Meta-Path Generation Online for Heterogeneous Network Embedding

Tao Liang*†, Jin Liu*†
*Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
†School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{liangtao0305, liujin}@iie.ac.cn

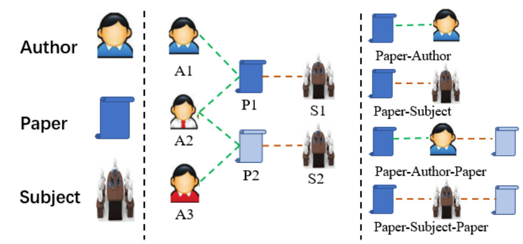
***Abstract*—Graph neural networks (GNNs), powerful deep representation learning methods for graph data, have been widely used in various tasks, such as recommendation systems and link prediction. Most existing GNNs are designed to learn node embeddings on homogeneous graphs. Heterogeneous information network (HIN) with various types of nodes and edges still faces great challenges for the heterogeneity and rich semantic information. To make full use of the heterogeneous information, many works try to manually design meta-paths, which are paths connected with two objects. They utilize meta-paths to capture more semantic information in heterogeneous graphs. However, manually designed meta-paths require domain knowledge and meta-path-based heterogeneous graph embedding methods only utilize the information of nodes with the same type, ignoring the impacts of the different types of nodes. We propose meta-path generation online for heterogeneous network embedding for all types of nodes, which can generate meta-paths and learn node embeddings simultaneously. Firstly, we exhaust all meta-paths within k-hop for specific nodes and apply a meta-path guided nodes aggregation. Secondly, we adopt an attention mechanism to select Top-N meta-paths with the largest attention coefficients for the semantic aggregation. The above two stages constitute one layer of our approach. Through stacking multi-layers, we can generate longer and more complex meta-paths. Without domain-specific preprocessing, extensive experiments on two datasets demonstrate that our proposed approach achieves better performance compared with other recent methods that require predefined meta-paths from domain knowledge.**

***Index Terms*—Meta-path, Heterogeneous network embedding, Graph neural network**

I. Introduction

Graph neural networks (GNNs), powerful deep representation learning methods for graph data, have been widely used in various tasks, such as recommendation systems [1], [2], link prediction [3]–[6] and similarity search [7], [8]. GNNs perform convolution in the spectral domain using the Fourier basis of a given graph [9], [10], or perform convolution directly on the graph using the basic graph structure by pass node features to neighbors [11], [12].

However, most existing GNNs are designed to learn node embeddings on homogeneous graphs. As a matter of fact, graphs in the real world usually have multiple types of nodes and edges, known as heterogeneous information network (HIN) [13]. As shown in Figure 1, a citation network has multiple types of nodes (e.g. author, paper, subject) and edges defined by their relations(e.g.Author-Paper, Paper-Subject). A naive approach is to ignore the node/edge types and treat them



(a) Node type (b) Graph schema (c) Meta-paths within 2-hop

Fig. 1. An illustrative example of a heterogeneous information network(ACM). Figure 1(a) Three types of nodes(i.e., author, paper, subject). Figure 1(b) A graph schema consists three types of nodes and two types of connections. Figure 1(c) all existing meta-paths within 2-hop neighborhoods for paper-type node.

as in a homogeneous graph. This is obviously a suboptimal solution since the model cannot leverage the heterogeneous information. To make full use of heterogeneous information, [14] proposes to manually design meta-paths [15], a composite relation connected with two objects, and decompose a heterogeneous graph into several homogeneous graphs defined by the meta-paths. Then GNNs can operate on the decomposed homogeneous graphs. This is a two-stage approach, which requires manually predefined meta-paths for each task. These meta-paths only utilize the information of nodes with the same type, ignoring the impacts of other different types of nodes. Moreover, downstream tasks will be severely affected by the choice of meta-paths.

Our approach can generate meaningful meta-paths for different nodes online and learn node embeddings on a heterogeneous graph in an end-to-end fashion. Specifically, our approach follows a two-stage structure: meta-path generation online and Top-N meta-paths selection, as shown in Figure 3.

Meta-path Generation Online. We exhaust all meta-paths within k-hop for a particular node in the first stage.s Taking the citation network ACM shown in Figure 1(c) as an example, there are 4 meta-paths within paper's 2-hop neighborhoods Paper-Author(PA), Paper-Subject(PS), Paper-Author-Paper(PAP), Paper-Subject-Paper(PSP). In addition, there is always one meta-path representing self-loop: Paper(P). Therefore, there are total 5 meta-paths within 2-hop neighborhoods. For each meta-path guided neighbors, we independently per-

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

form a node information aggregation, as shown in Figure 2. Then we obtain 5 meta-path-based embeddings representing these meta-paths.

Top-N Meta-paths Selection. Still taking ACM shown in Figure 1(c) as an example, paper classification in citation networks may benefit from meta-paths which are Paper-Author-Paper(PAP) or Paper-Author(PA). Therefore, it is impractical to treat different meta-paths equally, and it will weaken the semantic information provided by some meaningful meta-paths. For all meta-path-based embeddings, we adopt an attention mechanism to estimate the impact of semantics attached to each meta-path on the central node, as shown in Figure 2. Then we select Top-N meta-paths with the largest attention coefficients for the semantic aggregation.

