Expert Systems With Applications 238 (2024) 121821

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Integrating heterogeneous structures and community semantics for unsupervised community detection in heterogeneous networks

Yan Zhao, Weimin Li *, Fangfang Liu *, Jingchao Wang, Alex Munyole Luvembe

Department of Computer Engineering and Science, Shanghai University, Shanghai, China

ARTICLE INFO

Keywords:
Heterogeneous network
Community detection
Heterogeneous auto-encoder
Community semantic
Segmentation optimization

ABSTRACT

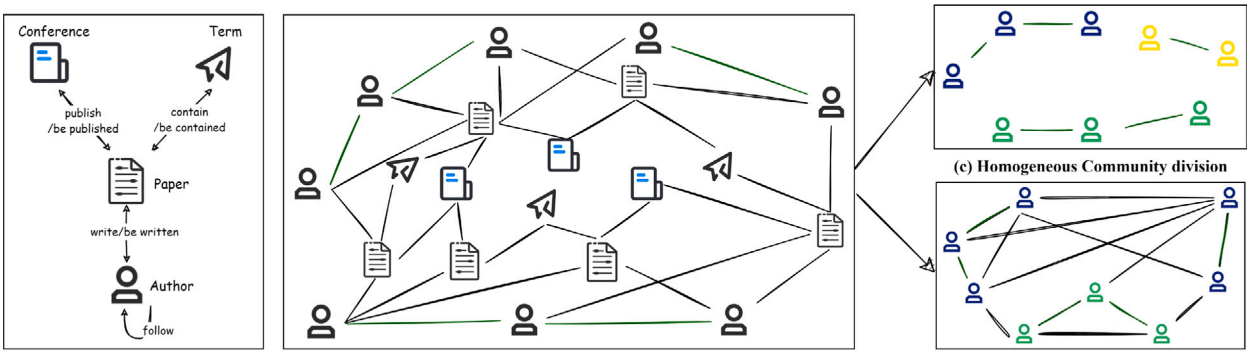
Community detection aims to discover hidden communities or groups in complex networks and is essentially unsupervised clustering behavior. However, most of the existing unsupervised methods are designed for homogeneous networks; therefore, they cannot effectively handle heterogeneous structures and rich semantic information. Under such a situation, it is difficult to accurately detect communities in heterogeneous networks that better reflect the real world. Therefore, this work aims to design an unsupervised framework to fuse heterogeneous structure information and interpret the rich semantics of the network in the form of community semantics. Thus, a heterogeneous network community detection method, called HAESF, is introduced. It includes two modules: the Heterogeneous Auto-Encoder (HAE) and the Semantic Factorization (SF) modules. In more detail, the HAE module adopts a hierarchical attention scheme to represent and aggregate the heterogeneous structure of the network. And it proposes the concept of heterogeneous information combinatorial graphs for structural reconstruction to achieve unsupervised detection. Concerning the SF module, it focuses on learning the semantic information in the network from the community point of view. It uses nonnegative matrix factorization to decompose the network features for obtaining community semantics. Once both modules are implemented, the objective of restricting community segmentation based on these semantics is achieved. The constraint is based on community semantic homogeneity to correct inaccurate node delineation. Furthermore, to improve the algorithm efficiency, a unified framework is designed to optimize the HAE and SF modules jointly. Within this new framework, the SF loss is innovatively used as a judgmental loss for selective segmentation optimizations, helping to obtain more reliable community detection results. As for the results, extensive experiments are performed on three public datasets. The findings show that HAESF outperforms the other popular unsupervised methods, where the composite score of HAESF is 11.73% ahead of the next best, demonstrating the proposed method's effectiveness.

1. Introduction

Network science is an interdisciplinary field that includes natural sciences, sociological sciences, computer sciences, engineering, and medicine (Khan & Niazi, 2017). DBLP, Twitter, and protein interaction networks are all objects of its research. It categorizes complex networks as homogeneous and heterogeneous based on network structure differences. Homogeneous networks contain only one type of node and edge. In contrast, heterogeneous networks contain various types of nodes and edges, leading to a more complex topology and richer semantic information than homogeneous networks. The difference is illustrated by the example of social networks. A homogeneous social network consists of only users, while the relationships between them are used as edges. Conversely, a heterogeneous social network can consist of three types of nodes: users, posts, and tags, and the edges can convey an array of relationships, such as following between users, liking posts, and associating tags. Clearly, heterogeneous networks better represent real networks and can be viewed as extensions of homogeneous networks. Referring to Fig. 1(a), a typical heterogeneous network is characterized by four types of nodes. As for Fig. 1(b), it presents a sample connection graph based on (a), where multiple nodes and edges form a complex network, with the green lines represent homogeneous relationships and the black lines represent heterogeneous relationships. As is evident, structural heterogeneity provides richer semantic information, which is crucial for describing the complexity of relationships and the diversity of nodes in the real world. It is particularly interesting to note that in today's era of big data, there is exponential growth in the types of nodes and the number of feature dimensions in complex networks.

* Corresponding authors.
E-mail addresses: orange2021@shu.edu.cn (Y. Zhao), wmli@shu.edu.cn (W. Li), ffliu@shu.edu.cn (F. Liu), static@shu.edu.cn (J. Wang), luvembe@shu.edu.cn (A.M. Luvembe).

https://doi.org/10.1016/j.eswa.2023.121821
Received 18 June 2023; Received in revised form 13 September 2023; Accepted 22 September 2023
Available online 6 October 2023
0957-4174/



(a) Network Schema   (b) Network instance diagram   (c) Homogeneous Community division   (d) Heterogeneous Community division

Fig. 1. Heterogeneous Network diagram.

Heterogeneous networks are becoming increasingly prevalent and relevant. Thus, the investigation of heterogeneous networks holds substantial theoretical and practical importance at this stage.

Moreover, detecting hidden communities in networks is a research hot-spot in network science. Sets of tightly connected nodes are divided into groups to reveal the structure and function of complex networks. Community detection in heterogeneous networks is one of the most direct and effective ways to study such networks. Additionally, it is an unsupervised process. For this reason, the main focus of this paper is to explore unsupervised community detection methods based on heterogeneous networks. In doing so, heterogeneous networks can be effectively analyzed and studied.

Currently, community detection methods, based on homogeneous networks, are quite mature and play a significant role in practical applications, such as the mining research groups in academic citation networks and the discovery of functional units in protein interaction networks (Bhowmick & Seah, 2015; Perianes-Rodríguez, Olmeda-Gómez, & Moya-Anegón, 2010). These methods are based on homogeneous networks modeled by a single type of node and do not take into account the heterogeneity of the network. Therefore, they have limitations when dealing with heterogeneous networks. To illustrate the limitation and the necessity of investigating heterogeneous methods, we take the heterogeneous network shown in Fig. 1(b) as an example and apply homogeneous and heterogeneous methods to detect communities on this network, respectively. The homogeneous method can only deal with a single type of node, so we extract only the Author node and the following relationship between Authors in Fig. 1(b) to construct the representation network. Fig. 1(c) illustrates the network structure and community segmentation obtained by the homogeneous method. The structure is simple with sparse connections between nodes. The communities derived from this network are isolated from each other, and the nodes within the communities are not well connected. This does not align with the actual network structure. The inability to effectively represent networks and thus accurately detect communities, which leads to the limitations of homogeneous methods when dealing with heterogeneous networks. Differently, heterogeneous method can be an effective solution to this limitation. Such method can handle multiple types of nodes, so we extracted Author nodes, Paper nodes, and semantic relationships between nodes to construct the representation network. Fig. 1(d) shows the heterogeneous approach results. For evaluation purposes, Paper nodes are hidden and the relevant semantic relationships are preserved, which remains in line with the network form in Fig. 1(c). Clearly, the heterogeneous method can represent richer network features. The community resulting from the method conforms to the community characteristics of "high cohesion and low coupling", and the node classification results are more comparable to reality. The results show that the heterogeneous method is much more effective than the homogeneous one. In actuality, the connecting edges among nodes of the same type in a heterogeneous network are quite sparse, as shown in Fig. 1(b). Analyzing and processing based on this sparse structure is challenging, which is the primary obstacle for homogeneous methods when it comes to dealing with heterogeneous networks. However, the heterogeneous method can alleviate this problem. By introducing various types of nodes, the method effectively preserves the complex topology and rich semantic relationships in the original network, allowing for more accurate detection of functional groups in the network. Ideally, we could introduce all heterogeneous nodes and semantic information in the network to obtain a more complete representation of the network and thus more accurate community segmentation (Han, Jin, Xing, Yang, & Xiong, 2023; Li, Ni, Wang, & Wang, 2022).

