Chapter 1
Introduction

Abstract Networks (or graphs) are ubiquitous in the real-world, such as social networks, academic networks, biological networks and so on. Heterogeneous information network (HIN), a.k.a., heterogeneous graph (HG), is an important type of network, which contains multiple types of nodes and edges. To date, the research of HG has attracted extensive attentions, the most important of which is the heterogeneous graph representation (HGR), a.k.a., heterogeneous network embedding. In this chapter, we first introduce some basic concepts and definitions in HG and emphasize the importance of graph representation learning in the field of data mining. Then, we analyse the unique challenges of HGR compared with homogeneous network. In the end, we briefly introduce the organization of this book.

1.1 Basic Concepts and Definitions

Before introducing HGR, we first give some basic definitions in HG. The first is information network, which is a template of the real-world networks. Specially, both homogeneous network and heterogeneous network can be seen as special cases of information network. We formally define them as follows:

Definition 1. Information Network [15]. An information network is defined as a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, in which $\mathcal{V}$ and $\mathcal{E}$ represent the node set and the link set, respectively. Each node $v \in \mathcal{V}$ and link $e \in \mathcal{E}$ is associated with their mapping functions $\phi(v) : \mathcal{V} \to \mathcal{A}$ and $\varphi(e) : \mathcal{E} \to \mathcal{R}$, where $\mathcal{A}$ and $\mathcal{R}$ denote the node types and link types, respectively. **Homogeneous Network (or Homogeneous Graph)** is an instance of the information network, with $|\mathcal{A}|$ = $|\mathcal{R}|$ = 1. **Heterogeneous Network (or Heterogeneous Graph)** requires $|\mathcal{A}| + |\mathcal{R}| > 2$, i.e., it contains different types of nodes and links.

Compared with homogeneous graph, heterogeneous graph has stronger expressive power, but also more complex. An example of heterogeneous academic graph is illustrated in Fig. 1.1a, which consists of four node types (Author, Paper, Venue,

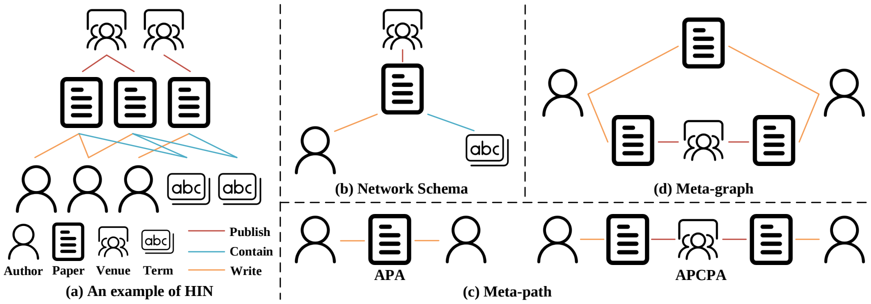

Fig. 1.1 A heterogeneous academic graph, including (a) four types of nodes (i.e. Author, Paper, Venue, Term) and three types of link (i.e., Publish, Contain, Write), (b) network schema, (c) meta-paths (i.e. Author-Paper-Author and Paper-Term-Paper), and (d) meta-graph.

and Term) and three link types (Author-Write-Paper, Paper-Contain-Term, and Conference-Publish-Paper). In the next, we will introduce some unique definitions in HG, including network schema (Fig. 1.1b), meta-paths (Fig. 1.1c) and meta-graph (Fig. 1.1d). Finally, we will give the definition of graph representation learning.

Since an HG contains multiple node types and link types, to understand the whole structure of it, it is necessary to provide a meta-level (or schema-level) description of the graph. Therefore, as the blueprint of HG, network schema is proposed to give an abstraction of the graph:

Definition 2. Network Schema of $\mathcal{G}$ is a directed graph $\mathcal{S} = (\mathcal{A}, \mathcal{R})$, which can be seen as a meta template of an HG with the node type mapping function $\phi(v) : \mathcal{V} \rightarrow \mathcal{A}$ and the link type mapping function $\varphi(e) : \mathcal{E} \rightarrow \mathcal{R}$. Fig. 1.1b illustrates the network schema of the academic graph.

Network schema describes the associations between different types of nodes. Based on it, we can further mine the higher-level semantics of the data. Therefore, meta-path [16] is further proposed to capture the higher-order relationships, i.e., semantics, between nodes. The definition of meta-path is given below:

Definition 3. Meta-path [16]. A meta-path m is based on a network schema $\mathcal{S}$, which is denoted as $m = A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$ (simplified to $A_1 A_2 \cdots A_{l+1}$) with node types $A_1, A_2, \cdots, A_{l+1} \in \mathcal{A}$ and link types $R_1, R_2, \cdots R_l \in \mathcal{R}$.

