

Article

# RGCLN: Relational Graph Convolutional Ladder-Shaped Networks for Signed Network Clustering

Anping Song \*, Ruyi Ji, Wendong Qi and Chenbei Zhang

School of Computer Science and Engineering, Shanghai University, Shanghai 200444, China

\* Correspondence: [apsong@shu.edu.cn](mailto:apsong@shu.edu.cn)

**Abstract:** Node embeddings are increasingly used in various analysis tasks of networks due to their excellent dimensional compression and feature representation capabilities. However, most researchers' priorities have always been link prediction, which leads to signed network clustering being under-explored. Therefore, we propose an asymmetric ladder-shaped architecture called RGCLN based on multi-relational graph convolution that can fuse deep node features to generate node representations with great representational power. RGCLN adopts a deep framework to capture and convey information instead of using the common method in signed networks—balance theory. In addition, RGCLN adds a size constraint to the loss function to prevent image-like overfitting during the unsupervised learning process. Based on the node features learned by this end-to-end trained model, RGCLN performs community detection in a large number of real-world networks and generative networks, and the results indicate that our model has an advantage over state-of-the-art network embedding algorithms.

**Keywords:** signed graphs; network embedding; sign graph convolution; community detection

## 1. Introduction

Networks are ubiquitous in our lives, covering all aspects of life, from social networks to transportation networks to biological systems. The edges between networks can indicate states or relationships such as friendship, cooperation, trust, and empowerment, or they can indicate hostility, aversion, distrust, and suppression. Compared with unsigned networks, signed networks can reflect more complex social relationships [1].

In order to tap the rich information contained in graphs, graph representation learning techniques have been extensively developed in recent years [2]. By learning the structural information of the graph, the node attribute information, and the external information, a low-dimensional feature representation of the nodes is obtained and then used for the analysis of downstream tasks. This approach has been widely used in the field of graph structure, but the research of these methods in signed networks has mainly focused on link prediction tasks [3–9], and there has been relatively little research related to community detection [5,10–13]. The node embedding methods on community detection tasks are also basically traditional methods and do not apply to the more popular deep learning frameworks or ideas currently available.

In traditional embedding methods, a significant portion of sign network embedding algorithms uses spectral methods [14], the central idea of which is to embed the original network into a low-dimensional space consisting of the first  $k$  eigenvectors of the eigenmatrix associated with a given network. However, it has been shown that matrix decomposition-based spectral methods have limited representation learning capability in capturing the highly nonlinear properties of complex network structures, and they also have a very high computational overhead. Thus, we designed an asymmetric ladder-shaped network framework based on signed networks combined with deep learning concepts to obtain node embedding representations that are friendly to signed clustering tasks.



**Citation:** Song, A.; Ji, R.; Qi, W.; Zhang, C. RGCLN: Relational Graph Convolutional Ladder-Shaped Networks for Signed Network Clustering. *Appl. Sci.* **2023**, *13*, 1367. <https://doi.org/10.3390/app13031367>

Academic Editor: Yu-Dong Zhang

Received: 22 December 2022

Revised: 9 January 2023

Accepted: 17 January 2023

Published: 19 January 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

The core idea of the algorithms currently developed for clustering problems in signed networks can be summarized by the well-known sociological theory, “balance theory” [15]. Balance theory can be summarized simply by saying that the friend of a friend and the enemy of an enemy are both friends. This makes most algorithms involving this theory minimal, excluding nodes that do not satisfy the theory. To address the above problem, our framework discards this very limiting theory and uses only deep networks to capture node information.

The node embedding algorithm proposed in this paper is based on a multi-relational graph convolutional structure and draws on the design concepts of fused spectral graph convolutional networks and U-Net [16] in GCLN [17]. Due to the particularity of graph convolution, there will be an over-smoothing problem during depth stacking, so most models only obtain first-order and second-order neighbor information [18,19]. However, experiments confirm that our framework can effectively avoid this problem during the deepening process. To better distinguish the different information represented by positive and negative links in signed networks, we use relational graph convolution in the ladder-shaped depth framework, which means that our framework is not strictly symmetric. Thanks to our abandonment of balance theory, there is no need to precisely follow balanced paths in the process of information aggregation and transfer, which significantly simplifies our framework.

To show better results in unsupervised tasks and to further avoid extreme divisions, we add size constraints. The soft constraint prevents individual loss-guided models from falling into local optima during training. We evaluate the effectiveness of RGCLN in several real-world networks and generative networks for semi-supervised and unsupervised tasks, and both end up with good experimental results.

The main contributions of our paper are as follows:

- A novel end-to-end optimized asymmetric ladder-shape framework is designed for signed networks, which can capture the deep hidden features and domain information of nodes and well solve the problem that graph convolution cannot be deeply superimposed.
- The framework behaves as an asymmetric structure, which effectively avoids the overfitting problem.
- The design concept of end-to-end optimization makes the framework more friendly to the signed clustering problem.
- The size constraint is defined and incorporated to avoid extreme divisions in unsupervised learning, which dramatically improves the performance of the model.

The rest of this paper is organized as follows. In Section 2, we discuss current work related to this paper and briefly introduce related research and recent advances in node embedding on signed networks. Section 3 introduces the model proposed in this paper, the RGCLN. In Section 4, we implement our proposed approach in experiments and show a comparison with the baseline approach. Section 5 concludes the paper and highlights future work.

## 2. Related Work

In this section, we state the work related to graph representation learning. A large amount of research work related to graph embedding already exists in unsigned networks [20,21]. The presence of negative chains in signed networks leads to the fact that methods in unsigned networks cannot be directly migrated and applied to signed networks. Therefore, all relevant algorithms for signed networks need to be redesigned and researched. Next, we mainly review various embedding algorithms related to signed networks. Ref. [22] used a signed Laplacian matrix and its normalized version for signed clustering. Ref. [23] proposed a multilevel signed network clustering framework based on a balanced normalized cut level (BNC), which solves the problem of the direct extension of a signed Laplacian from  $k > 2$  to  $k$ -way and can obtain high-quality clustering results quickly. Ref. [24] derived a simple normalized signed (SNS) graph Laplacian matrix and a balanced normalized signed (BNS) graph Laplacian matrix, and the embedding

representation obtained through the matrices have good node characterization capability. Ref. [13] proposed a regularized spectral clustering algorithm based on solving the generalized feature problem, which shows better robustness to noisy and sparse matrices compared to algorithms such as SNS and BNS.

