

**GRACE: A General Graph Convolution Framework for Atributed Graph Clustering**

BARAKEEL FANSEU KAMHOUA∗, The Chinese University of Hong Kong, LIN ZHANG∗, The Hong Kong University of Science and Technology,   
KAILI MA, The Chinese University of Hong Kong,   
JAMES CHENG2, The Chinese University of Hong Kong,   
BO LI, The Hong Kong University of Science and Technology,   
BO HAN, Hong Kong Baptist University,

Attributed graph clustering (AGC) is an important problem in graph mining as more and more complex data in real-world have been represented in graphs with attributed nodes. While it is a common practice to leverage both attribute and structure information for improved clustering performance, most existing AGC algorithms consider only a speciic type of relations, which hinders their applicability to integrate various complex relations into node attributes for AGC. In this paper, we propose GRACE, an extended graph convolution framework for AGC tasks. Our framework provides a general and interpretative solution for clustering many diferent types of attributed graphs, including undirected, directed, heterogeneous and hyper attributed graphs. By building suitable graph Laplacians for each of the aforementioned graph types, GRACE can seamlessly perform graph convolution on node attributes to fuse all available information for clustering. We conduct extensive experiments on 14 real-world datasets of 4 diferent graph types. The experimental results show that GRACE outperforms the state-of-the-art AGC methods on the diferent graph types in terms of clustering quality, time, and memory usage.

CCS Concepts: · **Information systems** → **Clustering and classiication**.

Additional Key Words and Phrases: Attributed graph clustering; Graph convolution

1 INTRODUCTION

Attributed graphs can be used to represent complex data in many application domains including social networks, web pages, and recommendation systems. Nodes in an attributed graph are associated with a number of attributes that describe the characteristics of objects, and edges (which can be of diferent types) represent the relationships among the objects.

Attributed graph clustering (AGC) is one of the most signiicant graph mining problems with many applications. AGC aims at partitioning the nodes in an attributed graph into a number of clusters such that (1) nodes in the same cluster should share similar attributes, and (2) nodes in diferent clusters should be minimally connected,

∗Both authors contributed equally to this research and are listed in alphabetical order. 2Corresponding Author

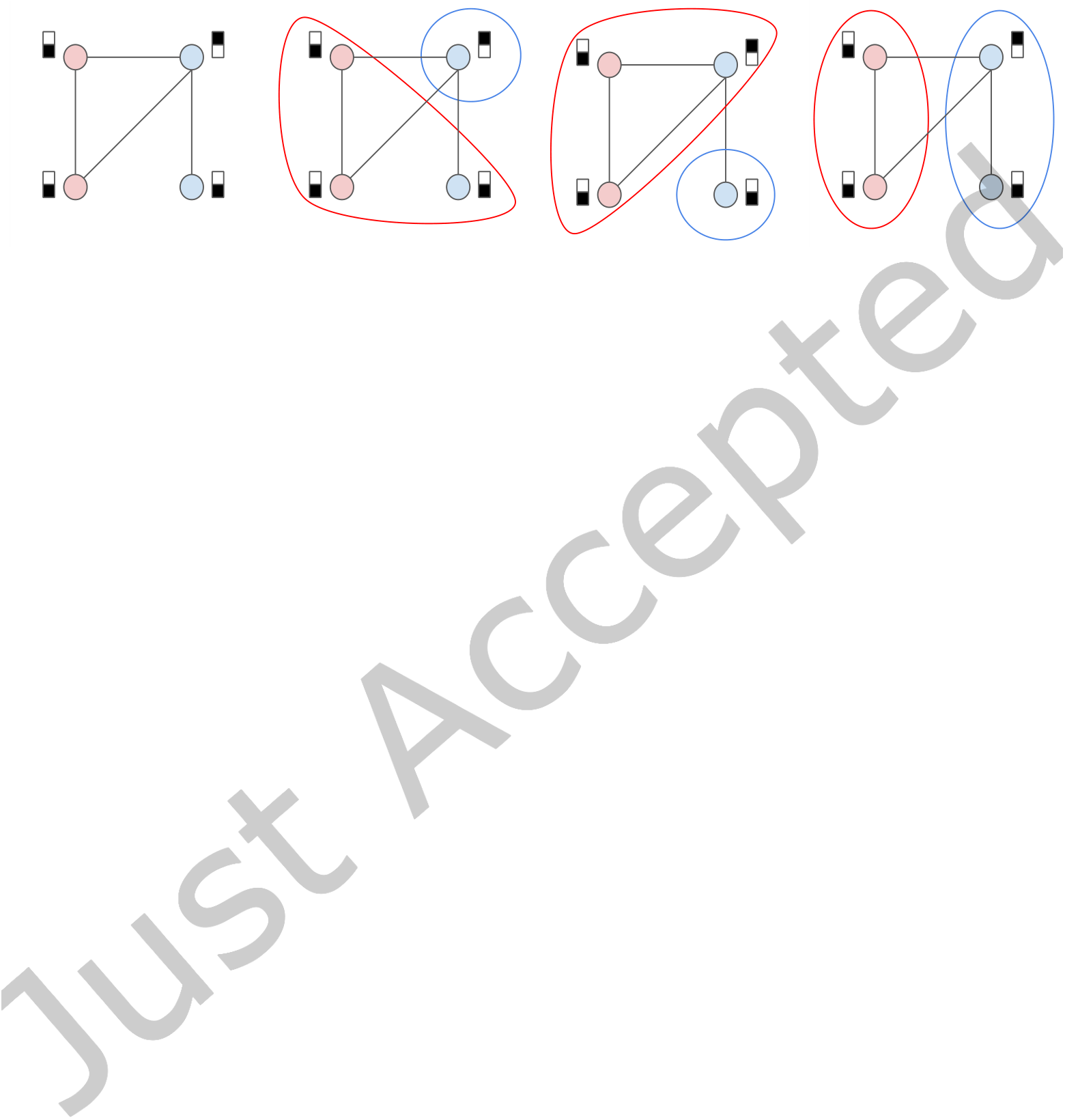
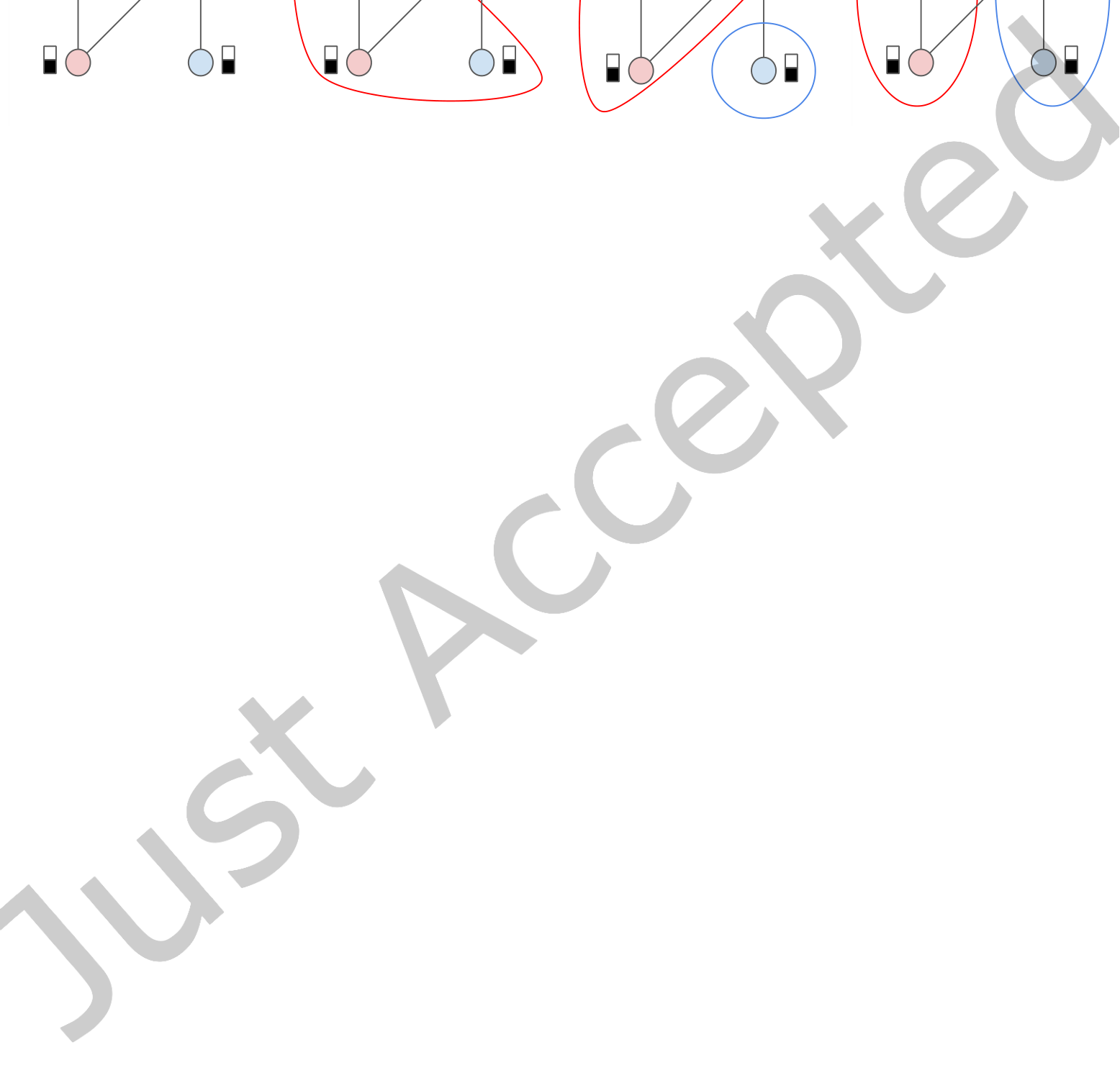
Authors’ addresses: Barakeel Fanseu Kamhoua, kamhoua@cse.cuhk.edu.hk, The Chinese University of Hong Kong, Hong Kong, ; Lin Zhang, lzhangbv@connect.ust.hk, The Hong Kong University of Science and Technology, Hong Kong, ; Kaili Ma, klma@cse.cuhk.edu.hk, The Chinese University of Hong Kong, Hong Kong, ; James Cheng, jcheng@cse.cuhk.edu.hk, The Chinese University of Hong Kong, Hong Kong, ; Bo Li, bli@cse.ust.hk, The Hong Kong University of Science and Technology, Hong Kong, ; Bo Han, bhanml@comp.hkbu.edu.hk, Hong Kong Baptist University, Hong Kong,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for proit or commercial advantage and that copies bear this notice and the full citation on the irst page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior speciic permission and/or a fee. Request permissions from permissions@acm.org.

© 2022 Association for Computing Machinery.

1556-4681/2022/6-ART $15.00   
<https://doi.org/10.1145/3544977>

ACM Trans. Knowl. Discov. Data.



|  |  |  |
| --- | --- | --- |
| 2 | • | Fanseu and Zhang et al. |

i.e., a cluster should have minimal edges connecting it to other clusters. Clustering results are very helpful to study and understand the graph data. For example, AGC can be used to discover social circles for social network analysis [34] or improve the search quality by ranking web pages associated with their cluster information [61].

|  |  |  |  |
| --- | --- | --- | --- |
| (a) Attributed Graph | (b) Attribute-based | (c) Structure-based | (d) Attribute & Structure |

Fig. 1. An example of an atributed graph with diferent clustering results. In (a), four nodes are connected with undirected edges and associated with two-dimensional atribute vectors. In (b), only node atributes are used for clustering, in (c) only the graph structure is used, and in (d) both node atributes and graph structure are used. The nodes are partitioned into two clusters (red and blue circles), and diferent node colors represent their ground truth classes.

