# TOTF: Missing-Aware Encoders for Clustering on Multi-View Incomplete Attributed Graphs

**Mengyao Li**[1], **Xu Zhou**[1*], **Jiapeng Zhang**[1*] and **Zhibang Yang**[1], **Cen Chen**[2], **Kenli Li**[1]
[1]Hunan University
[2]South China University of Technology
{lmy835, zhxu, zhangjp}@hnu.edu.cn, yangzb@ccsu.edu.cn, chencen@scut.edu.cn, lkl@hnu.edu.cn

## Abstract

As the network data in real life become multi-modal and multi-relational, multi-view attributed graphs have garnered significant attention. Numerous methods have achieved excellent performance in multi-view attributed graph clustering; however, they cannot efficiently handle incomplete attribute scenarios, which are prevalent in many real-life applications. Inspired by this, we investigate the problem of multi-view incomplete attributed graph clustering for the first time. In particular, the TOTF (Train Once Then Freeze) framework is designed to train missing-aware encoders that capture view-specific information while ignoring the impact of incomplete attributes, and then employs frozen encoders to uncover common information driven by clustering. After that, we propose a correlation strength-aware graph neural network on the basis of the inherent relationships among attributes to enhance accuracy. It is proven theoretically that traditional Generative Adversarial Networks (GANs) are unable to generate the unique real distribution. To address this issue, we further introduce the missing-position reminder mechanism into our intra-view adversarial games for better clustering results. Extensive experimental results demonstrate that our method achieves up to a 17% improvement in accuracy over the state-of-the-art methods. The source code is available at https://anonymous.4open.science/r/TOTF-main.

## 1 Introduction

As network data in real-world applications exhibit multi-modal and multi-relational characteristics [Qu *et al.*2017], multi-view attributed graphs [Lin and Kang2021, Lin *et al.*2023, Pan and Kang2021] have attracted considerable attention in practical domains such as social networks [Lu *et al.*2024], citation networks [Greenberg2009], and biological networks [Kermani *et al.*2022]. Notably, clustering on multi-view attributed graphs [Jin *et al.*2023, Fettal *et al.*2023] is a critical task aimed at finding a unified partition that divides nodes into several disjoint clusters. It plays a significant role in various applications, including recommendation systems, community detection, social network analysis, and ride-hailing demand prediction.


Figure 1: Multi-view incomplete attributed graph where some nodes are with incomplete attributes.

Although there are abundant researches about multi-view attributed graph clustering, they ignore incomplete attribute scenarios ragarding privacy concerns, system malfunctions, malicious attacks, or inadvertent omissions. Incomplete attributes are prevalent in many real-life applications [Jiang *et al.*2024]. For example, the research on Italian social networks about users interested in pharmaceutical products and health shows that there is a prevalence of missing values equal to 64.7% considering all social pages [Mariani *et al.*2024]. Additionally, in the U.S. healthcare information system, between 13.6% and 81% of data is outdated or lacking completeness [Jr. *et al.*2005].

In this paper, we focus on multi-view attributed graph clustering with incomplete attributes.

**Example 1.** *Fig. 1 illustrates an example of the multi-view incomplete attributed graph in the social network, containing $V$ views with incomplete attributes. The first view depicts the co-focus relationships on the Weibo platform, where Bob accidentally omits his favorite animal when filling out the form. The $V$-th view illustrates the colleague relationships, with Tom concealing his home address for privacy reasons. These incomplete attributes may affect the accuracy of clustering, thereby complicating the friend recommendation on the Weibo platform.*

**Prior Work and Limitations.** Existing researches fall into three classes. We list detailed differences in Table 1.

*Corresponding author

| Methods | Incomplete | Data Type | Incomplete Type | Without Label |
|---|---|---|---|---|
| MAGC, MCGC, MvAGC | ✗ | Multi-view attributed graph | ✗ | ✔ |
| ASD-VAE, GCNMF | ✔ | Single-view attributed graph | Attribute | ✗ |
| ProImp, ICMVC | ✔ | Euclidean data | Node | ✔ |
| TOTF (**Ours**) | ✔ | Multi-view attributed graph | Attribute | ✔ |

Table 1: Comparison of Prior Works

- **Multi-view attributed graph clustering.** Numerous works like MAGC [Lin *et al.*2023], MCGC [Pan and Kang2021], and MvAGC [Fan *et al.*2020] have been proposed for multi-view attributed graph clustering, however, these methods always assume ideal conditions and fail to effectively address incomplete attribute scenarios. As shown in Subsection 5.2, as the miss rate on the ACM dataset increases from 0.1 to 0.7, the accuracy of MvAGC decreases from 0.82 to 0.38, highlighting their struggle with incomplete attributes.
- **Single-view incomplete attributed graph learning methods.** This class of methods like ASD-VAE [Jiang *et al.*2024] and GCNMF [Taguchi *et al.*2021] involves incomplete attribute scenarios. However, they belong to supervised learning and rely on label information of node classification or link prediction. So they are not suitable for unsupervised tasks like clustering. Moreover, these methods are designed for single-view graphs and fail to deal with complex multi-view information.
- **Incomplete multi-view clustering methods.** This class of methods, like ProImp [Li *et al.*2023] and ICMVC [Chao *et al.*2024], is designed for multi-view Euclidean data, whose characteristics are totally different from graphs. Moreover, they focus on the incomplete view which means some samples are missing in several views, as a specific case of incomplete attributes. They are difficult to extend to deal with the more general problem (incomplete attributes).

**Challenges and Contributions.** Incomplete attributes bring new grim challenges. First, incomplete attributes may distort the inherent patterns and relationships within the data, making it challenging to accurately capture the true distribution of attributes and further affect clustering results. Second, although multi-view graphs can provide more complementary information, the coordination of different views poses a significant challenge. It is crucial to overcome the inconsistencies between these views, as well as those arising from incomplete attributes, to discover the consensus information that is essential for clustering.

In this paper, we propose the TOTF framework to address the problem of multi-view incomplete attributed graph clustering. To our knowledge, this is the first work focusing on this area. **Firstly**, we train a series of encoders that mitigate the impact of incomplete attributes. We design the Correlation Strength-Aware Graph Neural Network (CSAGNN) based on inherent relationships among attributed graphs. Incomplete attributes occur when some node attributes are missing; thus, while recovering these attributes, we extract information from both neighboring nodes and existing attributes within the node. However, few studies have explored dimensional relationships within graph neural networks. Therefore, we investigate the relationships among attributes via experiments to provide a rationale for CSAGNN. Additionally, we introduce a missing position reminder mechanism to address traditional GAN limitations in handling incomplete attributes and ensure real distribution acquisition. **Secondly**, we develop a clustering-driven common extractor to integrate multi-view information and learn consensus information essential for clustering tasks from view-specific information.