In addition to these Laplacian-based signed algorithms mentioned above, several studies have explored the possibilities of deep learning frameworks for embedding representations of signed network nodes. For example, ref. [25] used the sociological theory called balance theory to aggregate and transfer information between the nodes of a balanced path. In [26], a deep network embedding model is proposed to learn low-dimensional node vector representations with structural balance preservation. The model imposes a greater weight on the loss of negative chains, allowing the encoder to focus on capturing the more scarce negative connections. A new directed graph neural network model, SDGNN, was proposed in [27]. This model involves both balance theory and state theory and reconstructs all three types of features, link symbols, link directions, and directed triangles simultaneously, with a solid ability to learn node embeddings. The design of the three-part loss function of the SDGNN model has been explored in more depth by many subsequent studies. However, the embedding studies mentioned above have all focused on the link prediction task, and there is still much room for exploration in the community detection task.

Most of the existing signed network representation learning methods [28–31] were designed based on balance theory. SIN [1] first learned signed network embeddings using objective functions guided by balance theory. Ref. [6] proposed a node representation generation model based on adversarial learning using balance theory as the basis for determining pseudo-edge generation. Ref. [32] used a masked self-attentive layer to estimate the importance of different types of links and then aggregated more critical information from neighboring nodes to generate node embeddings based on balance theory. A recent research advancement related to our work is [33]. In the SSSNET model proposed in [33], a new interpretation of the social balance theory of signed network embeddings takes a neutral stance on whether the enemy of an enemy is a friend. Although the SSSNET model is a first step towards abandoning balance theory, it is ultimately limited by the balanced path. The algorithm proposed in this paper does not involve sociological theory but instead designs a practical depth framework to capture potential information between neighbors.

### 3. The Proposed Model

We first introduce some definitions that will be covered in the paper and then detail the specifics and details of the proposed framework.

#### 3.1. Notations

Before presenting the details of the framework, we introduce some definitions and notations that will be used in our paper. Let  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  be a signed network, where  $V = \{v_1, v_2, \dots, v_m\}$  is the set of  $m$  nodes, and  $\varepsilon \subset V \times V$  is the set of links. In particular, any link  $e_{ij} \in \varepsilon$  can be 1 or  $-1$ , where  $e_{ij} = 1$  denotes a positive link between  $v_i$  and  $v_j$ , while  $e_{ij} = -1$  denotes a negative link. We denote the adjacency matrix of the signature network  $G$  by  $\mathbf{A} \in \mathbb{R}^{n \times n}$ , where  $A_{ij} = 1$  indicates a positive link from  $v_i$  to  $v_j$ ,  $A_{ij} = -1$  indicates a negative link, and otherwise  $A_{ij} = 0$  indicates no link from  $v_i$  to  $v_j$ .  $D$  is the degree matrix, and the values on the diagonal are the degrees of each point. The task of clustering is to divide  $G$  into  $K$  clusters, where the nodes between clusters are as similar as possible, and the nodes between clusters should remain different. The model in this paper verifies the performance of unsupervised and semi-supervised clustering, where the seed nodes are identified with labels before training. The set of seed nodes is denoted as  $V_{\text{seed}} \subseteq V_{\text{train}} \subset V$ , where  $V_{\text{train}}$  is the set of all training nodes.

### 3.2. Multi-Relational Graph Convolutional Ladder-Shaped Networks

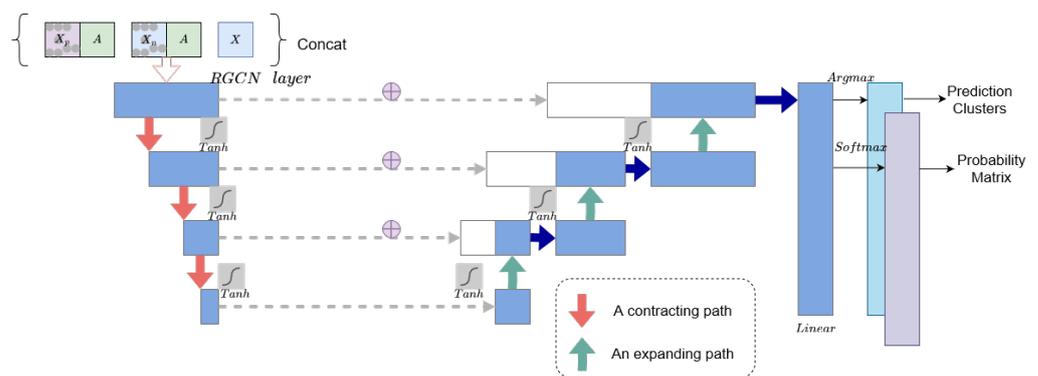
Due to the large variability between non-Euclidean data and data such as images, in this way, many excellent frameworks in the image domain cannot be used for graph-structured data. In [17], the authors proposed a graph convolutional ladder network (GCLN) to solve the over-smoothing problem by borrowing the well-known U-Net structure from medical image segmentation. The GCLN is a symmetric architecture with a GCN layer consisting of a systolic path and an extended path. Three contextual feature channels allow the contextual features captured from the systolic path to be fused with the localization information learned through the extended path. However, the GCLN framework is only applicable to general networks. We extend its design idea to signed networks and design the RGCLN architecture using multi-relational graph convolution instead of base convolution to address the over-smoothing problem under signed network clustering.

Next, we will focus on the multi-relational graph convolution ladder-shaped network framework proposed in this paper. We set the underlying graph convolution as multi-relational graph convolution, which differs from the standard graph convolution shared parameter settings by using different weights for two different types of edges: positive and negative edges. We do not make any assumptions about the positive and negative relations and only use the network to fit the deep structure information. The main part of our RGCLN model is shown in Figure 1, where it can be found that there are also contraction paths and expansion paths. However, in the contraction path, the number of neurons needs to be reduced to one-third of the previous layer. We define the message-passing process. Specifically, the node  $v$  of the  $l^{th}$  layer is denoted as  $Z_v^{(l)}$ , and its formula is defined as follows:

$$h_v^{(l)} = \left[ z_u^{(l-1)}, \sum_{u \in N_v^+} Z^{(l-1)}, \sum_{u \in N_v^-} Z^{(l-1)} \right]$$

$$Z_v^{(l)} = \sigma \left( h_v^{(l)} \cdot w^{(l)} + b^{(l)} \right) \tag{1}$$

where  $w^{(l)}$  denotes the weight matrix of the  $l^{th}$  layer, and  $b^{(l)}$  denotes the deviation vector of the  $l^{th}$  layer.



**Figure 1.** Framework of RGCLN.

Due to the convolution of multi-relational graphs, each convolution process has three parts: the aggregation of node features, positive domain information, and negative domain information, which makes the dimensionality of nodes expand to three times the original one after this process. Therefore, the settings of neurons on the expanding path need to refer to the settings of neurons on the contracted path and keep the principle of dimensionality consistency. This design is due to the specificity of the multi-relational graph convolution on the one hand and the need to aggregate the information on the corresponding side of the contraction path for the expansion path on the other. This makes the structure a non-strictly symmetric ladder-shaped network. We use Tanh as the activation function because it has been experimentally shown to work better when it is used.