Apparently, heterogeneous methods are essential for detecting communities in heterogeneous networks. Furthermore, such methods have practical importance and applicability. For instance, in social networks, in addition to user–user interactions, the "user-post-user" heterogeneous edges can also be utilized to indirectly acquire interactions among users, and content-based interactions can help in building more meaningful interest groups. Therefore, this paper's objective is to propose a novel community detection method based on heterogeneous networks, which can effectively learn the heterogeneous structure and semantic information within the network in an unsupervised manner, enabling accurate detection of the communities.

In existing works, most unsupervised methods are developed for homogeneous networks (Jin et al., 2021; Lyu, Shi, Sun, & Lin, 2022). For instance, WCD considers the effect of second-order neighbors on node similarity and uses deep sparse auto-encoder for community detection in weighted networks (Li, Jiang, et al., 2021). To further account for node attributes, GUCD is proposed, which is the first unsupervised community detection method based on GCN (He, Song, et al., 2021). GUCD uses GCNs embedded with Markov Random Fields as the encoder (Jin, Liu, Li, He, & Zhang, 2019), and designs a dual encoder in the decoder part for reconstructing network topology and node attributes respectively. Wang et al. also proposed an attention-based auto-encoder to adequately represent the variability among nodes (Wang, Pan, et al., 2019). However, these existing unsupervised methods can only deal with network models consisting of a single type of nodes, without considering complex heterogeneous nodes in real networks. Moreover, these methods tend to focus on the topology for community detection. Nevertheless, this strategy is challenging to apply in heterogeneous networks with large amounts of semantic information. In other words, homogeneous unsupervised methods are not suitable to handle heterogeneous networks due to their inherent limitations. Therefore, CPGNN innovatively uses contextual paths to extract heterogeneous node information. However, CPGNN is a supervised method that requires prior knowledge (Luo, Fang, Cao, Zhang, & Zhang, 2021). Consequently, it is challenging and rational to incorporate heterogeneous information into existing unsupervised work to achieve effective unsupervised community detection of heterogeneous networks.

To address the above-listed challenges, we propose the Heterogeneous Auto-Encoder with Semantic Factorization (HAESF) method, which fuses the heterogeneous structure with the semantic information. The method sufficiently learns the characteristics of heterogeneous networks. It not only heterogeneously handles multiple types of nodes, but also effectively exploits the rich semantic information in the network.

In summary, the contributions and innovations of this paper include the following:

(1) HAESF proposes a unified framework for aggregating the Heterogeneous Auto-Encoder (HAE) and the Semantic Factorization (SF) modules. The joint learning strategy ensures that heterogeneous structures are successfully integrated within community semantics;
(2) HAE is deployed for aggregating heterogeneous structural features, where a two-layer attention mechanism is used to learn the heterogeneous structure. For effective unsupervised learning, the Heterogeneous Information Combination Graph (HICG) is used as a target object for reconstruction;
(3) SF is concerned with learning semantic information, where a double-layer NMF is used to represent the semantic information of the network in the form of communities. Moreover, the consistency of the community semantics and the node attributes is used to semantically constrain community detection, assuming the community semantic homogeneity;
(4) In the joint optimization part, this paper innovatively proposes judgmental loss, based on the SF loss, to make reliability judgments, thus achieving selective segmentation optimization;
(5) Extensive experiments have been conducted to demonstrate the HAESF method's effectiveness.

The rest of this paper is organized as follows. In Section 2, we provide a brief overview of the relevant works. In Section 3, the concepts and definitions involved in the paper are explained. A detailed description of the proposed method HAESF is presented in Section 4. Section 5 is the experimental part that validates the effectiveness of HAESF. In Section 6, the paper is summarized.

2. Related work

This section begins with an overview of representative methods that already exist. It then details two methods related to the HAESF model: the Nonnegative Matrix Factorization (NMF)-based method and the Graph Neural Network (GNN)-based method.

2.1. Methodological overview

Currently, many complex network-based community detection methods have been proposed with excellent performance (Li, Zhu, et al., 2021; Naderipour, Fazel Zarandi, & Bastani, 2022). Existing methods include modularity-based methods, spectral clustering-based methods, Label Propagation Algorithm (LPA)-based methods, NMF-based methods, GNN-based methods, etc. Modularity-based methods aim to maximize the modularity of the community but are prone to under or over-division (Newman, 2006). Spectral clustering-based methods employ the feature vectors obtained from the spectral decomposition of the similarity matrix for community segmentation, which has better robustness and high-dimensional data processing capability but requires higher complexity for large networks (Li, He, Kloster, Bindel, & Hopcroft, 2018; Xu, Zhuang, Li, & Zhou, 2018). LPA is a propagation method that uses the similarity between nodes to continuously propagate label information to classify unlabeled nodes (Raghavan, Albert, & Kumara, 2007). Despite its simplicity and efficiency, this method is subject to falling into the local optimal solution, besides being sensitive to the label of the initial nodes. Aside from these classical methods, there are several novel perspectives to consider (Li, Zhang, Zhou, & Jin, 2023; Shang, Zhang, Zhang, Feng, & Jiao, 2022). For example, influence maximization-based methods assess nodes' influence in a network (Sun, Sun, Liu, Cao, & Yang, 2023; Zhang, Li, Wei, Liu, & Li, 2022). They consider the nodes with the most significant influence as the core nodes of the community, to which they seek to extend the community. But such methods may ignore some small and marginal groups.

Compared to the above methods, the NMF-based methods are simpler and more efficient. NMF does not require initial labeling, is not prone to segmentation problems, is easy to implement, and, most importantly, has superior interpretability. Therefore, it is widely used for many types of community detection tasks, including, but not limited to, attribute network detection (Qin, Jin, He, Gabrys, & Musial, 2017), supervised detection (Liu et al., 2017), multi-layer network detection (Gligorijević, Panagakis, & Zafeiriou, 2018), overlapping community detection, etc. In practice, the methods introduced have a wide range of applications. However, with the increasing network complexity, we find that these methods perform community detection by learning the network topology or node attributes independently (Guo et al., 2022; Li, Tang, Tang, Cao, & Zhang, 2022), and this processing strategy is unable to fully integrate the network topology and attribute features, which may lead to the problem of missing features or mismatch between structure and attributes. Meanwhile, the development of GNN has brought a turnaround to this problem. GNN has demonstrated excellent capabilities in representing graph data in recent years, performing well in tasks such as classifying nodes, predicting links, and detecting communities of interest (Su et al., 2022). GNN-based methods efficiently integrate the topology of the network with the attributes of the nodes, solving the problem of mismatch between topology and attributes while achieving sufficient learning of network features. Such methods have stable and excellent performance on complex networks with heterogeneous structures and rich semantics.

Inspired by the existing work, this paper presents an effective attempt to combine the NMF with the GNN.

2.2. NMF-based community detection

NMF is an approximate method for decomposing a non-negative data matrix into its basis matrix and its coefficient matrix. As a general rule, a sample of data can be represented as a linear combination of the features in the base matrix. The interpretability, simplicity, and efficiency of this approach make it widely used in a variety of areas, including community detection (Luo, Liu, Jin, Zhou, & Zhou, 2021; Pompili, Gillis, Absil, & Glineur, 2014).

For instance, NMFjGO dynamically incorporates attribute information into the topology, dealing with possible heterogeneity between attributes and topology (Huang, Zhong, Wang, Gong, & Ma, 2020). For complex networks, however, a shallow NMF such as the one used in NMFjGO is insufficient to abstract the complex mapping between the network structure and the community structure. Accordingly, DeepANMF presents an auto-encoder-like model (Ye, Chen, & Zheng, 2018), which consists of multiple NMF layers for achieving community delineation through layer-by-layer abstraction of the network structure. Note that this approach relies solely on topology structures and ignores node semantics. To efficiently learn complex structural and semantic information from networks, researchers have combined NMF with deep learning models. An example is ANMF embeds NMF into the GAE framework to impose more constraints on network reconstruction and resolve the mismatch between attributes and structures (Zhao et al., 2022). However, NMF is still linear in this case and has limited capabilities in handling complex structures. Thus, NMFGAAE combines NMF with GAE and can detect communities effectively while resolving the inherent linearity of NMF (He, Zheng, et al., 2021). However, the attention mechanism used in this method could not work better for heterogeneous networks containing several types of nodes.

As described above, the linear model NMF faces the obstacle of limited representation capability when dealing with complex networks. The combination with a deep network model can alleviate this problem. The specific model should be determined by the actual network situation.

2.3. GNN-based community detection

GNN is a deep learning technique that processes graph-structured data. The GNN-based methods for detecting communities have demonstrated promising results in several studies (Yuan, Zeng, Zuo, & Wang, 2023).