In summary, we highlight the contributions as follows:

- To the best of our knowledge, this is the first work to focus on multi-view attributed graph clustering with incomplete attributes.
- We design the TOTF (Train Once Then Freeze) framework to deal with multi-view incomplete attributed graph clustering which trains missing-aware encoders first and then freezes their parameters to further discover common information driven by clustering.
- We propose the correlation strength-aware graph neural network based on the findings of the inherent relationships among attributes through experiment investigation.
- We introduce the missing position reminder mechanism to discriminators and prove that the distribution of generated attributes is the same as the real distribution of attributes when achieving equilibrium.
- Extensive experimental results demonstrate that our method achieves significant accuracy improvements across different levels of incompleteness and is less affected by incomplete attributes.

The rest of this paper is organized as follows. In Section 2, we review related works. Section 3 gives the problem formulation. Section 4 describes our method in detail. Then, we conduct extensive experiments and analyze results in Section 5. Finally, Section 6 concludes this paper.

## 2 Related Work

In this section, we review related works. Incomplete multi-view clustering methods are typically designed for Euclidean data with missing samples and are difficult to extend to this general problem with incomplete attributes. Thus, we mainly focus on two other tasks: multi-view attributed graph clustering and incomplete single-view attributed graph learning.

### 2.1 Multi-View Attributed Graph Clustering

Recently, numerous studies have emerged on multi-view attributed graph clustering. MAGC [Lin *et al.*2023] utilizes the self-expressiveness property to integrate different views. MCGC [Pan and Kang2021] learns a consensus graph regularized by the graph contrastive loss. MvAGC [Lin and Kang2021] accelerates clustering by introducing sampled anchors. LMGEC [Fettal *et al.*2023] takes clustering and representation learning within a unified framework, reducing the overall time consumption. These methods always assume

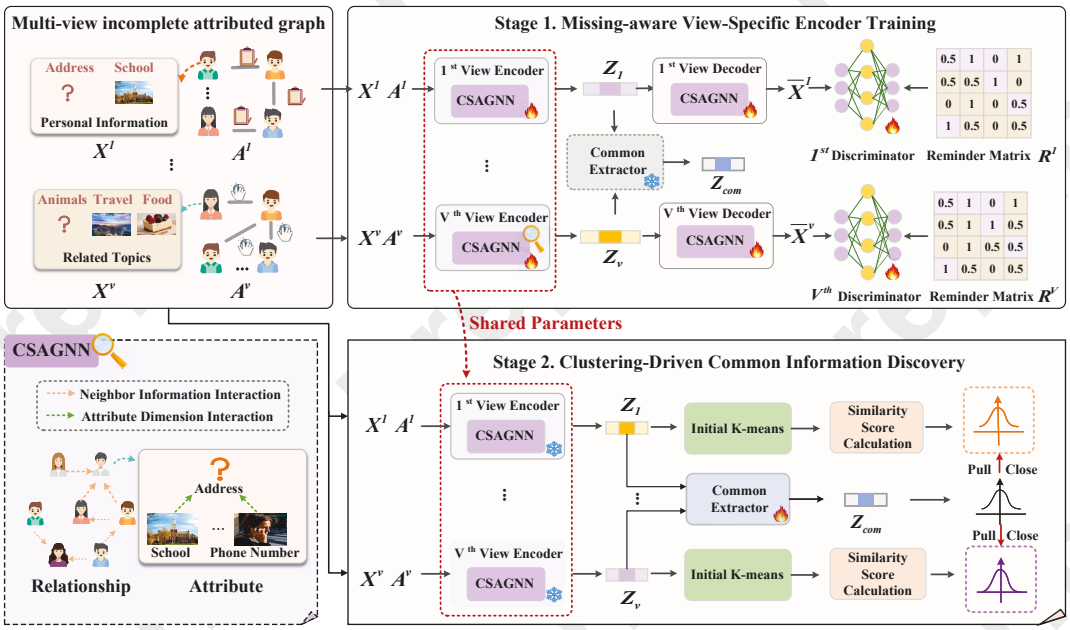



Figure 2: The TOTF Framework.

an ideal scenario where all nodes have complete attributes. However, attributes may be incomplete in real-world scenarios due to various factors, such as personal privacy concerns, accidental damage, or intentional deletion. They fail to effectively address incomplete attribute scenarios.

### 2.2 Single-View Incomplete Attributed Graph Learning

Single-view incomplete attributed graph learning has been extensively researched. GCNMF [Taguchi *et al.*2021] utilizes the Gaussian Mixture Model to represent missing features. ASD-VAE [Jiang *et al.*2024] learns a shared latent space from which the missing values can be imputed. However, these methods are unsuitable for multi-view incomplete attributed graph clustering. First, most methods belong to supervised learning methods, and rely on label information of classification and link prediction, making them unsuitable for unsupervised tasks like clustering. Second, these methods are specifically designed for single-view incomplete attributed graphs and cannot handle the complexities of multiple views.

## 3 Problem Formulation

In this section, we introduce our new problem formulation: Multi-view incomplete attributed graph and Multi-view incomplete attributed graph clustering.

**Definition 1.** *(Multi-view incomplete attributed graph)*
*A multi-view incomplete attributed graph can be defined as* $G = \{\Phi, \boldsymbol{A}^1, ..., \boldsymbol{A}^V, \boldsymbol{X}^1, ..., \boldsymbol{X}^V, \boldsymbol{M}^1, ..., \boldsymbol{M}^V\}$, *where* $\Phi$ *represents the node set.* $\boldsymbol{X}^v \in \mathbb{R}^{d_v \times n}$ denotes the attribute matrix for $v$-$th$ view with $d_v$ dimensions. And $\boldsymbol{A}^v$ represents the adjacency matrix where $\boldsymbol{A}^v = \{a^v_{i,j}\} \in \mathbb{R}^{n \times n}$ and if there exists an edge between nodes $v_i$ and $v_j$ in the $v$-$th$ view, $a^v_{i,j} = 1$, otherwise, $a^v_{i,j} = 0$. $\boldsymbol{M}^v$ is the missing marker matrix in the $v$-$th$ view, where $\boldsymbol{M}^v_{i,j} = 0$ if the $j$-$th$ attribute of the node $v_i$ in the $v$-$th$ view is unknown.