### 3.3. Objective Function and Discussions

Our proposed framework is an end-to-end unified training framework with different loss functions designed for clustering problems in semi-supervised and unsupervised cases. By training the network model, we are able to achieve the clustering of similar nodes with different nodes to achieve a light effect while the loss is gradually reduced.

#### 3.3.1. Global Size Constraints

A fundamental advantage of deep learning is its flexibility, but extremes can occur if left unconstrained, especially in clustering problems where extreme divisions may occur. To circumvent this overfitting situation, similar to that in images, this chapter explores an algorithm for constrained clustering. Some previous work has explored uniform distributions, which are only applicable to specific clusters. Here, a more general constraint is described.

In this paper, the default maximum cluster size will not exceed half of the number of nodes, and the maximum number of clusters will be penalized if it exceeds this maximum limit. This definition is consistent with the distribution law of most clusters and can effectively improve the robustness and reliability of the model. The size constraint is divided into hard and soft constraints; the former means that the model must satisfy this constraint to get clustering results, while the soft constraint is not satisfied. Soft constraints are usually much better than hard constraints, so soft constraints are used here. Let the total number of clusters be  $k$ , the total number of training instances be  $n$ , the ratio of the desired maximum number of cluster nodes to the total number of nodes be  $up$ , and  $q_{ic}$  can be considered as the degree of instance  $i$  belongs to cluster  $c$ ; then the global size constraint loss function is defined as:

$$\mathcal{L}_{SC} = \min \left[ \sum_{c \in \{1, \dots, k\}} \cdot \left( \max \left( \sum_{i=1}^n q_{ic} / n \right) - up \right) \right], up = 0.5 \tag{2}$$

The loss function in this paper operates by minimizing the difference between the desired maximum number of clusters and the actual maximum number of clusters. Considering a more specific case, the constraint defines that the relevant hyperparameters can be self-adjusting. The upper limit defined as half of the total number of nodes is the best assumption based on the real situation of the experimental part of the dataset, which is also consistent with the cluster distribution pattern of most real-world or generative networks associated with signed networks. The value of  $up$  can be adjusted to achieve the best constraint based on real-world conditions.

#### 3.3.2. Self-Supervised Loss

The loss function under the self-supervised task contains two components: a differentiable probabilistic balanced normalized cut (PBNC) loss proposed in [33], on which the size-constrained loss (SC) is defined and introduced. The differentiable probabilistic balanced normalized cut loss is related to the (non-differentiable) balanced normalized cut loss proposed in [23]. The PBNC loss formula is defined as follows.

$$\mathcal{L}_{PBNC} = \sum_{k=1}^K \frac{\left( \mathbf{P}_{(:,k)} \right)^T \left( \mathbf{D}^+ - \mathbf{A} \right) \mathbf{P}_{(:,k)}}{\left( \mathbf{P}_{(:,k)} \right)^T \mathbf{D} \mathbf{P}_{(:,k)}} \tag{3}$$

where  $\mathbf{P}_{(:,k)}$  denotes the  $k^{th}$  column of the expansion of the probability matrix  $\mathbf{P}$  and  $\mathbf{D}$ ,  $\bar{D}_{ii} = \sum_{j=1}^n |A_{ij}|$ , and  $\mathbf{D}^+ - \mathbf{A}$  denotes the non-classical Laplacian graph with all positive edges in the part of  $\mathbf{A}$ .  $\mathbf{D}^+$  is the diagonal matrix associated with the  $\mathbf{A}^+$  matrix.  $\mathcal{L}_{PBNC}$  is similar to the concept of the unhappy ratio in that it is designed so that positive edges are distributed within clusters, and negative edges appear in clusters between them. Edges

that violate this rule are considered incorrectly partitioned, and thus losses such as the unhappy ratio or PBNC are calculated.

However, when using only PBNC as loss self-supervised training, there exists an extreme division of all nodes into a cluster. To avoid this situation, the global size constraint mentioned above is added to the self-supervised learning module to help the network adjust.

$\mathcal{L}_{PBNC}$  and  $\mathcal{L}_{PBNC}$  constitute the loss of the self-supervised part of our model, and the two parts of the loss are assigned weights through the parameter  $\gamma_c$ , which takes values in the range  $0 \leq \gamma_c \leq 1$ . The final total loss function is defined as:

$$\mathcal{L} = (1 - \gamma_c)\mathcal{L}_{PBNC} + \gamma_c\mathcal{L}_{SC} \tag{4}$$

### 3.3.3. Supervised Loss

In the definition of the semi-supervised scene loss function, we minimize the cross-entropy loss  $\mathcal{L}_{CE}$  and the triplet state loss as well as Equation (5). The definition can be used when the labels of some nodes are known. By setting a certain number of seed nodes  $V^{seed}$  in advance, the similarity between the nodes of any unknown cluster class and the seed nodes can be obtained. Suppose two societies are drawn from the divided societies in the middle state of the training process, one of which has two seed nodes  $i$  and  $j$ , and the other one has a node  $k$ . Applying the similarity calculation formula, we can find that the relationship between  $i$  and  $j$  must be closer than that between  $i$  and  $k$ , i.e., the distance is closer. The formula is defined as follows:

$$\mathcal{L}_{triplet} = \frac{1}{|T|} \sum_{(v_i, v_j, v_k) \in T} \text{ReLU}(\text{CS}(z_i, z_j) - \text{CS}(z_i, z_k) + \alpha) \tag{5}$$

where  $T \subseteq V^{seed} \times V^{seed} \times V^{seed}$  is a node triplet,  $v_i$  and  $v_j$  are the two seed nodes set, and  $v_k$  is a node randomly selected from different clusters. We use the cosine similarity  $\text{CS}(v_i, v_j)$  to measure the similarity of the two nodes.  $\alpha \geq 0$  is used as a bound [34]. The other part of the loss uses the cross-entropy loss  $\mathcal{L}_{CE}$ , and the two parts of the loss are balanced by the parameters  $\gamma_s$  and  $\gamma_t$ , which takes values in the range of  $0 \leq \gamma_s, \gamma_t \leq 1$ . Then, the total loss under the semi-supervision of the final RGCLN model is defined as follows.

$$\mathcal{L} = \mathcal{L}_{PBNC} + \gamma_s(\mathcal{L}_{triplet} + \gamma_t\mathcal{L}_{CE}) \tag{6}$$

## 4. Experiments

In this section, we conduct experiments on several real-world networks and stochastic block generation networks of different sizes to evaluate our proposed model from all aspects and verify that our model performs equally well on sparse networks and large data sets with noise. We analyze the performance of all algorithms by plotting the mean and standard error of 20 replications.