In this paper, we propose a novel **M**eta-Path **G**eneration **O**nline for **H**eterogeneous Network **E**mbedding, named MGOHE. Our framework can generate meaningful meta-paths for different nodes online and learn effective node embeddings simultaneously. Furthermore, through stacking multi-layers, our framework can generate longer and more complex meta-paths. After that, the overall framework, including meta-path generation online and Top-N meta-paths selection, can be optimized via back propagation in an end-to-end fashion. The contributions of our work are summarized as follows:

(1) We propose a novel framework MGOHE, which can simultaneously generate meaningful meta-paths within k-hop and learn effective node embeddings on heterogeneous information networks for specific tasks.
(2) We use an attention mechanism to select Top-N meta-paths within k-hop for generating meaningful multi-hop connections without domain knowledge, which can avoid the redundant information brought by noise meta-paths.
(3) We evaluate the performance of the proposed MGOHE on the two real-world datasets: DBLP, ACM. The extensive experiments prove that our MGOHE improves the performance of classification task and clustering task. And our MGOHE can generate meaningful meta-paths like other predefined approaches, which also improve the interpretability of meta-paths.

II. Related Work

A. Graph Neural Network

Graph neural networks (GNNs) were introduced in [16], which aim to extend the deep neural network to process arbitrary graph-structured data. GNNs are categorized into two approaches: spectral [9], [10] and non-spectral method [11], [12], [17]. On one hand, [9] proposed a method to perform convolution in the spectral domain using the Fourier basis of a given graph. [10] uses a localized first-order approximation of spectral graph convolution to simplify the GNNs. On the other hand, non-spectral approaches define convolution operations directly on the graph, operating on spatially close neighbors. [11] introduces self-attention to learn the importance between nodes and its neighbors and fuse the neighbors' information for graph representation learning.

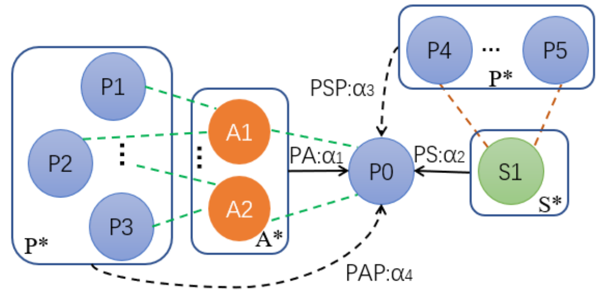


Fig. 2. This figure shows the node information aggregation process of MGOHE. α_i represents relative attention value of a meta-path. Green and orange dashed lines indicate edges. $\{P, A, S\}^*$ represents information aggregation at a specific node level. The black solid lines with arrow represent a meta-path aggregation from 1-hop neighborhoods. The black dashed lines with arrow represent a meta-path aggregation from k-hop neighborhoods, in this case $k = 2$.

B. Network Embedding

Network embedding, mapping a graph data into a latent space, has been widely used in many downstream network tasks. Traditionally, hand-crafted features have been used, such as graph kernel [18] and graph statistics [19]. Like other fields, hand-crafted features are not flexible and perform poorly. To overcome the drawback, recent network embedding methods, such as Deepwalk [20] and Node2vec [21], via random walks [22] on graphs to generate corpus and then feed them into a skip-gram [23] model to learn node embeddings, have managed to achieve better performance. However, these methods learn node embeddings solely based on the graph structure. To improve scalability or performance, attention mechanism on neighbors [11], [12], generalized convolution based on spectral convolution [24] have been studied. GraphSAGE [12] has proposed learnable aggregator functions which summarize information over a fixed size node neighbor. GAT [11] uses self-attention to learn the importance between nodes and its neighbors and fuses the neighbors to learn node embeddings. Although these methods show excellent performance, they all suffer from a common limitation that they cannot apply to the heterogeneous graphs, and can only deal with homogeneous graphs.

Heterogeneous information networks embedding learn the node embeddings for a graph with various types of nodes and edges. In current works, heterogeneous graph embedding techniques mostly rely on meta-path that is a composite relation connected with two objects. Metapath2vec [25] learns graph representation by using meta-path-based random walk and it only utilizes one meta-path scheme which is hard to mine comprehensive semantic information. ESim [8] accepts predefined meta-paths as guidances to learn graph representation in a user-prefered embedding space for similarity search. Although ESim can utilize multiple meta-paths, it needs to conduct grid search [26] to find the optimal weights for the meta-paths. HAN [17] learns graph representation by decomposing a heterogeneous graph into several homogeneous graph constructed by predefined meta-paths. It can learn the optimal weights of different meta-paths through an attention mechanism. However, these mata-path-based approaches face challenges on path construction and selection, which require domain knowledge and thus might not be able to capture all meaningful relations for each task. In addition, performance can be greatly affected by the choice of meta-paths. Unlike these approaches, without any domain knowledge, our approach can automatically generate meaningful meta-paths for different nodes online and learn the node embeddings on a heterogeneous graph in an end-to-end fashion.