**Definition 2.** *(Multi-view incomplete attributed graph clustering)*
*Given a multi-view incomplete attributed graph* $G = \{\Phi, \boldsymbol{A}^1, ..., \boldsymbol{A}^V, \boldsymbol{X}^1, ..., \boldsymbol{X}^V, \boldsymbol{M}^1, ..., \boldsymbol{M}^V\}$, *clustering on multi-view attributed graph aims to find a unified partition fitting all views to divide the nodes of the graph* $G$ *into* $k$ *clusters* $(C_1, C_2, ..., C_k)$.

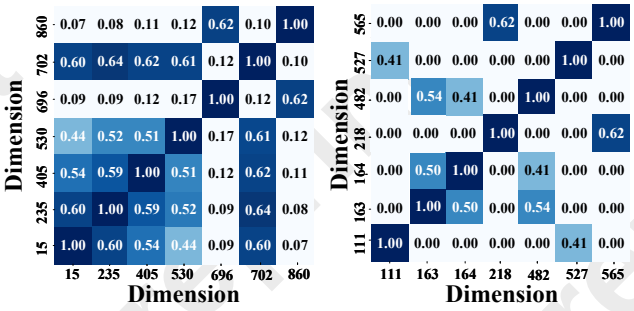
## 4 Proposed Method

To cluster on the multi-view incomplete attributed graph, we design a novel framework in which we train the missing-aware encoders first and then freeze their parameters. In this section, we describe our model in detail. We summarize the frequently used notations in Appendix A.1[3].

### 4.1 Overall Framework of TOTF

The TOTF framework consists of two stages: Missing-aware View-Specific Encoder Training and Clustering-Driven Common Information Discovery, as shown in Fig. 2. In Stage 1, we train a series of missing-aware, view-specific encoders that effectively capture the true distribution in each view while disregarding missing attributes. To address incomplete attributes, we design the Correlation Strength-Aware Graph Neural Network (CSAGNN), which imputes missing values from neighboring nodes and existing dimensions within the same node. Additionally, we implement a missing position reminder mechanism in our intra-view adversarial games to

[3]https://anonymous.4open.science/r/TOTF-main




(a) Wiki (b) Cora

Figure 3: Dimensional correlation analysis.

enhance view-specific encoders accurately capturing real attribute distributions despite incompleteness. In Stage 2, we freeze the parameters of the trained encoders and utilize the common extractor with clustering-driven self-supervised loss to derive common information beneficial for clustering.

## 4.2 Correlation Strength-aware Graph Neural Network

Graph neural networks (GNNs) [Lu *et al.*2024] have been widely applied. However, conventional GNNs like GCN and GAT primarily focus on message passing among neighboring nodes, neglecting the inherent relationships within attribute dimensions. This raises several pertinent questions: ***Is there a relationship between the dimensions of node attributes? Can the remaining dimensions be utilized to recover any missing dimensions for the same node?***

**Observation.** To address these questions, we employ the Kendall coefficient [Valencia *et al.*2019] to analyze inherent correlations between different dimensions in attributed graph datasets, as illustrated in Fig. 3. The larger the absolute value of the coefficient, the stronger the correlation. We find that there exist correlations between different dimensions, and the degree of correlation varies across these dimensions.

For instance, in Fig. 3(a), the correlation coefficient between the 702nd and 235th dimensions in Wiki dataset is 0.64, indicating a strong relationship between two dimensions. And the correlation coefficient between the 702nd and 860th dimensions is only 0.10, suggesting a weak relationship. Therefore, if the 702nd dimension is missing, the 235th dimension of the same node can offer more valuable information for recovery compared to that from the 860th dimension.

Inspired by this observation, we propose the Correlation Strength-Aware Graph Neural Network (CSAGNN), which both incorporates messages between attribute dimensions and neighboring nodes, as shown in Fig. 2. In our method, each encoder and decoder contain three layers of CSAGNN, and the 1-$th$ layer of CSAGNN in the $v$-$th$ view is defined as

$$CSAGNN(\boldsymbol{X}^v, \boldsymbol{A}^v) = \boldsymbol{A}^v Tnn[(\lambda_2 \boldsymbol{X}^v + \lambda_1 \boldsymbol{X}^v \boldsymbol{W}_1^v)\boldsymbol{W}_2^v] \quad (1)$$

where $\boldsymbol{X}^v$ and $\boldsymbol{A}^v$ denote the attributed matrix and the adjacency matrix in the $v$-$th$ view. $\boldsymbol{W}_1^v \in \mathbb{R}^{d_v \times d_v}$ is the dimension interaction matrix in the $v$-$th$ view. $\boldsymbol{W}_2^v \in \mathbb{R}^{d_v \times \hat{d}_1^v}$ is the weighted matrix mapping into the hidden space in the $v$-$th$ view where $\hat{d}_1^v < n$ is the output dimension in the in the 1-$th$ layer. $\lambda_2$ and $\lambda_1$ denote weight factors for original features and enhanced features, subject to the constraint $\lambda_1 + \lambda_2 = 1$. $Tnn(*)$ is the Tanh activation function. CSAGNN can not only integrate information from neighbors, but also capture the relationships among attribute dimensions, enabling it to use existing attributes for each node. It can initially impute missing attributes and obtain enhanced features, mitigating noise interference before mapping to latent spaces. As a result, compared with traditional GNNs, CSAGNN is more suitable for scenarios with incomplete attributes.

## 4.3 Intra-view Adversarial Games with the Missing Position Reminder Mechanism

**Intra-view Adversarial Games.** The autoencoder, consisting of a view-specific encoder and decoder, acts as the generator in the intra-view adversarial game, with the discriminator functioning as the opposing entity. The autoencoder aims to generate data closely matching the true distribution, while the discriminator attempts to distinguish between real and imputed data. However, incomplete attributes pose a challenge for autoencoders in capturing the unique true distribution within each view. ***In Subsection 4.7, Theorem 1 shows that incomplete attributes lead to multiple data distributions satisfying the condition in equilibrium, bringing difficultity to capture the unique true data distribution.***