### 4.1. Signed Stochastic Block Models

We use signed stochastic block models (SSBM) to generate signed networks with different characteristics that can be artificially defined as containing  $k$  number of blocks and  $n$  nodes. Our generative model can be denoted by  $\text{SSBM}(n, K, p, \rho, \text{and } \eta)$ .  $p \in \{0.001, 0.01, 0.1\}$  denotes the edge creation probability, and we set all pairs of nodes within and between clusters to have the same edge probability. The (approximate) ratio between the maximum and minimum cluster sizes is  $\rho$ , and the signed flip probability is denoted by  $\eta$ . We define three sizes of network graphs with  $n = 1000, 10,000,$  and  $30,000$  to test the performance of the model for different sizes of graphs.

#### 4.2. Real-World Data

We tested six real-world signed networks (Sampson [35], Rainfall [36], S&P1500 [37], PPI [38], Epinions [1], and Slashdot [1]) and summarize the specific information of the dataset in Table 1. The context of the Sampson dataset is that the social interactions between a group of monks recorded by Sampson as a visual experimenter contain four social relationships (respect/disrespect, like/dislike, positive influence/negative influence, and praise/blame), which we combined into a network. The Rainfall dataset records rainfall series about each region of Australia, with edge weights obtained from two-by-two correlations, which are finally processed into an extended network with nodes of  $n = 306$ . S&P1500 considers the daily prices of  $n = 1193$  stocks in the S&P1500 index for the period 2003 to 2015, and we construct an extended network with stocks as nodes and correlations between them as edges. PPI describes a network of protein–protein interactions, with activation or inhibition relationships between interacting proteins as network features. Epinions is a consumer review site about whether users trust each other, and we construct an extended network of trust relationships between users. Slashdot is a website known for its technology-related news site known for its user community; Slashdot allows users to mark each other as friends or foes.

**Table 1.** Statistics of six real-world networks.  $E^+$  and  $E^-$  denote positive and negative links, respectively.

Dataset	N	E	$E^+$	$E^-$
Sampson	25	166	148	18
Rainfall	306	93,636	64,408	29,228
S&P1500	1193	1,423,249	1,069,319	353,930
PPI	3058	11,860	7996	3864
Epinions	41,178	609,489	508,449	101,040
Slashdot	40,825	448,240	339,415	108,825

We give the labels for each dataset. In particular, for Rainfall, we used the label of the SPONGE algorithm as the valid label, so SPONGE did not participate in the algorithm performance comparison under the Rainfall network. For the PPI, Epinions, and Slashdot networks without ground truth, we used self-supervised loss for training and finally used the unhappy ratio as an evaluation metric. For these three networks without ground truth, we set the number of clusters to 10 to facilitate comparison. For the rest of the networks, Sampson had 5 clusters, Rainfall used 6 clusters set by SPONGE, and S&P1500 used a sector member network with 10 clusters.

#### 4.3. Baselines

Here we present some existing state-of-the-art signed network embedding methods. The baselines are as follows:

- A: A partitioning algorithm based on the direct relation of the symmetric adjacency matrix denoted as  $A^* = \frac{1}{2}(A + (A)^T)$ .
- SNS [24] and DNS [24]: SNS and DNS are two spectral methods based on the random walk normalized Laplacian, both extended from the unsigned random walk normalized graph Laplacian [39].
- L: the Laplacian matrix  $L^*$ , using a signed network adjacency matrix.
- $L_{\text{sym}}$ : This is the symmetric normalized version of L.
- BNC [23] and BRC [23]: direct extensions of the signed Laplacian to the k-way clustering problem. BNC and BRC denote two algorithms for optimizing balanced ratio cuts and balanced normalized cut objectives, respectively.

- SPONGE [13]: introduces a principle and theoretically sound spectral method for k-way clustering in signed graphs and proposes a regularized spectral clustering algorithm based on solving the generalized feature problem.
- SPONGE<sub>sym</sub> [13]: a variant of SPONGE that uses the minimum of k generalized eigenvectors to generate embeddings, called the symmetric Laplacian operator of SPONGE.
- SSSNET [33]: redefines the balance path in balance path and asks “is the enemy of the enemy?” A neutral position is taken, and it then proposes an end-to-end GNN framework for semi-supervised signature network clustering.

#### 4.4. Evaluation Metrics

To evaluate the performance of various algorithms on different datasets, we used two widely used metrics, namely the adjusted Rand index (ARI) [40] and the normalized mutual information (NMI) metric [41].

Given a set of  $n$  objects  $S = \{O_1, O_2, \dots, O_n\}$ , let the external evaluation criteria be  $U = \{u_1, \dots, u_R\}$ , and the clustering result be  $v = \{v_1, \dots, v_C\}$ , satisfying  $\bigcup_{i=1}^R u_i = S = \bigcup_{j=1}^C v_j$ ,  $u_i \cap u_{i^*} = \emptyset = v_i \cap v_{i^*}$ , where  $1 \leq i \neq i^* \leq R$  and  $1 \leq j \neq j^* \leq C$ . Then set four statistics.  $a$  is in  $U$  belonging to the same category and also belongs to the same category in  $V$ ;  $b$  is the number of data point pairs that belong to the same category in  $U$  but are different in  $V$ .  $c$  is the number of data point pairs that do not belong to the same category in  $U$  but belong to the same category in  $V$ , and  $d$  is the number of data point pairs that do not belong to the same category in  $U$  but also do not belong to the same category in  $V$ . The Rand coefficient [42] is defined as:

$$RI = \frac{a + d}{a + b + c + d} \tag{7}$$

Hubert and Arabie proposed the adjusted Rand index in 1985. The adjusted Rand index assumes that the hyper distribution of the model is a stochastic model. Then, the adjusted Rand index is defined as:

$$E(RI) = E\left(\sum_{ij} \binom{n_{ij}}{2}\right) = \left[\sum_i \binom{n_i}{2} \sum_j \binom{n_j}{2}\right] / \binom{n}{2} \tag{8}$$

$$\max(RI) = \frac{1}{2} \left[\sum_i \binom{n_i}{2} \sum_j \binom{n_j}{2}\right]$$

$$ARI = \frac{RI + E(RI)}{MAX(RI) - E(RI)} \tag{9}$$

In order to evaluate the similarity between the real partition and the partition obtained by using the algorithm to divide the network, NMI is widely used for evaluation. Assuming that A and B are two partitions of a network, the NMI between A and B is written as:

$$NMI = \frac{-2 \sum_{i=1}^{C_A} \sum_{j=1}^{C_B} C_{ij} \log(C_{ij}N / C_i.C_j)}{\sum_{i=1}^{C_A} C_i \log(C_i / N) + \sum_{j=1}^{C_B} C_j \log(C_j / N)} \tag{10}$$

where  $N$  is the number of network nodes, and  $C$  is the fusion matrix.  $C_{ij}$  equals the number of nodes shared by community  $i$  in partition  $A$  and by community  $j$  in partition  $B$ .  $C_A(C_B)$  is the number of communities in structure  $A(B)$  and  $C_i.(C_j)$  is the sum of elements of  $C$  in row  $i$ (column  $j$ ).  $NMI(A,B) = 1$  means that the  $A$  and  $B$  partitions are exactly the same.

