DOI: https://doi.org/10.24297/ijrem.v10i0.8403

Recent Approaches of Partitioning a Set into Overlapping Clusters, Distance Metrics, and Evaluation Measures

Gursimran Pal*, Sahil Kakkar

Mata Gujri Khalsa College Kartarpur (Jalandhar), Punjab, India*

Department of Computer Science & Engineering, Guru Jambheshwar University, Hisar, Haryana, India

*amgur12321@gmail.com, enthusiast.sahil@yahoo.in

Abstract

This paper reviews recently proposed overlapping co-clustering approaches and related evaluation measures. An overlap captures multiple views of the partitions in data set, hence is more expressive than traditional flat partitioning approaches. We present a graph-theoretic formulation of co-clustering, which allows nodes to possess multiple memberships and hence finds usage in diverse applications like text mining, web mining, collaborative filtering, and community detection. We also study proposed quality measures specifically adjusted to overlapping scenarios.particular subject.

Keywords: Partitioning Approaches, Graph Theory, Co-Clustering, Distance Measures, Quality Measures, High-Dimensional Metric Space

1 Introduction

When dealing with systems and networks of large size and complexity, clusters provide a natural summarization of the underlying system. The size and dimensionality of data generated has shown rapid growth, due to popularity of internet and social media. Although clustering is not a new data mining technique, this rise in volume and variety of data has made it more relevant than ever before. Clusters may be groups of frequently interacting individuals in social networks [1], [2], sets of web pages dealing with similar topics [3], [4], groups of genes having similar expression profiles [5], [6]. Most of such clustering applications are identified by occurrence of frequent overlaps among the discovered groups. This property is in effect due to inherent multiple memberships of the nodes forming the clusters. Thus traditional partition-based approaches to clustering fail to mine such multiplicity of representation. In this paper, we review some of the recently proposed approaches to overlapping co-clustering. This shift in paradigm also calls for required modification in traditional quality measures used for flat partitioning. We present in later sections, the measures adjusted to the overlapping case.

Since distance metric is inversely proportional to similarity metric, from now, onwards we will use the term *Similarity Metric*. One popular similarity metric that is used for comparing categorical data, is Jaccard Index. Given two sets S and T, Jaccard Similarity is defined as:

$$\mathbb{J}(S,T) = \frac{|S \cap T|}{|S \cup T|}$$

However, this metric renders the running cost of any algorithm prohibitively expensive due to two limitations:

- High-dimensionality of samples
- The large number of sample-pairs to compare, ${}^{n}C_{2}$

As dimensionality increases, the probability of occurrence of common features between samples decreases, making it difficult to ascertain similarity. This phenomenon is called *curse of dimensionality* [7]. To solve this

problem, randomized dimensionality reduction techniques like Minwise Hashing (MH) [8] and Weighted Minwise Sampling (WMS) [9] have been recently proposed in the literature. These techniques probabilistically reduce the dimension of samples from thousands to a small number, say K, while preserving the Jaccard similarity among samples.

Even if dimensions are reduced to a smaller number K, the large number of samples make the comparison task computationally expensive. A general approach to handle this task is Locality-Sensitive Hashing (LSH) [10]. The theory of LSH states that given the hash (MH or WMS) $h(x)$ of the sample x, for any pair of samples x and y, the probability of hash collision is given by:

$$\Pr\big(h(x) = h(y)\big) = \mathbb{J}(x, y)$$

This approach of comparing samples uses hash-table and is linear in number of samples, hence avoids ${}^{n}C_2$ number of comparisons.