**Missing Position Reminder Mechanism.** Inspired by [Yoon *et al.*2018] and [Miao *et al.*2021], we equip discriminators with the missing position reminder mechanism to address the above issue. We design the missing position reminder matrix $\boldsymbol{R}^v \in \mathbb{R}^{n \times d_v}$ to indicate positions of missing attributes in each view. This allows discriminators to effectively differentiate between real and generated data, while enabling the autoencoder to accurately capture the true data distribution. However, a high reminder rate has drawbacks; for instance, if all missing locations are indicated, the discriminator may make accurate judgments without any training. Thus, the missing position reminder matrix only indicates some locations of missing attributes. The missing position reminder matrix $\boldsymbol{R}^v \in \mathbb{R}^{n \times d_v}$ in the $v$-$th$ is defined as follows:

$$\boldsymbol{R}^v = \boldsymbol{K}^v \odot \boldsymbol{M}^v + 0.5(\boldsymbol{1} - \boldsymbol{K}^v) \quad (2)$$

where $\boldsymbol{K}^v \in \mathbb{R}^{n \times d_v}$ is a matrix of random variables consisting of 0 and 1, where the number of 1 is decided by the reminder rate $h$. When $\boldsymbol{R}_{ij}^v = 0$, we have $\boldsymbol{M}_{ij}^v = 0$, indicating a missing attribute at this position. When $\boldsymbol{R}_{ij}^v = 1$, we have $\boldsymbol{M}_{ij}^v = 1$, indicating the attribute exists. However, when $\boldsymbol{R}_{ij}^v = 0.5$, it is unclear whether the attribute is missing. The missing position reminder matrix helps the autoencoder capture true distribution within each view, disregarding the impact of incomplete attributes. In Subsection 4.7, we prove that with reminder matrices, the distribution of generated attributes is the same as the real distribution at equilibrium.

## 4.4 Clustering-driven Common Extractor

After Stage 1, we obtain encoders that capture view-specific features while are insulated from incomplete attributes. In

Stage 2, instead of using imputed data from Stage 1, we utilize the original data with missing values as input. The original data is passed through frozen encoders to generate view-specific representations. The clustering-driven common extractor consists of two fully connected layers and exponential linear units, designed to discover common information from view-specific representations.

Inspired by [Chen *et al.*2022], we introduce the clustering-driven loss to guide the common extractor to discover common information conducive to clustering. First, we perform the k-means to obtain cluster centers in each view and use the student's t-distribution [Chen *et al.*2022] to compute similarity scores $Q_1, ..., Q_V$ and $Q_{com}$ among node representations and cluster centers in each view and consensus view. Then, we calculate the common category assignment distribution $P_{com}$ by regularizing the similarity score in the consensus view. After that, the clustering-driven loss is defined as:

$$\begin{aligned} L_C &= \sum_{v=1}^{V}[KL(\boldsymbol{P}_{com}||\boldsymbol{Q}_v) + KL(\boldsymbol{P}_{com}||\boldsymbol{Q}_{com})] \\ &= \sum_{v=1}^{V}\sum_{i}\sum_{j} p_{ij}^{com} log\frac{p_{ij}^{com}}{q_{ij}^{v}} + \sum_{v=1}^{V}\sum_{i}\sum_{j} p_{ij}^{com} log\frac{p_{ij}^{com}}{q_{ij}^{com}} \end{aligned} \tag{3}$$

where $KL(*||*)$ is the KL divergence. The first item aims to pull the common category assignment distribution close to the similarity score of each view. The second item tries to minimize the distance between the similarity scores and the category assignment distribution in the consensus view, then we can utilize clustering information to optimize $\boldsymbol{Z}_{com}$.

### 4.5 Training Strategy

**Training Strategy in Stage 1.** To capture the view-specific information, we utilize the reconstruction loss and the intra-view adversarial game loss. The reconstruction loss for the $v$-$th$ view, which attempts to minimize differences between the reconstructed and original attributes, is as follows:

$$L_{re}^{v} = ||\boldsymbol{M}^{v} \odot \boldsymbol{X}^{v} - \boldsymbol{M}^{v} \odot \hat{\boldsymbol{X}}^{v}||_2 \tag{4}$$

where $\boldsymbol{M}^v$ is the missing marker matrix in the $v$-$th$ view, $\hat{\mathbf{X}}^v$ is the reconstructed attributed matrix in the $v$-$th$ view. $||*||_2$ represents the L2 norm. In the intra-view adversarial game, the training loss of the discriminator in the $v$-$th$ view with the missing position reminder mechanism is as follows:

$$\begin{aligned} L_D^v = &- E[\boldsymbol{M}^v \odot log(D_v(\overline{\boldsymbol{X}}^v, \boldsymbol{R}^v)) \\ &+ (1-\boldsymbol{M}^v) \odot log(1-D_v(\overline{\boldsymbol{X}}^v, \boldsymbol{R}^v))] \end{aligned} \tag{5}$$

where $D_v$ is the discriminator in the $v$-$th$ view. $\overline{\boldsymbol{X}}^v$ is the imputed data via $\overline{\boldsymbol{X}}^v = (1 - \boldsymbol{M}^v)\hat{\boldsymbol{X}}^v + \boldsymbol{M}^v\boldsymbol{X}^v$. The loss of the autoencoder in the $v$-$th$ view for Stage 1 is as follows:

$$L_G^v = L_{re}^v(\boldsymbol{X}^v, \hat{\boldsymbol{X}}^v) - (1 - \boldsymbol{M}^{\boldsymbol{v}} log(D_v(\overline{\boldsymbol{X}}^v, \boldsymbol{R}^v))). \tag{6}$$

In this stage, we first optimize the discriminator and then optimize the autoencoder until the intra-view adversarial game achieves the equilibrium state.

**Training Strategy in Stage 2.** After Stage 1, we can obtain a series of encoders that capture view-specific features effectively while mitigating the impact of missing information. Then we utilize the trained missing-aware view-specific encoders in Stage 1 to generate view-specific representations.

**Algorithm 1:** The TOTF Algorithm

**Input:** Multi-view incomplete attributed graph $\{\boldsymbol{A}^1, ..., \boldsymbol{A}^V, \boldsymbol{X}^1, ..., \boldsymbol{X}^V, \boldsymbol{M}^1, ..., \boldsymbol{M}^V\}$, the number of clusters $k$.
**Output:** $k$ clusters.

**Initialization:** Randomly initialization.
**// Stage-1: Missing-aware View-Specific Encoder Training**
**while** *not converged* **do**
  Obtain $\boldsymbol{Z}_v$ via encoders and reconstruct $\hat{\mathbf{X}}^v$.
  Impute data from $\overline{\boldsymbol{X}}^v = (1 - \boldsymbol{M}^v)\hat{\mathbf{X}}^v + \boldsymbol{M}^v\boldsymbol{X}^v$.
  Calculate $\boldsymbol{R}^v$ via Eq. (2).
  Distinguish imputed data $\overline{\boldsymbol{X}}^v$ via discriminators.
  Calculate $L_D^v$ and update discriminators.
  Calculate $L_G^v$ and update generators.
**end**
**// Stage-2: Clustering-Driven Common Information Discovery**
**while** *not converged* **do**
  Generate $\boldsymbol{Z}_v$ of each view via frozen encoders.
  Generate $\boldsymbol{Z}_{com}$ via common extractor.
  Employ K-means to obtain cluster centers for each view and the consistent view.
  Calculate $\{\boldsymbol{Q}_1, ..., \boldsymbol{Q}_V, \boldsymbol{Q}_{com}\}$ and $\boldsymbol{P}_{com}$.
  Calculate $L_{CE}$ and update common extractor.
**end**
Perform k-means on $\boldsymbol{Z}_{com}$ to get the $k$ clusters.
**return** *k clusters.*

The clustering-driven common extractor can capture common information $\boldsymbol{Z}_{com}$ that is conducive to clustering via the common loss $L_{com} = \sum_{v=1}^{V} ||\boldsymbol{Z}_{com} - \boldsymbol{Z}_v||_2$ and the clustering-driven self-supervised loss:

$$L_{CE} = L_C + L_{com}, \tag{7}$$

where $L_{CE}$ is the loss of common extractor in Stage 2.

