**Article:**

Su, H-L, Li, Z-P, Zhu, X-B et al. (2023) Hierarchical Graph Neural Network Based-on Semi-implicit Variational Inference. IEEE Transactions on Cognitive and Developmental Systems, 15 (2). pp. 887-895. ISSN: 2379-8920

https://doi.org/10.1109/tcds.2022.3193398

This article has been accepted for publication in IEEE Transactions on Cognitive and Developmental Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2022.3193398



# Hierarchical Graph Neural Network Based-on Semi-implicit Variational Inference

Hai-Long Su*, Zhi-Peng Li, Xiao-Bo Zhu, Li-Na Yang, Valeriya Gribova, Vladimir Fedorovich Filaretov, Anthony Cohn, De-Shuang Huang*, *Fellow, IEEE,*

***Abstract*—Graph neural network(GNN) has obtained outstanding achievements in relational data. However, these data have uncertain properties, for example, spurious edges may be included. Recently, Variational graph autoencoder(VGAE) has been proposed to solve this problem. However, the distributional assumptions in the variational family restrict the variational inference (VI) flexibility and they define variational families using mean-field, which can not capture complex posterior distributional. To solve the above question, in this paper, we proposed a novel GNN model based on semi-implicit variational inference (SIVI), which can embed the node to the latent space to improve VI flexibility and enhance VI expressiveness with mixing distribution. Specifically, to approximate the true posterior, a variational posterior was given utilizing a semi-implicit hierarchical variational framework, which can model complex posterior. Moreover, an iterative decoder is used to better capture graph properties. Besides, due to the hierarchical structure in our model, it can incorporation neighbour information between nodes. Experiments on multiple data sets, our method has achieved state-of-the-art results compared to other similar methods. Particularly, on the citation dataset Citeseer without features, our method outperforms VGAE by nine percentage.**

***Index Terms*—Latent variable, Variation inference, Graph neural network, Semi-implicit model, Hierarchical frame.**

## I. Introduction

CONVOLUTIONAL neural networks (CNNs) have made great achievements in the past ten years in the fields of speech [1], image [2, 3] and other fields [4]. However, CNNs can only handle normalized data, e.g., grids, sequences, i.e., Euclidean space data, which has translation invariance. Nevertheless, in reality there is a lot of data that is in non-European space, such as social network data, protein and protein (PPI) interaction data, transportation data, etc [8]. To process this data efficiently, graph neural networks (GNNs) have been proposed and also achievements huge progress in many fields, such as social networks [5], recommendation systems [6], protein-protein interaction [7] and action prediction [8][9]. Kipf *et.al* [10] proposed graph convolutional network (GCN), which learns the first-order approximate of nodes and followed a nonlinear activation function to learn graph representation. Considered neighbourhood information between nodes, Hamilton *et.al* [11] proposed GraphSAGE, a inductive representation learning method, which can aggregation neighbourhood information by aggregate function. GraphSAGE is designed to generate low-dimensional vector representations of nodes and is particularly useful for graphs with rich information on node attributes. Different neighbour nodes have different weights, specifying different weights to different nodes in the neighbourhood that can prevent redundant information from being aggregate, hence, Veličković *et.al* [12] proposed graph attention networks(GATs) use masked self-attentional layers. Besides, based on the GNN, some interesting works were also proposed by scholars, such as graph pool [13], place classification [14], and facial expression recognition [15].

Although it's very effective in dealing with relational data, there are some challenges. For example, graphs are very huge in nature, graph structural information is ignored. To eradicate the uncertain problem, Bayesian-based approaches were proposed. For instance, Zhang *et.al* proposed BGCN [16] that incorporation uncertainly graph information via parametric random graph model. However, BGCN relies heavily on the selection of random graphs, which ignore the node features and training labels. To overcome this drawback, Pal *et.al* [17] introduced a non-parametric BGCN, which uses the node features, training labels, and observed graph for posterior inference. Following this idea, many Bayesian-based approaches are proposed [18–20].