However, clustering attributed graphs poses signiicant challenges. On the one hand, traditional methods that focus on either node attributes or graph structure can lead to poor clustering results, as only one type of information is utilized in clustering the graph. An illustrating example of AGC is given in Figure 1. In attribute-based clustering, as shown in Figure 1(b), nodes with the same attributes are grouped together but many connections exist between these two clusters. In structure-based clustering, as shown in Figure 1(c), nodes belonging to diferent clusters are minimally connected, but nodes within the same cluster have diferent attributes. In addition, the clustering results obtained from attribute-based or structure-based clustering are inconsistent with each other, and there is no simple solution to choose one of these two diferent results, nor to combine them [21].

On the other hand, diferent approaches that combine node attributes and graph structure information have been proposed to produce better clustering results [77, 84, 88]. As shown in Figure 1(d), such approaches return better clusters by balancing the attribute and structure similarities properly. However, existing AGC approaches are mainly designed for speciic types of attributed graphs and cannot be easily generalized to other types. For example, AGC methods designed for attributed graphs with simple undirected edges cannot be easily extended to capture asymmetric, higher-order, and heterogeneous relationships, which are represented by directed, hyper-, and multi-relational edges, respectively. This is because undirected edges are only used to represent symmetric pairwise relationships and do not capture higher order relationships, e.g., in a social network, friendship is undirected, while the inluencerśfollower relationship is directed, social tagging that involves a group of users is modelled by hyper-edges, and multi-relational edges can describe rich user interactions on diferent social media sites.

Some existing methods have been extended to handle more types of graphs, but the generalization to many types of graphs remains a challenging problem. For example, on heterogeneous graphs, the recent state-of-the-art SpectralMix [53] can be easily extended to undirected graphs as well as directed graphs by simply using a single adjacency in place of the sum in the structure part of its objective function. However, it is not clear how to extend it to hypergraphs since the structure part of its objective function assumes that an edge only connects exactly two nodes.

ACM Trans. Knowl. Discov. Data.

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 3 |

For AGC on undirected graphs (Undi-AGC), graph convolution-based models have been shown to achieve state-of-the-art performance1[49, 84]. In an efort to extend the graph convolution-based methods in order to tackle AGC on hyper-graphs (hyper-AGC), in our prior work [27] we provide solid theoretical analysis on the excellent performance of graph convolution on AGC for simple undirected graphs, and we extended the results to clustering hypergraphs and proposed a model (GRAC) for hyper-AGC.

In this paper, we further extend our prior work [27] to propose a general graph convolution-based framework, called GRACE2, for AGC on diferent types of edges, i.e., a general framework for clustering attributed graphs with directed, hyper-, or multi-relational edges (or any graphs with a well deined graph Laplacian).

GRACE is the irst graph convolution-based algorithm proposed to work on all types of graphs with a well deined Laplacian. The graph convolution is a iltering operation that is used to de-noise graph signals by allowing speciic graph signals contained in the node attributes to pass through, where graph signals correspond to columns of the node attribute matrix and the graph convolution ilter corresponds to functions of the graph Laplacian [58]. As such, GRACE consists mainly of two phases, a de-noising (graph convolution) phase and a clustering phase. To integrate both phases and decide how much de-noising is needed, GRACE utilizes the notion of cluster compactness on the de-noised node attributes (i.e., how close the nodes in the same cluster are to each other with respect to their node attributes).

Based on graph convolution, we irst show from a Graph Signal Processing perspective [58] that GRACE can be interpreted as a two-step algorithm guided by the compactness measure, with the irst step being the de-noising step and the second being the clustering step, where the compactness measure controls what level of de-noising is needed. Second, we show that GRACE is easy to understand with respect to its efects on AGC on diferent types of graphs. Speciically, GRACE helps ind the trade-of between ensuring that nodes in diferent clusters are as minimally connected as possible, while at the same time ensuring that nodes in the same cluster have attributes that are not too dissimilar to each other. In addition, GRACE is easy to implement as it only consists of two (easy-to-implement) phases, a graph convolution phase and then a clustering phase, both of which have a relatively low time complexity.

2 BACKGROUND

In this section, we introduce the background of attribute-based and structure-based clustering algorithms. We also give the preliminaries of graph convolution and four diferent types of graph structures. We summarize the frequently used notations in Table 1.

2.1 Atribute-based Clustering

Clustering attributed graphs involves both node attributes and graph structure to measure the similarity between diferent nodes. In attribute-based clustering, as shown in Fig. 1(b), each node

|  |  |  |
| --- | --- | --- |
| 4 | • | Fanseu and Zhang et al. |

Table 1. Notations

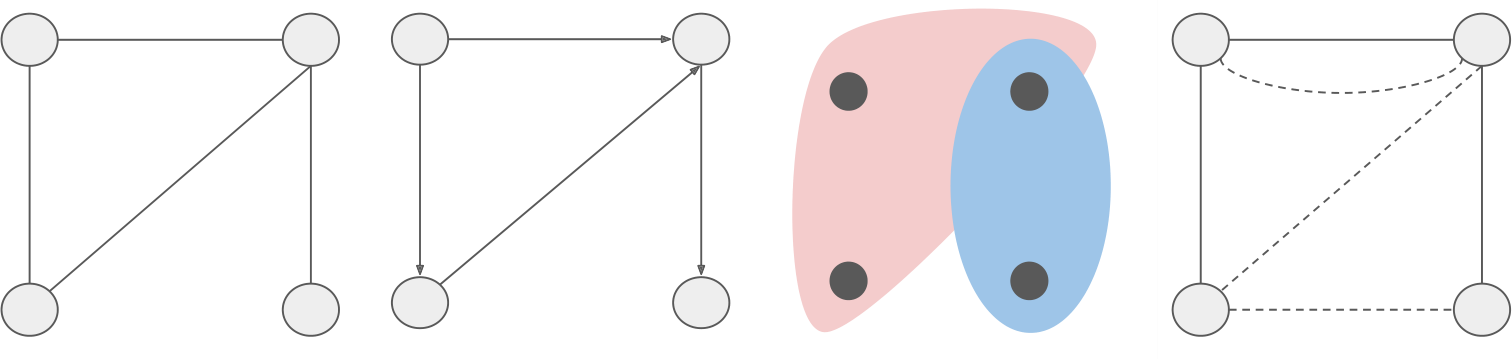
|  |  |
| --- | --- |
| Name | Description |

|  |  |
| --- | --- |
| V |  |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 5 |

In a similar manner, the optimal continuous solution of

|  |  |  |
| --- | --- | --- |
| 6 | • | Fanseu and Zhang et al. |



|  |  |  |  |
| --- | --- | --- | --- |
| (a) Undirected Graph | (b) Directed Graph | (c) Hyper Graph | (d) Multi-relational Graph |

Fig. 2. Examples of graph structures with four types of edges. In (a) and (b), pairwise nodes are connected with undirected and directed edges, respectively. In (c), hyper-edges are used to connect two or more nodes. In (d), solid lines and dashed lines represent two types of relations.

Otherwise,

|  |  |  |  |
| --- | --- | --- | --- |
| 4 | GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 7 |
| GRAPH CONVOLUTION BASED ATTRIBUTED GRAPH CLUSTERING |

In this section, we present a graph convolution based algorithm for Undi-AGC speciically. Then we provide a theoretical analysis on how graph convolution improves the clustering performance.

4.1 Overview

Given an attributed undirected graph

|  |  |  |
| --- | --- | --- |
| 8 | • | Fanseu and Zhang et al. |

**Algorithm 1** GRAC: Graph convolution based Undi-AGC algorithm

**Input:** Attributed graph

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 9 |

the solution

|  |  |  |
| --- | --- | --- |
| 10 | • | Fanseu and Zhang et al. |

Table 2. Graph Laplacians

|  |  |
| --- | --- |
| Name | Deinition |

|  |  |
| --- | --- |
| symmetric (undirected) graph Laplacian |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 11 |

To design an eicient hyper-graph convolution, we leverage the non-linear hyper-graph Laplacian [8, 27, 80]

by converting an attributed hyper-graph into an undirected graph as follows:

• We ind two farthest nodes inside each hyper-edge, i.e., (

|  |  |  |
| --- | --- | --- |
| 12 | • | Fanseu and Zhang et al. |

**Algorithm 2** GRACE: A general graph convolution framework for AGC

**Input:** Attributed graph

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 13 |

Table 3. Statistics of the datasets.

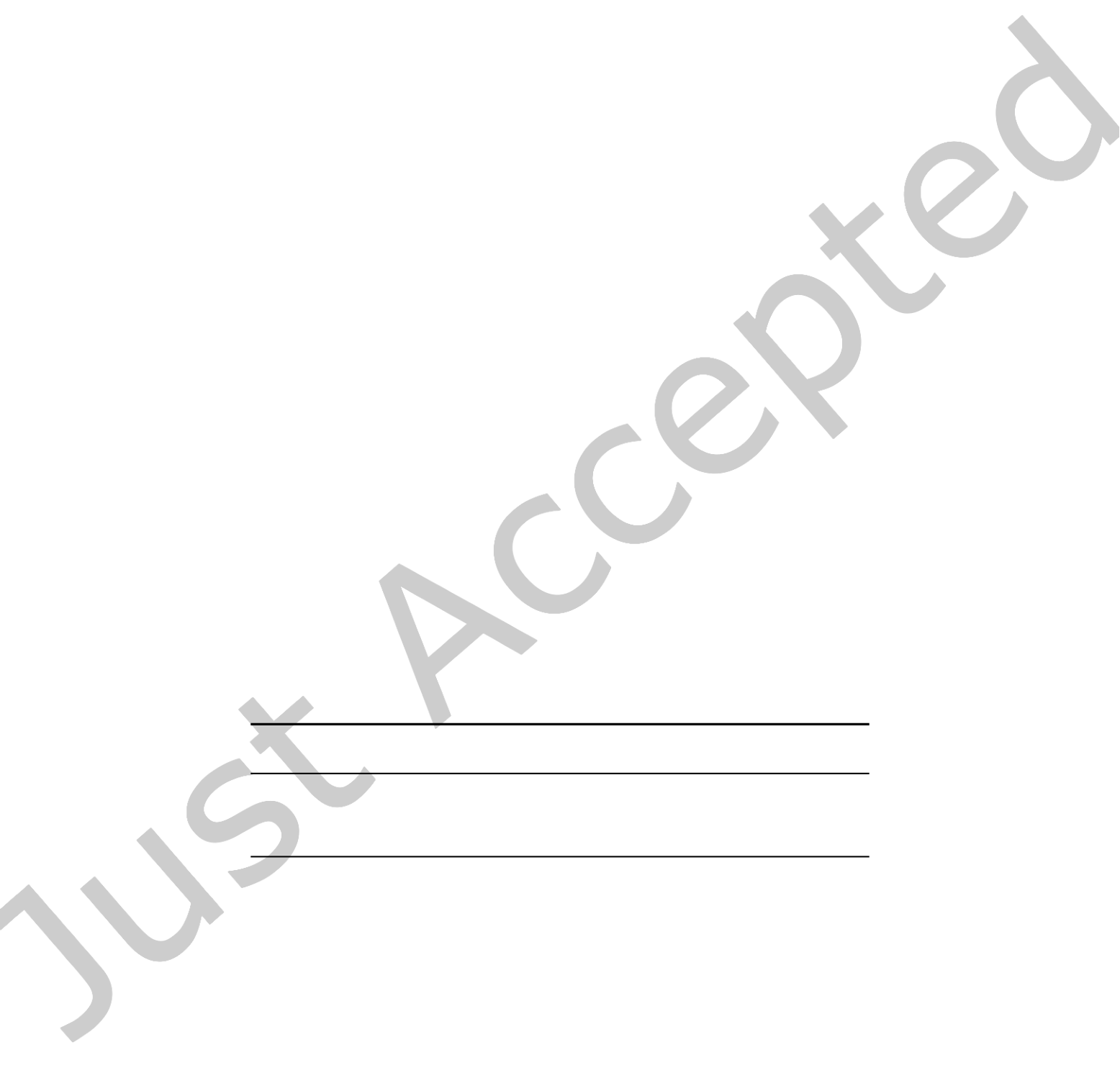
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Dataset | Type | #Node | #Edge | |R| | #Attribute | #Class |
|  | Undirected Cora | Undi | 2708 | 5429 | - | 1433 | 7 |
| Citeseer | Undi | 3327 | 4732 | - | 3703 | 6 |
| Pubmed | Undi | 19717 | 44338 | - | 500 | 3 |
| Wiki | Undi | 2405 | 17981 | - | 4973 | 17 |
|  | Cora-ML | Di | 2995 | 8416 | - | 2879 | 7 |
| Directed Citeseer | Di | 3312 | 4715 | - | 3703 | 6 |
|  | Cora Co-citation | Hyper | 2708 | 1579 | - | 1433 | 7 |
| Citeseer Co-citation | Hyper | 3327 | 1079 | - | 3703 | 6 |
| Pubmed Co-citation | Hyper | 19717 | 7963 | - | 500 | 3 |
| Cora Co-authorship | Hyper | 2708 | 1072 | - | 1433 | 7 |
| DBLP Co-authorship | Hyper | 43413 | 22535 | - | 1425 | 6 |
|  | ACM | MR | 8916 | 2240042 | 2 | 1870 | 3 |
| IMDB | MR | 9717 | 80216 | 2 | 2000 | 3 |
| Hete DBLP | MR | 26128 | 12055179 | 3 | 334 | 4 |
| 6 | EXPERIMENTS |  |  |  |  |  |  |