#### 4.5. Results on Dataset

We first report the results on six real-world networks, and the results of all methods on the association partitioning task are shown in Table 2. The first three datasets reflect the performance of the model under the semi-supervised task, which highlights the superiority

of our deep model from the side. The other three datasets reflect the unsupervised task, and it is clear that our model also shows superior performance under the unsupervised signed clustering task.

**Table 2.** Clustering performance on the real dataset; best outcomes are in bold. The first three columns are evaluated using ARI, where higher scores represent better classification results, and the rest of the dataset is evaluated by the unhappy ratio (%), where lower is better.

Dataset	Sampson	Rainfall	S&P1500	PPI	Epinions	Slashdot
A	0.32 ± 0.1	0.62 ± 0.09	0.2 ± 0.02	0.57 ± 0.0	0.6 ± 0.0	0.59 ± 0.02
SNS	0.11 ± 0.05	0.29 ± 0.03	0.0 ± 0.0	0.47 ± 0.0	0.16 ± 0.0	0.24 ± 0.0
DNS	0.3 ± 0.1	0.65 ± 0.04	0.06 ± 0.01	0.47 ± 0.0	0.17 ± 0.0	0.24 ± 0.0
L	0.16 ± 0.05	0.45 ± 0.08	0.06 ± 0.01	0.47 ± 0.0	0.18 ± 0.03	0.24 ± 0.0
L <sub>sym</sub>	0.35 ± 0.09	0.54 ± 0.05	0.24 ± 0.04	0.47 ± 0.0	0.17 ± 0.0	0.32 ± 0.09
BNC	0.32 ± 0.12	0.61 ± 0.07	0.04 ± 0.01	0.47 ± 0.0	0.17 ± 0.0	0.24 ± 0.0
BRC	0.21 ± 0.11	0.47 ± 0.05	0.0 ± 0.0	0.52 ± 0.0	0.16 ± 0.0	0.24 ± 0.0
SPONGE	0.36 ± 0.11	-	0.29 ± 0.05	0.48 ± 0.0	0.17 ± 0.0	0.25 ± 0.01
SPONGE <sub>sym</sub>	0.34 ± 0.11	0.73 ± 0.08	0.32 ± 0.04	0.46 ± 0.0	0.17 ± 0.0	0.24 ± 0.0
SSSNET	0.54 ± 0.11	0.76 ± 0.11	<b>0.63 ± 0.04</b>	0.21 ± 0.01	0.18 ± 0.01	0.24 ± 0.0
RGCLN	<b>0.56 ± 0.18</b>	<b>0.81 ± 0.11</b>	0.59 ± 0.06	<b>0.15 ± 0.01</b>	<b>0.14 ± 0.03</b>	<b>0.24 ± 0.01</b>

RGCLN outperforms the SSSNET model, which is also based on a deep GNN model, proving our initial idea that following the setting of the balance principle in a single way makes it difficult to consider the overall community structure from a global perspective. Although the SSSNET model makes an effort to consider the balancing principle at this point, it only adopts a partially balanced path. However, from the experimental results, we can try to abandon sociological theory altogether and reap unexpected good results instead. On the S&P1500 dataset, our model is slightly lower than SSSNET, which due to S&P1500, owns many balanced triangulations.

Figure 2 shows a visualization of the meaningful sorted adjacency matrix found by the algorithm under the Rainfall dataset. Here,  $k = 6$ , except for algorithms such as L and brc, which divide the number of communities to be less than a set value, so all algorithms recover essential information in the cluster.

Next, we conducted experiments on the SSBM. We designed multiple sets of experiments and multiple evaluation metrics, and the results are shown in Figure 3. We set two groups of node sizes  $n = \{1000, 10,000, 30,000\}$  and different parameters. Figure 3a shows that the performance of the division with a relatively small number of nodes gradually decreases as the Rand coefficient is adjusted and the flip probability  $\eta$  increases; the decreasing trend is more moderate in the NMI metrics, but all outperform the benchmark algorithm with the same variables. Figure 3b shows the case where the number of nodes is high, but the number of clusters is all low.

When the noise level is low, both the benchmark algorithm and RGCLN maintain extremely high clustering levels, but as the noise gradually increases, only SSSNET, SPONGE<sub>sym</sub> and RGCLN still maintain good performance, which corroborates the excellent noise immunity of the model proposed in this chapter.

The number of network nodes and clusters is higher in Figure 3c. The relevant performance of the model under large-scale sparse networks is reflected in Figure 3d. The accuracy of the benchmark algorithm in the clustering task and the robustness of the model are observed by varying the generation probability of the edges, i.e., the parameter  $p$ . When  $p = 0.001$ , the network is relatively sparse when algorithms such as BRC have completely failed, while the RGCLN model still exhibits the best performance for different network densities and noise levels. Although there exist individual models that are not

very different from the RGCLN model in such settings, in general, RGCLN still has the best performance and is the most stable.

