Explore Co-clustering on Job Applications

Qingyun Wan
qywan@stanford.edu

Motivation

In online job serving platform like Linkedin, Indeed and etc., Job recommendations are usually generated based on matching users and job posting features which are complicated as feature space can be very large. If users forget to update their profiles online, we even need to infer features based on other information. In this case, it is desirable to detect innate groupings of users and jobs based on more direct and truthful information - job applications. which can improve recommendation quality ultimately.

The goal of this project is to explore the effectiveness of **co-clustering** users and jobs and compare with **one-way clustering** on users based on job applications.

Data

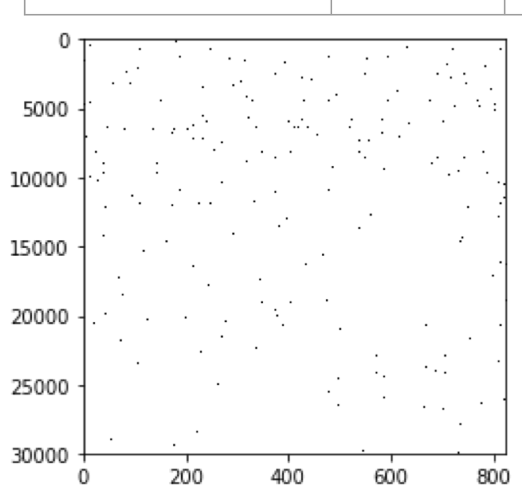
Kaggle Job Recommendation Challenge

- ~ 1.6m unique job applications
- ~ 360k unique jobs
- ~ 320k unique job applicants

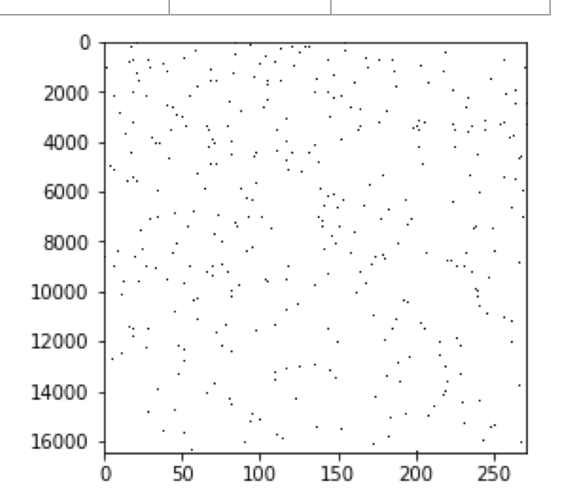
Preprocess

- constructed 0-1 user-job matrices based on job application with 2 different densities

Minimum # of Job Applications Per Job	Density	# of Rows (users)	# of Columns (jobs)	# of Non-zero Entries (applications)
75	0.32%	29957	823	79666
100	0.73%	16449	271	32873



Lower Density



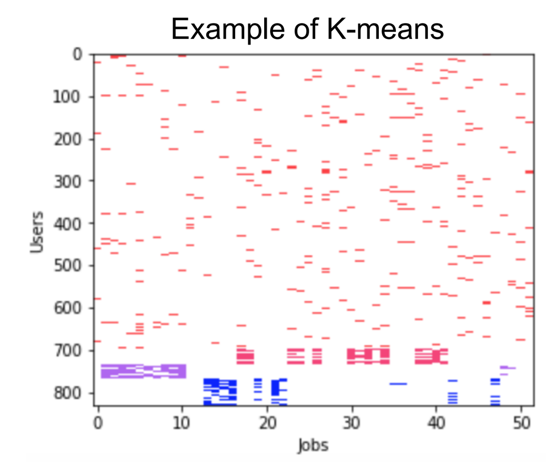
Higher Density

- split job applications for each density into 5 partitions for 5-fold cross validation

Methodology

Baseline K-means

Apply one-way clustering on users using K-means. Each training data point is a user represented in one-hot encoded vector based on the applied jobs.