C. Meta-path Generation Automatically

Meta-path has been widely used in heterogeneous information network to capture semantic information, such as meta-path-based link prediction [27], [28], similarity measure [29], [30], node classification [17], [25], recommendation [31] and so on. However, most existing works manually select meta-paths by domain experts and thus might not be able to capture all meaningful relations. Hence, how to automatically construct or select meta-paths with rich semantics becomes an urgent problem to solve.

AMIE [32] proposes an approach that can automatically discover meta-paths. Given an example of a pair of nodes, AMIE utilizes a similarity aggregation function and a greedy framework to generate meta-paths that can optimally explain the relationship between these node pairs. However, it chooses the best meta-path for a pair of nodes, not the global. AMPG [33] employs a greedy algorithm to select the most relevant meta-path at each step according to the similarity score. Unlike these approaches, we exhaust all existing meta-paths within k-hop neighborhoods for a specific node online and then adopt an attention mechanism to select the Top-N meaningful meta-paths. In the process of selecting the meta-paths, we integrate node aggregation and meta-path aggregation for node embedding, which will promote each other.

III. THE PROPOSED FRAMEWORK

The goal of our framework, MGOHE, is to simultaneously generate meaningful meta-paths within k-hop and learn node embeddings on heterogeneous information networks for specific tasks. We begin this section by introducing the notations and definitions used in the rest of the paper followed by a brief background on network embedding. Finally, we describe our online meta-path generation method and an attention mechanism for Top-N meta-paths selection. Figure 3 presents the whole framework of MGOHE.

A. Preliminaries

Heterogeneous Information Network [13]. A heterogeneous information network is a specific graph which contains either multiple types of nodes or multiple types of edges, we denote a graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of objects and $\mathcal{E}$ is a set of links. A heterogeneous graph is also associated with a node type mapping function $\phi : \mathcal{V} \rightarrow \mathcal{A}$ and a link type mapping function $\psi : \mathcal{E} \rightarrow \mathcal{R}$. $\mathcal{A}$ and $\mathcal{R}$ denote the sets of the node types and the sets of the edge types respectively, where $|\mathcal{A}| + |\mathcal{R}| > 2$.

Meta-path [15]. A meta-path ρ is defined as a path in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$ (abbreviated as $A_1 A_2 \cdots A_{l+1}$) which describes a composite relation $R = R_1 \circ R_2 \circ \cdots \circ R_l$ between objects A_1 and A_{l+1}, where $\circ$ denotes the composition operator on relations. Note that we define the node itself as the meta-path within 0-hop.

Meta-path guided neighbors. Given a node i and a meta-path ρ in a heterogeneous graph, the meta-path guided neighbors $\mathcal{N}_i^\rho$ of node i are defined as the set of nodes which connect with node i via meta-path ρ.

B. Network Embedding

In the field of network embedding, no matter homogeneous graph embedding GCN [10], GAT [11], Deepwalk [20] or heterogeneous embedding HAN [17], they all share the same pipline: (1) find neighbors. (2) information aggregation of neighbors. Deepwalk [20] utilizes random walk to find neighbors and then adopts co-occurrence of neighbors as the features for information aggregation. GCN [10] and GAT [11] both choose 1-hop neighborhoods on the graph as neighbors. During information aggregation phase, GAT specifies different weights to neighbors while the GCN simply averages over neighbors. HAN [17] uses meta-path guided nodes of the same type with center node as neighbors. And then employs an attention mechanism similar to GAT for node information aggregation. We follow the HAN's strategy and use meta-path guided nodes with any type as neighbors where the meta-path is generated online. Then we adopt an attention mechanism similar to GAT for node information aggregation to obtain multiple meta-path-based embeddings. Finally, we adopt another attention mechanism to select Top-N meta-path-based embeddings for the semantic aggregation.

C. Meta-path Generation Online

Previous works [17], [34] perform graph convolution on the meta-path graphs which require manually predefined meta-paths. They only gather information from nodes of the same type. Instead, our proposed method generates meta-paths for given data and tasks online and performs graph convolution on the generated meta-path graphs. In this way, we can generate more meaningful meta-paths and utilize more subtle information from other types of nodes.

When performing information aggregation for a node, we exhaust all meta-paths within the node's k-hop neighborhoods online. In particular, we set k=2 in experiments and the reason will be explained later. Then we perform information aggregation on each meta-path guided neighbors to obtain a semantic embedding representing the meta-path.

Inspired by GAT [11], we use a self-attention mechanism to assign varying levels of importance to different neighbors. For the different feature spaces of different types of nodes(e.g. node with type ϕ_i), we design the node type-specific transformation matrix $\mathbf{W}_{\phi_i}$ to project the features into the same feature space.