For example, LGNN uses a non-backtracking operator to define the adjacency of edges to solve the community detection problem in a supervised setting (Chen, Li, & Bruna, 2017). Graph Attention Network (GAT) is a GNN-based attention mechanism to learn the importance between nodes and their neighbors, and to aggregate neighbor information for community detection (Veličković et al., 2017). Nevertheless, GAT does not account for the heterogeneity of nodes or edges, so it cannot handle heterogeneous networks. HAN based on hierarchical attention addresses this shortcoming, which employs meta-paths for learning network structure and heterogeneous information (Wang, Ji, et al., 2019), and then uses hierarchical attention to learn the representation of heterogeneous structure and semantics.

Despite the excellent performance of the above methods in supervised community detection, community detection is essentially an unsupervised task, so we use Graph Auto-Encoder (GAE) for it. GAE allows the user to choose from various GNNs as encoders. For example, VGAE chose the GCN-Encoder, which employs two convolutional layers in the encoder part to learn the Gaussian distribution of the data (Kipf & Welling, 2016). With variational inference, the model performs well even with missing data. On this basis, VGAER is improved by using modularity information as a higher-order feature of the node to jointly optimize the structure with lower-order features (Qiu, Huang, Xu, & Li, 2022). Furthermore, GAT-Encoder uses attention networks as an encoder to distinguish the importance of different nodes (Luvembe, Li, Li, Liu, & Xu, 2023). However, GAE methods do not effectively handle heterogeneous network structures, so they cannot be applied to heterogeneous networks.

Consequently, this paper proposes a new heterogeneous auto-encoder to achieve representation learning of heterogeneous structures and structure semantics. Additionally, deep model training results are often difficult to interpret, and NMF's high interpretability can precisely compensate for this deficiency. Therefore, this paper adopts this win-win combination of NMF and GAE as a unified framework for learning heterogeneous networks.

3. Preliminaries and notations

Definition 1 (*Heterogeneous Network*)**.** A complex network containing nodes or edges of different types, where each node and edge has different properties and meanings. When a complex network is represented by a graph $G = (V, E)$, V denotes the set of nodes, $|V| = N$, E denotes the set of edges, and the type mapping functions of nodes and edges are $f(V) \rightarrow \Phi$ and $g(E) \rightarrow \Psi$ respectively. A network is said to be heterogeneous when the sum of the number of types of nodes and edges in the network is greater than 2, i.e., $|\Phi| + |\Psi| > 2$, and homogeneous when $|\Phi| + |\Psi| = 2$.

Definition 2 (*Master Node and Slave Node*)**.** In a heterogeneous network, there are several types of nodes, but usually, only a single type has a dominant role and is referred to as the master node. Other nodes are called slave nodes and act as a bridge to the master node. For instance, in Fig. 1(a), we consider the Author to be the master node, playing the dominant role, and the Paper, Conference, and Term to be slave nodes. The master node is the object of community detection, and the slave nodes provide heterogeneous information to complement the structure and meaning of the master node.

Table 1
Notations and explanations.

Notation	Explanation
X	Node feature representation
Θ_k	kth meth-path
G_k	Semantic graph based on Θ_k
α_{uv}^k	Weight of the node pair (u, v) in the G_k
$\vec{z}_u^{\,k}$	Latent representation of the node u in the G_k
β_k	Weight of the G_k
Z	Final embedding representation
U	Community membership matrix
S	Community semantic matrix
ρ	Judgment threshold

Definition 3 (*Meta-path*)**.** Meta-paths are paths that connect two specific entities within a network and are abstract representations of particular semantics. A meta-path is usually represented by $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \ldots \xrightarrow{R_l} A_{l+1}$, where A_i denotes the node through which the path passes, and $R_1 R_2 \ldots R_l$ represents the composite relationship from the source node A_1 to the target node A_{l+1}. As part of this study, we set A_1 and A_{l+1} as the master nodes, and intermediate nodes on the pathway as the slave nodes. We use meta-paths to obtain the structural relationships between the two master nodes. Depending on how many slave nodes are involved, meta-paths can be divided into short-range and long-range.

Short-Range Meta-Path: There is one slave node passing through the meta-path, as defined by APA.

Long-Range Meta-Path: A meta-path that passes through more than one slave node, as in APCPA. It is generally true that short-range meta-paths connect node pairs more tightly than long-range meta-paths.

Meta-path neighbor: In a meta-path instance, two master nodes A and B are considered to be each other's meta-path neighbors, and we treat meta-path neighbors as structural neighbors that can convey semantic information.

Definition 4 (*Master Node Semantic Graph*)**.** Using meta-paths for master node-oriented representation learning of heterogeneous networks, we can obtain homogeneous graphs that show edge relationships between master nodes. Since this homogeneous graph is extracted under specific meta-path semantics, we refer to it as the master node semantic graph.

Definition 5 (*Heterogeneous Information Combination Graph, HICG*)**.** HICG is obtained by linearly combining multiple master node semantic graphs, which preserves structural features and semantic information well. The theoretical basis is that a meta-path is a generalized abstraction of the semantics of the network, allowing the master node semantic graphs to be understood as mapping different views of the same network. A HICG structure is shown in Fig. 1 (d), where only the master node and meta-path relationships are preserved and the structural heterogeneity information is hidden.

Definition 6 (*Community Semantic Homogeneity Hypothesis*)**.** The hypothesis assumes that nodes within a community have similar semantic features, which is to say that nodes within a community share certain semantic features consistent with the overall semantics of the community. Our hypothesis is based on the assumption of homogeneity, which we use to explain the consistency between node attributes and community semantics.

Some notations and explanations are given in Table 1.

4. The proposed model

In this section, the proposed HAESF model is described in detail. As shown in Fig. 2, the model can be divided into three modules. (a) Preprocessing module. This module deals with heterogeneous networks

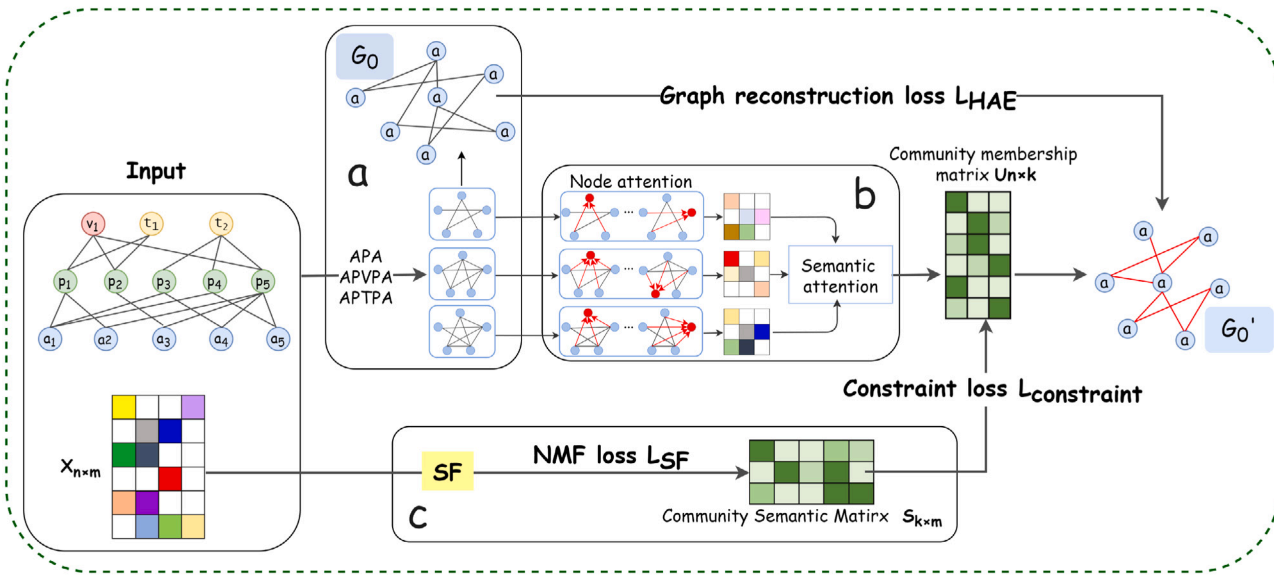


Fig. 2. The framework of HAESF. Pre-processing module (a) processes the original network to preserve heterogeneous information. HAE module (b) performs unsupervised representation learning and obtains a community partition U. SF module (c) obtains community semantics S and uses it to constrain U.

based on meta-paths and defines HICG to preserve heterogeneous structures. This module is not part of formal training. (b) Heterogeneous Auto-Encoder module (HAE). The module can efficiently aggregate the heterogeneous network structure and contextual information to detect communities in an unsupervised manner. (c) Semantic Factorization module (SF). This part processes the network feature information and presents it in the form of community semantics. The semantic information obtained from SF is used to enforce constraints on community detection. In the formal training process, HAE and SF are aggregated into a unified framework. Moreover, a judgmental segmented optimization is proposed to improve the performance of the algorithms.