Variational autoencoders(VAEs) is a popular method for unsupervised representation learning of high dimensional data [21]. Inspired by VAEs, variational graph autoencoders(VGAEs) [22] was proposed. VGAE uses GCN as an encoder and a simple inner product as the decoder. VGAE assumes the variational posterior is Gaussian distribution, which is restricted variational inference flexibility. And inner product decoder limits the ability to generate models. Recently, semi-implicit variational inference (SIVI) [24] and normalizing flow (NF) [25–27] are proposed, which provide flexibility posterior distribution and effective optimization. Expansion

This work was supported by the grant of National Key R&D Program of China (No. 2018AAA0100100) and partly supported by grants from the National Science Foundation of China, Nos. 61732012, 61932008, 62002266, and 62073231, and by the Scientific & Technological Base and Talent Special Program of the Guangxi Zhuang Autonomous Region, GuiKe AD18126015 and by "BAGUI Scholar" Program of Guangxi Province of China.

Hai-Long Su, Zhi-Peng Li, and Xiao-Bo Zhu are with Institute of Machine Learning and Systems Biology, School of Electronics and Information Engineering, Tongji University, Caoan Road 4800, Shanghai 201804, China. (e-mail: hailongsu@tongji.edu.cn; Zhipeng_li@tongji.edu.cn; xiaobozhu_tj@163.com)

Li-Na Yang is with the School of Computer, Electronics and Information, Guangxi University, Nanning 530004, China. (e-mail: lnyang@gxu.edu.cn)

Valeriya Gribova and Vladimir Fedorovich Filaretov are with Institute of Automation and Control Processes, Far Eastern Branch of Russian Academy of Sciences, Russia. (e-mail: gribova@iacp.dvo.ru; filaretov@inbox.ru).

Anthony Cohn is with Automated Reasoning in the School of Computing, University of Leeds, England. (e-mail:A.G.Cohn@leeds.ac.uk).

De-Shuang Huang is with Big Data and Intelligent Computing Research Center, Guangxi Academy of Science, Nanning, 530007, China, and Institute of Machine Learning and Systems Biology, Tongji University. (dshuang@tongji.edu.cn)

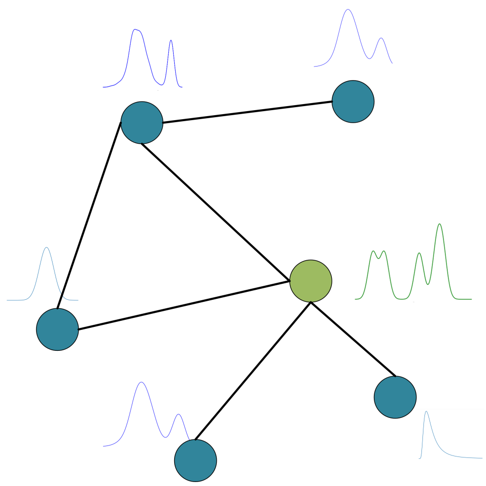

Fig. 1. Node distribution propagation. Our method can diffuse the node's distributions between neighbors.

SIVI to graph domain, SIG-VAE [28] was proposed. SIG-VAE introduced a semi-implicit model to learn the posterior, which can well simulation the complex posterior with heavy tails, skewness, multimodality characteristics.

Through parameterizing VAEs by GNN, Grover *et.al* [29] proposed Graphite, a framework for large graph nodes representation. Graphite uses forward message passing for encoding nodes to latent space and reverse message passing for decoding. Despite the excellent results Graphite obtained, however, in the encoder step, Graphite assumes the posterior is isotropic Gaussian, which results in high variance, and it restricts the expressive ability, hence, it can not model complex posterior.