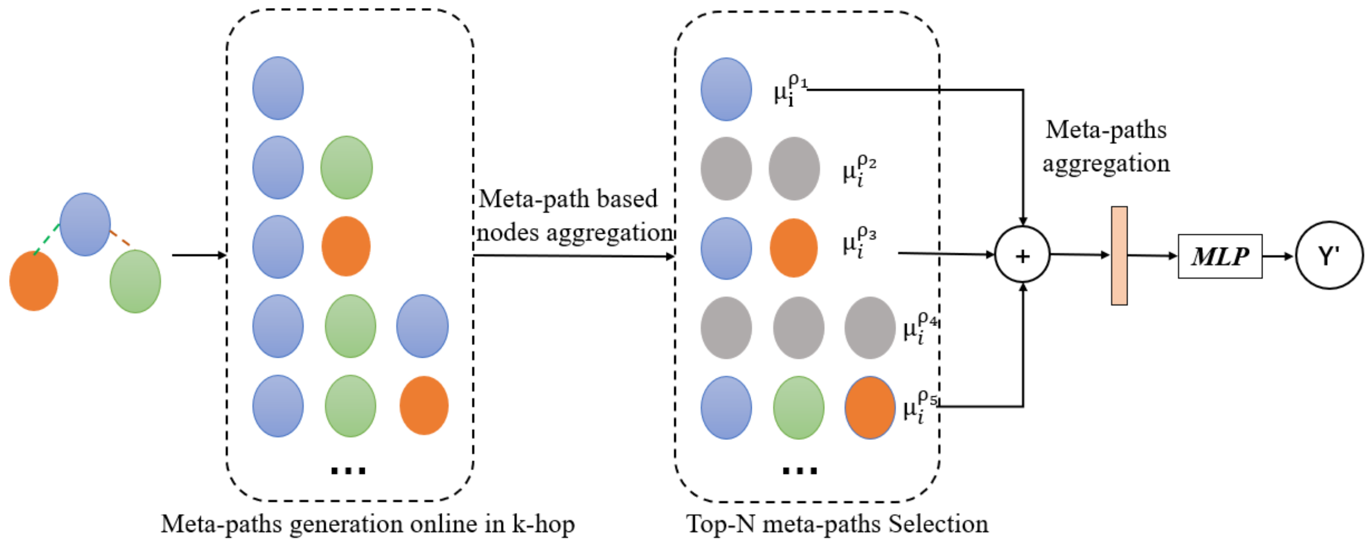


Fig. 3. This figure shows an end-to-end architecture of our MGOHE. There are two stages in the framework. Firstly, we exhaust all meta-paths in a particular node's k-hop neighborhoods. Meta-path guided neighbors are all homogeneous, and we employ a meta-path-based nodes information aggregation. Secondly, we adopt an attention mechanism to select the Top-N meaningful meta-paths for the final aggregation. μ_i^ρ is the attention coefficient for each meta-path.

$$\mathbf{h}_i' = \mathbf{W}_{\phi_i} \cdot \mathbf{h}_i \tag{1}$$

Where $\mathbf{h}_i$ and $\mathbf{h}_i'$ are the original and projected feature of node i respectively. Through type-specific projection operation, the attention can handle arbitrary types of nodes.

For the node j in set $\mathcal{N}_i^\rho$ of the node i, we leverage a self-attention to learn the attention coefficients of node j, which indicates the importance of node j to node i. Then we normalize them across all neighbors on meta-path ρ with the softmax function:

$$\alpha_{ij}^\rho = \frac{\exp\left(\mathbf{a}_\rho^{\mathrm{T}} \cdot \left[\mathbf{h}_i' \| \mathbf{h}_j'\right]\right)}{\sum_{j \in \mathcal{N}_\rho^k} \exp\left(\mathbf{a}_\rho^{\mathrm{T}} \cdot \left[\mathbf{h}_i' \| \mathbf{h}_j'\right]\right)} \tag{2}$$

where $\mathbf{a}_\rho$ is a parametrized weight vector for meta-path ρ, $\|$ denotes the concatenate operation. Then we perform nodes information aggregation based on each meta-path ρ, and obtain a meta-path-based embedding:

$$\mathbf{z}_i^\rho = \sigma\left(\sum_{j \in \mathcal{N}_i^\rho} \alpha_{ij}^\rho \cdot \mathbf{h}_j'\right) \tag{3}$$

where $\sigma(\cdot)$ denotes the Leakyrelu activation function. Similar to [17], we employ multi-head attention to stabilize the learning process. Specifically, we conduct meta-path generation online once. Then we repeat meta-path based node aggregation Q times and concatenate the learned embeddings as the meta-path-based embedding:

$$\mathbf{z}_i^\rho = \prod_{q=1}^{Q} \sigma\left(\sum_{j \in \mathcal{N}_i^\rho} \alpha_{ij}^\rho \cdot \mathbf{h}_j'\right) \tag{4}$$

D. *Top-N Meta-paths Selection*

Every node in a heterogeneous graph contains multiple semantic information. Each semantic is revealed by a particular meta-path. Since we exhaust all meta-paths within k-hop neighborhoods in previous stage, which will introduce useless or redundant information, the most meaningful Top-N meta-paths will be selected for the semantic information aggregation. To address the challenge of meta-path selection and semantic aggregation, we delicately design an attention mechanism to automatically learn different levels of importance to meta-paths for a particular node. Then we will select Top-N meta-paths with the largest attention coefficients and fuse them for the particular node and task. Taking all meta-path-based embeddings $\{z_i^1, z_i^2, \ldots z_i^\rho\}$ of node i as input, the learned weights of each meta-path $\{w_i^1, w_i^2, \ldots w_i^\rho\}$ can be shown as follows:

$$\{w_i^1, w_i^2, \ldots w_i^\rho\} = \mathrm{att}_{\text{meta-path}}\{z_i^1, z_i^2, \ldots z_i^\rho\} \tag{5}$$