We conduct experiments to show the efectiveness, eiciency and generality of GRACE for clustering diferent types of attributed graphs.

6.1 Datasets

We used 14 datasets in our experiments with 4 diferent graph types: undirected (Undi), directed (Di), Hyper, and multi-relational (MR) graphs. The statistics of the datasets are summarized in Table 3. #Node and #Edge are the number of nodes and edges. |R| represents the number of edge types for MR graphs. #Attribute is the dimension of the node attribute vector. #Class is the number of classes i.e., the number of ground-truth labels.

**For undirected graphs**, Undirected Cora, Citeseer, Pubmed, and Wiki are commonly used to evaluate undirected AGC algorithms [66, 84]. The irst three datasets are citation networks, where nodes are publications and they are connected if one cites another (ignoring the directionality). Wiki is a web-page network whose nodes are websites and are connected if one links another. The nodes in Undirected Cora and Citeseer are associated with binary word vectors, while the nodes in Pubmed and Wiki are associated with tf-idf weighted word vectors. Pubmed is connected, i.e., there is a path from any node to any other node, while Undirected Cora, Citeseer and Wiki are not. Moreover, the edges of Undirected Cora, Citeseer, and Pubmed are not weighted, i.e.,



|  |  |  |
| --- | --- | --- |
| 14 | • | Fanseu and Zhang et al. |

**For multi-relational graphs**, we used three multi-relational graphs: ACM, IMDB, and Hete DBLP [53, 70], which are constructed from heterogeneous graphs. ACM consists of 3025 papers (P), 5835 authors (A) and 56 subjects (S). To cluster papers into diferent groups, we used the PAP (i.e., papers-authors-papers) and PSP (i.e., papers-subjects-papers) meta-paths to construct the two-type multi-relational (MR) graph for ACM. The IMDB consists of 3550 movies (M), 4441 actors (A), and 1726 directors (D). To cluster movies, we used the MAM and MDM meta-paths to construct the two-type multi-relational (MR) graph for IMDB. Finally, DBLP consists of 4057 authors (A), 14328 papers (P), 20 conferences (C), and 7723 terms (T). To cluster authors, we constructed the three-type multi-relational (MR) graph for DBLP based on the APA, APCPA, and APTPA meta-paths.

6.2 Evaluation Metrics and Experimental Setups

In our experiments, we used four unsupervised measures: IntraFD and InterED, and their approximations, i.e., IntraFD\_approx and InterED\_approx (see Section 4.3) to evaluate the clustering quality w.r.t. node attributes and graph structure, respectively. For all the measures, the lower their values the better is the quality of the clustering result. Lower IntraFD and IntraFD\_approx mean that nodes with similar attributes are more likely grouped in the same cluster, while lower InterED and InterED\_approx mean that less edges are connected between diferent clusters. In addition, we also used three commonly used supervised measures to evaluate the clustering results based on the ground-truth labels: clustering accuracy (Acc), normalized mutual information (NMI), and f1-macro score (F1) [49, 66, 67, 71, 74, 84]. For all supervised measures, a higher value indicates a better clustering quality.

We conducted all experiments on a Linux system with an Intel (R) Xeon (R) Silver 4114 CPU at 2.20GHz with 256GB of RAM, except the experiments with multi-relational graphs. This is because the state-of-the-art baseline SpectralMix [53] was implemented in Java3and incurred an execution error in our Linux server. To solve this issue, we ran all experiments for multi-relational graphs on a windows system with an Intel(R) Xeon(R) CPU E5-1620 v2 @ 3.70GHz with 16G of RAM. We report the running time (in seconds) averaged over 10 runs of each dataset (excluding data loading and pre-processing ), and the maximum process memory usage (in MB) of each algorithm. We will release the code and data for reproducibility of the results.

6.3 Baselines and Parameter setings

Table 4. Properties of the algorithms.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Baseline | Input | Undi | Di | Hyper | MR |
| K-Means | attribute | ✓ | ✓ | ✓ | ✓ |
| N-cut | structure | ✓ | ✓ | ✓ | ✓ |
| ARGE | both | ✓ | ✓ | ✓ | ✓ |
| ARVGE | both | ✓ |
| AdaGCN | both | ✓ |
| GNMF | both |
| ✓ | ✓ |
| JNMF | both |
| SpectralMix | both |
| GRACE | both | ✓ | ✓ | ✓ | ✓ |

We compared GRACE with the baselines on diferent types of graph structures. Their main characteristics are summarized in Table 4, and the parameter settings are introduced below.

3<https://gitlab.cs.univie.ac.at/yllis19cs/spectralmixpublic/-/tree/master/SpectralMix>

ACM Trans. Knowl. Discov. Data.

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 15 |

**K-means [44] and N-cut [26]** are clustering algorithms that use only node attribute or graph structure information. K-means can simply be applied on all datasets because it ignores any type of graph structures. N-cut was originally designed for clustering simple undirected graphs. However, as stated in Section 5, it can be used to cluster other types of graphs with properly designed graph Laplacians. Speciically, we selected the fast spectral normalized directed Laplacian [35]

|  |  |  |
| --- | --- | --- |
| 16 | • | Fanseu and Zhang et al. |

same weights to all relations (except when the user has prior knowledge of which relation is most important), while ensuring that�|R|



|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 17 |

the state-of-the-art AdaGCN algorithm can produce better clustering results with much less time and memory cost.

Following the idea of AdaGCN, GRACE exploits efective and eicient graph convolution on undirected graphs to combine node attribute and graph structure information. It remarkably outperforms K-Means and N-cut, and signiicantly saves time and memory cost compared to ARGE and ARVGE. Furthermore, it improves the state-of-the-art performance of AdaGCN while being comparable in terms of time and memory cost. Especially for the Wiki dataset, the improvements of GRACE are 15.6%, 10.1% and 7.7% in terms of Acc, NMI, and F1, respectively. Importantly, GRACE can deal with various types of graph structures (as shown below), while AdaGCN was proposed for undirected graphs only.

6.4.2 Performance on directed graphs. We then evaluated the clustering performance on two directed graphs, Cora-ML and Directed Citeseer, and their results are given in Table 7.

Table 7. Clustering performance on directed datasets.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Baselines |  |  | Cora-ML | | |  |  |  | Directed Citeseer | | |  |
|  | Acc% | NMI% | F1% | Time (Sec) | | Mem (MB) | Acc% | NMI% | F1% | Time (Sec) | | Mem (MB) |
| K-Means | 50.62 | 32.30 | 47.00 | 03.40 | | **285.10** | 41.06 | 18.74 | 37.99 | 04.98 | | **205.60** |
| N-cut | 52.35 | 32.56 | 48.58 | **00.37** | | 296.23 | 46.49 | 21.27 | 43.15 | **00.40** | | 307.53 |
| GNMF | 68.21 | 43.58 | 68.09 | 03.95 | | 240.93 | 63.86 | 36.97 | 60.57 | 10.23 | | 295.55 |
| JNMF | 50.62 | 24.79 | 48.68 | 64.52 | | 278.44 | 64.46 | 36.59 | 60.55 | 135.19 | | 330.49 |
| GRACE | **75.09** | **60.73** | **71.48** | | 22.79 | 309.64 | **68.69** | **42.97** | **63.79** | | 10.98 | 421.81 |

The table shows that neither K-Means (attribute-based) nor N-cut (structure-based) algorithm can produce satisfying clustering results on attributed directed graphs. N-cut on these two datasets always outperforms K-Means when the directional information of graph structure has been utilized. The GNMF and JNMF algorithms, which combine node attribute and graph structure information using their respective objective functions, can generally obtain better clustering results compared to the K-Means and N-cut competitors. The results from GNMF and JNMF are comparable to each other, while GNMF usually takes less time and memory due to its simpler objective function and hence faster update rule. By applying graph convolution on directed graphs, GRACE consistently achieves the best clustering performance. The improvements of GRACE in terms of NMI are 17.2% on Cora-ML and 6.0% on Directed Citeseer. Though GRACE takes more time and memory space compared to GNMF, the diferences are negligible as both directed datasets are very small. We will present the eiciency of GRACE later on larger hyper-graphs and multi-relational graphs.

6.4.3 Performance on hyper-graphs. We then analyzed the clustering results on two Co-citation datasets4(as shown in Table 8), and two Co-authorship datasets (as shown in Table 9).

Table 8. Clustering performance on Co-citation datasets.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Baselines |  |  | Cora Co-citation | | |  |  |  | Pubmed Co-citation | | |  |
|  | Acc% | NMI% | F1% | Time (Sec) | | Mem (MB) | Acc% | NMI% | F1% | Time (Sec) | | Mem (MB) |
| K-Means | 33.27 | 13.07 | 22.12 | **01.48** | | **179.64** | 59.47 | 31.10 | 58.14 | **02.64** | | **195.45** |
| N-cut | 30.02 | 01.12 | 07.39 | 02.41 | | 343.35 | 39.95 | 00.07 | 19.09 | 105.20 | | 12525.82 |
| GNMF | 43.91 | 22.58 | 40.56 | 20.92 | | 337.52 | 55.61 | 20.84 | 56.98 | 147.76 | | 8828.43 |
| JNMF | 45.31 | 23.01 | 42.61 | 36.95 | | 382.82 | 51.61 | 14.04 | 49.22 | 603.02 | | 10751.11 |
| GRACE | **55.39** | **34.65** | **50.50** | | 04.88 | 283.75 | **60.89** | **31.14** | **60.17** | | 04.23 | 448.30 |

4The comparison on the Citeseer Co-citation dataset is omitted, and it has very similar results as seen in [27].