Inspired by SIVI and SIG-VAE, we proposed a hierarchical graph neural network based on semi-implicit variational inference. Specifically, to handle Graphite's distribution assumptions and inability to model complex distributions, we proposed a hierarchical frame for our model in the encoder step, enhancing the expressiveness of posterior distribution for nodes in the latent space. For do this, we construct stochastic multi-layers hierarchically and inject random noise at every layer. After doing this, the output of GNN is random variables rather than deterministic, hence, uncertainty in the structure of a graph can be measured. More specifically, when obtain node's latent posterior distributions, the distributions of their neighboring nodes are incorporated simultaneously, which is very important for graph representation, as shown in Fig.1. The distributions of aqua green nodes can propagation to olive node. In the decoder step, following Graphite, reverse message passing is used to construct the graph. For model inference, we derive a lower evidence lower bound (ELBO) followed by SIVI [24].

The contributions of our proposed method are as follows:

1) We proposed a new GNN model, which can propagation uncertainty between neighbourhood nodes. Contrary to determinant GNN, such as GCN and GAT, our method can metric the uncertainty of graph structure, which is very important for information aggregation.
2) We proposed a new encoder framework for variational posterior learning. Specifically, different from the traditional encoder, which assumes posterior to be Gaussian distribution, we use a hierarchical design for model to learn the parameters of posterior, then, we can obtain a more expressive posterior, which can model complex graph data.
3) We use our proposed method to perform link prediction task and conduct extensive experiments on 3 citation datasets(ie, Cora, Cisteer, Pubmed) and 4 different scenario datasets, our method obtains start-of-the-art results compared with the baseline method.

## II. PRELIMINARIES

Given a graph $\mathbf{G} = (\mathbf{E}, \mathbf{V})$ where $\mathbf{E}$ and $\mathbf{V}$ refer to edges and nodes of the graph respectively. Additionally, the feature matrix of the graph defined as a *m*-dimensional signal $\mathbf{X} \in \mathbb{R}^{m\times n}$ associated with each node. In this paper, graph structure is represented by symmetric adjacency matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ where $n = |\mathbf{V}|$ and $\mathbf{A}_{i,j} = 1$ denote there is an edge between node $i$ and $j$ and $\mathbf{A}_{i,j} = 0$ on the contrary.

### A. Graph Neural Networks

Sperduti *et.al* [30] applied neural network to solve directed acyclic graphs, which is regarded as a motivation of GNN. The notion of GNN was declared by Gori *et.al* firstly [31]. After that, Scarselli *et.al* [32] and Gallicchio *et.al* [33] further developed GNN. The intuitive idea behind GNN is message passing between nodes and their neighbourhood by an iterative manner until a fixed point appears or converges. However, this process is inefficient computation, hence many works attempts to overcome this problem [34] [35].

Drawing on the ideas of CNNs, convolutional GNN (ConvGNN) has been proposed. There are two streams of ConvGNN: spectral-based and spatial-based approaches. The first spectral-based method was proposed by Bruna [36], which introduced a graph convolution based on spectral graph theory. Inspired by this method, many works have been raised recently [10, 37, 38]. Spatial-based approaches have been proposed recently by scholars. Compared with spectral-based methods, which involved spectral graph theory, spatial-based approaches operate directly on the nodes of graph. The spatial-based approach maximizes the use of the rich information of the nodes, hence, many researchers focus on this area. For example, GraphSAGE [12] and GAT [11] are proposed benefits from effective node aggregation. In order to capture high-order information of neighbour nodes, multi-hop neighbourhood aggregation is essential. Hence, based on Weisfeiler-Leman(WL) [39], high order GNN has proposed [40, 41], which extension 1-WL test to k-WL test to capture multi-hop information of nodes.Furthermore, many VAE-based [42, 43] GNNs are also proposed. Such as VGAE [22], GraphVAE [44], simple-GVAE [45], CGVAE [46].

### B. Semi-implicit Variational Inference

The vanilla variational inference (VI) has the following drawbacks: 1) optimization difficulties when learning posterior distributions, 2) cannot resolve the distribution with skewness, kurtosis, multimodality, and other characteristics. To overcome the above questions, Semi-implicit Variational Inference(SIVI) [24] has been proposed. SIVI is similar to HVM proposed by Ranganath *et.al* [47]. SIVI assumes that posterior parameters are derived from an implicit distribution rather than being analytic. SIVI also has a hierarchical structure, which can capture complex posterior distribution. More specifically, assuming $\varphi \sim q_\psi(\varphi)$, where $\psi$ denotes distribution parameter can be inferred, the SIVI distribution for $Z$ can be defined as a hierarchical manner: $Z \sim p(Z|\varphi), \varphi \sim q_\psi(\varphi)$. In order to obtain a tractable variational posterior, SIVI derives a lower bound for ELBO [48–51] to optimize the variational parameters.