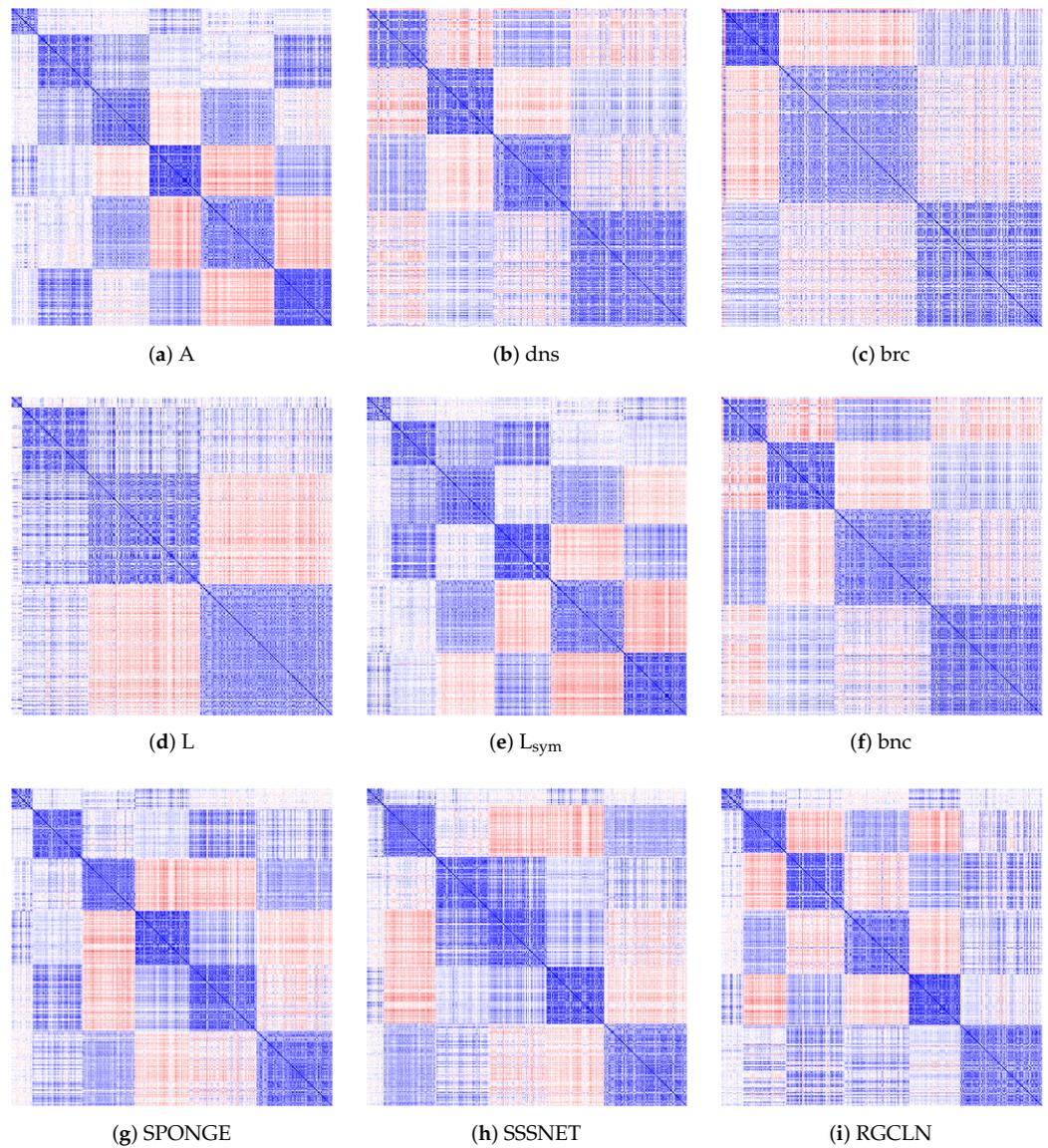


Figure 2. Sorted adjacency matrix of Rainfall dataset ( $k = 6$ ).

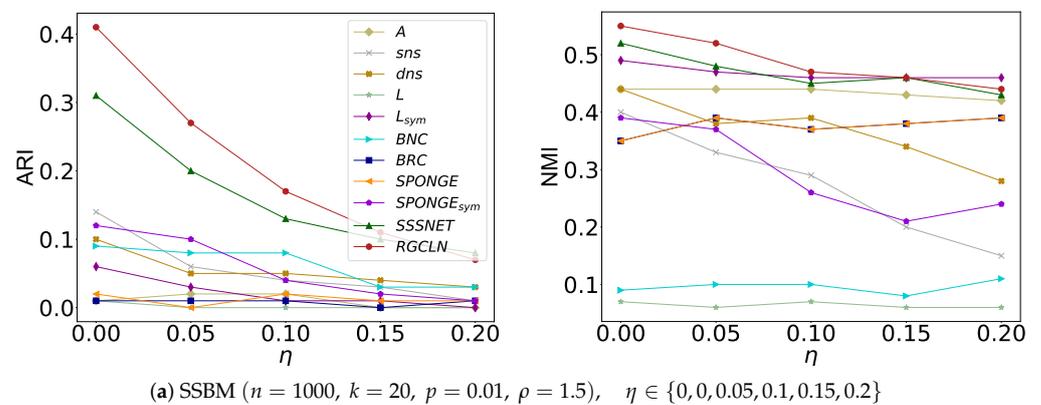
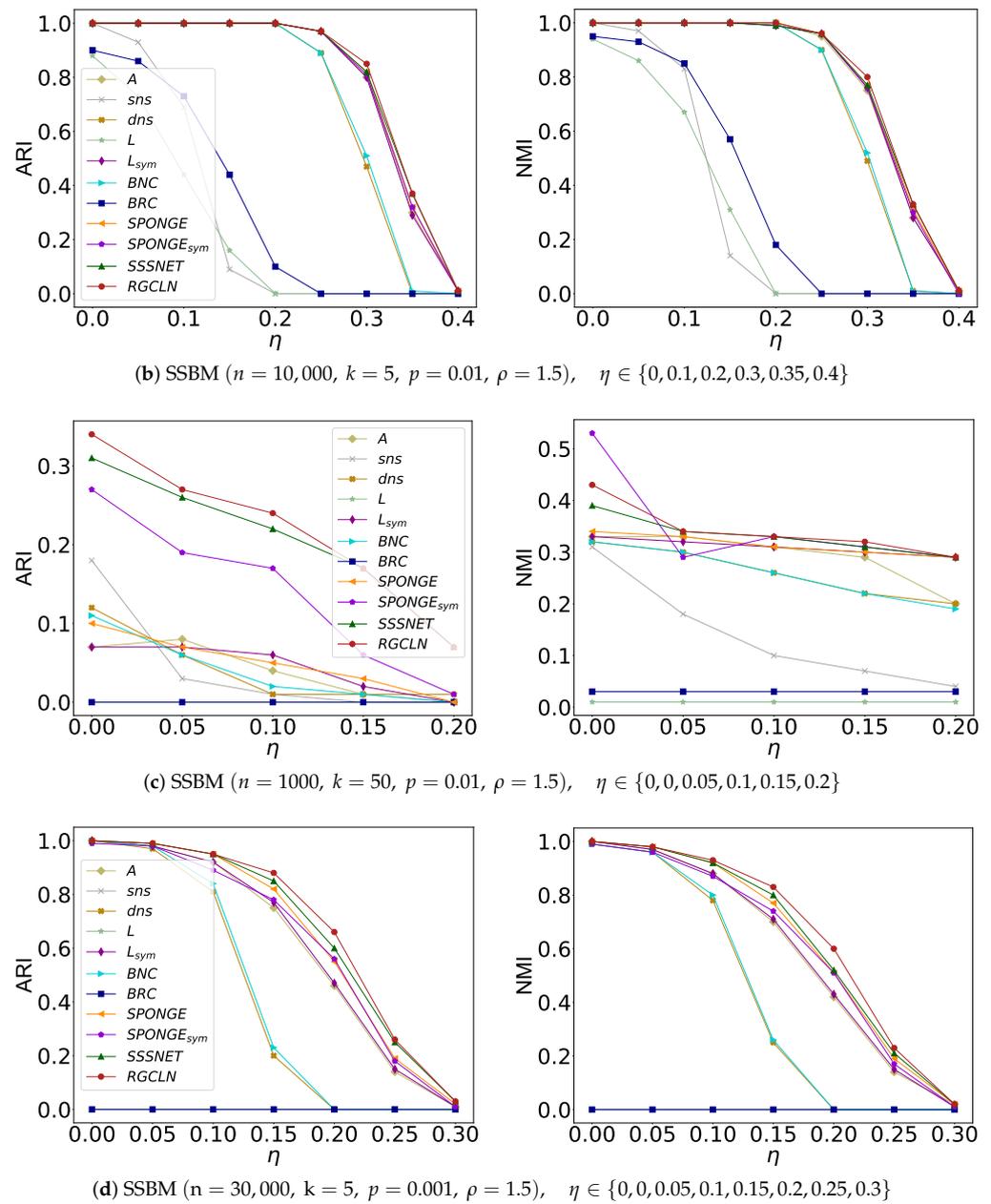


Figure 3. Cont.