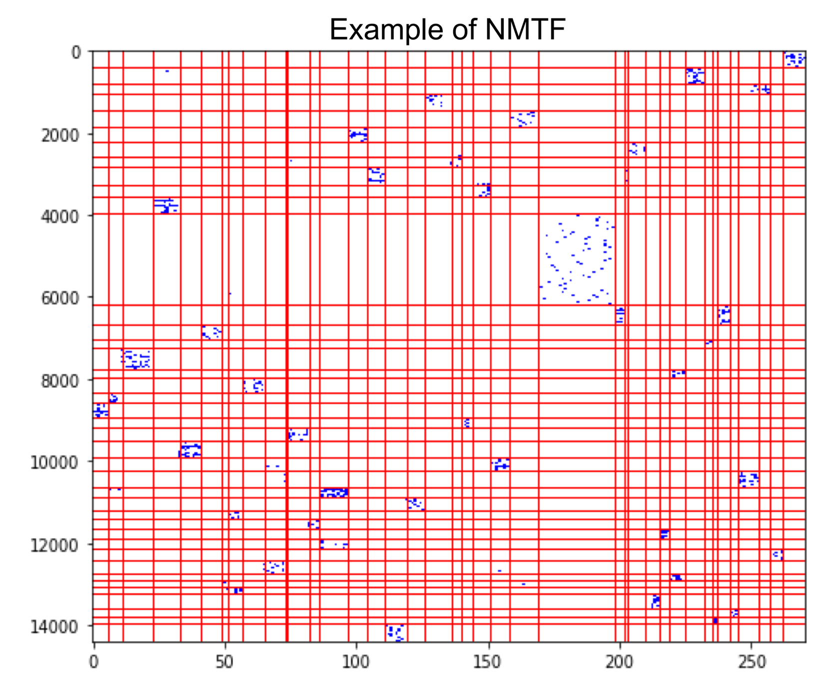
Co-clustering

• Nonnegative Matrix Tri-factorization (NMTF)

Given the user-job matrix X, NMTF does nonnegative 3-factor decomposition of it and the objective function derived from bi-orthogonal Nonnegative Matrix Factorization is

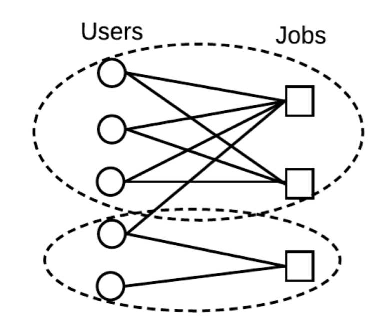
$$\min_{F\geq 0, G\geq 0, S\geq 0} \|X - FSG^T\|^2, \ \ s.t. \ F^T F = I, \ G^T G = I.$$

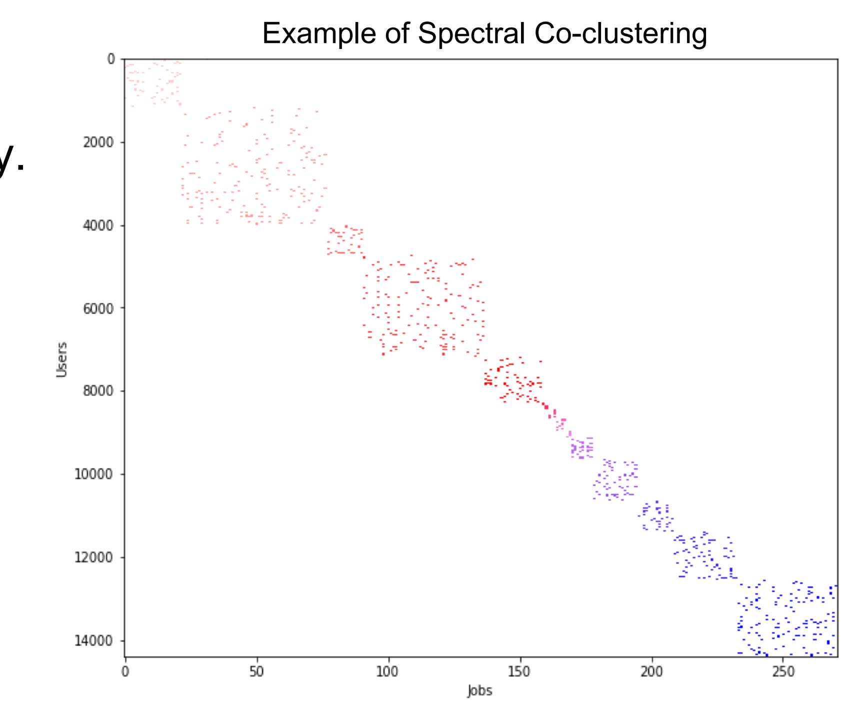
where F is the cluster indicator matrix of clustering rows and G is the indicator matrix of clustering columns. It corresponds to simultaneously clustering rows and columns of X.



• Spectral Co-clustering

Given the 0-1 user-job matrix, convert it to a undirected bipartite graph which has two sets of vertices representing users and jobs respectively. An edge exists if a user has an application of a job.





A cut between two clusters is defined by the total number of edges between them. The objective is partitioning the graph to achieve the minimum of a normalized version of all cuts, which corresponds to the best co-clustering.