4.1. Pre-processing

This module is proposed to retain the heterogeneous information in the network. It is important to note that in heterogeneous networks, the relationships between homogeneous nodes tend to be relatively sparse, which is the main reason why existing homogeneous methods perform poorly. To solve this problem, heterogeneous information must be introduced to enrich the network structure.we need to introduce heterogeneous information into the network.

Our basic approach is to preserve relationships between heterogeneous nodes using meta-paths, as shown in Fig. 2(a). Using the predefined meta-paths $(\Theta_1, \Theta_2, \dots, \Theta_K)$, we extract the master node semantic graphs $(G_1, G_2, \dots, G_K)$ and input them into HAE.

Our advanced approach is to propose the concept of the Heterogeneous Information Combination Graph (HICG). In this paper, we use structure reconstruction to achieve unsupervised community detection, specifically the reconstruction of the network structure associated with the master node. Unlike homogeneous networks, we do not have a determined master node graph as the reconstruction object. Thus, we innovatively proposed the HICG as a reconstruction target. HICG is a linear combination of the master node semantic graphs $(G_1, G_2, \dots, G_K)$. Since these semantic graphs represent different views of a heterogeneous network, their effective combination can preserve the structural features and semantic information of the network. The combination is as follows:

$$G_0 = \theta_1 * G_1 + \theta_2 * G_2 + \cdots + \theta_K * G_K \tag{1}$$

where the combination coefficients $(\theta_1, \theta_2, \dots, \theta_K)$ represent the weights of different meta-paths. Meta-path' importance varies according to their semantics and length, so they must be given weights according to their contributions.

4.2. Heterogeneous Auto-Encoder

HAE is a model for learning node features and detecting communities in heterogeneous networks. Since Auto-Encoder (AE) is a commonly used unsupervised means, in this paper, AE will be given a heterogeneous treatment to make it applicable to heterogeneous networks. As shown in Fig. 2 (b), HAE consists of two components: Heterogeneous encoder and Decoder. The former is based on a two-layer self-attention mechanism for efficient aggregation of heterogeneous structural and semantic information, and the latter is based on HICG to achieve unsupervised community detection.

4.2.1. Two-layer attention-based Heterogeneous Encoder

Since different nodes in the same semantic graph, as well as different semantic graphs, have different importance, we propose a two-layer attention mechanism. The first layer is used to learn attention between nodes, and the second layer is used to learn self-attention between semantic graphs. The input is the set of master node semantic graphs $(G_1, G_2, \dots, G_K)$. Output is a coarse community membership matrix U, the matrix is then optimized using semantic constraints.

A. Attention Between Nodes

We use a GAT network to perform self-attentive learning to determine the importance of nodes under independent semantics. As input, we have semantic graphs $(G_1, G_2, \dots, G_K)$ and node features $X = \{\vec{x}_1, \vec{x}_2, \dots, \vec{x}_N\}$, $\vec{x}_u \in \mathbf{R}^M$, where N denotes the number of nodes and M denotes the dimensions of the features.

We need to compute the weights between nodes under each semantic graph independently. Semantic graphs and meta-paths have a one-to-one correspondence. To begin with, the similarity between a node and its meta-path neighbors is computed in the following manner:

$$e^k_{uv} = a(W\vec{x}_u, W\vec{x}_v) \tag{2}$$

where e^k_{uv} denotes the importance of the node pair (u, v) in the semantic graph G_k, a is the shared attention mechanism and W is the learnable model parameter.

For ease of processing, we normalize the importance of node pairs using SoftMax to obtain the weights α_{uv}^k:

$$\alpha_{uv}^k = softmax(e_{uv}^k) = \frac{\exp(e_{uv}^k)}{\sum_{i\in N_u}\exp(e_{ui}^k)} \tag{3}$$

where N_u denotes the set of neighbors of node u.

Subsequently, we obtain a vector representation of the nodes by aggregation in the following way:

$$\vec{z}_u^{\,k} = \sigma\left(\sum_{v\in N_u}\alpha_{uv}^k W\vec{x}_v\right) \tag{4}$$

where $\vec{z}_u^{\,k}$ represents the vector representation of the node u in the semantic graph G_k and σ represents the activation function.

Since the multi-head attention mechanism is more effective at capturing node features, we calculate multiple attention heads in parallel, contact the output, and transform it linearly to obtain the final representation of the nodes:

$$\vec{z}_u^{\,k} = \|_{d=1}^{D}\sigma\left(\sum_{v\in N_u}\alpha_{uv}^{k^d} W^d\vec{x}_v\right) \tag{5}$$

where $\|$ represent contact operation, D is the number of attention heads, $\alpha_{uv}^{k^d}$ is attention weight computed by dth attention head, W^d is a model parameter.

A single semantic graph can be represented by the following: $Z^k = \left\{\vec{z}_1^{\,k}, \vec{z}_2^{\,k}, \dots, \vec{z}_N^{\,k}\right\}$, the combined representation of multiple semantic graphs is: $Z = \left\{Z^1, Z^2, \dots, Z^K\right\}$, where K is the number of semantic graphs.

B. Attention Between Semantic Graphs

To measure the contribution of different semantic graphs to the heterogeneous network representation, we determine the importance w_k of semantic graph G_k based on the above node representation Z^k.

$$w_k = \frac{1}{|V|}\sum_{i\in V} q^T \cdot \tanh\left(W\cdot\vec{z}_k + b\right) \tag{6}$$

where V is the node set, q is the semantic-level attention vector, W is the weight matrix and b is the bias.

SoftMax is used to normalize the semantic graph weight coefficients β_k:

$$\beta_k = \frac{\exp\left(w_k\right)}{\sum_{p=1}^{K}\exp\left(w_p\right)} \tag{7}$$

Finally, to obtain the complete heterogeneous graph representation, the node representations of the different semantic graphs are weighted and summed.

$$Z = \sum_{k=1}^{K}\beta_k\cdot Z_k \tag{8}$$

To derive a rough community membership matrix U, we perform clustering based on the node representation Z.

4.2.2. HICG-based decoder

Based on U, we reconstruct the heterogeneous structure. This operation is implemented using inner products:

$$\hat{A} = sigmoid(UU^T) \tag{9}$$

HAE loss is measured by the binary cross-entropy measure, which is calculated as follows:

$$L_{HAE} = -\frac{1}{n^2}\sum_{u,v} A_{uv}\log\hat{A_{uv}} + (1-A_{uv})\log(1-\hat{A_{ij}}) \tag{10}$$

where $\hat{A}$ denotes the reconstructed graph structure and A denotes the structure of the HICG G_0, here the adjacency matrix is used to represent the network structure.

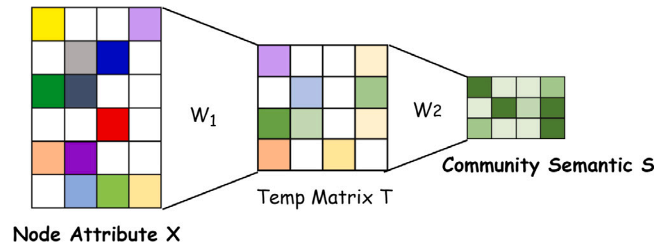


Fig. 3. Double-layer NMF.

4.3. Semantic factorization

SF module preserves the semantic information of the network in the form of community semantics. Since heterogeneous networks are semantically rich, we introduce this module to handle the network feature. To improve the interpretability of the obtained communities, we use the NMF method. As shown in Fig. 2(c), we use NMF to process the node attribute features and represent the result of the decomposition as a community semantic matrix S. S serves as a constraint to optimize the community segmentation result U. The theory followed by this module is the homogeneity assumption of the community semantics, which was introduced in Section 3.

As nodes have rich attribute features, shallow NMF may be unable to map them effectively, and the factorization of single-layer NMF is uncertain. Consequently, the module adopts a double-layer NMF for decomposition, as shown in Fig. 3.

The process can be decomposed into two consecutive single-level NMF processes:

$$\begin{aligned} X &\approx W_1 T \\ T &\approx W_2 S \end{aligned} \tag{11}$$

where W_1, W_2 represent coefficient matrixes, and T as a temporary transition matrix that does not represent a specific meaning.

For calculation purposes, in practice, we optimize the double-layer NMF directly:

$$X \approx W_1 W_2 S \tag{12}$$

The loss function for semantic factorization is as follows:

$$\begin{aligned} L_{SF} &= \|X - W_1W_2S\|_F^2 \\ s.t.\ \ & W_1, W_2 \geq 0, S \geq 0 \end{aligned} \tag{13}$$

The parameter update formula is as follows:

$$\begin{aligned} W_1 &= W_1 \odot \frac{XS^TW_2^T}{W_1W_2SS^TW_2^T} \\ W_2 &= W_2 \odot \frac{W_1^TXS^T}{W_1^TW_1W_2SS^T} \\ S &= S \odot \frac{W_2^TW_1^TX}{W_2^TW_1^TW_1W_2S} \end{aligned} \tag{14}$$

Based on the above update process, we can obtain a stable and credible semantic matrix S. The next step is semantic constraints on community detection. We constrain and correct the community membership matrix U in HAE under the assumption of community semantic homogeneity. The loss function is shown below:

$$L_{constraint} = \|X - US\|_F^2 \tag{15}$$

Optimizing the loss $L_{constraint}$, the heterogeneity between the community membership matrix U and the community semantic matrix S can be minimized, which allows the affiliation of nodes to be accurately determined and corrected at the level of community semantics. HAE and SF are associated with $L_{constraint}$, which ensures consistency and coordination between the two modules.