2 Co-clustering

Co-clustering (also known as bi-clustering) is a natural evolution of the subspace clustering paradigm. It simultaneously clusters the rows (objects) and columns (attributes) of a dataset. Unlike traditional single-mode clusters, co-clusters correspond to arbitrary subspaces of objects and attributes, shown below as rectangles:

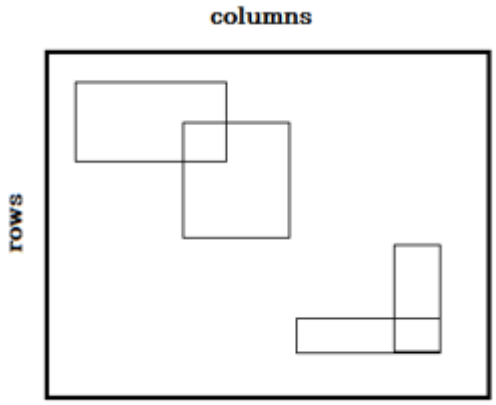

Figure 0-1 Co-clusters

For sparse high-dimensional data, like in word-document data [11], co-clustering is more effective than single-mode clustering because it does not group objects based on global correlation as groups of objects are present in lower subspaces. Co-clustering is equivalent to implicit and adaptive dimensionality reduction and noise removal (irrelevant dimensions removal) leading to better clustering results [12] because the objects (features) get described by feature-clusters (object-clusters) rather than just features (objects).

2.1 Graph-theoretic Formulation

Co-clustering finds application in diverse set of problems such as text mining [11], [13], collaborative filtering [14] and community detection [1], [2]. These problem areas are identified by information presented in form of categorical data. In categorical data, samples are usually described by presence or absence of some features, forming a binary/categorical feature vector corresponding to each sample. This type of data can be expressed by a sparse matrix. Formally, given a set O of objects and the set F of features, with $|O| = n$ & $|F| = m$, a co-cluster C is a triple (O', F', R), with $O' \subseteq O$, $F' \subseteq F$ & $R \subseteq O \times F$ that can be described as

$$C(O', F, ' R) = \{ O' \times F' \mid o \in O', f \in F', (o, f) \in R\} \qquad (1)$$

Here O' and F' are object clusters and feature clusters, respectively and relation R defines the structural type of co-cluster (binary connectivity in case of binary data) connecting object o and feature f co-occurring in the co-cluster. Notice that augmenting the relation R extends the traditional similarity-based clustering to a more general and flexible "pattern-based framework". In the simplest case of binary data, this connectivity information R between two disjoint sets (objects & features) can be modelled by the edge-connectivity between type-O and type-F nodes of the corresponding bipartite graph. Given an undirected unweighted bipartite graph along with its edge-connectivity information in the form of binary adjacency matrix $D_{n\times m}$ with n number of type-O nodes and m number of type-F nodes, a co-clustering solution aims to find cluster-pairs (O', F') of subset nodes such that every node-pair (o, f) exists in R, i.e., is edge-connected.

Notice that the above formulation does not restrict an object o or feature f from subscribing to more than one clusters. This leverages the possibility to represent multiple memberships as overlapping groups. Other alternative formulations such as (k, l)-Co-clustering which seek to partition the data into k object-clusters and l feature-clusters, as such fail to detect overlapping groups. In this paper, we specifically focus on co-clustering approaches following the formulation given in Eq. 1.

3 Overlapping Approaches to Co-clustering

In this section, we review overlapping co-clustering algorithms proposed in literature, which span into different fields like text mining, web mining, community detection, etc. Most of the algorithms base their formulation from Eq. 1.

3.1 Scalable Overlapping Co-clustering of Word-Document Data

In this paper [11], the author has proposed a scalable co-clustering algorithm based on Locality-Sensitive Hashing technique [10] to find co-clusters of words and documents. Following steps summarize the complete algorithm in brief:

- In the first step, objects are hashed to n possible co-clusters such that objects having at least p features in common are assigned the same hash-key. Objects having same hash-key form a candidate cluster.
- In the second step, for each candidate co-cluster, a feature that is associated to a minimum fraction of co-cluster's objects is inserted in an iterative manner.
- In final step, for each feature-set in candidate co-cluster, a document that contains all features is associated to the candidate co-cluster, leading to almost complete binary connectivity within each co-cluster.
- Since the algorithm experimentally leads to thousands of distinct co-clusters, a graph-partitioning routine kMETIS [15] is used to partition the corresponding graph into k different clusters.

