in each cluster). Refer to the analysis of Afinn score in section (4).

We can see with the above example that most of the articles resides in cluster 0 (826 articles assigned as cluster 0 for k=3 and 881 articles assigned as cluster 1 for k=6), and from the 20 terms displayed, we can guess that the topic/theme of this Concordia website is related to "school" in general.

The naming I would give to each cluster are:

k=3:

- Cluster 0: Concordia Engineering department
- Cluster 1: Concordia events and services
- Cluster 2: research: sustainable campus

k=6:

- Cluster 0: Concordia Engineering department
- Cluster 1: Concordia events and services
- Cluster 2: research: software testing
- Cluster 3: research: sustainable campus
- Cluster 4: research: user network
- Cluster 5: Concordia Stores (?)

3) Assess Clusters' Behavior

In the k-means clustering algorithm, the value of k represents the number of clusters that the algorithm will attempt to identify within the data. As the value of k increases, the algorithm will attempt to divide the data into more clusters, which means that the clusters will be more specific and potentially more accurate in terms of capturing the inherent structure of the data.

From the above example, cluster 0 and 1 for both k=3 and k=6 seemed to have identical theme as many articles were distributed into this category. Moreover, the ratio of distribution for cluster 0

and cluster 1 for both k=3 and k=6 is similar: majority of the articles stayed in the same category even after k=6 clustering.

The content of the top 20 clustered terms were also interesting to note: for both cluster 0 and 1 with k=3 and k=6 carried similar outputs. Since the website is about Concordia University and majority of the contents are dedicated to introducing departments, school systems, services, and events, which is very predictable to have words related to school information with majority of articles categorized in it. From 3rd cluster, the theme seems to slightly deviate from school: according to the output cluster terms, it seems like some research or proposals about what university is aiming for in the future, due to the world "solution", "university", "campus", "future", and all the other technical words. It can be observed that cluster 3 at k=3 is further distributed for k=6.

It is important to note that, however, increasing the value of k can also increase the risk of overfitting, where the clusters become too specific and do not generalize well to new data. In this example case, the distribution of clustering was a lot biased, leaning mostly to cluster 0 and 1. As the Concordia website hosts mostly about school related information, unlike website with various topics like online news magazine, the overfitting problem was expected. For this Concordia website, making more distributions on cluster 0 and 1 or making cluster 2-6 under one cluster and naming it as "unrelated school info" would have solved the overfitting problem.

4) Sentiment Scores Analysis

A lot of clusters resulted in sentiment score of 0.0. This is because Concordia website is mostly and purely about informing Concordia university, which rarely includes positive or negative statements. However, not all the clusters had sentiment score of 0. For cluster 2 with k=3, sentiment score was 3.0 due to the optimistic words like "future", "sustainable", or "smart". Meanwhile, cluster 2 with k=6 displayed -1.0 negative sentiment score, possibly due to the emergence of "risk" word.

5) Usability of Sentiment Analysis

Applying sentiment scores can be very useful on website with sentiment or emotion expressed, such as movie or literature reviews, customer service, marketing, and research. Nevertheless, school website is an informational text which has the purpose of giving information and avoids emotive language as much as possible. Therefore, it might not be useful for if the objective of the analysis is to determine how people feel about the University.

However, sentiment analysis could also be used to identify patterns or trends in the way that people talk about the school online, which could help the school understand what aspects of the school are most important and how it is perceived by the community. For example, based on the texts in the cluster collections, each text in cluster is accurately related to Concordia University. The clusters contain terms such as "Concordia," "school," "university," "student," and "academic," which suggest that the text is related to various aspects of the university, such as its programs, services, events, and research, which is a distinct pattern that can be observed from Concordia University website thanks to the help of sentiment analysis.