Here $\mathrm{att}_{\text{meta-path}}$ denotes the function which performs the meta-path attention. It shows the importance of each meta-path for a specific node. We first transform meta-path-based embeddings through a nonlinear transformation.(e.g. one-layer MLP in experiments). Then we measure the importance of those meta-paths with an attention vector q. The importance of each meta-path, denote as w_i^ρ, is shown as follows:

$$w_i^\rho = \mathbf{q}^{\mathrm{T}} \cdot \tanh\left(\mathbf{W}^\rho \cdot \mathbf{z}_i^\rho + \mathbf{b}^\rho\right) \tag{6}$$

where W is the weight matrix, b is the bias vector, q is the attention vector. Note that for meaningful comparisons, all above parameters are shared for all meta-paths. After obtaining the importance of each meta-path, we normalize them via softmax function.

$$\mu_i^\rho = \frac{\exp\left(w_i^\rho\right)}{\sum_{\rho=1}^{P} \exp\left(w_i^\rho\right)} \tag{7}$$

Obviously, the higher μ_i^ρ is, the more important meta-path ρ is. So we select Top-N meta-paths with the largest attention coefficients for the semantic aggregation. With the learned weights as coefficients, we can fuse these meta-path-based embeddings to obtain the final embedding $\mathbf{Z}_i$ as follows:

$$\mathbf{Z}_i = \sigma\left(\sum_{\rho \in Top-N} \mu_i^\rho \cdot \mathbf{z}_i^\rho\right) \tag{8}$$

where $\sigma(\cdot)$ denotes the activation function elu. The final embedding is aggregated by Top-N meta-path-based embeddings.

We can apply the final embedding to particular tasks with different loss function. For node classification, we minimize the Cross-Entropy over all labeled node between the ground-truth and the prediction:

$$Loss = -\sum_{l \in \mathcal{Y}_L} \mathbf{Y}^l \ln\left(\mathbf{C} \cdot \mathbf{Z}^l\right) \tag{9}$$

where C is the parameter of the classifier, $\mathcal{Y}_L$ is the set of node indices that have labels, $\mathbf{Y}^l$ and $\mathbf{Z}^l$ are the labels and embeddings of labeled nodes.

IV. EXPERIMENTS

In this section, we evaluate the embedding quality of the proposed MGOHE and analyze the generated meta-paths using two large real-world datasets. We conduct extensive experiments and analysis to answer the following research questions: **Q1.** Can the proposed method learn effective node embeddings for classification and clustering? **Q2.** Can the proposed method generate meaningful meta-paths? Can the Top-N strategy select meaningful meta-paths?

A. Datasets

To evaluate the effectiveness of the proposed MGOHE, we use two heterogeneous graph datasets with multiple types of nodes and edges. Table I shows the detailed describtions of the dataset used here. **DBLP** contains four types of nodes (papers(P), authors(A), conferences(C), terms(T)), six types of edges (PA, AP, PC, CP, PT, TP), and research areas of authors as labels. **ACM** contains three types of nodes (papers(P), authors(A), subjects(S)), four types of edges (PA, AP, PS, SP), and categories of papers as labels. Each paper feature in the two datasets is the elements of a bag-of-words represented of keywords. The features of other types of nodes come from the additive aggregation of the papers connected to them.

B. Baselines

We compare MGOHE with several recent network embedding methods:

- **Deepwalk** [20]: A random walk based embedding method for the homogeneous graphs. Here we ignore the heterogeneity of nodes and perform Deepwalk on the whole heterogeneous graph.
- **Metapath2vec** [25]: A heterogeneous graph embedding method which performs meta-path-based random walk and utilizes skip-gram [23] to embed the heterogeneous graphs.
- **GCN** [10]: A semi-supervised graph convolutional network for the homogeneous graphs.
- **GAT** [11]: A graph neural network which uses the attention mechanism on the homogeneous graphs.
- **HAN** [17]: A heterogeneous graph embedding method which exploits manually selected meta-paths. Here, we test HAN on the selected meta-paths as described in [17].

C. Implementation Details

We set the embedding dimension to 64 for all the above methods for a fair comparison. We fix the number of layer to 2 and use Adam optimizer for all models. For random walk based models including Deepwalk and Metapath2vec, a walk length is set to 100 per node for 1000 iterations and the window size is set to 5 with 7 negative samples. For meta-path-based approaches, we follow the settings in HAN [17], e.g. PAP and PSP for ACM, APA, APCPA and APTPA for DBLP. For HAN, we use all the meta-paths. For Metapath2vec, GCN and GAT, we test all the meta-paths and report the best performance. All models are trained using a learning rate of $\eta = 0.005$, with anneals every 50 epochs by $\eta/2$ until 400 epochs were reached. For proposed MGOHE, we set the dimension of the attention vector q to 8, the number of attention head to 4. We use early stopping with patience of 80 that means we stop training if the validation loss does not decrease for 80 consecutive epochs. Since the 2-hop neighborhoods of a node must be connected to itself, which is the most meaningful meta-path, we set k for k-hop to 2. In order to capture more subtle semantic information, we select the Top-5 meta-paths with the largest attention coefficients for every layer. We use a structure similar to resnet [35], which means that we concat the embedding of the first layer and the embedding of the second layer as the final embedding.