Though the proposed algorithm is scalable, it does not guarantee 100% coverage of entire dataset, i.e. some objects or features may not belong to any of the discovered co-clusters.

3.2 An Effective Approach on Overlapping Structures Discovery for Co-clustering

In this paper [16], authors proposed a novel *overlapping pattern search* (OPS) strategy based on discriminative feature (object) set identification, given a non-overlapping partition of the data matrix. A discriminative feature function is defined which evaluates the contribution of a column-cluster in distinguishing given row-cluster from other row-clusters, based on difference of density between co-occurring block and average density of all blocks in the corresponding column group. Symmetrically, discriminative object function evaluates the contribution of a row-cluster in distinguishing given column-cluster from other column-clusters. Given a row-cluster (column-cluster) to be discriminated, the set of column-clusters (row-clusters) for which discriminative feature (object)

function is greater than zero is called Discriminative Feature Set (Discriminative Object Set) of that distinguished row-cluster (column-cluster).

The OPS strategy is then applied in the following manner:

- In the first step, a partition-based co-clustering like [17], [18] is applied on data matrix to obtain non-overlapping partitions in the form of row-clusters and column-clusters.
- For each row *r* of data matrix and each row-cluster *I* not containing *r*, if fraction of *I*'s features, shared by *r*, is more than a user-specified threshold α, then *r* is added to the co-cluster *I*.
- For each column *c* of data matrix and each column-cluster *J* not containing *r*, if fraction of *J*'s objects, shared by *r*, is more than a user-specified threshold β, then *c* is added to the co-cluster *J*.

Though the algorithm successfully discovers inherent overlaps among objects and features, it is computationally expensive because of element-wise comparisons, hence doesn't scale for large data sets.

3.3 A Hash-based Co-clustering Algorithm for Categorical Data

In this approach [13], the author relaxes the constraint in Eq. 1 to maximize the number of elements within the co-clusters, reformulating it as:

$$C(O', F', R) = \{ O' \times F' \mid o \in O', f \in F', |R'| \geq \rho \,.\, |O' \times F'| \} \qquad (2)$$

where the fraction ρ accounts for tolerable sparsity of features/objects within a co-cluster.

The steps of proposed algorithm are highlighted as following:

- In the first step, data is optionally pre-processed to remove features more frequent than given threshold.
- Then initial set of seed clusters are obtained by applying LSH technique [10] for Jaccard distance.
- For each seed co-cluster, the new dataset is derived at, which contains union of feature-set of each object and union of object-set of each feature.
- Next, the InClose [19] algorithm is applied to obtain set of co-clusters containing at least a minimum number of features and objects.
- For each generated co-cluster, objects and features are inserted into co-cluster so that sparsity is not more than ρ, as per Eq. 2.
- In the last step, co-clusters sharing exactly the same feature-sets are merged.

3.4 Trawling the Web for emerging Cyber-communities

Community discovery in networks requires representation of the interaction among same kinds of entities. Mathematically, this translates to an undirected unweighted bipartite graph with square adjacency matrix $D_{n \times n}$ such that $|O| = n$ & $|F| = n$. Since $O = F$, ideally only upper triangular matrix is sufficient for analysis. Discovering communities in this setting translates to discovering the sets of nodes in the corresponding bipartite graph that show complete binary connectivity, i.e., *completely connected subgraphs* or *bicliques*. Again, this formulation does not prohibits the subgraphs to overlap. For practical purposes, a dense subgraph with density greater than user-specified threshold can be used as a community candidate

In this approach [20], the authors base their algorithm on the hypothesis that websites that should be part of the same community are frequently co-cited. Web-community is represented as dense directed bipartite graph with each directed edge representing citation-link from the source page (citing the target page) to the target page (being cited by the source page).