Different meta-paths capture the semantic relationships from different views. For example, the meta-path of "APA" indicates the co-author relationship and "APCPA" represents the co-conference relation. Both of them can be used to formulate the proximity over authors. Although meta-path can be used to depict the connections over nodes, it fails to capture a more complex relationship, such as motifs [10]. To address this challenge, meta-graph [6] is proposed to use a directed acyclic graph of node and link types to capture more complex relationship between two HG nodes.

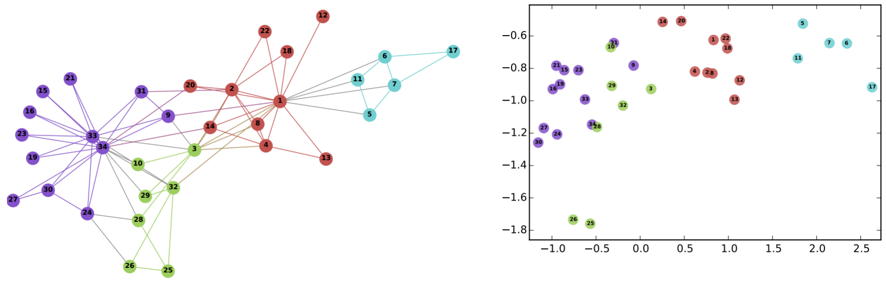

Fig. 1.2 A toy example of graph representation. **Left**: the input Karate graph. **Right**: the output node representations. Image is extracted from DeepWalk [11].

Definition 4. Meta-graph [6]. A meta-graph $\mathcal{T}$ can be seen as a directed acyclic graph (DAG) composed of multiple meta-paths with common nodes. Formally, meta-graph is defined as $\mathcal{T} = (V_{\mathcal{T}}, E_{\mathcal{T}})$, where $V_{\mathcal{T}}$ is a set of nodes and $E_{\mathcal{T}}$ is a set of links. For any node $v \in V_{\mathcal{T}}, \phi(v) \in \mathcal{A}$; for any link $e \in E_{\mathcal{T}}, \varphi(e) \in \mathcal{R}$.

An example of meta-graph is shown in Fig. 1.1d, which can be regarded as the combination of meta-path “APA” and “APCPA”, reflecting a high-order similarity of two nodes. Note that a meta-graph can be symmetric or asymmetric [28].

The research of graph is always an important topic in machine learning. However, due to the non-Euclidean property, traditional heuristic methods suffer from high computational cost and low parallelizability [2], which cannot be used for real applications. Therefore, a critical challenge for this field is to find effective data representation. Through graph representation learning, the nodes are projected into vectors and can be incorporated with the advanced machine learning technologies and tasks. We formalize the problem of graph representation learning as follow.

Definition 5. Graph Representation Learning [2], also known as network embedding, aims to learn a function $\Phi : \mathcal{V} \rightarrow \mathbb{R}^d$ that embeds the nodes $v \in \mathcal{V}$ in a graph into a low-dimensional Euclidean space where $d \ll |\mathcal{V}|$.

A toy example of graph representation learning is shown in Fig. 1.2. Through graph representation learning, the complex network in the non-Euclidean space is projected into a low-dimensional Euclidean space. Therefore, the high computational cost and low parallelizability issues are well solved. In the next, we will give a brief review of the recent development of graph representation learning.

1.2 Graph Representation Learning

In the aforementioned chapter, we refer that the analysis of graphs suffers from the high computational complexity and low parallelizability issues. To deal with these problems, graph representation learning is proposed and rapidly becomes the major tool in network analysis [2, 23].

Previous graph representation learning methods focus on preserving the structural information of the graph. For example, Deepwalk [11] uses random walk to generate node sequences and then employs the skip-gram model to lean the co-occurrence of nodes within a window, thus capturing the local structures. LINE [17] preserves both the first- and second-order structure similarities, node2vec [5] extends Deepwalk to global structures with a Depth-first Sampling (DFS) and Breadth-first Sampling (BFS), M-NMF [19] learns the community structures and AROPE [29] preserves arbitrary order proximity through Singular Value Decomposition (SVD). In particular, Qiu et al. [12] proves that most existing graph representation learning methods can be unified into a matrix factorization framework.

Further, some methods begin to incorporate the rich node/edge attributes in node representation. TADW [24] jointly factorizes the adjacency matrix and text matrix to fuse the structural and attribute information. DANE [4] enforces the structural representations and attribute representations to be consistent, so that the learned representations can capture these two kinds of information at the same time. ANRL [30] designs a neighbor enhancement autoencoder. It aims to reconstruct the target neighbors and attributes to model the structural and attribute information.