Algorithm 1 HAESF

Require: Graph $G = (V, E)$, Attribute X, Semantic graphs $(G_1, G_2, \ldots G_K)$, Parameter: λ, μ, ρ
Ensure: Communities Set $C = \{C_1, C_2, \ldots, C_k\}$

for $epoch = 1, 2, \ldots$ **do**
  for each semantic graph **do**
    for each node $v \in V$ **do**
      Compute attention between nodes
    end for
    Compute attention between semantic graphs
  end for
  Generate Community membership matrix U and get L_{HAE}
  Compute community semantic matrix S via executing Eq. (14) and get L_{SF}
  Impose semantic constraints on U by Eq. (15)
  Beginning of reliability judgment
  if $L_{SF} \leq \rho$ **then**
    Jointly optimize by minimizing Eq. (16)
  end if
  if $L_{SF} > \rho$ **then**
    Individually optimize L_{HAE} by minimizing Eq. (17)
  end if
end for
for each node $v \in V$ **do**
  $q = argmaxU_{ik}$
  $C_q = C_q \cup \{v\}$
end for
return C

4.4. Model optimization

Since L_{SF} and $L_{constraint}$ depend on the community semantic matrix, however, the decomposition process of NMF is unstable and stochastic. The double-layer NMF makes the process more stable, but the semantic matrix obtained from each decomposition iteration is not completely reliable. For this reason, instead of directly optimizing the L_{SF}, we use an innovative approach to increase the robustness of our model. We treat it as a judgmental condition. That is, we define L_{SF} as a judgmental loss.

Definition 7 (*Judgmental Loss*)**.** The loss is used as a judgment condition for selective optimization, rather than direct optimization. The next optimization method is chosen based on the judgment result of the loss.

Definition 8 (*Judgment Threshold*)**.** The threshold is used to evaluate the reliability of results relating to judgment loss. If the judgment loss is less than the threshold, we consider the results as reliable, which can be used for community detection and the selection of joint optimization methods. Conversely, if the judgment loss is more than the threshold, it is considered unreliable, and we resort to a single optimization approach.

Specifically, we set a reasonable threshold ρ. When $L_{SF} \leq \rho$, the current semantic matrix is considered reliable, and the losses of HAE and SF are combined for joint optimization:

$$\min Loss = \lambda L_{HAE} + \mu L_{constraint} \tag{16}$$

where λ, μ are hyperparameters to adjust the weights between the two losses.

When $L_{SF} \geq \rho$, this semantic matrix is not reliable and we do not adopt it. In this case only the loss L_{HAE} is used for optimization:

$$\min Loss = \lambda L_{HAE} \tag{17}$$

During the optimization process, the threshold is continuously updated until the most reliable community semantic matrix is determined.

The detailed optimization algorithm can be found in Algorithm 1.

5. Experiments

This section is an experimental section where the dataset, benchmark model, and test experiments are described in detail.

5.1. Datasets and baselines

5.1.1. Datasets

Datasets for DBLP, ACM, and IMDB are representative of heterogeneous networks.DBLP is a typical academic paper database. It contains information on papers in computer science and related fields. DBLP consists of four types of nodes: Authors, Paper, Conference, and Term. These nodes form a complex relationship. A subset of the DBLP was used for this study. ACM is also a member of the academic papers network, which contains papers from various academic conferences and journals sponsored by ACM. There are three types of nodes in the database, Author, Paper, and Subject. IMDB is an online film database containing a large number of film and television productions, consisting of three types of nodes: Movie, Actor, and Director.

All nodes in the above dataset have unique community identifiers. Detailed data information can be found in Table 2.

5.1.2. Baseline models

HAESF is compared with some popular unsupervised methods to verify its effectiveness. Since unsupervised methods are rare in heterogeneous networks, we have selected several methods that have demonstrated success in homogeneous networks.

DAEGC (Wang, Pan, et al., 2019): A graph attention auto-encoder method, which uses KL scatter for self-optimal learning of node representations and optimizes both node embedding and clustering simultaneously.

VGAE (Kipf & Welling, 2016): A variational graph auto-encoder that can incorporate latent variables and yield interpretable node representations from a Gaussian distribution.

VGAER (Qiu et al., 2022): A modified version of VGAE, VGAER adds modularity as a high-order feature and changes the reconstruction objective.

GAER (Qiu et al., 2022): A graph attention-based method that introduces modularity information as high-order features and then uses an attention network to learn these features. It is a non-variable version of VGAER.

DANMF (Ye et al., 2018): A multi-layer NMF approach similar to Auto-encoder, where both Encoder and Decoder are composed of multiple one-layer NMFs, thus addressing the issue of shallow NMFs failing to map complex network relationships.

HAN (Wang, Ji, et al., 2019): A supervised community detection method for heterogeneous networks that uses hierarchical attention to learn node representations under different meta-paths. We embed HAN as an encoder component in the Auto-Encoder framework to enable unsupervised learning.

5.2. Comparison experiments

5.2.1. Evaluation metrics

The algorithm performance is evaluated using four commonly used unsupervised clustering metrics. The metrics are Normalized Mutual Information (NMI), Adjusted Rand Coefficient (ARI), Accuracy (ACC), and F1-score. We used ARI and NMI to assess the similarity between the results of community detection and the actual distribution of nodes. ACC measures the algorithm's accuracy in predicting node labels within different communities, while F1-score combines accuracy and recall to assess its performance more comprehensively.

Table 2
Statistics of datasets.

Dataset	Type(Number)	Master-Node	Feature	Relation	Meta-path
DBLP	Author(A,4957) Paper(P, 14328) Conference(C,20) Term(T, 8789)	Author	334	A-P P-C P-T	APA APCPA APTPA
ACM	Paper(P,3025) Author(A,5835) Subject(S,56)	Paper	1232	P-A P-S	PAP PSP
IMDB	Movie(M,4780) Actor(A,5841) Director(D,2269)	Movie	1830	M-A M-D P-T	MAM MDM

5.2.2. Experimental setup

To ensure a fair comparison, parameters were set to their optimal values for each comparison experiment. In particular, for DAEGC, we set the regularization parameter to 10 and divided the experiments into pre-training and formal training stages following their original settings. DANMF's regularization parameter and layer number are set to 1 and m-128-64-k, respectively. VGAE's learning rate and number of layers are set at 0.01 and m-32-16-k, correspondingly. VGAER has a layer count of m-64-32-k. GAER uses the same parameter settings as VGAER. In HAN, 2 and 8 are the regularization parameters and attention heads, respectively. For HAESF, the number of layers is set to m-128-64-k, dropout to 0.5/0.6, the learning rate is 0.005, and the epoch upper limit to 400.

5.2.3. Performance analysis

As previously stated, these methods are not capable of handling complex heterogeneous structures. To address this issue, we added a network pre-processing stage to each algorithm, using the processed homogeneous graphs as input to these methods. The experimental results are presented in Tables 3–5. All results were obtained by repeating the experiment 10 times and then taking the average.

Table 3 shows the experimental results based on DBLP, and it can be seen that HAESF outperforms other comparison methods in all metrics, with HAESF's NMI scores being 13%, 99.7%, 39.2%, 49%, 15.8%, and 46.2% higher than DAEGC, DANMF, GAER, VGAE, VGAER, and HAN, respectively. Despite performing extremely well in unsupervised community detection in recent years, DAEGC and VGAER still perform somewhat worse than HAESF, demonstrating the effectiveness of HAESF in learning heterogeneous networks. Compared to some other comparative methods, this method performs significantly better. The superiority of HAESF over the heterogeneous method HAN is also evident, which shows the criticality and effectiveness of the proposed community semantic constraints. As shown in Table 4, HAESF obtains the best results on all metrics except ARI and has a significant advantage over the second-best metric. On the relatively inferior ARI, our scores are also fairly comparable.