The authors exploit these co-citations in the web-graph to extract initially all overlapping completely-connected subgraphs. In later stage, these subgraphs are expanded by inserting near-neighbour nodes. Steps followed in Trawling are summarized as follows:

- As pre-processing step, pages that are unusually highly referenced on the Web are deleted (e.g., Yahoo!)
- An inclusion-exclusion strategy is adopted, in which at every step, either a page is eliminated from the contention or an (*i*, *j*)-constrained biclique is reported, assuring useful progress at every step (either detect a community or prune the data).
- For each point $x \notin c_t$, it is added to the community c_t if it is edge-connected to at least minimum fraction of nodes in the community c_t.
- Since web crawl data is very big, the algorithm requires to be executable in a small number of steps. Where in each step, data is processed as a stream from disk and then stored back after processing.

4 Evaluation Measures

Two commonly used measures for comparing two clustering partitions are Normalized Mutual Information (NMI) [21] and F-measure [22]. Given two partitions *X* and *Y* of the data matrix, the NMI score $I_{norm}(X: Y)$ is defined as:

$$I_{norm}(X:Y) = \frac{H(X) + H(Y) - H(X,Y)}{(H(X) + H(Y))/2}$$

where $H(X)$ is the entropy of random variable *X* associated with partition *X* and $H(Y)$ is the entropy of random variable *Y* associated with partition *Y*, whereas $H(X, Y)$ is the joint entropy. NMI equals 1 when both partitions *X* and *Y* coincide.

However, this definition does not extends to the case where clusters in the partition overlap with one another. For comparing partitions with overlapping classes/clusters, an extention of NMI has been proposed in [23], according to which,

$$NMI(X,Y) = 1 - \frac{1}{2}[H(X|Y)_{norm} + H(Y|X)_{norm}]$$

where

$$H(X|Y)_{norm} = \frac{1}{C_X}\sum_k \frac{min_{l\in\{1,2,\ldots|C_Y|\}} H(X_k|Y_l)}{H(X_k)}$$

and

$$H(Y|X)_{norm} = \frac{1}{C_Y}\sum_k \frac{min_{l\in\{1,2,\ldots|C_X|\}} H(Y_k|X_l)}{H(Y_k)}$$

Here, $H(X|Y)$ and $H(Y|X)$ are conditional entropies respectively and $|C_X|$ and $|C_Y|$ are the number of clusters/classes in partitions *X* and *Y* respectively.

F-measure is defined as the harmonic mean of Precision and Recall, where precision is the fraction of objects in the cluster that share the same class and recall is the fraction of objects in the class that share the same cluster. However, precision and recall lose their meaning in the overlapping setting because multiple-memberships of objects lead to ambiguity. A work around this problem has been used in the form of pairwise precision and pairwise recall in [24], [25].

- Pairwise Precision: fraction of object pairs co-occurring in the same cluster that also share at least one class.
- Pairwise Recall: fraction of object pairs that sharing at least one class that also co-occur in the same cluster

F-measure is then defined as:

$$F = \frac{2 \times Pairwise\ Precision \times Pairwise\ Recall}{Pairwise\ Precision + Pairwise\ Recall}$$

5 Conclusions

In this paper, we reviewed the overlapping co-clustering formulations for categorical data. We saw that often in practice, the constraints cannot be strict and require to be relaxed by introducing density or sparsity throttles. Overlapping co-clustering translates to finding bicliques when dealing with community detection in networks. Quality measures like NMI and F-measure cannot be directly applied to overlapping cases. Modifications to these measures need to consider the sharing of more than one classes by the objects. This review helps foster the learning of essential ingredients required to design a scalable overlapping co-clustering algorithm.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. J. Chen, O. R. Zaiane, and R. Goebel, "Detecting Communities in Large Networks by Iterative Local Expansion," in International Conference on Computational Aspects of Social Networks, 2009. CASON '09, 2009, pp. 105–112.
2. X. Wang, L. Tang, H. Gao, and H. Liu, "Discovering Overlapping Groups in Social Media," in 2010 IEEE International Conference on Data Mining, 2010, pp. 569–578.
3. V. Crescenzi, P. Merialdo, and P. Missier, "Clustering Web pages based on their structure," Data & Knowledge
4. L. Yi, B. Liu, and X. Li, "Eliminating Noisy Information in Web Pages for Data Mining," in Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 2003, pp. 296–305.
5. Y. Cheng and G. Church, "Biclustering of expression data," Proc Eighth Int Conf Intell Syst Mol Biol, vol. 8, pp. 93–103, Dec. 1999.
6. J. Yang, H. Wang, W. Wang, and P. Yu, "Enhanced biclustering on expression data," in Third IEEE Symposium on Bioinformatics and Bioengineering, 2003. Proceedings, 2003, pp. 321–327.
7. S. Har-Peled, P. Indyk, and R. Motwani, Approximate Nearest Neighbor: Towards Removing the Curse of Dimensionality. 2012.
8. Rajaraman and J. D. Ullman, Mining of Massive Datasets. New York, NY, USA: Cambridge University Press, 2011, p. 87.
9. Shrivastava, "Exact Weighted Minwise Hashing in Constant Time," arXiv:1602.08393 [cs], Feb. 2016.