**Figure 3.** Stochastic block generation network with different sizes and different parameter settings.

#### 4.6. Ablation Studies

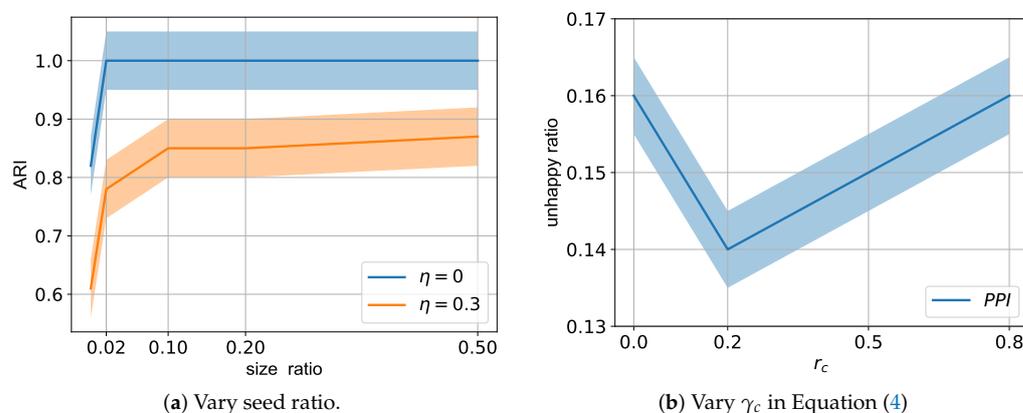
In order to verify the impact of our model on the performance of the whole framework during message delivery and whether its own features are involved in the delivery, we design two variants.  $RGCLN_1$  only considers aggregating messages of positive and negative neighbors, while  $RGCLN_2$  needs to aggregate its own features, messages of positive neighbors and messages of negative neighbors during the delivery process. Under the same settings, we verified the friendliness of these two variants in the downstream clustering task for six real datasets. The results are shown in Table 3, where it can be found that including one’s own features is more helpful for the model in most networks. Unless otherwise stated, all references to the RGCLN model in this paper refer to  $RGCLN_2$ .

**Table 3.** Two variants for validating the validity of ontology features. The first three rows are evaluated with the ARI metric, while the remaining dataset without the ground truth is evaluated with the unhappy ratio.

Dataset	Metric	RGCLN <sub>1</sub>	RGCLN <sub>2</sub>
Sampson	ARI	<b>0.65 ± 0.18</b>	0.56 ± 0.10
Rainfall	ARI	0.76 ± 0.13	<b>0.81 ± 0.11</b>
S&P1500	ARI	0.52 ± 0.06	<b>0.59 ± 0.06</b>
PPI	Unhappy ratio	0.19 ± 0.06	<b>0.15 ± 0.01</b>
Epinions	Unhappy ratio	0.14 ± 0.01	0.14 ± 0.03
Slashdot	Unhappy ratio	0.24 ± 0.00	0.24 ± 0.01

#### 4.7. Parameter Sensitivities

In this section, we investigate the sensitivity of the model to the parameters. Figure 4a shows the model in a semi-supervised environment, exploring the effect of the ratio of seed nodes on the results, where only a tiny-sized seed needs to be set to achieve the desired results. Figure 4b shows the evaluation of the help of the size constraint for training in an unsupervised environment. The PPI network clusters best with  $\gamma_c$  set to 0.2, but we use the default setting of 0.5 to obtain more general results and consider that the primary purpose of including size constraints is to prevent extreme clustering results rather than to optimize the metric evaluation results. In general, our model is not very sensitive to changes in hyperparameters, and the main features are still reflected in the network. The main features are in the network architecture design and training optimization.



**Figure 4.** Hyperparametric analysis (a,b). Figure (a) in SSBM ( $n = 10,000, k = 5, p = 0.01, \rho = 1.5$ ) and Figure (b) in PPI network environment. Figure (a) compares the ARI, while Figure (b) compares the unhappy ratio tested.

## 5. Conclusions

In our paper, we propose a deep ladder network based on relational graph convolution that can perform unsupervised and semi-supervised learning end-to-end on signed network community partitioning tasks. In contrast to traditional balance theory-based representation learning approaches, our proposed model learns the intrinsic laws and representation hierarchy of sample data purely from a deep framework. This goes beyond the common thinking in the field and provides an entirely new method of research. Extensive experimental results on generated and real-world networks demonstrate the effectiveness of RGCLN, achieving optimal accuracy and solving domain problems.

For future work, we will try to extend RGCLN to the detection of directed signed networks since the directional information given will have more positive effects on signed clustering. Another direction worthy of continued research is with respect to time series

prediction. It would also be interesting to study the evolution of node states and relationships through dynamic signed network data elements containing time series to infer target-predictive information with the application or research value.