With the development of deep learning, the emerging graph neural networks (GNNs) show powerful capability in combining the network structures and node attributes. Graph convolutional networks (GCNs) [7] is one of the most representative work, which designs a convolutional operator in spatial domain to filter the node attributes by network structures. Graph Attention Networks (GAT) [18] uses self-attention to learn the important of nodes in fusing the attributes of neighbors. Klicpera et al. propose Predict then Propagate (PPNP) [8], which incorporates personalized Pagerank into GNNs and alleviates the over-smoothing problem. SGC [22] simplifies the design of GCN through decoupling the transformation step and aggregation step, which not only reduces the parameters of GNNs but also accelerates the training process.

In addition to the structural and attribute information, recently, some researchers tend to explore the semantic information from the multiple node/edge types in graphs [14], leading to the research of HGR.

1.3 Heterogeneous Graph Representation Learning and Challenges

Different from homogeneous graph representation learning that mainly needs to preserve the structural information, heterogeneous graph representation learning aims to preserver the structural and semantic information simultaneously. However, due to the heterogeneity of HG, HGR imposes more challenges to this problem, which are illustrated below.

- **Complex structure** (the complex HG structure caused by multiple types of nodes and edges). In a homogeneous graph, the fundamental structure can be considered as the so-called first-order, second-order, and even higher-order structure [11, 17, 19]. All these structures are well defined and have good intuition. However, the structure in HG will dramatically change depending on the selected relations. Let's still take the bibliographic network in Fig. 1.1a as an example, the neighbors of one paper will be authors with the "write" relation, while with "contain" relation, the neighbors become terms. Complicating things further, the combination of these relations, which can be considered as a higher-order structure in HG, will result in different and more complicated structures. How to efficiently and effectively preserve these complex structures is thereby an urgent need but it is still a significant challenge in HGR, and current efforts have been made towards the meta-path structure [3] and meta-graph structure [27], etc.

- **Heterogeneous attributes** (the fusion problem caused by the heterogeneity of attributes). Since the nodes and edges in a homogeneous graph have the same type, each dimension of the node or edge attributes has the same meaning. In this situation, node can directly fuse the attributes of its neighbors. However, in heterogeneous graph, the attributes of different types of nodes and edges may have different meanings [26, 20]. For example, the attributes of author can be the research fields, while paper may use keywords as attributes. Therefore, how to overcome the heterogeneity of attributes and effectively fuse the attributes of neighbors is an important challenge in HGR.
- **Application dependent** (the domain knowledge hidden in HG structures and attributes). HG is closely related to the real-world applications, while many practical problems remain unsolved. For example, constructing an appropriate HG may require sufficient domain knowledge in a real-world application. Also, meta-path and/or meta-graph are widely used to capture the structure of HG. However, unlike homogeneous graph, where the structure, e.g., the first-order and second-order structure is well defined, meta-path selection may also need prior knowledge. Furthermore, to better facilitate the real-world applications, we usually need to elaborately encode the side information, e.g., node attributes [20, 26, 21, 25] or more advanced domain knowledge [13, 1, 9] to the HGR process.

1.4 Organization of the Book

This book is written to comprehensively review the development of HGR and introduce the state-of-the-art methods. We first summarize existing works from two perspectives: method and technique, and introduce some open sources of this field. Then we introduce the state-of-the-arts models of each category in detail. Part one focuses on the four main HGR models. Part two introduces the development of HGR on real-world industrial scene. Finally, we discuss the future research direction of HGR and summarize the content of this book.

The remainder of this book is organized as follows. In Chapter 2, we summarize the developments of HGR, including taxonomy, technique and open sources. In

Chapter 3-6, we categorize existing HGR methods into four categories, including structured-preserved HGR, attribute-assisted HGR, dynamic HGR and some other emerging topics. In each chapter, we will introduce their unique challenges and designs in detail. In Chapter 7-9, we further explore the transformativeness of existing HGR methods that have been successfully deployed in real-world applications, e.g., recommendation, text mining, cash-out user detection, etc. In Chapter 10, we forecast the future research directions in this field.