### 4.6 The TOTF Algorithm

In Stage 1, we train the missing-aware view-specific encoders. Each encoder converts incomplete data to latent embeddings $\boldsymbol{Z}_v$, while decoders reconstruct specific information $\hat{\boldsymbol{X}}^v$. By combining original and reconstructed data, we obtain imputed data. Then, discriminators differentiate between imputed and original data via the missing position reminder matrix. We calculate losses for discriminators and generators and update them sequentially. Upon convergence, we achieve view-specific encoders that are insensitive to missing information. In Stage 2, we freeze the encoder parameters and generate common representations $\boldsymbol{Z}_{com}$ through a common extractor. We apply k-means to obtain cluster centers and compute similarity scores for each view and the consistent view. After that, we calculate the common category assignment distribution $P_{com}$. Then, we update the common extractor until convergence. Finally, k-means is performed on common representations $\boldsymbol{Z}_{com}$ to obtain $k$ clusters.

| Dataset | Missing Rate | m=0.1 | | | | m=0.3 | | | | m=0.5 | | | | m=0.7 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ACC | NMI | F1 | ARI | ACC | NMI | F1 | ARI | ACC | NMI | F1 | ARI | ACC | NMI | F1 | ARI |
| Cora | AGCN0 | 0.625 | 0.437 | 0.557 | 0.366 | 0.607 | 0.415 | 0.473 | 0.352 | 0.559 | 0.369 | 0.458 | 0.269 | 0.495 | 0.273 | 0.404 | 0.258 |
| | AGCNM | 0.551 | 0.371 | 0.446 | 0.238 | 0.535 | 0.360 | 0.418 | 0.211 | 0.563 | 0.352 | 0.441 | 0.285 | 0.515 | 0.291 | 0.421 | 0.282 |
| | SDCN0 | 0.530 | 0.318 | 0.446 | 0.270 | 0.411 | 0.212 | 0.284 | 0.133 | 0.332 | 0.137 | 0.266 | 0.066 | 0.282 | 0.041 | 0.144 | 0.025 |
| | SDCNM | 0.457 | 0.236 | 0.308 | 0.204 | 0.417 | 0.180 | 0.341 | 0.151 | 0.355 | 0.136 | 0.280 | 0.076 | 0.306 | 0.017 | 0.109 | 0.004 |
| | ASD-VAE | 0.314 | 0.165 | 0.299 | 0.098 | 0.227 | 0.155 | 0.325 | 0.074 | 0.243 | 0.055 | 0.211 | 0.026 | 0.225 | 0.010 | 0.155 | 0.006 |
| | MvAGC0 | 0.348 | 0.147 | 0.308 | 0.076 | 0.287 | 0.067 | 0.205 | 0.046 | 0.281 | 0.029 | 0.110 | 0.015 | 0.284 | 0.031 | 0.108 | 0.001 |
| | MvAGCM | 0.323 | 0.137 | 0.272 | 0.078 | 0.283 | 0.104 | 0.264 | 0.045 | 0.276 | 0.040 | 0.209 | 0.029 | 0.274 | 0.037 | 0.132 | 0 |
| | MAGC0 | 0.692 | 0.532 | 0.632 | 0.442 | 0.621 | 0.481 | 0.584 | 0.343 | 0.568 | 0.453 | 0.499 | 0.275 | 0.500 | 0.394 | 0.448 | 0.180 |
| | MAGCM | 0.694 | 0.528 | 0.634 | 0.445 | 0.646 | 0.500 | 0.601 | 0.353 | 0.561 | 0.453 | 0.489 | 0.300 | 0.522 | 0.400 | 0.461 | 0.212 |
| | Ours | 0.741 | 0.557 | 0.691 | 0.525 | 0.740 | 0.556 | 0.688 | 0.530 | 0.727 | 0.547 | 0.689 | 0.489 | 0.614 | 0.454 | 0.609 | 0.362 |
| ACM | AGCN0 | 0.779 | 0.545 | 0.794 | 0.568 | 0.754 | 0.535 | 0.711 | 0.539 | 0.629 | 0.363 | 0.560 | 0.353 | 0.386 | 0.011 | 0.134 | 0.199 |
| | AGCNM | 0.791 | 0.558 | 0.780 | 0.561 | 0.723 | 0.477 | 0.678 | 0.472 | 0.716 | 0.435 | 0.677 | 0.442 | 0.585 | 0.256 | 0.562 | 0.261 |
| | SDCN0 | 0.777 | 0.507 | 0.775 | 0.530 | 0.781 | 0.530 | 0.772 | 0.537 | 0.674 | 0.348 | 0.647 | 0.370 | 0.522 | 0.194 | 0.505 | 0.186 |
| | SDCNM | 0.787 | 0.530 | 0.787 | 0.549 | 0.751 | 0.552 | 0.702 | 0.538 | 0.742 | 0.443 | 0.743 | 0.455 | 0.573 | 0.282 | 0.433 | 0.330 |
| | ASD-VAE | 0.363 | 0.277 | 0.236 | 0.205 | 0.381 | 0.271 | 0.254 | 0.237 | 0.336 | 0.116 | 0.201 | 0.097 | 0.284 | 0.002 | 0.133 | 0 |
| | MvAGC0 | 0.820 | 0.512 | 0.822 | 0.545 | 0.813 | 0.474 | 0.813 | 0.525 | 0.602 | 0.294 | 0.484 | 0.303 | 0.384 | 0.009 | 0.307 | 0.010 |
| | MvAGCM | 0.821 | 0.498 | 0.822 | 0.546 | 0.819 | 0.489 | 0.819 | 0.540 | 0.620 | 0.284 | 0.539 | 0.309 | 0.351 | 0.001 | 0.174 | 0.000 |
| | MAGC0 | 0.857 | 0.572 | 0.853 | 0.625 | 0.851 | 0.559 | 0.848 | 0.613 | 0.636 | 0.389 | 0.513 | 0.370 | 0.632 | 0.393 | 0.513 | 0.374 |
| | MAGCM | 0.858 | 0.575 | 0.854 | 0.628 | 0.863 | 0.580 | 0.860 | 0.641 | 0.841 | 0.538 | 0.837 | 0.592 | 0.636 | 0.391 | 0.513 | 0.372 |
| | Ours | 0.884 | 0.656 | 0.885 | 0.693 | 0.875 | 0.637 | 0.876 | 0.673 | 0.843 | 0.561 | 0.844 | 0.601 | 0.802 | 0.483 | 0.804 | 0.517 |
| Wiki | AGCN0 | 0.360 | 0.294 | 0.230 | 0.137 | 0.299 | 0.207 | 0.144 | 0.076 | 0.350 | 0.276 | 0.214 | 0.131 | 0.399 | 0.355 | 0.229 | 0.140 |
| | AGCNM | 0.420 | 0.344 | 0.285 | 0.214 | 0.423 | 0.374 | 0.284 | 0.160 | 0.380 | 0.305 | 0.222 | 0.186 | 0.388 | 0.340 | 0.222 | 0.176 |
| | SDCN0 | 0.255 | 0.143 | 0.105 | 0.066 | 0.265 | 0.157 | 0.114 | 0.086 | 0.268 | 0.159 | 0.123 | 0.085 | 0.247 | 0.144 | 0.093 | 0.064 |
| | SDCNM | 0.236 | 0.129 | 0.083 | 0.056 | 0.257 | 0.141 | 0.084 | 0.072 | 0.225 | 0.114 | 0.064 | 0.056 | 0.238 | 0.133 | 0.098 | 0.058 |
| | ASD-VAE | 0.157 | 0.237 | 0.149 | 0.053 | 0.156 | 0.151 | 0.108 | 0.044 | 0.154 | 0.056 | 0.073 | 0.013 | 0.114 | 0.011 | 0.056 | 0.002 |
| | MvAGC0 | 0.519 | 0.503 | 0.394 | 0.255 | 0.468 | 0.471 | 0.331 | 0.206 | 0.322 | 0.305 | 0.129 | 0.095 | 0.175 | 0.021 | 0.029 | 0.001 |
| | MvAGCM | 0.503 | 0.491 | 0.397 | 0.274 | 0.481 | 0.464 | 0.344 | 0.233 | 0.408 | 0.361 | 0.242 | 0.097 | 0.237 | 0.167 | 0.052 | 0.039 |
| | MAGC0 | 0.514 | 0.484 | 0.353 | 0.214 | 0.452 | 0.434 | 0.252 | 0.169 | 0.444 | 0.411 | 0.242 | 0.179 | 0.377 | 0.380 | 0.198 | 0.114 |
| | MAGCM | 0.497 | 0.510 | 0.420 | 0.262 | 0.506 | 0.514 | 0.417 | 0.280 | 0.501 | 0.510 | 0.425 | 0.289 | 0.506 | 0.506 | 0.442 | 0.254 |
| | Ours | 0.549 | 0.526 | 0.463 | 0.361 | 0.534 | 0.512 | 0.451 | 0.341 | 0.549 | 0.526 | 0.471 | 0.350 | 0.543 | 0.504 | 0.456 | 0.346 |