Table 5 is taken from experiments on IMDB. GAER achieved the best results on both NMI and ARI metrics, and HAESF achieved the best scores on ACC. We found that the scores based on IMDB are generally lower compared to the first two datasets. This is strongly related to IMDB's data characteristics. To prove this, we performed data statistics on these three networks, and the statistical metrics include feature sparsity and structural sparsity. The metric scores for DBLP, ACM, and IMDB are [3.60%, 22.05%], [4.47%, 41.49%], and [0.20%, 0.14%], respectively. Compared to other datasets, IMDB has very sparse features and structure. A large number of models are incapable of processing efficiently on sparse networks, and HAESF is no exception. Despite this, HAESF's performance is still competitive on a relative basis.

Table 3
Comparison of DBLP.

	DAEGC	DANMF	GAER	VGAE	VGAER	HAE	HAENMF
NMI	0.6620	0.0026	0.4629	0.3883	0.6406	0.4097	**0.7612**
ARI	0.7137	0.0012	0.5221	0.6179	0.7769	0.3897	**0.8051**
ACC	0.8676	0.3015	0.4891	0.5381	0.6518	0.6283	**0.9172**
F1	0.8507	0.1457	0.3883	0.3588	0.6219	0.5095	**0.9094**

Table 4
Comparison of ACM.

	DAEGC	DANMF	GAER	VGAE	VGAER	HAE	HAENMF
NMI	0.4685	0.3820	0.4109	0.2929	0.4108	0.4132	**0.5733**
ARI	0.3533	0.3245	**0.6483**	0.6109	0.6483	0.3461	0.4949
ACC	0.6073	0.6509	0.6616	0.6115	0.6616	0.6522	**0.7140**
F1	0.5098	0.6566	0.3427	0.2911	0.3427	0.6659	**0.7156**

Table 5
Comparison of IMDB.

	DAEGC	DANMF	GAER	VGAE	VGAER	HAE	HAENMF
NMI	0.0848	0.0024	**0.1911**	0.0010	0.0001	0.0123	0.0430
ARI	0.0861	−0.0034	**0.3567**	0.3422	−0.0034	0.0138	0.0548
ACC	0.4336	0.3453	0.2103	0.3415	0.3453	0.4306	**0.4444**
F1	**0.4979**	0.3105	−0.0083	−0.0001	0.3105	0.3254	0.3688

5.2.4. Discussion

From the results on both DBLP and ACM datasets, the NMI score of HAESF is much higher than the other methods, which shows the superiority of the method. However, the method only ranks third in terms of NMI score on IMDB, which is a limitation of the method. To evaluate the continued competitiveness of HAESF's comprehensive performance across multiple datasets, we utilize the Friedman test to determine significance (Kasihmuddin, Jamaludin, Mansor, Wahab, & Ghadzi, 2022; Zamri, Azhar, Mansor, Alway, & Kasihmuddin, 2022).

The test formulates hypothesis H_0: there is no significant difference in performance between the algorithms, which holds at the significant level $\alpha \geq 0.05$. Using the NMI scores obtained from the comparison experiments as representative indicators for the significance test, the statistics are calculated as shown below:

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_{j=1}^{k} r_j^2 - \frac{k(k+1)^2}{4}\right] \tag{18}$$

where N, k are the number of datasets and algorithms, respectively, and r_j is the average ranking of the jth algorithm. We compute the Friedman statistic to be 13.429 with $\rho = 0.037 < \alpha$, so the H_0 is rejected, proving that there are significant differences between several algorithms. To observe the comparison effect more intuitively, we visualize the results of this test using a boxplot, as shown in Fig. 4.

HAESF performs significantly better than other algorithms. We can see that the upper bound and the median of HAESF are significantly higher than the other boxes. This indicates that the algorithm outperforms the other algorithms both in terms of best and average performance. It is notable that DANMF consistently performs worse than the

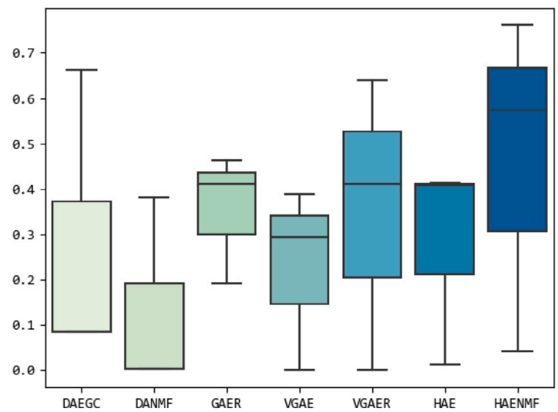


Fig. 4. Significance testing.

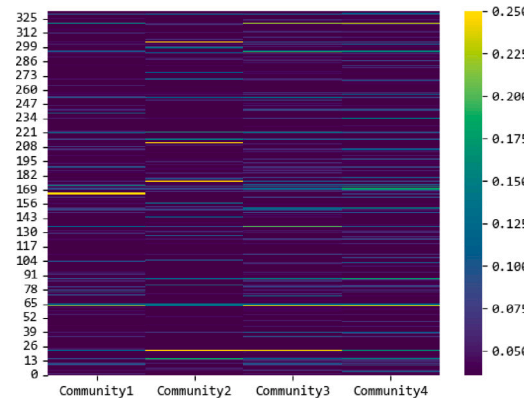


Fig. 5. Community interpretability.

other algorithms. Because for semantically information-rich heterogeneous networks, considering only the network topology is not sufficient to support the algorithm in making the correct division choice. This is in contrast to homogeneous networks, which rely mainly on topology for community decomposition, and where semantic information is always used as a supplementary resource.

Overall, HAESF has obvious superiority and stability in unsupervised community detection for heterogeneous networks. Compared to other unsupervised methods, HAESF performs significantly better than baselines on different metrics. Moreover, the robust performance of HAESF on different datasets is sufficient to demonstrate the effectiveness of the algorithm, which is capable of performing unsupervised community detection on heterogeneous networks with excellent results.

5.3. Community interpretability analysis

To illustrate the interpretability of the obtained communities, we visualized the semantic matrix of communities in the form of a heat map. Fig. 5 shows the distribution of the 334 features in the DBLP dataset across the four different communities and the semantic preferences of the communities. The closer the color is to yellow, the stronger the community's semantic features.

According to Fig. 5, the semantic characteristics of each community and its representative features can be identified. For example, the representative feature of community 1 is feature 165, feature 64 can be considered as an auxiliary feature, and the heat values of other features are lower, which proves that the semantics of this community is more concentrated. Community 2 has four representative features, which proves that this community is active and semantically rich. Community 3 and Community 4 are similar in that the representative semantics are not prominent, but there are multiple features with a heat value greater than 0.175, which shows that the community has complex semantics. In real-world scenarios, these semantic features can be used to identify community types and make further judgments. For example, semantically focused communities tend to form opinion leaders, which facilitates community management and information control. Communities with complex semantics are less likely to form a unified opinion, and community members are loose, which is not conducive to community stability. In short, the community obtained from the HAESF model has high interpretability, which is critical for practical applications.

5.4. Ablation experiments

To verify the validity of the individual components, we conducted ablation experiments on DBLP and visualized the clustering results, as shown in Fig. 6.

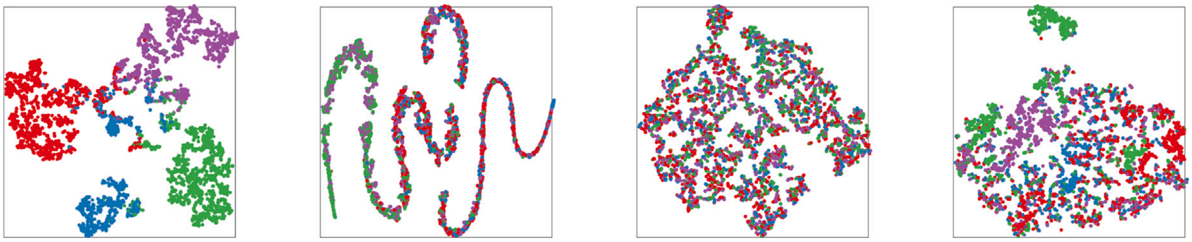
Fig. 6(a) shows the clustering result obtained by HAESF. It can be seen that the nodes are roughly divided into 4 separate communities. Most of the nodes are correctly divided, and a small portion of the nodes of the mixed communities in the middle are entangled with each other. The overall division is fairly clear. Fig. 6(b) removes the SF module which decomposes the semantics of the communities. The clustering result relying on the HAE alone drops significantly, with nodes from the purple and green communities mixed, and nodes from the blue and red communities mixed, which demonstrates that the SF is valid and that community semantics can be trusted. It is clear from Fig. 6(c) that, without HAE, community detection is solely based on SF and node attributes. There is no tendency for clustering, demonstrating the importance of heterogeneous structure in network learning. It also demonstrates that an independent SF is insufficient to handle semantic-rich heterogeneous networks. Fig. 6(d) removes the node attributes and relies only on the topology for learning. We can observe that only a very small fraction of the nodes cluster into groups, and most remain discrete, indicating the importance of node attributes for heterogeneous networks. In a homogeneous network, good clustering can be achieved solely by considering its topology, whereas in a heterogeneous network, attribute information of nodes is equally important as its topology.