D. Classification and Clustering

Table II shows the performance of MGOHE and other node classification baselines. Table III shows the performance of MGOHE and other clustering baselines. By analysing the result of our experiments, we will answer the question **Q1**.

Classification. In the classification task, we use KNN classifier with k=5 to perform node classification. In order to prove the reliability of the experimental data, we repeat KNN for 10 times to report the average of Macro-F1 and Micro-F1 in Tabel II. GAT tries to assign different weights to neighbors while GCN simply averages over neighbors. So

TABLE I
STATISTICS OF THE DATASETS

Dataset	Relations(A-B)	Number of A	Number of B	Number of A-B	Feature	Train	Val	Test
DBLP	Paper-Author	14328	4057	19645	6044	800	400	2857
	Paper-Conf	14328	20	14328				
	Paper-Term	14327	8789	88420				
ACM	Paper-Author	3025	6028	10055	1524	600	300	2125
	Paper-Subject	3025	73	3025				

TABLE II
QUANTITATIVE RESULTS (%) ON THE NODE CLASSIFICATION TASK

Datasets	Metrics	Deepwalk	Metapath2vec	GCN	GAT	HAN	**MGOHE**
DBLP	Macro-F1	86.31	91.23	87.62	90.28	91.69	**93.41**
	Micro-F1	85.03	91.57	88.83	90.75	92.74	**93.78**
ACM	Macro-F1	83.85	72.41	86.79	88.14	91.13	**92.79**
	Micro-F1	83.29	72.74	87.34	88.39	91.18	**92.84**

TABLE III
QANTITATIVE RESULTS (%) ON THE NODE CLUSTERING TASK

Datasets	Metrics	Deepwalk	Metapath2vec	GCN	GAT	HAN	**MGOHE**
DBLP	NMI	72.25	73.34	74.11	65.85	73.48	**76.79**
	ARI	79.95	78.43	79.39	69.01	79.04	**82.56**
ACM	NMI	48.01	23.37	55.40	64.87	71.10	**76.42**
	ARI	44.48	16.86	58.61	70.79	75.82	**82.31**

GAT usually performs better than GCN. HAN is a modified GAT for heterogeneous graph. With the guide of multiple meta-paths, the HAN performs better than GAT. This result shows that meta-path can introduce more semantic information. Interestingly, Metapath2vec is better than Deepwalk on DBLP and the opposite on ACM. Metapath2vec introduces meta-path on heterogeneous graph random walk compared with deepwalk. This result shows that meaningful meta-path can capture more semantic information while meaningless meta-path may cause adverse effects on performance. The MGOHE is modified from HAN, trying to generate meaningful meta-paths for specific datasets and tasks. We observe that our MGOHE achieves the best performance on all the datasets against other methods. It demonstrates that in classification task, the MGOHE can learn more meaningful meta-paths and make full use of heterogeneous information for effective node embedding.

Clustering. We also perform clustering task to evaluate the embeddings learned by the above algorithms. We use the same ground-truth as in node classification. And we adopt NMI and ARI to evaluate the quality of the clustering results. Since the performance of KMeans is affected by initial centroids, we repeat the process for 10 times and report the average result in Tabel III. Note we use the embeddings trained for classification task. We find that the results do not show the same trend in the clustering task and the classification task. For example, Metapath2vec is better than GCN in classification task and the opposite in clustering task. This inconsistency may come from the reason that the embeddings is optimized by classification loss. We also find that the performance on different datasets varies greatly. For example, the clustering performance of Metapath2vec, GCN and HAN is similar on DBLP while HAN is much better than the other two on ACM. On one hand, this shows the effectiveness of manually selected meta-paths. On the other hand, this manually-selected meta-paths have a lot of uncertainty and is more susceptible to the influence of experts. We observe that our MGOHE achieves the best performance on all the datasets against other methods. It proves that MGOHE is better at generating meta-paths that are actually meaningful without being affected by domain knowledge. It also proves the stability and effectiveness of GMOHE.

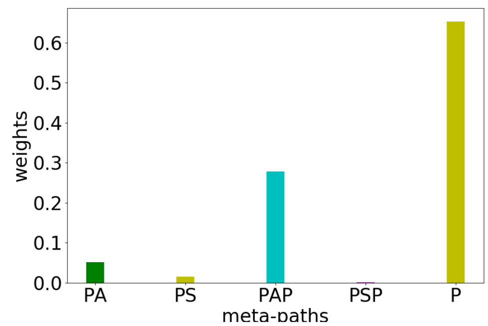
E. Meta-Path Generation

We compare the predefined meta-paths with the meta-paths generated by MGOHE to discuss the question **Q2**.