Table 2: Clustering results in different missing rates.

### 4.7 Theoretical Analysis

**Theorem 1.** *In intra-view adversarial games, incomplete attributes can hinder the generator from generating the desired distribution (the unique true attributed distribution) when achieving the equilibrium state, as multiple data distributions may satisfy the equilibrium conditions.*

*Proof.* Based on [Yoon *et al.*2018] and [Miao *et al.*2021], we give the proof in Appendix A.2.

**Theorem 2.** *Given the missing position reminder matrix $\boldsymbol{R}^v$, the equilibrium of each view in intra-view adversarial games can be uniquely determined, where the distribution of generated attributes $\hat{p}(\boldsymbol{X}^v|\boldsymbol{M}^v)$ is the same as the real distribution $\hat{p}(\boldsymbol{X}^v|1)$ in each view, that is $\hat{p}(\boldsymbol{X}^v|\boldsymbol{M}^v) = \hat{p}(\boldsymbol{X}^v|1)$.*

*Proof.* We prove Theorem 2 based on [Yoon *et al.*2018] and [Miao *et al.*2021] in Appendix A.2.

**Time complexity analysis.** The time complexity of the TOTF algorithm is near $\mathcal{O}(n^2)$ shown in Appendix A.3.

## 5 Experiments

### 5.1 Experiment Setting

**Metrics and Datasets.** We use four evaluation metrics: ACC, F1-score, NMI, and ARI [Fan *et al.*2020]. We conduct experiments on five datasets-ACM [Lin and Kang2021], AMiner [Hong *et al.*2020], Cora, Citeseer, and Wiki [Fettal *et al.*2023] datasets. Details about datasets, hardware platform, and model parameters are shown in Appendix B.1.

**Comparison Algorithms.** The comparison methods contain multi-view attributed graph clustering methods (MAGC [Lin *et al.*2023] and MvAGC [Lin and Kang2021]) with zero filling and mean filling, single-view attributed graph clustering methods (SDCN [Bo *et al.*] and AGCN [Peng *et al.*]) with zero filling and mean filling, and single-view incomplete attributed graph learning methods (ASD-VAE [Jiang *et al.*2024]). For single-view methods, we execute them on each view and calculate the average results across all views. For ASD-VAE, we remove label-dependent components to adapt for unsupervised clustering tasks.

### 5.2 Comparison Experiments

We evaluate advanced methods on five datasets across different missing rates. Due to space constraints, we only present results for the Cora, ACM, and Wiki datasets in Table 2. Results for the Aminer and Citeseer datasets are available in Appendix B.2. The detailed observations are as follows.

Firstly, our method performs better than other advanced methods across different miss rates. When the miss rate is 0.1, the accuracy of TOTF is 5% higher than other methods in the ACM dataset, indicating TOTF has an excellent ability to discover common information via the clustering-driven loss. In Table 2, our method achieves at least a 40% improvement in accuracy over the Cora dataset compared to ASD-VAE, which shows that as a single-view incomplete attributed graph learning method, ASD-VAE is not well-suited for addressing multi-view incomplete attributed graphs. It may derive from two reasons: First, most of these are designed for supervised tasks that require labeling information, and without labeling information, their accuracy drops significantly.