ACM Trans. Knowl. Discov. Data.



|  |  |  |
| --- | --- | --- |
| 18 | • | Fanseu and Zhang et al. |

Table 9. Clustering performance on Co-authorship datasets.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Baselines |  |  | Cora Co-authorship | | |  |  |  | DBLP Co-authorship | | |  |
|  | Acc% | NMI% | F1% | Time (Sec) | | Mem (MB) | Acc% | NMI% | F1% | Time (Sec) | | Mem (MB) |
| K-Means | 33.27 | 13.07 | 22.12 | **01.46** | | **183.30** | 61.25 | 38.30 | 60.90 | **21.34** | | **1262.65** |
| N-cut | 29.25 | 02.00 | 09.50 | 07.42 | | 374.96 | 27.58 | 01.00 | 09.25 | 1751.96 | | 54187.64 |
| GNMF | 51.22 | 27.49 | 44.57 | 21.01 | | 327.47 | 63.71 | 47.91 | 58.38 | 1100.76 | | 41845.62 |
| JNMF | 49.41 | 28.63 | 44.29 | 11.16 | | 422.26 | 61.80 | 44.72 | 58.82 | 3642.91 | | 48216.73 |
| GRACE | **60.08** | **37.44** | **59.19** | | 11.10 | 276.61 | **64.49** | **56.27** | **65.82** | | 92.65 | 1763.06 |

From the tables, we observe that the attribute-based K-Means algorithm always outperforms the structure-based N-cut algorithm. This implies that node attributes are more informative than hyper-graph structure information for clustering on both the Co-citation and Co-authorship networks. Especially on the Pubmed Co-citation and DBLP Co-authorship datasets, N-cut produces very poor clustering results. Moreover, due to the dense resulting edges, and the time complexity of building the traditional hypergraph laplacian, N-cut can be very time and memory consuming on large hyper-graphs, e.g., Pubmed Co-citation and DBLP Co-authorship.

As the hyper-graph structure contains very little information that helps for clustering (as shown by the poor N-cut results), the GNMF and JNMF algorithms that simply combine the node attribute and hyper-graph structure information in their objective functions perform even worse than the attribute-based methods. For example, the performance of GNMF and JNMF are 10.3% and 17.1% lower than K-Means in terms of NMI on the Pubmed Co-citation dataset. Moreover, both GNMF and JNMF algorithms like N-cut also become time and memory consuming for large hyper-graphs, e.g., GNMF is about 50 times slower and takes about 30 times more memory than K-Means on the DBLP Co-authorship and the Pubmed Co-citation datasets.

GRACE consistently outperforms all the baselines on all hyper-graph datasets. Even for the Pubmed Co-citation network, it still well utilizes the inferior hyper-graph structure information to improve clustering performance of the attribute based clustering. GRACE outperforms the GNMF and JNMF algorithms by a large margin, e.g., 8.4% − 17.1% in terms of NMI, and is time and memory eicient even for large hyper-graphs. On the DBLP Co-authorship datasets, GRACE achieves the best performance, while using only 93 seconds and 1763 MB of memory, which are 39 times and 27 times less than JNMF.

6.4.4 Performance on multi-relational graphs. We inally compared the clustering performance on the three multi-relational graphs (their results are reported in Table 10). Due to the space limitation, we only present the running time on the largest Hete DBLP dataset.

Table 10. Clustering performance on multi-relational datasets.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Baselines |  | ACM |  |  | IMDB |  |  | Hete DBLP | | |  |
|  | Acc% | NMI% | F1% | Acc% | NMI% | F1% | Acc% | NMI% | | F1% | Time(Sec) |
| K-Means | 67.93 | 32.03 | 68.10 | 52.45 | 14.63 | 53.14 | 36.80 | 08.47 | 28.46 | | **02.41** |
| N-cut | 34.35 | 07.00 | 23.68 | 34.84 | 00.07 | 32.54 | 92.31 | 76.54 | 91.81 | | 03.42 |
| SpectralMix | **90.12** | **67.99** | **90.70** | 55.46 | **20.11** | 56.98 | 40.55 | 16.05 | 32.27 | | 22572.54 |
| GRACE | 88.89 | 65.08 | 89.08 | **62.87** | 18.44 | **62.95** | **92.33** | **76.68** | **91.83** | | 74.63 |

The table shows that the K-Means algorithm with only node attribute information produces inferior clustering results. The structure-based method, N-cut, performs poorly on ACM and IMDB but performs exceptionally well on Hete DBLP. This is because the three-type relations of Hete DBLP provide more information for clustering compared with the two-type relations of the ACM and IMDB datasets. The clustering accuracy of using N-cut on

ACM Trans. Knowl. Discov. Data.

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 19 |

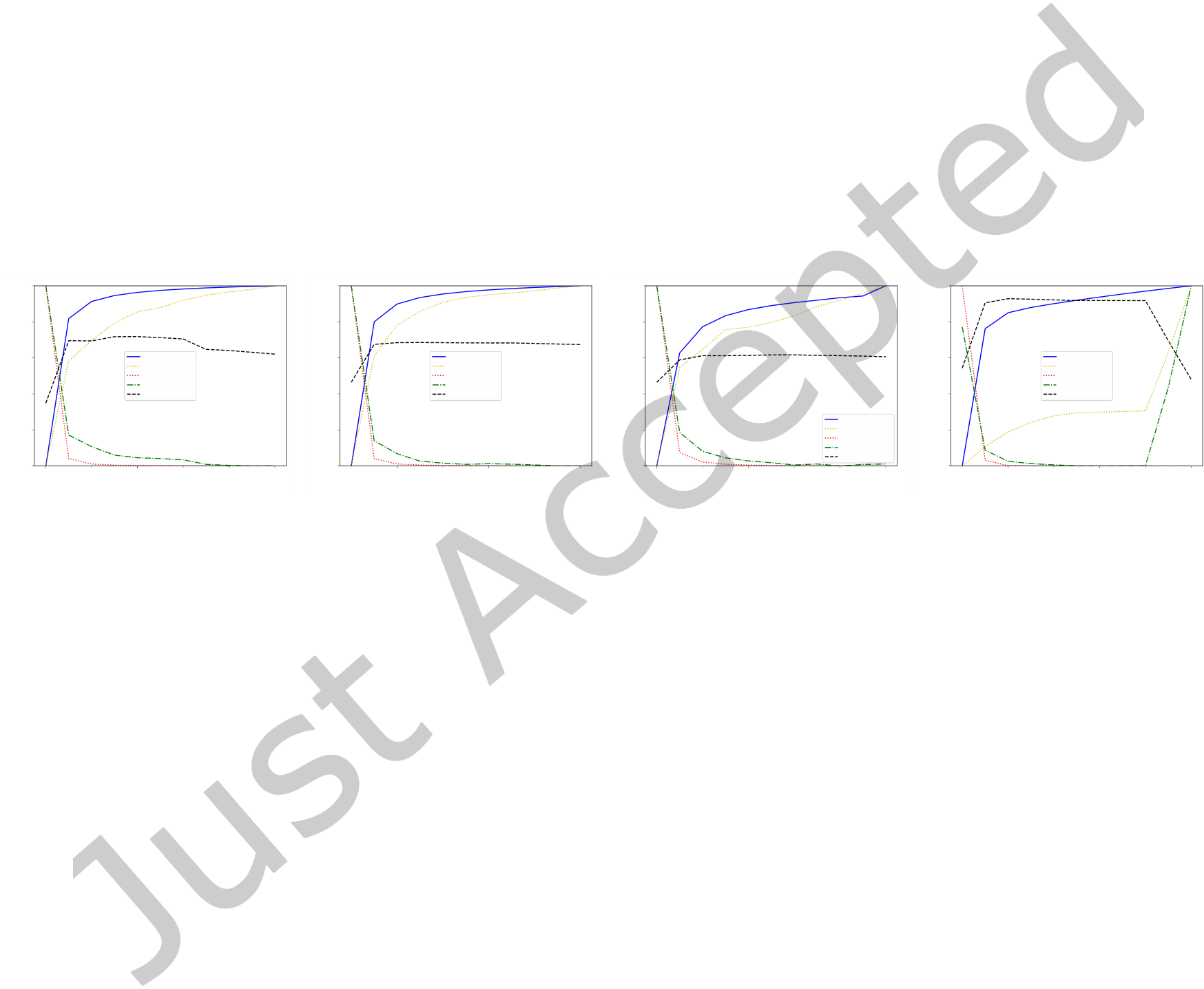
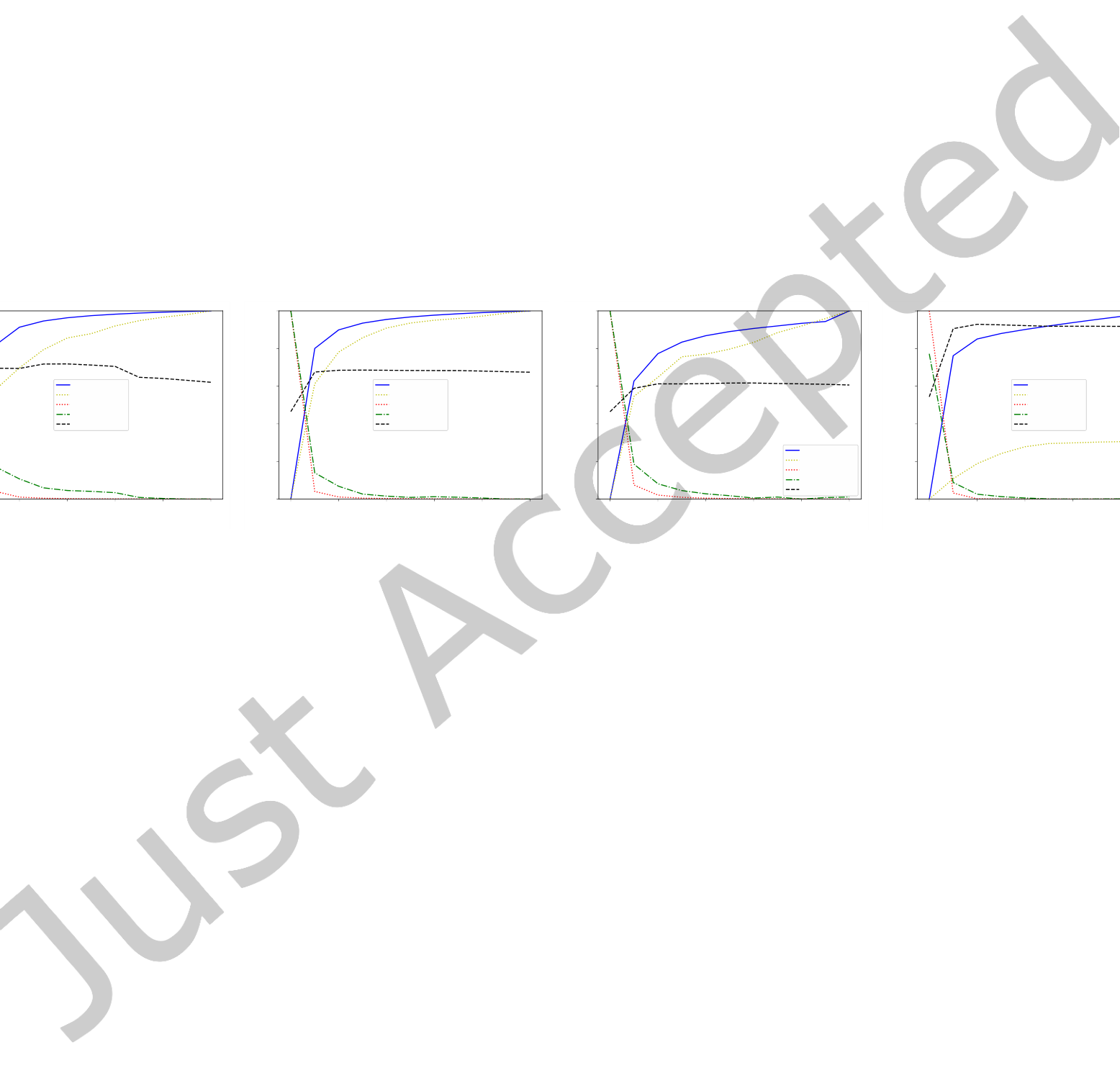
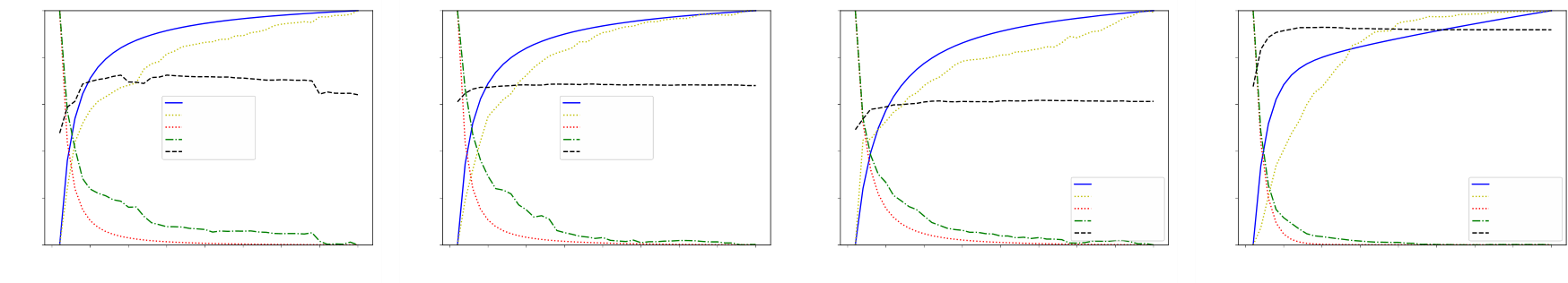
the adjacency matrix of each individual relation of Hete DBLP is, 30% for the APA relation, 88.4% for the APCPA relation and 52.7% for the APTPA relation, while that for the combined relations is 92.31% as shown in Table 10. In comparison, the clustering accuracy of using N-cut on each individual relation of ACM and IMDB is: (ACM) 34.8% for PAP and 35.1% for PSP; and (IMDB) 37.5% for MDM and 37.7% for MAM. The results show that the relations of Hete DBLP capture more information for clustering than those of ACM and IMDB do. The results also indicate that each relation of Hete DBLP captures diferent information and combining them gives better clustering results.