## III. Proposed Method

When we use probabilistic graphical model for graph representation learning, we are interested in learning a parameterized distribution over adjacency matrix $\mathbf{A}$ and the nodes feature $\mathbf{X}$ is added as conditioning evidence. In this paper, to achieve this goal, we induct a latent variable $\mathbf{Z}_i \in \mathbb{R}^k$ and the node feature $\mathbf{X}_i \in \mathbb{R}^m$ for each node $i \in 1, 2, \ldots, n$ alone with $\mathbf{A}_{i,j} \in \mathbb{R}$. Without loss of generality, we use concise representation $\mathbf{Z} \in \mathbb{R}^{n \times k}$, $\mathbf{X} \in \mathbb{R}^{n \times m}$, and $\mathbf{A} \in \mathbb{R}^{n \times n}$ for the variables, which are conditional independencies.

Then, through maximizing the marginal likelihood of the observed adjacency matrix $\mathbf{A}$ conditioned on $\mathbf{X}$, model parameters $\Theta$ can be obtained as follows:

$$\underset{\Theta}{\operatorname{argmax}} \log p_\Theta(\mathbf{A}|\mathbf{X}) = \log \int_{\mathbf{Z}} p_\Theta(\mathbf{A}, \mathbf{Z}|\mathbf{X}) d\mathbf{Z} \tag{1}$$

Here, $p(\mathbf{Z}|\mathbf{X})$ is a fixed prior distribution over the latent variable of every node associated with graph $\mathbf{G}$. However, equation (1) is intractable, hence, a variational posterior $q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X})$ is introduced to approximate the true posterior, which can be turned into an optimization problem. Therefore, we can obtain a tractable evidence lower bound (ELBO) to the above objective with parameters $\Phi$ :

$$\begin{aligned} \log p_\Theta(\mathbf{A}|\mathbf{X}) = & \underbrace{\mathbb{E}_{q_\Phi(\mathbf{Z}|\mathbf{A},\mathbf{X})}\left[\log \frac{p_\Theta(\mathbf{A}, \mathbf{Z}|\mathbf{X})}{q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X})}\right]}_{\textbf{ELBO}} \\ & + \underbrace{\mathbb{E}_{q_\Phi(\mathbf{Z}|\mathbf{A},\mathbf{X})}\left[\log \frac{q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X})}{p_\Theta(\mathbf{Z}|\mathbf{A}, \mathbf{X})}\right]}_{\mathbf{KL(q_\Phi(Z|A,X)||p_\Theta(Z|A,X))}} \end{aligned} \tag{2}$$

Since the $\mathbf{KL(q_\Phi(Z|A,X)||p_\Theta(Z|A,X)) \geqslant 0}$, ie, it's non-negative, hence, equation (2) can be rewritten as follows:

$$\log p_\Theta(\mathbf{A}|\mathbf{X}) \geqslant \mathbb{E}_{q_\Phi(\mathbf{Z}|\mathbf{A},\mathbf{X})}\left[\log \frac{p_\Theta(\mathbf{A}, \mathbf{Z}|\mathbf{X})}{q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X})}\right] \tag{3}$$

when the approximate posterior $q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X})$ matches the true posterior $p_\Theta(\mathbf{Z}|\mathbf{A}, \mathbf{X})$, the lower bound is tight. Therefore, in order to obtain the best approximate of true posterior, we can maximize the (3) to optimize the parameters. Hence, the question is how to solve the $q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X})$(*encoder*) and $p_\Theta(\mathbf{A}|\mathbf{Z}, \mathbf{X})$(*decoder*).