Second, single-view methods struggle with complex multi-view graphs and are unable to leverage the complementary information across different views effectively.

Secondly, our method is less affected by miss rates compared with other methods and performs well even in high missing rates. As for MAGC and MvAGC, they cannot handle multi-view incomplete attributed graphs effectively. In Table 2, as the missing rate increases, their accuracy declines sharply. On the ACM dataset, as the missing rate increases from 0.1 to 0.7, the ACC metric of MAGC decreases by more than 20%. AGCN and SDCN perform poorly across different rates because they are designed for single-view attributed graphs with complete attributes and are unable to effectively handle these complex scenarios. For our method, on the ACM dataset, when the miss rate reaches 0.7, the accuracy of our method remains above 80%, which is approximately 17% higher than other methods. This indicates that our method can effectively deal with incomplete attributes, and the trained missing-aware, view-specific encoder can capture features accurately and is insensitive to missing information.

### 5.3 Ablation Studies

To evaluate the significance of each component in our method, we have designed ablation experiments as follows:

**Case A: W/o (With/Without) $\boldsymbol{R}^v$.** Table 3 shows that discriminators with the missing position reminder mechanism outperform traditional discriminators, improving accuracy by 4% to 6% based on the ARI metric. This indicates that the missing position reminder mechanism allows the discriminator to distinguish between real and fake data more effectively, thereby guiding the generator to produce data that aligns with the true distribution more closely.

**Case B: W/o CSAGNN.** Table 3 shows that CSAGNN performs better than classical GNNs, improving accuracy by 4% to 9% on the ARI metric. It indicates that CSAGNN is more effective for incomplete attributed graphs as it can combine the information between dimensions and neighbors.

**Case C: W/o $L_{re}$.** Table 3 indicates that the reconstruction loss has a greater impact on all datasets. By minimizing the difference between the reconstructed data and the original data, encoders can better capture view-specific information.

**Case D: W/o $L_C$.** The results in Table 3 indicate that the clustering-driven self-supervised loss can improve clustering results 2% to 6% according to ARI metric. The combination of $L_C$ and $L_{com}$ makes the captured consensus information more beneficial for clustering tasks.

**Case E: W/o $L_{com}$.** $L_{com}$ can improve clustering results 2% to 6% according to ARI metric. $L_{com}$ can obtain the consensus information from the view-specific information.

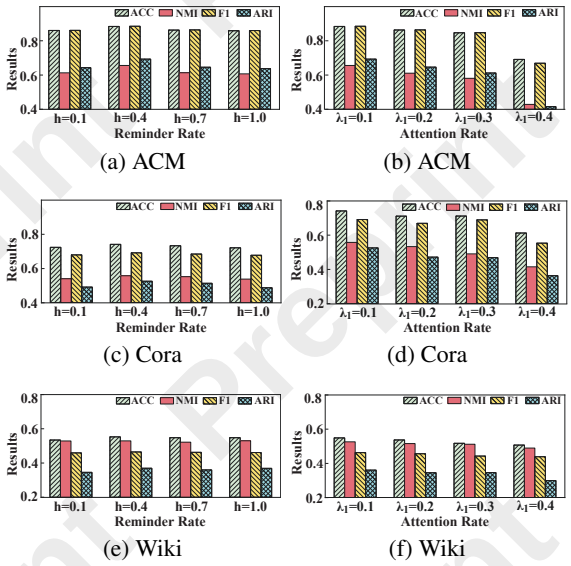
**Case F: W/o $\overline{\boldsymbol{X}}^v$.** As shown in Table 3, we find that we do not need to utilize the imputed data from Stage 1. The trained missing-aware view-specific encoders can effectively capture view-specific information while being insensitive to missing data, achieving better results than with imputed data.

### 5.4 Parameter Sensitivity Analysis

We analyze the parameter sensitivity of the missing position reminder mechanism and the CSAGNN. Figs. 4(a), (c), and (e) illustrate the variation in accuracy across different reminder rates when the missing rate is 0.1. There are only slight differences at varying reminder rates $h$. Notably, when $h = 0.4$, the clustering performance is generally optimal. However, when $h = 1$, there is a noticeable drop in the ARI metric on the ACM and Cora datasets, compared to $h = 0.4$. Thus, we set the reminder rate to 0.4 in our model. Figs. 4(b), (d), and (f) illustrate the variation in accuracy for different dimensional interaction coefficients when the missing rate is 0.1. Excessively large coefficients result in suboptimal outcomes, while the optimal performance is attained when $\lambda_1 = 0.1$. Thus, we adopt $\lambda_1 = 0.1$ for most datasets.

| Dataset | $m$=0.1 | A | B | C | D | E | F | Ours |
|---|---|---|---|---|---|---|---|---|
| ACM | ACC | 0.857 | 0.854 | 0.556 | 0.859 | 0.859 | 0.863 | 0.884 |
| | NMI | 0.608 | 0.603 | 0.276 | 0.605 | 0.608 | 0.613 | 0.656 |
| | F1 | 0.857 | 0.855 | 0.518 | 0.860 | 0.860 | 0.864 | 0.885 |
| | ARI | 0.636 | 0.631 | 0.293 | 0.637 | 0.638 | 0.645 | 0.693 |
| Cora | ACC | 0.719 | 0.701 | 0.472 | 0.728 | 0.730 | 0.715 | 0.741 |
| | NMI | 0.543 | 0.512 | 0.279 | 0.549 | 0.544 | 0.531 | 0.557 |
| | F1 | 0.677 | 0.667 | 0.465 | 0.679 | 0.683 | 0.674 | 0.691 |
| | ARI | 0.482 | 0.435 | 0.209 | 0.508 | 0.506 | 0.483 | 0.525 |
| Wiki | ACC | 0.511 | 0.510 | 0.413 | 0.522 | 0.519 | 0.539 | 0.549 |
| | NMI | 0.502 | 0.503 | 0.456 | 0.507 | 0.513 | 0.517 | 0.526 |
| | F1 | 0.433 | 0.426 | 0.367 | 0.451 | 0.440 | 0.453 | 0.463 |
| | ARI | 0.328 | 0.328 | 0.221 | 0.325 | 0.337 | 0.349 | 0.361 |

Table 3: Ablation Studies.




(a) ACM (b) ACM (c) Cora (d) Cora (e) Wiki (f) Wiki

Figure 4: Parameter Sensitivity Analysis

## 6 Conclusion

In this paper, we first focus on multi-view incomplete attributed graph clustering. We propose the TOTF framework which trains missing-aware encoders first to capture view-specific information and then freezes their parameters to discover common information available for clustering. Furthermore, we propose the correlation strength-aware graph neural network to discover inherent relationships of attributes. Also, we introduce the missing position reminder mechanism for discriminators. Experiments illustrate that our method can perform excellent performances across all incomplete rates.