10. Gionis, P. Indyk, and R. Motwani, "Similarity Search in High Dimensions via Hashing," in Proceedings of the 25th International Conference on Very Large Data Bases, San Francisco, CA, USA, 1999, pp. 518–529.
11. F. O. D. Franca, "Scalable Overlapping Co-clustering of Word-Document Data," in Eleventh International Conference on Machine Learning and Applications (ICMLA), 2012, vol. 1, pp. 464–467.
12. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic Co-clustering," in Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 2003, pp. 89–98.
13. F. O. de França, "A Hash-based Co-Clustering Algorithm for Categorical Data," arXiv:1407.7753 [cs], Jul. 2014.
14. T. George and S. Merugu, "A scalable collaborative filtering framework based on co-clustering," in Fifth IEEE International Conference on Data Mining (ICDM'05), 2005, p. 4 pp.-pp.
15. G. Karypis, "METIS and ParMETIS," in Encyclopedia of Parallel Computing, D. Padua, Ed. Springer US, 2011, pp. 1117–1124.
16. W. Lin, Y. Zhao, P. S. Yu, and B. Deng, "An Effective Approach on Overlapping Structures Discovery for Co-clustering," in Web Technologies and Applications, L. Chen, Y. Jia, T. Sellis, and G. Liu, Eds. Springer International Publishing, 2014, pp. 56–67.
17. D. Chakrabarti, S. Papadimitriou, D. S. Modha, and C. Faloutsos, "Fully Automatic Cross-associations," in Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 2004, pp. 79–88.
18. Long, Z. (Mark) Zhang, and P. S. Yu, "Co-clustering by Block Value Decomposition," in Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, New York, NY, USA, 2005, pp. 635–640.
19. S. Andrews, "In-Close2, a high performance formal concept miner," in Conceptual Structures for Discovering Knowledge : 19th International Conference on Conceptual Structures, ICCS 2011, Derby, UK, July 25-29, 2011. Proceedings, S. Andrews, S. Polovina, R. Hill, and B. Akhgar, Eds. Derby: Springer, 2011, pp. 50–62.
20. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "Trawling the Web for emerging cyber-communities," Computer Networks, vol. 31, no. 11–16, pp. 1481–1493, May 1999.
21. L. N. F. Ana and A. K. Jain, "Robust data clustering," in 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings, 2003, vol. 2, p. II-128-II-133 vol.2.
22. C.j. Van Rijsbergen, "Foundation of evaluation," Journal of Documentation, vol. 30, no. 4, pp. 365–373, Apr. 1974.
23. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," New J. Phys., vol. 11, no. 3, p. 33015, 2009.
24. Banerjee, C. Krumpelman, J. Ghosh, S. Basu, and R. J. Mooney, "Model-based Overlapping Clustering," in Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, New York, NY, USA, 2005, pp. 532–537.
25. S. Gregory, "A Fast Algorithm to Find Overlapping Communities in Networks," in Machine Learning and Knowledge Discovery in Databases, W. Daelemans, B. Goethals, and K. Morik, Eds. Springer Berlin Heidelberg, 2008, pp. 408–423.