Experiments

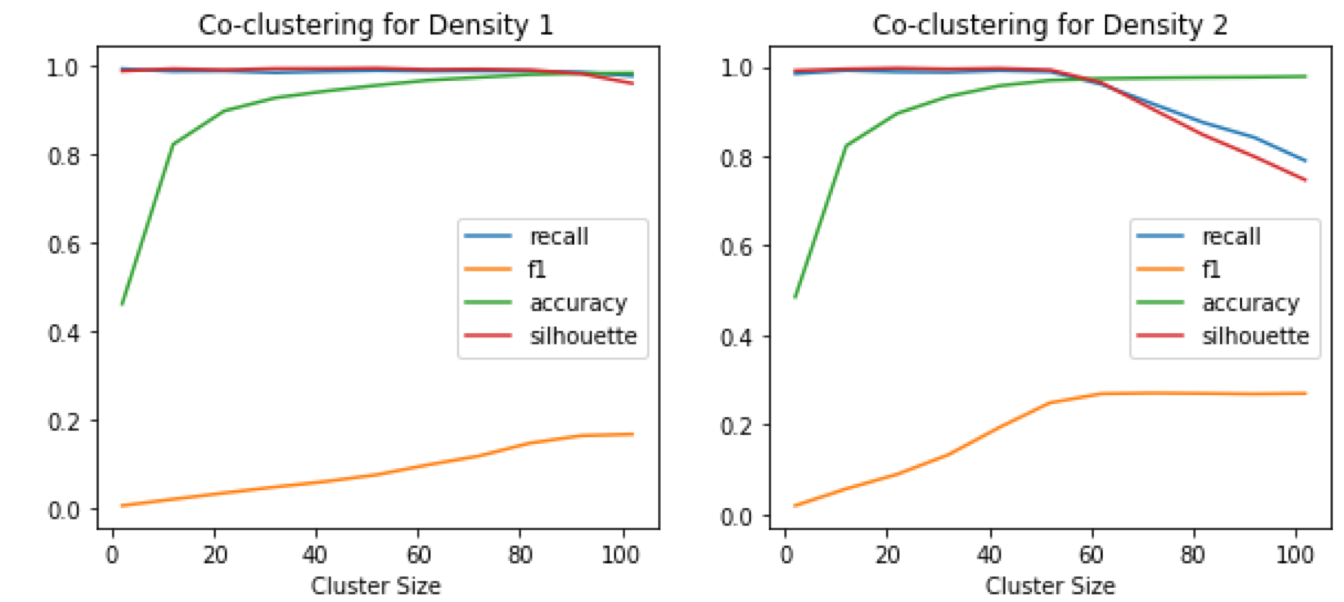
Since the labels for job seekers are unknown but only job applications,

- to validate the effectiveness of different clustering methods, I used the job applications in the testing set and computed recall, accuracy and F1 score against trained clusters.
- to find the optimal number of clusters, use the silhouette method to find the number that yields to the maximum silhouette score.

Result

	Density 1 (Lower Density)			Density 2 (Higher Density)		
	recall	accuracy	F1 score	recall	accuracy	F1 score
NMTF	0.984	0.975	0.131	0.989	0.968	0.237
Spectral Co-clustering	0.985	0.985	0.141	0.948	0.969	0.213
Baseline K-means	0.971	0.570	0.009	0.754	0.525	0.016

Effectiveness Comparision



Cluster Size Analysis with 5-fold Cross Validation

Discussion & Future

- Compared to one-way clustering, co-clustering is more stable with higher F1 score and more accurate with much less false positive.
- From the visual comparison, co-clustering has more balanced cluster size than K-means, which is more ideal for job recommendation by supplying more focused pools of jobs to recommend.
- Unlike co-clustering, K-means matches users more strictly, limiting potential jobs that might be suitable for users.
- Both co-clustering methods are slow because they both leverage matrix decomposition, it's beneficial to explore more scalable co-clustering methods so that we can co-cluster efficiently on more sparse dataset.

[1] Ding C, Li T, Peng W, et al. Orthogonal nonnegative matrix t-factorizations for clustering[C]//Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006: 126-135.
[2] Dhillon I S. Co-clustering documents and words using bipartite spectral graph partitioning[C]//Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001: 269-274.
[3] Ungar L H, Foster D P. Clustering methods for collaborative filtering[C]//AAAI workshop on recommendation systems. 1998, 1: 114-129.
[4] Long B, Zhang Z M, Yu P S. Co-clustering by block value decomposition[C]//Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005: 635-640.

Presentation video link:
https://www.youtube.com/watch?v=3zPD-QipAnw