## Acknowledgments

The research was supported by the Major Research Plan of National Natural Science Foundation of China (Grant No. 92055213), the National Natural Science Foundation of China (Grant Nos. U23A20317, 62172146, 62402481), the Natural Science Foundation of Hunan Province (Grant No. 2023JJ10016, 2023JJ30083), and the Provincial Key R&D Program of Hunan (2024AQ2025).

## References

[Bo *et al.*, ] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. Structural deep clustering network. In *Proceedings of The Web Conference 2020*, page 1400–1410, New York, NY, USA, April. Association for Computing Machinery.

[Chao *et al.*, 2024] Guoqing Chao, Yi Jiang, and Dianhui Chu. Incomplete contrastive multi-view clustering with high-confidence guiding. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 11221–11229, Vancouver, Canada, February 2024. AAAI Press.

[Chen *et al.*, 2022] Jianpeng Chen, Zhimeng Yang, Jingyu Pu, Yazhou Ren, Xiaorong Pu, Li Gao, and Lifang He. Shared-attribute multi-graph clustering with global self-attention. In *Neural Information Processing - 29th International Conference, ICONIP 2022, Virtual Event, November 22-26, 2022, Proceedings, Part I*, pages 51–63, Virtual Event, November 2022. Springer.

[Fan *et al.*, 2020] Shaohua Fan, Xiao Wang, Chuan Shi, Emiao Lu, Ken Lin, and Bai Wang. One2multi graph autoencoder for multi-view graph clustering. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 3070–3076, Taipei, April 2020. ACM / IW3C2.

[Fettal *et al.*, 2023] Chakib Fettal, Lazhar Labiod, and Mohamed Nadif. Simultaneous linear multi-view attributed graph representation learning and clustering. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023*, pages 303–311, Singapore, February 2023. ACM.

[Greenberg, 2009] Steven Greenberg. How citation distortions create unfounded authority: Analysis of a citation network. *BMJ (Clinical research ed.)*, 339:b2680, 2009.

[Hong *et al.*, 2020] Huiting Hong, Hantao Guo, Yucheng Lin, Xiaoqing Yang, Zang Li, and Jieping Ye. An attention-based graph neural network for heterogeneous structural learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 4132–4139, New York, NY, USA, February 2020. AAAI Press.

[Jiang *et al.*, 2024] Xinke Jiang, Zidi Qin, Jiarong Xu, and Xiang Ao. Incomplete graph learning via attribute-structure decoupled variational auto-encoder. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining, WSDM 2024, Merida, Mexico, March 4-8, 2024*, pages 304–312, Merida, Mexico, March 2024. ACM.

[Jin *et al.*, 2023] Shan Jin, Zhikui Chen, Shuo Yu, Muhammad Altaf, and Zhenchao Ma. Self-augmentation graph contrastive learning for multi-view attribute graph clustering. In *Proceedings of the 2023 Workshop on Advanced Multimedia Computing for Smart Manufacturing and Engineering, AMC-SME 2023, Ottawa ON, Canada, 29 October 2023*, pages 51–56, Ottawa ON, Canada, October 2023. ACM.

[Jr. *et al.*, 2005] Donald W. Miller Jr., John D. Yeast, and Robin L. Evans. Missing prenatal records at a birth center: A communication problem quantified. In *AMIA 2005, American Medical Informatics Association Annual Symposium, Washington, DC, USA, October 22-26, 2005*, Washington, USA, October 2005. AMIA.

[Kermani *et al.*, 2022] Ali Golzadeh Kermani, Ali Kamandi, and Ali Moeini. Integrating graph structure information and node attributes to predict protein-protein interactions. *J. Comput. Sci.*, 64:101837, 2022.

[Li *et al.*, 2023] Haobin Li, Yunfan Li, Mouxing Yang, Peng Hu, Dezhong Peng, and Xi Peng. Incomplete multi-view clustering via prototype-based imputation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, pages 3911–3919, Macao, China, August 2023. ijcai.org.

[Lin and Kang, 2021] Zhiping Lin and Zhao Kang. Graph filter-based multi-view attributed graph clustering. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 2723–2729, Montreal, Canada, August 2021. ijcai.org.

[Lin *et al.*, 2023] Zhiping Lin, Zhao Kang, Lizong Zhang, and Ling Tian. Multi-view attributed graph clustering. *IEEE Trans. Knowl. Data Eng.*, 35(2):1872–1880, 2023.

[Lu *et al.*, 2024] Hu Lu, Haotian Hong, and Xia Geng. Deep self-supervised attributed graph clustering for social network analysis. *Neural Process. Lett.*, 56(2):130, 2024.

[Mariani *et al.*, 2024] Paolo Mariani, Andrea Marletta, and Matteo Locci. Missing values and data enrichment: an application to social media liking. *Comput. Stat.*, 39(1):217–237, 2024.

[Miao *et al.*, 2021] Xiaoye Miao, Yangyang Wu, Jun Wang, Yunjun Gao, Xudong Mao, and Jianwei Yin. Generative semi-supervised learning for multivariate time series imputation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021,*

*The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 8983–8991, Virtual Event, February 2021. AAAI Press.

[Pan and Kang, 2021] Erlin Pan and Zhao Kang. Multi-view contrastive graph clustering. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 2148–2159, December 2021.

[Peng *et al.*, ] Zhihao Peng, Hui Liu, Yuheng Jia, and Junhui Hou. Attention-driven graph clustering network. In *Proceedings of the 29th ACM International Conference on Multimedia*, page 935–943, New York, NY, USA, October. Association for Computing Machinery.

[Qu *et al.*, 2017] Meng Qu, Jian Tang, Jingbo Shang, Xiang Ren, Ming Zhang, and Jiawei Han. An attention-based collaboration framework for multi-view network representation learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, pages 1767–1776, Singapore, November 2017. ACM.

[Taguchi *et al.*, 2021] Hibiki Taguchi, Xin Liu, and Tsuyoshi Murata. Graph convolutional networks for graphs containing missing features. *Future Gener. Comput. Syst.*, 117:155–168, 2021.

[Valencia *et al.*, 2019] Dalia Valencia, Rosa E. Lillo, and Juan Romo. A kendall correlation coefficient between functional data. *Adv. Data Anal. Classif.*, 13:1083–1103, 2019.

[Yoon *et al.*, 2018] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. GAIN: missing data imputation using generative adversarial nets. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pages 5675–5684, Stockholm, Sweden, July 2018. PMLR.