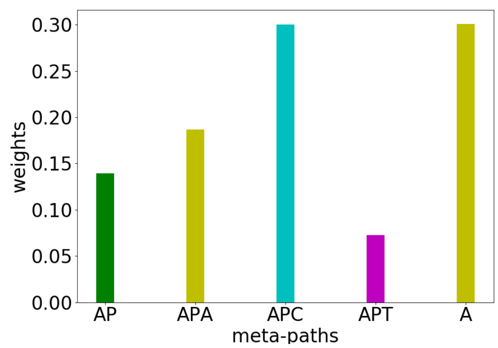
Essentially, we exhaust all meta-paths within k-hop neighborhoods, including self-loop connection. During training, we use the attention mechanism to optimize the weights of different meta-paths. When inferencing, we calculate the average weight of the corresponding meta-paths on all nodes. We argue that the average weight of all meta-paths reflect the importance of the meta-path in real scenarios. Taking paper in the ACM dataset as an example, there are 4 meta-paths(PA, PS, PAP, PSP, as shown in Figure 1(c) within 2-hop neighborhoods and one meta-path(P) representing self-loop. After training 100 epoches, we count the average weights of all meta-paths. As shown in Figure 4, meta-path P takes the maximum weight, which corresponds to our prior perception and the self-loop connection in HAN [17] simultaneously. PAP is the suboptimal meta-path and others take little weights. HAN [17] adopts PAP and PSP which strongly rely on experts knowledge. In experiments, MGOHE's weights is approximately equal with HAN's weights for PAP and PSP. However, MGOHE doesn't

TABLE IV
COMPARISON WITH PREDEFINED META-PATHS AND TOP-N META-PATHS BY MGOHE.

Datasets	Predefined Meta-path	Meta-path generated by MGOHE(Top-3)
DBLP	APCPA,APA,APTPA	APC*, APA*, AP*
ACM	PAP,PSP	PAP*, PA*, PS*



(a) Different weights for ACM meta-paths



(b) Different weights for DBLP meta-paths

Fig. 4. Different weights for meta-paths

require extensive domain knowledge. Furthermore, through stacking multi-layers, our framework can generate longer and more complex meta-paths.

We make a comparision with predefined meta-paths and Top-N meta-paths generated by MGOHE in Table IV. For example, PA is a meta-path found by MGOHE within one-layer and PA* indicates more complex meta-paths found by MGOHE within multi-layers. As shown in Table IV, the predefined meta-paths originated from domain knowledge are consistent with Top-N meta-paths generated by MGOHE. This shows that MGOHE are capable of learning the importance of meta-paths for particular task. More interestingly, MGOHE discovers important meta-paths that are not in the predefined meta-path set. For example, in the ACM dataset MGOHE ranks PA and PS as important meta-paths, which is not included in the predefined meta-path set. It makes sense that paper's subject(label to predict) is relevant to the venue where the author participated and the area of the conference. So we believe that the interpretability of MGOHE provides useful insight in node classification and clustering by the attention scores on meta-paths.

V. CONCLUSION

Our proposed Meta-Path Generation Online for Heterogeneous Network Embedding(MGOHE) can be applied to heterogeneous graph of arbitrary nodes without any domain expert knowledge. In MGOHE, we try to automatically generate meta-paths during the learning process to resolve the dependence of meta-paths construction on domain knowledge. Besides, the proposed Top-N strategy selects more meaningful meta-paths while excluding meaningless ones. Our framework can generate meaningful meta-paths for different nodes online and learn effective node embeddings simultaneously. Without any domain knowledge, experimental results including classification and clustering demonstrate the effectiveness of MGOHE. By analyzing the generated meta-paths and the learned attention weights of the meta-paths, the proposed MGOHE can generate meaningful meta-paths automatically. This has also proven its potentially good interpretability. However, the exhaustive way to search the optimal meta-paths is time-consuming. Therefore, in future work, we will explore heuristics to automatically generate meaningful meta-paths.

REFERENCES

[1] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu, "Heterogeneous Information Network Embedding for Recommendation," *arXiv e-prints*, p. arXiv:1711.10730, Nov 2017.

[2] W. X. Z. Binbin Hu, Chuan Shi and P. S. Yu., "Leverage meta-path based context for top-n recommendation with a neural co-attention model," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. London, United Kingdom: ACM, 2018. [Online]. Available: https://dl.acm.org/citation.cfm?id=3219965

[3] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, "Co-author relationship prediction in heterogeneous bibliographic networks," in *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining*, ser. ASONAM '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 121–128. [Online]. Available: http://dx.doi.org/10.1109/ASONAM.2011.112

[4] J. Zhang, P. S. Yu, and Z.-H. Zhou, "Meta-path based multi-network collective link prediction," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '14. New York, NY, USA: ACM, 2014, pp. 1286–1295. [Online]. Available: http://doi.acm.org/10.1145/2623330.2623645

[5] J. Zhang, P. S. Yu, and Y. Lv, "Organizational Chart Inference," *arXiv e-prints*, p. arXiv:1507.06841, Jul 2015.

[6] H. Chen, H. Yin, W. Wang, H. Wang, Q. V. H. Nguyen, and X. Li, "Pme: Projected metric embedding on heterogeneous networks for link prediction," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '18. New York, NY, USA: ACM, 2018, pp. 1177–1186. [Online]. Available: http://doi.acm.org/10.1145/3219819.3219986

[7] J. Shang, M. Qu, J. Liu, L. M. Kaplan, J. Han, and J. Peng, "Meta-path guided embedding for similarity search in large-scale heterogeneous information networks," *ArXiv*, vol. abs/1610.09769, 2016.

