CSC 177-01 Data Warehousing and Data Mining (Spring 2019)

Mini-Project 2: Clustering

Due at 2:00 pm, Friday, March 8, 2019

Demo Session: class time, Friday, March 8, 2019

In this project you will use **the cleaned data you achieved in Project 1** to practice with three clustering algorithms: k-means, MAX-based agglomerative clustering, and SSE-based agglomerative clustering.

1. Clustering of Twitter Users (40 pts)

In the first problem we will look into the clustering of users. To apply clustering, we first need to represent each user as a vector of integers with the frequency (i.e., how many times) a user has used each hashtag/handle. Use *tfidfVectorzier()* in Sci-kit learn to create this representation. **See hints in the last section.**

Task 1.1 (20 pts): Let's apply clustering and compare the clustering result against a known ground truth. **In the file "clinton_trump_user_classes.txt", we have the ground truth "class" membership for each user id in the data. Class 0 corresponds to Trump followers, while class 1 corresponds to Clinton followers.**

Run the k-means algorithm (K=2) and the two different variations of the agglomerative clustering algorithm (MAX-based and SSE-based).

Compute the confusion matrix, precision, recall, and F-measure for (1) k-means, (2) MAX-based agglomerative clustering, and (3) SSE-based agglomerative clustering. **Compare their performance and include your conclusions in your report. See sample code in lab 4.**

Task 1.2 (10 pts): For k-means, look at the two centers (centroids) and print the top-30 hashtags/handles with the highest tfidf values.

Task 1.3 (10 pts): Show the **two respective word clouds** of the two centers (centroids) by using hashtags/handles and their tfidf values. **Hint: Use function fit_words() that comes with wordcloud**

Can you draw some conclusion from the most frequent hashtags/handles in each cluster about what differentiates the two clusters?

2. Clustering of Hashtags/handles (30 pts)

In the second problem, we will look into clustering of hashtags/handles. Represent each hashtag/handle as a vector of integers with the frequency of the hashtag/handle for each user. **Hint: You may transpose the matrix you achieved in the first problem to construct this representation.**

Task 2.1 (10 pts) First, you apply the k-means algorithm. Create a plot of the SSE error of the k-means algorithm as a function of the number of clusters, for k up to 20, in order to determine the optimal number of clusters.

Task 2.2 (20 pts) Run the k-means algorithm for the optimal number of clusters you identified in the last task. Print some hashtags/handles in each cluster. From the hashtags/handles in each cluster, try to deduce what is the topic it concerns. **Include your conclusions in your report.**

3. Grading Breakdown

Your grade for this project will be based on the soundness of your design, the novelty of your work, and the effort you put into the project.

Implementation	**70 pts**
Your report	**20 pts**
In-class defense	**10 pts**

4. Deliverables:

(1) All your source code/figure in Python Jupyter notebook.

(2) Your report in PDF format, with the name and id of each team member, course title, assignment id, and due date on the first page. As for length, I would expect a **report with more than one page**. Your report should include the following sections (but not limited to):

1. **Problem Statement**
2. **Methodology**
3. **Experimental Results and Analysis**
4. **Task Division and Project Reflection**

In the section "**Task Division and Project Reflection**", describe the following:

- who is responsible for which part,
- challenges your group encountered and how you solved them
- and what you have learned from the project as a team.

10 pts will be deducted for missing the section of task division and project reflection.

Notice: please don't send me the original dataset. ☺

All the files must be submitted **by team leader** on Canvas before

2:00 pm, Friday, March 8, 2019

NO late submissions will be accepted.

5. Demo Session:

Each team member must demo your work during the scheduled demo session. Each team have **three minutes** to demo your work in class. The following is how you should allocate your time:

- Findings/results (1 minute)
- Task division (1 minute)
- Challenges encountered and what you have learned from the project (1 minutes)

Failure to show up in defense session will result in **zero** point for the project.

6. Teaming:

Students must work in teams of 2 or 3 people. Think clearly about who will do what on the project. Normally people in the same group will receive the same grade. However, the instructor reserve the right to assign different grades to team members depending on their contributions. So you should choose partners carefully!

Hints:

- To obtain vector representation of each user, you may group all the hashtags/handles by each user and then create a dataframe where each line has a user and all her hashtags/handles. Then call *tfidfVectorzier()* on that dataframe. See sample code here:

  df_hashtag_agg = df.groupby('UserID')['Hashtags'].sum()
  df_ready_for_sklearn = pd.DataFrame({'User_id': df_hashtag_agg.index, 'All_hashtags': df_hashtag_agg.values})

- If you experience low memory issues when using *tfidfVectorzier()*, adjust the parameters ***max_df, min_df,*** and ***max_features*** appropriately.

- For Numpy matrix transpose, see this link:

  https://docs.scipy.org/doc/numpy/reference/generated/numpy.transpose.html

- Pandas supports high-performance SQL join operations. Use Pandas function ***pd.merge()*** to merge (or to say, join) two dataframes based on values in one particular column. See an example here:

  https://chrisalbon.com/python/data_wrangling/pandas_join_merge_dataframe/