On ACM, the state-of-the-art method, SpectralMix, outperforms GRACE in terms of clustering quality. However, GRACE still achieves competitive clustering quality, and we remark that GRACE is orders of magnitude faster than SpralMix, which took 13,558 seconds compared to the 10 seconds of GRACE for clustering ACM. However, on IMDB and Hete DBLP, GRACE outperforms SpectralMix (with a signiicant margin on Hete DBLP). The performance diference on Hete DBLP is because SpectralMix takes into account the multi-relational graph structure and node attributes alternatively in its update rule, and as such, the contribution from the informative graph structure of Hete DBLP is weakened. By using multi-relational graph convolution, GRACE can achieve comparable performance on ACM and IMDB, and produce much better performance on Hete DBLP (51.8% improvement in terms of Acc) compared with SpectralMix. Furthermore, GRACE is about 300 times faster than SpectralMix on the large Hete DBLP dataset. Note that the running time of SpectralMix on ACM and IMDB are 13,557.88 and 51.89 seconds, respectively, while those of GRACE are 9.60 and 2.88 seconds, respectively.

6.4.5 Conclusions of performance comparison. In conclusion, the clustering results on various datasets show that GRACE generally outperforms other algorithms on all types of attributed graphs, while remaining competitively time and memory eicient compared to methods that do not combine node attribute and graph structure. By using graph convolution, GRACE can signiicantly outperform the methods that focus on either node attribute or graph structure, and perform much faster and takes much less memory compared to other competitors that combine the information in other manners. The results demonstrate that our graph convolution based framework works well on various graph structures. This in fact shows the capability and universality of GRACE in capturing complex and diverse real-world relationships.

6.5 Sensitivity Study

Now we discuss the sensitivity of GRACE’s hyper-parameters on diferent graph types. We report the trends of diferent measures (i.e., Acc, IntraFD, IntraFD\_approx, InterED, and InterED\_approx) when only varying the length of hops or the graph iltering rate using four representative datasets: Undirected Cora, Directed Citeseer, Citeseer Co-citation, and Hete DBLP. All measures (except Acc) are unit normalized to it in the range [0, 1] for a fair comparison. Moreover, same Laplacians, normalizations and



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 20 | • | Fanseu and Zhang et al. | | | | | | | | Unit Normalized Measures | 1.0 | 0 | IntraFD\_approx | | | | | | | 40 | Unit Normalized Measures | 1.0 | IntraFD\_approx | | | | | | | | 1.0 | 0 5 | 10 | 15 | 20 | 25 | 30 | IntraFD\_approx | |
| 1.0 |
| IntraFD\_approx | | | | | | | |
| 0.8 | 0.8 | 0.8 | 0.8 |
| 0.6 | 0.6 | 0.6 | 0.6 |
| IntraFD  InterED\_approx | | | | | | | | IntraFD  InterED\_approx | | | | | | |
| 0.4 | InterED | | | | | | | | 0.4 | InterED | | | | | | | 0.4 | 0.4 |
| Accuracy | | | | | | | | Accuracy | | | | | | |
| 0.2 | 0.2 | 0.2 | 0.2 |
| IntraFD | | | | | | | | IntraFD | |
| Unit Normalized Measures | 0 | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 0.0 | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 0.0 | InterED\_approx | | | | | | | | Unit Normalized Measures | InterED\_approx | |
| InterED | | | | | | | | InterED | |
| 0.0 | Accuracy | | | | | | | | 0.0 | Accuracy | |
| 0 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 35 | 40 |
| number of hops (L) | | | | | | | | number of hops (L) | | | | | | | number of hops (L) | | | | | | | | number of hops (L) | | | | |
| (a) Undirected Cora | | | | | | | | (b) Directed Citeseer | | | | | | | (c) Citeseer Co-citation | | | | | | | | (d) Hete DBLP | | | | |

Fig. 4. Efects of the length of hops.

important to determine a proper length of hops to achieve the best performance. As such, we will later check the efectiveness of the stopping algorithm used in GRACE in Section 6.6.

6.5.2 Sensitivity study on the graph filtering rate. We then investigate the impact of the graph iltering rate while ixing the length of hops as 25. The results are given in Figure 5.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1.0 | IntraFD\_approx | 1.0 | IntraFD\_approx | 1.0 | 1.0 | IntraFD\_approx |
| 0.8 | 0.8 | 0.8 | 0.8 |
| 0.6 | 0.6 | 0.6 | 0.6 |
| IntraFD | IntraFD | IntraFD |
| 0.4 | InterED\_approx | 0.4 | InterED\_approx | 0.4 | 0.4 | InterED\_approx |
| InterED | InterED | InterED |
| Accuracy | Accuracy | Accuracy |

IntraFD\_approx

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0.2 | Unit Normalized Measures | 0.2 | Unit Normalized Measures | 0.2 | IntraFD | 0.2 |
| Unit Normalized Measures | InterED\_approx | Unit Normalized Measures |

InterED   
Accuracy

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.0 | 0.0 | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 | 0.0 | 0.0 | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 | 0.0 | 0.0 | 0.2 | 0.4 | 0.6 | 0.8 | 1.0 | 0.0 | 0.0 | 0.2 | 0.4 | 0.6 | 0.8  ) | 1.0 |
| graph filtering rate (alpha) | | | | graph filtering rate (alpha) | | | | graph filtering rate (alpha) | | | | | graph filtering rate (alpha) | | |
| (a) Undirected Cora | | | | (b) Directed Citeseer | | | | (c) Citeseer Co-citation | | | | | (d) Hete DBLP | | |

Fig. 5. Efects of the graph filtering rate.

The igures on diferent graph types show that InterED\_approx and InterED decrease rapidly and gradually stabilize at 0 (except on Hete DBLP), while IntraFD\_approx and IntraFD start growing from 0 and inally stabilize as the graph iltering rate increases. As expected, the clustering performance in terms of Acc increases and then decreases. The trends it the analysis in Theorem 4.1 that graph convolution ilters the attributive noise guided by graph structure. The higher graph iltering rate improves the clustering quality in terms of structure but harms in terms of attribute. On Hete DBLP, InterED starts growing and being discrepant from InterED\_approx when the graph iltering rate becomes too large, which is caused by over-iltering or over-smoothing the node attributes. Therefore, we set a relatively small graph iltering rate and balance between the IntraFD and InterED measures by adjusting the length of hops.

6.6 Ablation Study

To measure the impact of the stopping algorithm and graph Laplacian selection, we conduct ablation studies on each component following the same experimental settings in Section 6.5.

6.6.1 Ablation study on the stopping algorithm. To measure the impact of the stopping algorithm, we let graph convolution run for 40 iterations, and report the trends of Acc, NMI, F1, and Compactness. We mark the iteration (with a pink line) when the stopping criterion is fulilled, i.e., the compactness of clusters changes very little. The results are reported in Figure 6.

ACM Trans. Knowl. Discov. Data.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1.0 | 1.0 | GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | | • | 21 |
| 1.0 | 1.0 |

Compactness

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0.8 | 0.8 | 0.8 | Accuracy | 0.8 |
| F1-score |

NMI

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0.6 | 0 | Compactness | | | | Unit Normalized Measures | 0.6 | 0 | Compactness | | | | Unit Normalized Measures | 0.6 | 0 | 10 | 20 | 30 | 40 | 0.6 | 0 | Compactness | | | |
| 0.4 | 0.4 | 0.4 | 0.4 |
| 0.2 Unit Normalized Measures | Accuracy | | | | 0.2 | Accuracy | | | | 0.2 | 0.2 Unit Normalized Measures | Accuracy | | | |
| F1-score | | | | F1-score | | | | F1-score | | | |
| 0.0 | NMI | | | | 0.0 | NMI | | | | 0.0 | 0.0 | NMI | | | |
| 10 | 20 | 30 | 40 | 10 | 20 | 30 | 40 | 10 | 20 | 30 | 40 |
| number of hops (L) | | | | number of hops (L) | | | | number of hops (L) | | | | number of hops (L) | | | |
| (a) Undirected Cora | | | | (b) Directed Citeseer | | | | (c) Citeseer Co-citation | | | | (d) Hete DBLP | | | |

Fig. 6. Efects of the stopping algorithm.

The igures show that the compactness of the clusters at each iteration indeed decreases exponentially. It can also be observed that as the compactness is deceasing, Acc, NMI, and F1 are all increasing and then reach the peak as the compactness stabilizes before they drop. Therefore, choosing the stopping criterion at the point when the compactness stabilizes enables us to pick a good balance point for the number of hops or iterations. The igures also validate the efectiveness of our stopping algorithm as the pink lines in the plots (corresponding to the number of hops chosen by our stopping algorithm) are always at the near-best Acc values. In some cases, as the graph iltering rate is set quite small, the performance will maintain after the pink lines, in which the early stopping algorithm helps save the running time by returning good clusters earlier.

