Digital Media Laboratory

Weekly Report December 27, 2021

Ali J. Alaee
alialaee98@gmail.com

Report No. 22
00/09/29- 00/10/05

Abstract

- Analyzing the User Similarity Matrix for Sparsity
- Finding appropriate method for clustering
- Getting results for the Spectral Clustering

Description

- **Sparsity**
  Since the Similarity Matrix occupies 2.6 GB, the sparsity check was a critical issue. It turns out that more than 85% of the elements are 0, therefore, a sparse matrix (csr) is constructed just in case.
  Another interesting fact about the data is that 1720 of 18781 selected users have 0 similarity with all other users; meaning that we can eliminate them from our clustering problem. However, in the analysis stated in this report they are not yet removed.

- **Clustering Methods**
  Last week, we discussed about using K-means algorithm for clustering the users. However, the Similarity Matrix does not represent users as points in a multi-dimensional space, but contains the similarity of each pair. Consequently, other clustering algorithms that work only based on the pair-wise similarities should be selected instead. These methods are:
  - Spectral Clustering: The algorithm takes the top k eigenvectors of the input matrix corresponding to the largest eigenvalues, then runs the k-mean algorithm on the new matrix. This is the method that I have used for getting results.
  - Hierarchical Clustering: Hierarchical clustering, also known as hierarchical cluster analysis, is an algorithm that groups similar objects into groups. The implementation is only based on an input distance matrix.

- **Results**
  The implementation of the Spectral Clustering on the User Similarity Matrix is available on Similarity_Matrix_Analysis.ipynb. Three experiments were obtained:

Experiment	Input Number of Clusters	Number of Members for Each Cluster
1	2	17930, 851
2	3	13728, 5046, 7
3	4	13134, 4964, 680, 3

Next Week

- Removing useless users from the Similarity Matrix, and experimenting the Spectral Clustering once again.
- If this way of clustering is accepted, then we could start learning different embeddings for each set of users.

References