5.5. Parameter sensitivity analysis

To assess the robustness of the model, we performed sensitivity tests on the DBLP for the parameters involved in the loss function. Extensive experiments have shown that a combination of parameters $(50, 0.25)$ achieves optimal community detection on this dataset. For comparison purposes, we fixed μ to 0.25 when testing parameter λ, and similarly fixed λ to 50 when testing parameter μ.

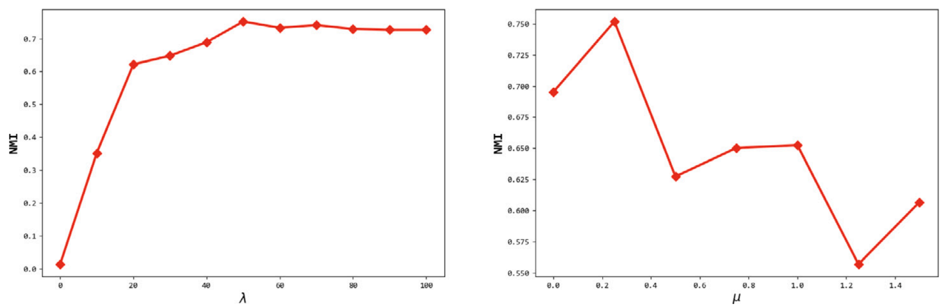
The results of parameter perception for λ are shown in Fig. 7(a). It is easy to find that when $\lambda \geq 20$ the NMI score always stays within a relatively excellent and stable interval. When $\lambda = 50$, the HAESF achieves the most optimal NMI value. Therefore, we believe that the model is not sensitive to the parameter λ. As long as the value of λ is within a reasonable range, the model can obtain a relatively effective clustering performance. The experimental results of the parameter μ are shown in Fig. 7(b). When $\mu = 0.25$, the NMI score far exceeds the other values. The NMI score fluctuates significantly when μ is changed, but is still within the interval $[0.55, 0.75]$, which is still an acceptable and excellent score when compared with the performance of other algorithms.

Overall, we can achieve better clustering results by tuning the model parameters, but as long as the parameter values are within a reasonable range, our model will be able to achieve satisfactory clustering results. In short, our model is not sensitive to these two loss parameters, which demonstrates its strong robustness.



(a) HAESF (b) HAE (c) SF (d) Topology

Fig. 6. DBLP-based visualization of ablation results. Each dot represents an author and the different colors represent the community they belong to.



(a) NMI score of the parameter λ (b) NMI score of the parameter μ

Fig. 7. DBLP-based parameter sensitivity testing.

5.6. Model convergence analysis

We tested the convergence of the model on the DBLP to verify the learning capability of the model, and the results are shown in Fig. 8.

Fig. 8(a) illustrates the convergence of the model under the NMI metric. DAEGC converges first and its NMI value remains high, primarily due to an additional pre-training phase in DAEGC. The VGAE reaches convergence very quickly but scores poorly. Although there is a peak in VGAER, the final level is comparable to that of VGAE. This consistency can be explained by the variational idea employed by both models to learn the distribution of the data, however, this idea does not work well on DBLP. In contrast, GAER achieves better results, demonstrating that modularity information can be meaningful for detecting communities. However, the model is prone to oscillations during training and is less stable. The HAN converged rapidly after 60 epochs, indicating that the heterogeneous attention network can learn heterogeneous structures quickly, but with limited accuracy. Due to the addition of the unstable SF module, the HAESF model converged slightly slower than the other models, converging after roughly 200 epochs, however, it scored significantly higher than the other models.

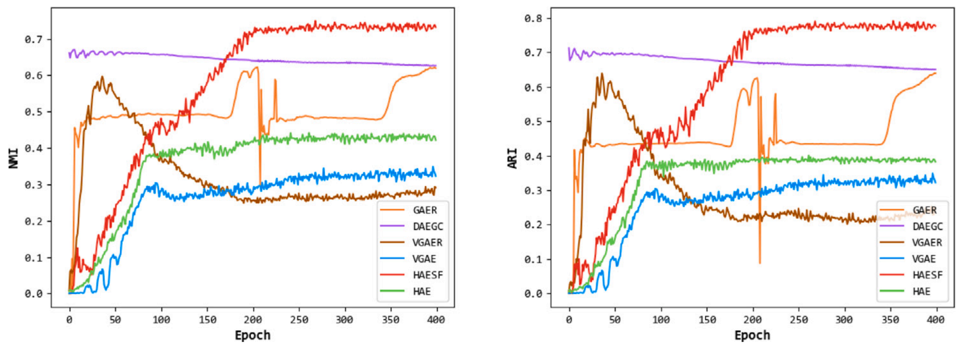
Fig. 8(b) shows the ARI scores of each model with epoch iteration, which are highly consistent with Fig. 8(a). The results obtained with different metrics show that HAESF has significant convergence and stability. It can produce the most accurate clustering results while maintaining fast and stable convergence. The consistency in the performance of different metrics allows us to believe with a relatively high degree of confidence that the method proposed in this paper has significant stability and validity.

6. Conclusion and future work

This paper investigates unsupervised community detection methods based on heterogeneous networks and proposes a new framework, the HAESF model. This model performs sufficient learning related to heterogeneous network characteristics, including effective processing of heterogeneous nodes and full use of semantic information. which fundamentally solves the recurrent problem of existing methods (e.g., struggling to achieve effective community detection for heterogeneous networks). Firstly, the HAE module is mainly used to learn heterogeneous structures and implement unsupervised methods where we innovatively propose the concept of HICG and combine it with the two-layer attention mechanism to achieve unsupervised community detection. Secondly, the SF focuses on the learning and fusion of semantic information. We assumed that the community semantic is homogeneous and they are obtained by double-layer NMF that are used to constrain the partitioning of nodes. Finally, we propose a new selective segmentation optimization to combine both modules for efficient joint training, and we define the SF loss as a judgment loss and use it to make reliability judgments. The proposed approach improves the accuracy of community detection. Compared to the existing unsupervised methods, HAESF shows superior and stable performance on different datasets. However, the method does not have the most accurate performance on all datasets, such as IMDB. The data on IMDB has obvious characteristics such as feature dispersion and sparse structure. This makes the method proposed in this paper unable to learn the nodes efficiently, and thus difficult to perform accurate community detection. This is a limitation of the method. Overall, HAESF's unsupervised nature makes it more practical and generalizable than existing heterogeneous methods. It can be easily applied to any community task in heterogeneous networks without prior knowledge.

Concerning the future work, we will continue to refine the HAESF and the unsupervised framework, which can be accomplished in the following aspects:

(1) In terms of learning network features, the HAESF's ability to learn sparse networks should be improved. As previously stated, HAESF struggles to perform effective community detection for networks with sparse features and structure (e.g., IMDB); therefore, the introduction of data augmentation can be considered to compensate for the shortcoming;
(2) A more generalized variant of the NMF should be proposed in the SF module. To explain the design concept and innovation significance in a simplified way, the basic double-layer NMF is chosen in this paper, and opting for this choice ensures both



(a) NMI score (b) ARI score

Fig. 8. DBLP-based model convergence testing.

stability and effective decomposition, but the application scenarios are limited. That is why a more general variant should be proposed. It should satisfy as many practical needs as possible. For example, if the network features are too complex and shallow decomposition cannot accurately capture their relationships, multi-layer NMF will be used to extract the network features through layer-by-layer abstractions. Or, depending on the network characteristics, different variants can be chosen (Hedjam, Abdesselam, & Melgani, 2021; Wu, Kwong, Zhou, Jia, & Gao, 2018), such as SNMF, ONMF, or NMF with imposed feature constraints, where we impose feature or structural constraints on the NMF to steer the direction of community segmentation;

(3) Regarding the model framework, pre-processing modules can be combined into a unified joint optimization framework. HAESF treats the preprocessing and formal training parts independently, which may lead to performance loss when dealing with unknown networks. Therefore, in the future, we will consider unifying these two parts and dynamically adjusting the weight of each semantic graph in the HICG during the training process, so the network structure can be preserved in a better way. In addition, the unnecessary inherent losses in the unsupervised process can be reduced. This unified framework not only ensures training continuity but also improves the algorithm's efficiency.

CRediT authorship contribution statement

Yan Zhao: Conceptualization, Methodology, Data curation, Visualization, Writing – original draft. **Weimin Li:** Conceptualization, Resources, Supervision, Writing – review & editing. **Fangfang Liu:** Supervision, Formal analysis. **Jingchao Wang:** Validation, Writing – review & editing. **Alex Munyole Luvembe:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2022YFC3302600).