6.6.2 Ablation study on graph Laplacian variants. We constructed variants of graph Laplacians to validate the efectiveness of GRACE’s graph Laplacians choices on diferent graph types. The results are given in Figure 7, where the green curves represent the Acc results of GRACE, and the red curves (and magenta curve in Figure 7(d)) for the graph Laplacian variants. We used the same alpha values for each dataset as in Section 6.3.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1.0 | 0 | without selfloops | | | | Acc | 1.0 | 0 | Fast spectral directed Laplacian | | | | Acc | 1.0 | hypergraph laplacian | | | | | 1.0 | 0 | 10 | equal weights [0.5, 0.5] | | |
| 0.8 | 0.8 | 0.8 | 0.8 |
| 0.6 | 0.6 | 0.6 | 0.6 |
| Acc | 0.4 | 0.4 | Acc |
| 0.4 | 0.4 |
| 0.2 | 0.2 | 0.2 | 0.2 | greater weight on PAP [0.7, 0.3] | | |
| 0.0 | with selfloops | | | | 0.0 | Symmetric directed Laplacian | | | | 0.0 | non-linear laplacian with mediators | | | | | 0.0 | greater weight on PLP [0.3, 0.7] | | |
| 10 | 20 | 30 | 40 | 10 | 20 | 30 | 40 | 0 | 10 | 20 | 30 | 40 | 20 | 30 | 40 |
| number of hops (L) | | | | number of hops (L) | | | | number of hops (L) | | | | | number of hops (L) | | |
| (a) Undirected Cora | | | | (b) Directed Citeseer | | | | (c) Citeseer Co-citation | | | | | (d) ACM | | |

Fig. 7. Comparison of diferent graph Laplacians on diferent graph types.

On Undirected Cora, we compared the performance of using the symmetric graph Laplacian

|  |  |  |
| --- | --- | --- |
| 22 | • | Fanseu and Zhang et al. |

eicient as shown in Tables 9 and 8. Finally, on ACM, the multi-relational graph Laplacian

|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 23 |

this category we have RankClus [60] which focuses on bi-typed heterogeneous networks by simultaneously addressing ranking and clustering on heterogenous graphs, NetClus [62] which extends RanClus to heterogenous graphs with more than two types of nodes, SI-Cluster [91] which introduces a vertex based similarity measure and proposes an algorithm to iteratively reine the clusters based on this measure, and NMF based methods [51].

There are two main drawbacks to these methods. First, the two sources of information, i.e., the node attributes and the graph structure, do not always contain exactly the same information, but rather are complementary to each other (as seen in Section 6, the results of K-Means are always diferent to those of N-cut). As such, focusing on one source of information only (e.g., graph structure) will neglect the information contained in the other (e.g., the node attributes). Moreover, once clusters are obtained independently from the diferent sources of information, it is not clear how to combine them both, nor which one to choose.

7.2 Methods that combine graph structure and node atributes

AGC methods that combine graph structure and node attributes can be categorized into four approaches: hybrid distance-based, probabilistic model-based, nonnegative matrix factorization (NMF)-based, and node embedding-based approaches.

Hybrid distance-based methods design a uniied distance that combines node attributes and graph structure information. For example, the linear combination of attributive distance (e.g., euclidean distance) and structural distance (e.g., the shortest path). Based on the hybrid distance, there are two ways to perform clustering: (1) we can directly apply distance-based clustering algorithms (e.g., K-Medoids or K-Means [44]); (2) or we can utilize the hybrid distance information to construct a new graph (e.g., a K-NN graph [12] or edge-weighted graph [48]), and then apply graph clustering algorithms. This category cannot easily be extended to other types of graphs because the explicit structural distance measures are usually designed for a speciic graph type, and to extend it to another graph type one will need to redeine the structural distance measure. For example, on undirected graphs, in SA-Cluster [89] and its extended versions [10, 90], the uniied neighborhood random walk distance is computed on a node-augmented graph with new attributive nodes and edges, and the K-Medoids algorithm is then applied to ind clusters. This new augmented graph is built on the assumption that the original graph structure was an undirected graph and it is not obvious how to build it on heterogeneous graphs or hypergraphs. Another example is the distance based state-of-the-art SpectralMix [53] on heterogeneous graphs. It can easily be reduced to work on undirected graphs, by setting the number of multi-relational graph adjacency matrices to 1 and using the undirected graph adjacency as the multi-relational graph adjacency matrix, but it is not clear how to extend it to hypergraphs, since in hypergraphs each edge can contain much more than a single node, but the structural distance used in SpectralMix’ joint objective only accommodates simple edges linking two nodes. On heterogeneous graphs, MvAGC [40] denoises the graph signals (node attributes) and then proposes a graph learning algorithm to learn node similarities and uses the resulting similarity graph for clustering. The main diference between MvAGC and GRACE is that in MvAGC, the convolution (denoising) module is not directly connected to the graph learning and clustering modules. Thus, although MvAGC presents a novel approach to clustering, i.e., denoising and learning similarities and then clustering, their work is not directed towards understanding the efects of graph convolution for clustering. One of the limitations of MvAGC is that the graph iltering phase, which is shown to be essential for clustering in our work, is only treated as a prepossessing phase, and hence it is diicult to see why the optimal

|  |  |  |
| --- | --- | --- |
| 24 | • | Fanseu and Zhang et al. |

Probabilistic model-based methods consider both node attributes and graph structure as probabilistic variables, and model the community memberships of each node as latent variables. From this view, attributed graph clustering can be transformed into a community membership estimation problem. There are three ways to model the statistical relationships among node attributes (X), graph structure (G) and community memberships (F): (1) node attributes generate community memberships and then community memberships generate graph structure (i.e.,



|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 25 |

social recommendation, one could use the combined user embedding for user clustering or the item embedding for item clustering. Compared with GRACE, it is non-trivial to generalize GraphRec to graph-types such as hypergraphs or multi-relational graphs, because in this case one would need to completely redesign the message passing scheme of GraphRec to adapt to multiple nodes in a single edge or message passing across diferent adjacency matrices. Moreover, it is also diicult to analyze what the efects of GraphRec on clustering may be (e.g., why and how it achieves a certain efect on the intra-cluster feature distance or inter-cluster edge density), even when it is paired with a suitable clustering loss function. This is because it is not clear how we may incorporate GraphRec’s message passing scheme in the analysis even if we follow a similar approach as we do in our paper. FeSoG [41] proposes to employ federated learning to address the privacy concerns in social recommendation based on the fact that most graph neural networks for this task require a centralized storage of the social links and item interactions of users. The attention based graph neural network proposed in FeSoG is similar to GraphRec (discussed above), since FesoG is an attention network with a similar procedure to learn the embedding (besides the federated learning it employs). Therefore, FeSoG shares the same limitations as GraphRec compared with GRACE. KCGN [23] addresses issues that most social recommendation neural networks do not handle, i.e., (1) the interaction between items, (2) the presence of multi-typed user item interactions, and (3) the dynamic nature of user-item interactions. To address these issues, KCGN designs a new message passing framework and a new loss function that incorporate (a) multi-typed dynamic user-item interaction encoding, and (b) knowledge-aware user-user and item-item inter-dependent relations, where (a) captures the dynamic relationship between user and items and (b) captures the local relationship among users and the local relationship among items. KGCN uses a message passing framework to fuse user-item interactions and a graph leaning framework to fuse user-user interactions and item-item interactions in a coupled way. In contrast, GRACE uses graph convolution to fuse both user-user interactions and user-item (user-attributes in our case) interactions and then capture the interdependence in both by building the similarity matrix for clustering. An advantage of GRACE has over KCGN (and the previous discussed models as well) for the clustering task is that, as discussed in Sections 4 and 6, we show the direct efects of graph convolution on the fundamental measures of clustering and can thus use graph convolution to directly guide the clustering, in addition to the fact that no training is needed. Moreover, GRACE is easily generalized to diferent graph types so long as we have a well deined laplacian and a suitable low-pass ilter, which is not the case for KCGN as it is hard to extend KCGN to another graph type such as hypergraphs. However, KCGN incorporates the dynamic nature of user-item interactions in their framework, which is an interesting direction to consider for GRACE as its future work.

Some major laws of these models are: (1) some of these models as noted are proposed speciically for speciic types of graphs and cannot be easily extended to other types, (2) Some of the state-of-the-art methods on each type of graphs (e.g., SpectralMix on heterogeneous graphs) are extremely time costly to run, (3) no study exists on how such a general method afects clustering on the diferent types of graphs and no performance nor analysis guarantees have been provided. As a result of these drawbacks, the beneits of diferent types of graph structures (undirected, directed, hyper, heterogeneous, etc.) used to store information have never been compared with each other. Thus, our aim in this work has been to address these challenges by (1) proposing an easy and general convolution framework for clustering on all types of graphs7, (2) proposing a convolution model GRACE based on the framework which is both fast and memory eicient, and (3) providing theoretical and experimental analysis and guaranties of our model and framework in general on all types of graphs.

7.3 Diferences between GRACE and GRAC

A preliminary work of this paper is GRAC [27], as GRACE can be considered as a generalization of GRAC. There are however key diferences between GRACE and GRAC.

7An additional beneit of our framework is that it can be utilized by any convolution-based algorithms or NMF-based algorithms.

ACM Trans. Knowl. Discov. Data.

|  |  |  |
| --- | --- | --- |
| 26 | • | Fanseu and Zhang et al. |

First, GRACE is a framework for handling AGC on diferent types of attributed graphs with diferent types of edges, while GRAC was tailored speciically for hypergraphs. To achieve this, we use the convolution proposed by GRAC and propose GRACE as a general framework. We make use diferent ilters for GRACE for each diferent type of graphs in order to address the diferent challenges imposed by diferent types of relations (edges) on the diferent types of graphs.

Second, We provide theoretical guarantees for the efects of GRACE on AGC by following GRAC in analyzing the efects of the graph convolution on AGC. We modify the analysis from GRAC. Speciically, we modify the efects of the convolution on IntraFD. This can be seen by the fact that Equation (11) in this work is diferent from Equation (9) in [27]. This leads to the more precise conclusion that when



|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 27 |

[2] Esra Akbas and Peixiang Zhao. 2017. Attributed Graph Clustering: an Attribute-aware Graph Embedding Approach. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, Sydney, Australia, July 31 - August 03, 2017, Jana Diesner, Elena Ferrari, and Guandong Xu (Eds.). ACM, 305ś308. <https://doi.org/10.1145/3110025.3110092>  
[3] Reid Andersen, Fan Chung, and Kevin Lang. 2007. Using PageRank to Locally Partition a Graph. Internet Mathematics 4 (01 2007), 35ś64.

<https://doi.org/10.1080/15427951.2007.10129139>  
[4] Song Bai, Feihu Zhang, and Philip H. S. Torr. 2019. Hypergraph Convolution and Hypergraph Attention. CoRR abs/1901.08150 (2019).

arXiv[:1901.08150](https://arxiv.org/abs/1901.08150) <http://arxiv.org/abs/1901.08150>  
[5] Mikhail Belkin and Partha Niyogi. 2003. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. Neural Comput.

15, 6 (2003), 1373ś1396. <https://doi.org/10.1162/089976603321780317>  
[6] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang. 2010. Graph regularized nonnegative matrix factorization for data representation. IEEE transactions on pattern analysis and machine intelligence 33, 8 (2010), 1548ś1560.

[7] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S. Huang. 2011. Graph Regularized Nonnegative Matrix Factorization for Data Representation. IEEE Trans. Pattern Anal. Mach. Intell. 33, 8 (2011), 1548ś1560. <https://doi.org/10.1109/TPAMI.2010.231>  
[8] T.-H. Hubert Chan and Zhibin Liang. 2020. Generalizing the hypergraph Laplacian via a difusion process with mediators. Theor.

Comput. Sci. 806 (2020), 416ś428. <https://doi.org/10.1016/j.tcs.2019.07.024>  
[9] Siheng Chen, Aliaksei Sandryhaila, José M. F. Moura, and Jelena Kovacevic. 2014. Signal denoising on graphs via graph iltering. In 2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP). 872ś876. <https://doi.org/10.1109/GlobalSIP.2014.7032244>[10] Hong Cheng, Yang Zhou, and Jefrey Xu Yu. 2011. Clustering Large Attributed Graphs: A Balance between Structural and Attribute Similarities. ACM Trans. Knowl. Discov. Data 5, 2 (2011), 12:1ś12:33. <https://doi.org/10.1145/1921632.1921638>  
[11] Fan Chung. 2005. Laplacians and the Cheeger Inequality for Directed Graphs. Annals of Combinatorics 9 (04 2005), 1ś19. [https:](https://doi.org/10.1007/s00026-005-0237-z)  [//doi.org/10.1007/s00026-005-0237-z](https://doi.org/10.1007/s00026-005-0237-z)   
[12] TA Dang and Emmanuel Viennet. 2012. Community detection based on structural and attribute similarities. In International conference on digital society (icds). 7ś12.