**Author Contributions:** Conceptualization, A.S. and R.J.; methodology, A.S. and R.J.; software, R.J.; validation, R.J.; formal analysis, A.S. and R.J.; data curation, R.J.; writing—original draft preparation, R.J.; writing—review and editing, A.S. and R.J.; visualization, W.Q. and C.Z.; supervision, W.Q. and C.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset is available upon request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Leskovec, J.; Huttenlocher, D.; Kleinberg, J. Signed networks in social media. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, GA, USA, 10–15 April 2010; pp. 1361–1370.
- Liu, H.; Zhang, Z.; Cui, P.; Zhang, Y.; Cui, Q.; Liu, J.; Zhu, W. Signed Graph Neural Network with Latent Groups. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Virtual Event, 14–18 August 2021; pp. 1066–1075.
- Javari, A.; Derr, T.; Esmailian, P.; Tang, J.; Chang, K.C.C. Rose: Role-based signed network embedding. In Proceedings of the Web Conference 2020, Taipei, Taiwan, 20–24 April 2020; pp. 2782–2788.
- Chen, Y.; Qian, T.; Liu, H.; Sun, K. “Bridge”: Enhanced Signed Directed Network Embedding. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 773–782.
- Xu, P.; Hu, W.; Wu, J.; Liu, W.; Du, B.; Yang, J. Social trust network embedding. In Proceedings of the 2019 IEEE International Conference on Data Mining (ICDM), Beijing, China, 8–11 November 2019; pp. 678–687.
- Lee, Y.C.; Seo, N.; Han, K.; Kim, S.W. ASiNE: Adversarial Signed Network Embedding. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, 25–30 July 2020; pp. 609–618.
- Chen, X.; Yao, J.; Li, M.; Zhang, Y.; Wang, Y. Decoupled variational embedding for signed directed networks. *ACM Trans. Web (TWEB)* **2020**, *15*, 1–31. [[CrossRef](#)]
- Yan, D.; Zhang, Y.; Xie, W.; Jin, Y.; Zhang, Y. Muse: Multi-faceted attention for signed network embedding. *Neurocomputing* **2023**, *519*, 36–43. [[CrossRef](#)]
- Ma, L.; Ma, Y.; Lin, Q.; Ji, J.; Coello, C.A.C.; Gong, M. Snegan: Signed network embedding by using generative adversarial nets. *IEEE Trans. Emerg. Top. Comput. Intell.* **2020**, *6*, 136–149. [[CrossRef](#)]
- Wang, S.; Aggarwal, C.; Tang, J.; Liu, H. Attributed signed network embedding. In Proceedings of the 2017 ACM Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 137–146.
- Yang, B.; Cheung, W.; Liu, J. Community mining from signed social networks. *IEEE Trans. Knowl. Data Eng.* **2007**, *19*, 1333–1348. [[CrossRef](#)]
- Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. Line: Large-scale information network embedding. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1067–1077.
- Cucuringu, M.; Davies, P.; Glielmo, A.; Tyagi, H. SPONGE: A generalized eigenproblem for clustering signed networks. In Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, Naha, Japan, 16–18 April 2019; pp. 1088–1098.
- Mercado, P.; Tudisco, F.; Hein, M. Clustering signed networks with the geometric mean of Laplacians. *Adv. Neural Inf. Process. Syst.* **2016**, *29*. [[CrossRef](#)]
- Heider, F. *The Psychology of Interpersonal Relations*; Psychology Press: London, UK, 2013.
- Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*; Springer: Cham, Switzerland, 2015; pp. 234–241.
- Hu, R.; Pan, S.; Long, G.; Lu, Q.; Zhu, L.; Jiang, J. Going deep: Graph convolutional ladder-shape networks. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 2838–2845.
- Grover, A.; Leskovec, J. node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 855–864.
- Wang, X.; Cui, P.; Wang, J.; Pei, J.; Zhu, W.; Yang, S. Community preserving network embedding. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; pp. 203–209.
- Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.

21. Li, J.; Lu, K.; Huang, Z.; Zhu, L.; Shen, H.T. Transfer independently together: A generalized framework for domain adaptation. *IEEE Trans. Cybern.* **2018**, *49*, 2144–2155. [[CrossRef](#)] [[PubMed](#)]
22. Kunegis, J.; Schmidt, S.; Lommatzsch, A.; Lerner, J.; De Luca, E.W.; Albayrak, S. Spectral analysis of signed graphs for clustering, prediction and visualization. In Proceedings of the 2010 SIAM International Conference on Data Mining, Columbus, OH, USA, 29 April–1 May 2010; pp. 559–570.
23. Chiang, K.Y.; Whang, J.J.; Dhillon, I.S. Scalable clustering of signed networks using balance normalized cut. In Proceedings of the 21st ACM International Conference on Information and Knowledge Management, Maui, HI, USA, 29 October–2 November 2012; pp. 615–624.
24. Zheng, Q.; Skillicorn, D.B. Spectral embedding of signed networks. In Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, BC, Canada, 30 April–2 May 2015; pp. 55–63.
25. Derr, T.; Ma, Y.; Tang, J. Signed graph convolutional networks. In Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM), Singapore, 17–20 November 2018; pp. 929–934.
26. Shen, X.; Chung, F.L. Deep network embedding for graph representation learning in signed networks. *IEEE Trans. Cybern.* **2018**, *50*, 1556–1568. [[CrossRef](#)] [[PubMed](#)]
27. Huang, J.; Shen, H.; Hou, L.; Cheng, X. SDGNN: Learning node representation for signed directed networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual Event, 2–9 February 2021; pp. 196–203.
28. Derr, T.; Aggarwal, C.; Tang, J. Signed network modeling based on structural balance theory. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 557–566.
29. Jung, J.; Yoo, J.; Kang, U. Signed Graph Diffusion Network. *arXiv* **2020**, arXiv:2012.14191.
30. Kim, J.; Park, H.; Lee, J.E.; Kang, U. Side: Representation learning in signed directed networks. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 509–518.
31. Kumar, S.; Spezzano, F.; Subrahmanian, V.S.; Faloutsos, C. Edge weight prediction in weighted signed networks. In Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, Spain, 12–15 December 2016; pp. 221–230.
32. Li, Y.; Tian, Y.; Zhang, J.; Chang, Y. Learning signed network embedding via graph attention. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 4772–4779.
33. He, Y.; Reinert, G.; Wang, S.; Cucuringu, M. SSSNET: Semi-Supervised Signed Network Clustering. In Proceedings of the 2022 SIAM International Conference on Data Mining (SDM), Virtual Event, 28–30 April 2022; pp. 244–252.
34. Tian, Y.; Zhao, L.; Peng, X.; Metaxas, D. Rethinking kernel methods for node representation learning on graphs. *Adv. Neural Inf. Process. Syst.* **2019**, *32*. [[CrossRef](#)]
35. Sampson, S.F. A Novitiate in a Period of Change: An Experimental and Case Study of Social Relationships. Ph.D. Thesis, Cornell University, Ithaca, NY, USA, 1968.
36. Bertolacci, M.; Cripps, E.; Rosen, O.; Lau, J.W.; Cripps, S. Climate inference on daily rainfall across the Australian continent, 1876–2015. *Ann. Appl. Stat.* **2019**, *13*, 683–712. [[CrossRef](#)]
37. Yahoo! Finance. S&P 1500 Data. 2021. Available online: <https://finance.yahoo.com/> (accessed on 19 January 2021).
38. Vinayagam, A.; Zirin, J.; Roesel, C.; Hu, Y.; Yilmazel, B.; Samsonova, A.A.; Neumüller, R.A.; Mohr, S.E.; Perrimon, N. Integrating protein-protein interaction networks with phenotypes reveals signs of interactions. *Nat. Methods* **2014**, *11*, 94–99. [[CrossRef](#)] [[PubMed](#)]
39. Shi, J.; Malik, J. Normalized cuts and image segmentation, Pattern Analysis and Machine Intelligence. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 888–905.
40. Gates, A.J.; Ahn, Y.-Y. The impact of random models on clustering similarity. *J. Mach. Learn. Res.* **2017**, *18*, 1–28.
41. Wu, F.; Huberman, B.A. Finding communities in linear time: A physics approach. *Eur. Phys. J. B* **2004**, *38*, 331–338. [[CrossRef](#)]
42. Rand, W.M. Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.* **1971**, *66*, 846–850. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.