References

Bhowmick, S. S., & Seah, B. S. (2015). Clustering and summarizing protein-protein interaction networks: A survey. *IEEE Transactions on Knowledge and Data Engineering*, *28*(3), 638–658.

Chen, Z., Li, X., & Bruna, J. (2017). Supervised community detection with line graph neural networks. arXiv preprint arXiv:1705.08415.

Gligorijević, V., Panagakis, Y., & Zafeiriou, S. (2018). Non-negative matrix factorizations for multiplex network analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(4), 928–940.

Guo, K., Zhao, Z., Yu, Z., Guo, W., Lin, R., Tang, Y., et al. (2022). Network embedding based on biased random walk for community detection in attributed networks. *IEEE Transactions on Computational Social Systems*.

Han, Z., Jin, X., Xing, H., Yang, W., & Xiong, H. (2023). An effective heterogeneous information network representation learning framework. *Future Generation Computer Systems*.

He, D., Song, Y., Jin, D., Feng, Z., Zhang, B., Yu, Z., et al. (2021). Community-centric graph convolutional network for unsupervised community detection. In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence* (pp. 3515–3521).

He, C., Zheng, Y., Fei, X., Li, H., Hu, Z., & Tang, Y. (2021). Boosting nonnegative matrix factorization based community detection with graph attention auto-encoder. *IEEE Transactions on Big Data*, *8*(4), 968–981.

Hedjam, R., Abdesselam, A., & Melgani, F. (2021). NMF with feature relationship preservation penalty term for clustering problems. *Pattern Recognition*, *112*, Article 107814.

Huang, Z., Zhong, X., Wang, Q., Gong, M., & Ma, X. (2020). Detecting community in attributed networks by dynamically exploring node attributes and topological structure. *Knowledge-Based Systems*, *196*, Article 105760.

Jin, D., Liu, Z., Li, W., He, D., & Zhang, W. (2019). Graph convolutional networks meet markov random fields: Semi-supervised community detection in attribute networks. In *Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01* (pp. 152–159).

Jin, D., Yu, Z., Jiao, P., Pan, S., He, D., Wu, J., et al. (2021). A survey of community detection approaches: From statistical modeling to deep learning. *IEEE Transactions on Knowledge and Data Engineering*.

Kasihmuddin, M. S. M., Jamaludin, S. Z. M., Mansor, M. A., Wahab, H. A., & Ghadzi, S. M. S. (2022). Supervised learning perspective in logic mining. *Mathematics*, *10*(6), 915.

Khan, B. S., & Niazi, M. A. (2017). Network community detection: A review and visual survey. arXiv preprint arXiv:1708.00977.

Kipf, T. N., & Welling, M. (2016). Variational graph auto-encoders. arXiv preprint arXiv:1611.07308.

Li, Y., He, K., Kloster, K., Bindel, D., & Hopcroft, J. (2018). Local spectral clustering for overlapping community detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *12*(2), 1–27.

Li, S., Jiang, L., Wu, X., Han, W., Zhao, D., & Wang, Z. (2021). A weighted network community detection algorithm based on deep learning. *Applied Mathematics and Computation*, *401*, Article 126012.

Li, W., Ni, L., Wang, J., & Wang, C. (2022). Collaborative representation learning for nodes and relations via heterogeneous graph neural network. *Knowledge-Based Systems*, *255*, Article 109673.

Li, C., Tang, Y., Tang, Z., Cao, J., & Zhang, Y. (2022). Motif-based embedding label propagation algorithm for community detection. *International Journal of Intelligent Systems*, *37*(3), 1880–1902.

Li, W., Zhang, C., Zhou, X., & Jin, Q. (2023). Dynamic Multi-view Group Preference Learning for group behavior prediction in social networks. *Expert Systems with Applications*, [ISSN: 0957-4174] Article 120553.

Li, W., Zhu, H., Li, S., Wang, H., Dai, H., Wang, C., et al. (2021). Evolutionary community discovery in dynamic social networks via resistance distance. *Expert Systems with Applications*, *171*, Article 114536.

Liu, X., Wang, W., He, D., Jiao, P., Jin, D., & Cannistraci, C. V. (2017). Semi-supervised community detection based on non-negative matrix factorization with node popularity. *Information Sciences*, *381*, 304–321.

Luo, L., Fang, Y., Cao, X., Zhang, X., & Zhang, W. (2021). Detecting communities from heterogeneous graphs: A context path-based graph neural network model. In *Proceedings of the 30th ACM international conference on information & knowledge management* (pp. 1170–1180).

Luo, X., Liu, Z., Jin, L., Zhou, Y., & Zhou, M. (2021). Symmetric nonnegative matrix factorization-based community detection models and their convergence analysis. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(3), 1203–1215.

Luvembe, A. M., Li, W., Li, S., Liu, F., & Xu, G. (2023). Dual emotion based fake news detection: A deep attention-weight update approach. *Information Processing & Management*, *60*(4), Article 103354.

Lyu, C., Shi, Y., Sun, L., & Lin, C.-T. (2022). Community detection in multiplex networks based on evolutionary multi-task optimization and evolutionary clustering ensemble. *IEEE Transactions on Evolutionary Computation*.

Naderipour, M., Fazel Zarandi, M. H., & Bastani, S. (2022). Fuzzy community detection on the basis of similarities in structural/attribute in large-scale social networks. *Artificial Intelligence Review*, 1–35.

Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, *103*(23), 8577–8582.

Perianes-Rodríguez, A., Olmeda-Gómez, C., & Moya-Anegón, F. (2010). Detecting, identifying and visualizing research groups in co-authorship networks. *Scientometrics*, *82*(2), 307–319.

Pompili, F., Gillis, N., Absil, P.-A., & Glineur, F. (2014). Two algorithms for orthogonal nonnegative matrix factorization with application to clustering. *Neurocomputing*, *141*, 15–25.

Qin, M., Jin, D., He, D., Gabrys, B., & Musial, K. (2017). Adaptive community detection incorporating topology and content in social networks. In *Proceedings of the 2017 IEEE/ACM international conference on advances in social networks analysis and mining 2017* (pp. 675–682).

Qiu, C., Huang, Z., Xu, W., & Li, H. (2022). VGAER: Graph neural network reconstruction based community detection. arXiv preprint arXiv:2201.04066.

Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, *76*(3), Article 036106.

Shang, R., Zhang, W., Zhang, J., Feng, J., & Jiao, L. (2022). Local community detection based on higher-order structure and edge information. *Physica A. Statistical Mechanics and its Applications*, *587*, Article 126513.

Su, X., Xue, S., Liu, F., Wu, J., Yang, J., Zhou, C., et al. (2022). A comprehensive survey on community detection with deep learning. *IEEE Transactions on Neural Networks and Learning Systems*.

Sun, Y., Sun, X., Liu, Z., Cao, Y., & Yang, J. (2023). Core node knowledge based multi-objective particle swarm optimization for dynamic community detection. *Computers & Industrial Engineering*, *175*, Article 108843.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). Graph attention networks. arXiv preprint arXiv:1710.10903.

Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., et al. (2019). Heterogeneous graph attention network. In *The world wide web conference* (pp. 2022–2032).

Wang, C., Pan, S., Hu, R., Long, G., Jiang, J., & Zhang, C. (2019). Attributed graph clustering: A deep attentional embedding approach. arXiv preprint arXiv:1906.06532.

Wu, W., Kwong, S., Zhou, Y., Jia, Y., & Gao, W. (2018). Nonnegative matrix factorization with mixed hypergraph regularization for community detection. *Information Sciences*, *435*, 263–281.

Xu, Y., Zhuang, Z., Li, W., & Zhou, X. (2018). Effective community division based on improved spectral clustering. *Neurocomputing*, *279*, 54–62.

Ye, F., Chen, C., & Zheng, Z. (2018). Deep autoencoder-like nonnegative matrix factorization for community detection. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 1393–1402).

Yuan, S., Zeng, H., Zuo, Z., & Wang, C. (2023). Overlapping community detection on complex networks with graph convolutional networks. *Computer Communications*, *199*, 62–71.

Zamri, N. E., Azhar, S. A., Mansor, M. A., Alway, A., & Kasihmuddin, M. S. M. (2022). Weighted random k satisfiability for k=1, 2 (r2SAT) in discrete Hopfield neural network. *Applied Soft Computing*, *126*, Article 109312.

Zhang, C., Li, W., Wei, D., Liu, Y., & Li, Z. (2022). Network dynamic GCN influence maximization algorithm with leader fake labeling mechanism. *IEEE Transactions on Computational Social Systems*.

Zhao, Z., Ke, Z., Gou, Z., Guo, H., Jiang, K., & Zhang, R. (2022). The trade-off between topology and content in community detection: An adaptive encoder–decoder-based NMF approach. *Expert Systems with Applications*, *209*, Article 118230.