[13] Olivier Delalleau, Yoshua Bengio, and Nicolas Le Roux. 2005. Eicient Non-Parametric Function Induction in Semi-Supervised Learning.

In Proceedings of the Tenth International Workshop on Artiicial Intelligence and Statistics (proceedings of the tenth international workshop on artiicial intelligence and statistics ed.). Society for Artiicial Intelligence and Statistics, 96ś103.

[14] Inderjit S Dhillon, Yuqiang Guan, and Brian Kulis. 2004. Kernel k-means: spectral clustering and normalized cuts. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. 551ś556.

[15] Chris Ding and Xiaofeng He. 2004. K-means clustering via principal component analysis. In Proceedings of the twenty-irst international conference on Machine learning. 29.

[16] Chris Ding, Xiaofeng He, and Horst D Simon. 2005. On the equivalence of nonnegative matrix factorization and spectral clustering. In Proceedings of the 2005 SIAM international conference on data mining. SIAM, 606ś610.

[17] Rundong Du, Barry L. Drake, and Haesun Park. 2017. Hybrid Clustering based on Content and Connection Structure using Joint Nonnegative Matrix Factorization. CoRR abs/1703.09646 (2017). arXiv[:1703.09646](https://arxiv.org/abs/1703.09646) <http://arxiv.org/abs/1703.09646>  
[18] Wenqi Fan, Yao Ma, Qing Li, Jianping Wang, Guoyong Cai, Jiliang Tang, and Dawei Yin. 2022. A Graph Neural Network Framework for Social Recommendations. IEEE Transactions on Knowledge and Data Engineering 34, 5 (2022), 2033ś2047. [https://doi.org/10.1109/TKDE.](https://doi.org/10.1109/TKDE.2020.3008732)

[2020.3008732](https://doi.org/10.1109/TKDE.2020.3008732)   
[19] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2018. Hypergraph Neural Networks. AAAI 2019 (2018). [20] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

[21] Ting Guo, Shirui Pan, Xingquan Zhu, and Chengqi Zhang. 2018. CFOND: consensus factorization for co-clustering networked data. IEEE Transactions on Knowledge and Data Engineering 31, 4 (2018), 706ś719.

[22] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous Graph Transformer. CoRR abs/2003.01332 (2020).

arXiv[:2003.01332](https://arxiv.org/abs/2003.01332) <https://arxiv.org/abs/2003.01332>  
[23] Chao Huang, Huance Xu, Yong Xu, Peng Dai, Lianghao Xia, Mengyin Lu, Liefeng Bo, Hao Xing, Xiaoping Lai, and Yanfang Ye. 2021.

Knowledge-aware Coupled Graph Neural Network for Social Recommendation. Proceedings of the AAAI Conference on Artiicial Intelligence 35, 5 (May 2021), 4115ś4122. <https://ojs.aaai.org/index.php/AAAI/article/view/16533>  
[24] Po-Yao Huang, Robert Frederking, et al. 2019. RWR-GAE: Random Walk Regularization for Graph Auto Encoders. arXiv preprint arXiv:1908.04003 (2019).

[25] Zhichao Huang, Yunming Ye, Xutao Li, Feng Liu, and Huajie Chen. 2017. Joint Weighted Nonnegative Matrix Factorization for Mining Attributed Graphs. In Advances in Knowledge Discovery and Data Mining - 21st Paciic-Asia Conference, PAKDD 2017, Jeju, South Korea, May 23-26, 2017, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 10234), Jinho Kim, Kyuseok Shim, Longbing Cao, Jae-Gil Lee, Xuemin Lin, and Yang-Sae Moon (Eds.). 368ś380. <https://doi.org/10.1007/978-3-319-57454-7_29>  
[26] Jianbo Shi and J. Malik. 2000. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 8 (2000), 888ś905. <https://doi.org/10.1109/34.868688>

ACM Trans. Knowl. Discov. Data.



|  |  |  |
| --- | --- | --- |
| 28 | • | Fanseu and Zhang et al. |

[27] Barakeel Fanseu Kamhoua, Lin Zhang, Kaili Ma, James Cheng, Bo Li, and Bo Han. 2021. HyperGraph Convolution Based Attributed HyperGraph Clustering. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management.

[28] Thomas N Kipf and Max Welling. 2016. Variational Graph Auto-Encoders. NIPS Workshop on Bayesian Deep Learning (2016). [29] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classiication with Graph Convolutional Networks. In International Conference on Learning Representations (ICLR).

[30] Da Kuang, Chris Ding, and Haesun Park. 2012. Symmetric nonnegative matrix factorization for graph clustering. In Proceedings of the 2012 SIAM international conference on data mining. SIAM, 106ś117.

[31] Tarun Kumar, Sankaran Vaidyanathan, Harini Ananthapadmanabhan, Srinivasan Parthasarathy, and Balaraman Ravindran. 2018.

Hypergraph Clustering: A Modularity Maximization Approach. CoRR abs/1812.10869 (2018). arXiv[:1812.10869](https://arxiv.org/abs/1812.10869) [http://arxiv.org/abs/1812.](http://arxiv.org/abs/1812.10869)

[10869](http://arxiv.org/abs/1812.10869)   
[32] Daniel D. Lee and H. Sebastian Seung. 2000. Algorithms for Non-negative Matrix Factorization. In Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA, Todd K. Leen, Thomas G. Dietterich, and Volker Tresp (Eds.). MIT Press, 556ś562. <http://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization>[33] Marius Leordeanu and Cristian Sminchisescu. 2012. Eicient Hypergraph Clustering. AISTATS (01 2012).

[34] Jure Leskovec and Julian Mcauley. 2012. Learning to Discover Social Circles in Ego Networks. In Advances in Neural Information Processing Systems, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.), Vol. 25. Curran Associates, Inc., 539ś547.

[https://proceedings.neurips.cc/paper/2012/ile/7a614fd06c325499f1680b9896beedeb-Paper.pdf](https://proceedings.neurips.cc/paper/2012/file/7a614fd06c325499f1680b9896beedeb-Paper.pdf)   
[35] Chensheng Li, Xiaowei Qin, Xiaodong Xu, Dujia Yang, and Guo Wei. 2020. Scalable Graph Convolutional Networks With Fast Localized Spectral Filter for Directed Graphs. IEEE Access 8 (2020), 105634ś105644. <https://doi.org/10.1109/ACCESS.2020.2999520>  
[36] Jianxin Li, Hao Peng, Yuwei Cao, Yingtong Dou, Hekai Zhang, Philip S. Yu, and Lifang He. 2021. Higher-Order Attribute-Enhancing Heterogeneous Graph Neural Networks. CoRR abs/2104.07892 (2021). arXiv[:2104.07892](https://arxiv.org/abs/2104.07892) <https://arxiv.org/abs/2104.07892>[37] Pan Li and Olgica Milenkovic. 2017. Inhomogeneous Hypergraph Clustering with Applications. In Advances in Neural Information Processing Systems, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc., 2308ś2318. [https://proceedings.neurips.cc/paper/2017/ile/a50abba8132a77191791390c3eb19fe7-Paper.pdf](https://proceedings.neurips.cc/paper/2017/file/a50abba8132a77191791390c3eb19fe7-Paper.pdf) [38] Q. Li, Z. Han, and X.-M. Wu. 2018. Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning. In The Thirty-Second AAAI Conference on Artiicial Intelligence. AAAI.

[39] Xiang Li, Ben Kao, Zhaochun Ren, and Dawei Yin. 2019. Spectral Clustering in Heterogeneous Information Networks. In The Thirty-Third AAAI Conference on Artiicial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artiicial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artiicial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019. AAAI Press, 4221ś4228. <https://doi.org/10.1609/aaai.v33i01.33014221>  
[40] Zhiping Lin and Zhao Kang. 2021. Graph Filter-based Multi-view Attributed Graph Clustering. In Proceedings of the Thirtieth International Joint Conference on Artiicial Intelligence, IJCAI-21, Zhi-Hua Zhou (Ed.). International Joint Conferences on Artiicial Intelligence Organization, 2723ś2729. <https://doi.org/10.24963/ijcai.2021/375> Main Track.

[41] Zhiwei Liu, Liangwei Yang, Ziwei Fan, Hao Peng, and Philip S. Yu. 2021. Federated Social Recommendation with Graph Neural Network. ACM Trans. Intell. Syst. Technol. (nov 2021). <https://doi.org/10.1145/3501815> Just Accepted.

[42] Ulrike Luxburg. 2004. A Tutorial on Spectral Clustering. Statistics and Computing 17 (01 2004), 395ś416. [https://doi.org/10.1007/s11222-](https://doi.org/10.1007/s11222-007-9033-z) [007-9033-z](https://doi.org/10.1007/s11222-007-9033-z)   
[43] Yi Ma, Jianye Hao, Yaodong Yang, Han Li, Junqi Jin, and Guangyong Chen. 2019. Spectral-based Graph Convolutional Network for Directed Graphs. CoRR abs/1907.08990 (2019). arXiv[:1907.08990](https://arxiv.org/abs/1907.08990) <http://arxiv.org/abs/1907.08990>  
[44] James MacQueen et al. 1967. Some methods for classiication and analysis of multivariate observations. In Proceedings of the ifth Berkeley symposium on mathematical statistics and probability, Vol. 1. Oakland, CA, USA, 281ś297.

[45] James MacQueen et al. 1967. Some methods for classiication and analysis of multivariate observations. In Proceedings of the ifth Berkeley symposium on mathematical statistics and probability, Vol. 1. Oakland, CA, USA, 281ś297.

[46] Fragkiskos D. Malliaros and Michalis Vazirgiannis. 2013. Clustering and Community Detection in Directed Networks: A Survey. CoRR abs/1308.0971 (2013). arXiv[:1308.0971](https://arxiv.org/abs/1308.0971) <http://arxiv.org/abs/1308.0971>  
[47] Julian J. McAuley and Jure Leskovec. 2012. Learning to Discover Social Circles in Ego Networks. In Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States, Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q.

Weinberger (Eds.). 548ś556. <http://papers.nips.cc/paper/4532-learning-to-discover-social-circles-in-ego-networks>  
[48] Jennifer Neville, Micah Adler, and David Jensen. 2003. Clustering relational data using attribute and link information. In In Proceedings of the Text Mining and Link Analysis Workshop, 18th International Joint Conference on Artiicial Intelligence. 9ś15.

[49] Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. 2018. Adversarially Regularized Graph Autoencoder for Graph Embedding.. In IJCAI. 2609ś2615.

[50] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 701ś710.