[8] J. Shang, M. Qu, J. Liu, L. M. Kaplan, J. Han, and J. Peng, "Meta-Path Guided Embedding for Similarity Search in Large-Scale Heterogeneous Information Networks," *arXiv e-prints*, p. arXiv:1610.09769, Oct 2016.

[9] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, “Spectral networks and locally connected networks on graphs.”

[10] T. N. Kipf and M. Welling, “Semi-Supervised Classification with Graph Convolutional Networks,” *arXiv e-prints*, p. arXiv:1609.02907, Sep 2016.

[11] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph Attention Networks,” *arXiv e-prints*, p. arXiv:1710.10903, Oct 2017.

[12] W. L. Hamilton, R. Ying, and J. Leskovec, “Inductive Representation Learning on Large Graphs,” *arXiv e-prints*, p. arXiv:1706.02216, Jun 2017.

[13] Y. Sun and J. Han, “Mining heterogeneous information networks: a structural analysis approach,” *ACM SIGKDD Explorations Newsletter*, vol. 14, no. 2, pp. 20–28, 2013.

[14] X. Kong, B. Cao, P. S. Yu, Y. Ding, and D. J. Wild, “Meta path-based collective classification in heterogeneous information networks.”

[15] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, “Pathsim: Meta path-based top-k similarity search in heterogeneous information networks,” *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.

[16] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, Jan 2009.

[17] X. Wang, H. Ji, C. Shi, B. Wang, P. Cui, P. Yu, and Y. Ye, “Heterogeneous graph attention network.”

[18] S. V. N. Vishwanathan, N. N. Schraudolph, R. Kondor, and K. M. Borgwardt, “Graph kernels,” *Journal of Machine Learning Research*, vol. 11, no. 2, pp. 1201–1242, 2010.

[19] S. Bhagat, G. Cormode, and S. Muthukrishnan, “Node classification in social networks,” *Computer Science*, vol. 16, no. 3, pp. 115–148, 2011.

[20] B. Perozzi, R. Al-Rfou, and S. Skiena, “DeepWalk: Online Learning of Social Representations,” *arXiv e-prints*, p. arXiv:1403.6652, Mar 2014.

[21] A. Grover and J. Leskovec, “Node2vec: Scalable feature learning for networks,” in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’16. New York, NY, USA: ACM, 2016, pp. 855–864. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939754

[22] J. D. Noh and H. Rieger, “Random walks on complex networks,” *Physical Review Letters*, vol. 92, no. 11, p. 118701.

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” *arXiv e-prints*, p. arXiv:1301.3781, Jan 2013.

[24] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, “Geometric deep learning: Going beyond euclidean data,” *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, July 2017.

[25] Y. Dong, N. V. Chawla, and A. Swami, “metapath2vec: Scalable representation learning for heterogeneous networks,” in *KDD ’17*. ACM, 2017, pp. 135–144.

[26] S. M. Lavalle and M. S. Branicky, “On the relationship between classical grid search and probabilistic roadmaps,” vol. 23, no. 23, pp. 673–692, 2003.

[27] J. Li, B. Ge, K. Yang, Y. Chen, and Y. Tan, “Meta-path based heterogeneous combat network link prediction,” *Physica A Statistical Mechanics & Its Applications*, vol. 482, pp. 507–523.

[28] X. Cao, Y. Zheng, C. Shi, J. Li, and B. Wu, “Meta-path-based link prediction in schema-rich heterogeneous information network,” *International Journal of Data Science & Analytics*, vol. 3, no. 4, pp. 285–296.

[29] Y. Xiao, Y. Sun, B. Norick, T. Mao, and J. Han, “User guided entity similarity search using meta-path selection in heterogeneous information networks,” in *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2012.

[30] P. Do and P. Pham, “Dw-pathsim: a distributed computing model for topic-driven weighted meta-path-based similarity measure in a large-scale content-based heterogeneous information network,” *Journal of Information and Telecommunication*, vol. 3, no. 1, pp. 19–38.

[31] T. Liang, C. Liang, W. Jian, D. Hai, and A. Bouguettaya, “Meta-path based service recommendation in heterogeneous information networks,” in *International Conference on Service-Oriented Computing*, 2016.

[32] C. Meng, R. Cheng, S. Maniu, P. Senellart, and W. Zhang, “Discovering meta-paths in large heterogeneous information networks,” in *Proceedings of the 24th International Conference on World Wide Web*, ser. WWW ’15. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2015, pp. 754–764. [Online]. Available: https://doi.org/10.1145/2736277.2741123

[33] X. Cao, Y. Zheng, C. Shi, J. Li, and B. Wu, “Link prediction in schema-rich heterogeneous information network,” in *PAKDD*, 2016.

[34] Y. Zhang, Y. Xiong, X. Kong, J. Mi, and Y. Zhu, “Deep collective classification in heterogeneous information networks,” 04 2018, pp. 399–408.

[35] K. He, X. Zhang, S. Ren, and S. Jian, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.