References

1. Chen, T., Sun, Y.: Task-guided and path-augmented heterogeneous network embedding for author identification. In: WSDM, pp. 295–304. ACM (2017)
2. Cui, P., Wang, X., Pei, J., Zhu, W.: A survey on network embedding. IEEE Transactions on Knowledge and Data Engineering **31**(5), 833–852 (2018)
3. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: KDD, pp. 135–144. ACM (2017)
4. Gao, H., Huang, H.: Deep attributed network embedding. In: IJCAI, pp. 3364–3370. ijcai.org (2018)
5. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: KDD, pp. 855–864. ACM (2016)
6. Huang, Z., Zheng, Y., Cheng, R., Sun, Y., Mamoulis, N., Li, X.: Meta structure: Computing relevance in large heterogeneous information networks. In: KDD, pp. 1595–1604. ACM (2016)
7. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (2017)
8. Klicpera, J., Bojchevski, A., Günnemann, S.: Predict then propagate: Graph neural networks meet personalized pagerank. In: ICLR (Poster). OpenReview.net (2019)
9. Liu, Z., Zheng, V.W., Zhao, Z., Li, Z., Yang, H., Wu, M., Ying, J.: Interactive paths embedding for semantic proximity search on heterogeneous graphs. In: KDD, pp. 1860–1869. ACM (2018)
10. Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., Alon, U.: Network motifs: simple building blocks of complex networks. Science **298**(5594), 824–827 (2002)
11. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. In: KDD, pp. 701–710 (2014)
12. Qiu, J., Dong, Y., Ma, H., Li, J., Wang, K., Tang, J.: Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec. In: WSDM, pp. 459–467. ACM (2018)
13. Shi, C., Hu, B., Zhao, W.X., Yu, P.S.: Heterogeneous information network embedding for recommendation. IEEE Transactions on Knowledge and Data Engineering **31**(2), 357–370 (2018)
14. Shi, C., Li, Y., Zhang, J., Sun, Y., Yu, P.S.: A survey of heterogeneous information network analysis. IEEE Trans. Knowl. Data Eng. **29**(1), 17–37 (2017)
15. Sun, Y., Han, J.: Mining heterogeneous information networks: a structural analysis approach. SIGKDD Explorations **14**(2), 20–28 (2012)
16. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. Proceedings of the VLDB Endowment **4**(11), 992–1003 (2011)
17. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: WWW, pp. 1067–1077 (2015)
18. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. ICLR (2018)

19. Wang, X., Cui, P., Wang, J., Pei, J., Zhu, W., Yang, S.: Community preserving network embedding. In: AAAI (2017)
20. Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., Yu, P.S.: Heterogeneous graph attention network. In: WWW, pp. 2022–2032. ACM (2019)
21. Wang, X., Lu, Y., Shi, C., Wang, R., Cui, P., Mou, S.: Dynamic heterogeneous information network embedding with meta-path based proximity. IEEE Transactions on Knowledge and Data Engineering (2020)
22. Wu, F., Jr., A.H.S., Zhang, T., Fifty, C., Yu, T., Weinberger, K.Q.: Simplifying graph convolutional networks. In: ICML, *Proceedings of Machine Learning Research*, vol. 97, pp. 6861–6871. PMLR (2019)
23. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S.: A comprehensive survey on graph neural networks. IEEE Trans. Neural Networks Learn. Syst. **32**(1), 4–24 (2021)
24. Yang, C., Liu, Z., Zhao, D., Sun, M., Chang, E.Y.: Network representation learning with rich text information. In: IJCAI, pp. 2111–2117. AAAI Press (2015)
25. Yang, L., Xiao, Z., Jiang, W., Wei, Y., Hu, Y., Wang, H.: Dynamic heterogeneous graph embedding using hierarchical attentions. In: ECIR, *Lecture Notes in Computer Science*, vol. 12036, pp. 425–432. Springer (2020)
26. Zhang, C., Song, D., Huang, C., Swami, A., Chawla, N.V.: Heterogeneous graph neural network. In: KDD, pp. 793–803. ACM (2019)
27. Zhang, D., Yin, J., Zhu, X., Zhang, C.: Metagraph2vec: complex semantic path augmented heterogeneous network embedding. In: PAKDD, pp. 196–208. Springer (2018)
28. Zhang, W., Fang, Y., Liu, Z., Wu, M., Zhang, X.: mg2vec: Learning relationship-preserving heterogeneous graph representations via metagraph embedding. IEEE Transactions on Knowledge and Data Engineering (2020)
29. Zhang, Z., Cui, P., Wang, X., Pei, J., Yao, X., Zhu, W.: Arbitrary-order proximity preserved network embedding. In: KDD, pp. 2778–2786. ACM (2018)
30. Zhang, Z., Yang, H., Bu, J., Zhou, S., Yu, P., Zhang, J., Ester, M., Wang, C.: ANRL: attributed network representation learning via deep neural networks. In: IJCAI, pp. 3155–3161. ijcai.org (2018)