ACM Trans. Knowl. Discov. Data.



|  |  |  |
| --- | --- | --- |
| GRACE: A General Graph Convolution Framework for Atributed Graph Clustering | • | 29 |

[51] Yaoyao Qin, Caiyan Jia, and Yafang Li. 2016. Community detection using nonnegative matrix factorization with orthogonal constraint. In 2016 Eighth International Conference on Advanced Computational Intelligence (ICACI). 49ś54. <https://doi.org/10.1109/ICACI.2016.7449802>[52] J.A. Rodriguez. 2002. On the Laplacian Eigenvalues and Metric Parameters of Hypergraphs. Linear and Multilinear Algebra 50, 1 (2002), 1ś14. <https://doi.org/10.1080/03081080290011692> arXiv[:https://doi.org/10.1080/03081080290011692](https://arxiv.org/abs/https://doi.org/10.1080/03081080290011692)   
[53] Ylli Sadikaj, Yllka Velaj, Sahar Behzadi, and Claudia Plant. 2021. Spectral Clustering of Attributed Multi-Relational Graphs. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (Virtual Event, Singapore) (KDD ’21). Association for Computing Machinery, New York, NY, USA, 1431ś1440. <https://doi.org/10.1145/3447548.3467381>  
[54] Shota Saito, Danilo P. Mandic, and Hideyuki Suzuki. 2017. Hypergraph p-Laplacian: A Diferential Geometry View. CoRR abs/1711.08171 (2017). arXiv[:1711.08171](https://arxiv.org/abs/1711.08171) <http://arxiv.org/abs/1711.08171>  
[55] Venu Satuluri and Srinivasan Parthasarathy. 2011. Symmetrizations for Clustering Directed Graphs (EDBT/ICDT ’11). Association for Computing Machinery, New York, NY, USA, 343ś354. <https://doi.org/10.1145/1951365.1951407>  
[56] Srijan Sengupta and Yuguo Chen. 2015. SPECTRAL CLUSTERING IN HETEROGENEOUS NETWORKS. Statistica Sinica 25, 3 (2015), 1081ś1106. <http://www.jstor.org/stable/24721222>  
[57] Jinghan Shi, Houye Ji, Chuan Shi, Xiao Wang, Zhiqiang Zhang, and Jun Zhou. 2020. Heterogeneous Graph Neural Network for Recommendation. CoRR abs/2009.00799 (2020). arXiv[:2009.00799](https://arxiv.org/abs/2009.00799) <https://arxiv.org/abs/2009.00799>  
[58] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. 2013. The emerging ield of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. IEEE Signal Processing Magazine 30, 3 (2013), 83ś98.

<https://doi.org/10.1109/MSP.2012.2235192>  
[59] Heli Sun, Fang He, Jianbin Huang, Yizhou Sun, Yang Li, Chenyu Wang, Liang He, Zhongbin Sun, and Xiaolin Jia. 2020. Network Embedding for Community Detection in Attributed Networks. ACM Trans. Knowl. Discov. Data 14, 3 (2020), 36:1ś36:25. [https:](https://doi.org/10.1145/3385415)  [//doi.org/10.1145/3385415](https://doi.org/10.1145/3385415)   
[60] Yizhou Sun, Jiawei Han, Peixiang Zhao, Zhijun Yin, Hong Cheng, and Tianyi Wu. 2009. RankClus: Integrating Clustering with Ranking for Heterogeneous Information Network Analysis (EDBT ’09). Association for Computing Machinery, New York, NY, USA, 565ś576.

<https://doi.org/10.1145/1516360.1516426>  
[61] Yizhou Sun, Yintao Yu, and Jiawei Han. 2009. Ranking-based clustering of heterogeneous information networks with star network schema. Ranking-Based Clustering of Heterogeneous Information Networks with Star Network Schema, 797ś806. [https://doi.org/10.1145/](https://doi.org/10.1145/1557019.1557107)  [1557019.1557107](https://doi.org/10.1145/1557019.1557107)   
[62] Yizhou Sun, Yintao Yu, and Jiawei Han. 2009. Ranking-Based Clustering of Heterogeneous Information Networks with Star Network Schema. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Paris, France) (KDD ’09). Association for Computing Machinery, New York, NY, USA, 797ś806. <https://doi.org/10.1145/1557019.1557107>  
[63] Yuuki Takai, Atsushi Miyauchi, Masahiro Ikeda, and Yuichi Yoshida. 2020. Hypergraph Clustering Based on PageRank. 1970ś1978.

<https://doi.org/10.1145/3394486.3403248>  
[64] Zekun Tong, Yuxuan Liang, Changsheng Sun, Xinke Li, David Rosenblum, and Andrew Lim. 2020. Digraph Inception Convolutional Networks. In Advances in Neural Information Processing Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 17907ś17918. [https://proceedings.neurips.cc/paper/2020/ile/cfb6e2288a630c2a787a64ccc67097c-](https://proceedings.neurips.cc/paper/2020/file/cffb6e2288a630c2a787a64ccc67097c-Paper.pdf) [Paper.pdf](https://proceedings.neurips.cc/paper/2020/file/cffb6e2288a630c2a787a64ccc67097c-Paper.pdf)   
[65] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks.

International Conference on Learning Representations (2018). <https://openreview.net/forum?id=rJXMpikCZ> accepted as poster. [66] Chun Wang, Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Attributed Graph Clustering: a Deep Attentional Embedding approach. In Proceedings of the Twenty-Eighth International Joint Conference on Artiicial Intelligence, Sarit Kraus (Ed.).

Association for the Advancement of Artiicial Intelligence (AAAI), United States of America, 3670ś3676. [https://doi.org/10.24963/ijcai.](https://doi.org/10.24963/ijcai.2019/509)

[2019/509](https://doi.org/10.24963/ijcai.2019/509)   
[67] Chun Wang, Shirui Pan, Guodong Long, Xingquan Zhu, and Jing Jiang. 2017. Mgae: Marginalized graph autoencoder for graph clustering. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. 889ś898.

[68] Daixin Wang, Peng Cui, and Wenwu Zhu. 2016. Structural Deep Network Embedding. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi (Eds.). ACM, 1225ś1234. <https://doi.org/10.1145/2939672.2939753>[69] Xiao Wang, Di Jin, Xiaochun Cao, Liang Yang, and Weixiong Zhang. 2016. Semantic Community Identiication in Large Attribute Networks. In Proceedings of the Thirtieth AAAI Conference on Artiicial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, Dale Schuurmans and Michael P. Wellman (Eds.). AAAI Press, 265ś271. <http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11964>[70] Xiao Wang, Nian Liu, Hui Han, and Chuan Shi. 2021. Self-supervised Heterogeneous Graph Neural Network with Co-contrastive Learning. CoRR abs/2105.09111 (2021). arXiv[:2105.09111](https://arxiv.org/abs/2105.09111) <https://arxiv.org/abs/2105.09111>  
[71] Joyce Whang, Rundong Du, Sangwon Jung, Geon Lee, Barry Drake, Qingqing Liu, Seonggoo Kang, and Haesun Park. 2020. MEGA: multi-view semi-supervised clustering of hypergraphs. Proceedings of the VLDB Endowment 13 (01 2020), 698ś711. [https://doi.org/10.](https://doi.org/10.14778/3377369.3377378)

[14778/3377369.3377378](https://doi.org/10.14778/3377369.3377378)

ACM Trans. Knowl. Discov. Data.



|  |  |  |
| --- | --- | --- |
| 30 | • | Fanseu and Zhang et al. |

[72] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying Graph Convolutional Networks. In Proceedings of the 36th International Conference on Machine Learning. PMLR, 6861ś6871.

[73] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. 2019. A Comprehensive Survey on Graph Neural Networks. CoRR abs/1901.00596 (2019). arXiv[:1901.00596](https://arxiv.org/abs/1901.00596) <http://arxiv.org/abs/1901.00596>  
[74] Rongkai Xia, Y. Pan, Lei Du, and J. Yin. 2014. Robust Multi-View Spectral Clustering via Low-Rank and Sparse Decomposition. In AAAI. [75] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How Powerful are Graph Neural Networks?. In International Conference on Learning Representations. <https://openreview.net/forum?id=ryGs6iA5Km>  
[76] Wei Xu, Xin Liu, and Yihong Gong. 2003. Document clustering based on non-negative matrix factorization. In SIGIR 2003: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 28 - August 1, 2003, Toronto, Canada, Charles L. A. Clarke, Gordon V. Cormack, Jamie Callan, David Hawking, and Alan F. Smeaton (Eds.). ACM, 267ś273.

<https://doi.org/10.1145/860435.860485>  
[77] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. 2012. A model-based approach to attributed graph clustering.

Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, SIGMOD ’12 (05 2012). [https://doi.org/10.1145/](https://doi.org/10.1145/2213836.2213894)  [2213836.2213894](https://doi.org/10.1145/2213836.2213894)   
[78] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. 2012. A model-based approach to attributed graph clustering. In Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24, 2012, K. Selçuk Candan, Yi Chen, Richard T. Snodgrass, Luis Gravano, and Ariel Fuxman (Eds.). ACM, 505ś516. [https://doi.org/10.1145/](https://doi.org/10.1145/2213836.2213894)  [2213836.2213894](https://doi.org/10.1145/2213836.2213894)   
[79] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. 2014. GBAGC: A General Bayesian Framework for Attributed Graph Clustering. ACM Trans. Knowl. Discov. Data 9, 1 (2014), 5:1ś5:43. <https://doi.org/10.1145/2629616>  
[80] Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha Talukdar. 2019. HyperGCN: A New Method For Training Graph Convolutional Networks on Hypergraphs. In Advances in Neural Information Processing Systems (NeurIPS) 32. Curran Associates, Inc., 1509ś1520.

[81] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y. Chang. 2015. Network Representation Learning with Rich Text Information. In Proceedings of the Twenty-Fourth International Joint Conference on Artiicial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015, Qiang Yang and Michael J. Wooldridge (Eds.). AAAI Press, 2111ś2117. <http://ijcai.org/Abstract/15/299>[82] Jaewon Yang, Julian J. McAuley, and Jure Leskovec. 2013. Community Detection in Networks with Node Attributes. In 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, December 7-10, 2013, Hui Xiong, George Karypis, Bhavani M. Thuraisingham, Diane J. Cook, and Xindong Wu (Eds.). IEEE Computer Society, 1151ś1156. <https://doi.org/10.1109/ICDM.2013.167>  
[83] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. 2009. Combining Link and Content for Community Detection: A Discriminative Approach. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Paris, France) (KDD ’09). Association for Computing Machinery, New York, NY, USA, 927ś936. <https://doi.org/10.1145/1557019.1557120>[84] Xiaotong Zhang, Han Liu, Qimai Li, and Xiao-Ming Wu. 2019. Attributed Graph Clustering via Adaptive Graph Convolution. 4327ś4333.

<https://doi.org/10.24963/ijcai.2019/601>  
[85] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. 2006. Learning with Hypergraphs: Clustering, Classiication, and Embedding. Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference, 1601-1608 (2007) 19, 1601ś1608.

[86] Dengyong Zhou, Jiayuan Huang, and Bernhard Scholkopf. 2005. Learning from labeled and unlabeled data on a directed graph. Proceedings of the 22nd international conference on Machine learning (2005).

[87] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, and Maosong Sun. 2018. Graph Neural Networks: A Review of Methods and Applications. CoRR abs/1812.08434 (2018). arXiv[:1812.08434](https://arxiv.org/abs/1812.08434) <http://arxiv.org/abs/1812.08434>  
[88] Y. Zhou, Hong Cheng, and J. X. Yu. 2009. Graph Clustering Based on Structural/Attribute Similarities. Proc. VLDB Endow. 2 (2009), 718ś729.

[89] Yang Zhou, Hong Cheng, and Jefrey Xu Yu. 2009. Graph Clustering Based on Structural/Attribute Similarities. Proc. VLDB Endow. 2, 1 (2009), 718ś729. <https://doi.org/10.14778/1687627.1687709>  
[90] Yang Zhou, Hong Cheng, and Jefrey Xu Yu. 2010. Clustering Large Attributed Graphs: An Eicient Incremental Approach. In ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010, Geofrey I. Webb, Bing Liu, Chengqi Zhang, Dimitrios Gunopulos, and Xindong Wu (Eds.). IEEE Computer Society, 689ś698. <https://doi.org/10.1109/ICDM.2010.41>[91] Yang Zhou and Ling Liu. 2013. Social Inluence Based Clustering of Heterogeneous Information Networks (KDD ’13). Association for Computing Machinery, New York, NY, USA, 338ś346. <https://doi.org/10.1145/2487575.2487640>

ACM Trans. Knowl. Discov. Data.