### A. Encoder using semi-implicit variational inference

A typical approach of defining variational posterior is to use the mean-field (MF), then, the posterior can been written as $q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X}) \approx \prod_{i=1}^n q_{\Phi_i}(z_i|A, X)$, where $q_{\Phi_i}(z_i|A, X)$ is assumed to be Gaussian distribution. A high variance will occur, however, when the Gaussian distribution mismatching true posterior and it's restricting expressiveness of posterior following from the distributional assumptions [42]. To address the aforementioned problem, instead of using MF in previous work[52], we use a hierarchical semi-implicit framework to approximation the variational posterior. Specifically, in order to diffusion the uncertain between node and it's neighborhoods, first, we inject random noise at every layer of our encoder and concatenating it with nodes attributes, so that the output of GNN are random variables:

$$\mathbf{h}_p = \mathbf{GNN}_p(\mathbf{A}, CON(\mathbf{X}, \epsilon_p, \mathbf{h}_{p-1})) \tag{4}$$

where $\epsilon_p \sim q_p(\epsilon)$ is Bernoulli distribution of $N$-dimensional noise. $p$ denotes the number of layers, $\mathbf{h}_0 = \mathbf{X}$, $CON$ denote CONCAT operation. With the above operation, the output of GNN $\mathbf{H} = CON(\{\mathbf{h}_i\}_{i=1}^p)$ is a random variable. Hence, it has been used to measure the uncertainty of GNN. Then, we use this random $\mathbf{H}$ with $\mathbf{X}$ and $\mathbf{A}$ to learn the parameters of variational posterior:

$$\mu(\mathbf{A}, \mathbf{X}) = \mathbf{GNN}_\mu(CON(\mathbf{X}, \mathbf{h}_P), \mathbf{A}) \tag{5a}$$

$$\sigma(\mathbf{A}, \mathbf{X}) = \mathbf{GNN}_\sigma(CON(\mathbf{X}, \mathbf{h}_P), \mathbf{A}) \tag{5b}$$

$CON$ denote CONCAT operation. Through (5a) and (5b), we can obtain the parameters of variational posterior. Attributed to the equation (4), which can propagate the uncertainly through the different layers, hence, when to learn the parameters $\mu$ and $\sigma$ it can aggregation the neighbourhoods information of nodes, which is very important for aggregation nodes information. Finally, we can obtain the variational posterior as follows:

$$q_\Phi(\mathbf{Z}|\mathbf{A}, \mathbf{X}, \mu, \sigma) = \Pi_{i=1}^N q_\Phi(\mathbf{z}_i|\mathbf{A}, \mathbf{X}, \mu_i, \sigma_i) \tag{6}$$

where $q_\Phi(\mathbf{z}_i|\mathbf{A}, \mathbf{X}, \mu_i, \sigma_i) = \mathcal{N}(\mu_i(\mathbf{A}, \mathbf{X}), \sigma_i(\mathbf{A}, \mathbf{X}))$.

In contrast to Graphite [52], which uses the mean-field approximation to define variational family and assumes the posterior is isotropic Gaussian, which restricts the expression of posterior distribution. Moreover, its output of GNN is determination, which can not propagate uncertainly between the nodes. Our approach can propagate uncertainly between the nodes with the random output of GNN and aggregate the neighborhood information naturally. This can increase the power of expressiveness and capture the complex posterior. Our approach can be seen in Fig.2.

### B. Decoder using iterative manner

Following the [52] uses iterative manner for decoding, this paper also uses this approach. Specifically, given the latent variable $\mathbf{Z}$ and feature matrix $\mathbf{X}$, an intermediate weighted $\tilde{\mathbf{A}}$ can be calculated through the inner-product of $\mathbf{Z}$.

$$\tilde{\mathbf{A}} = \frac{\mathbf{Z}\mathbf{Z}^\mathbf{T}}{||\mathbf{Z}||^\mathbf{2}} + \mathbf{1}\mathbf{1}^\mathbf{T} \tag{7}$$

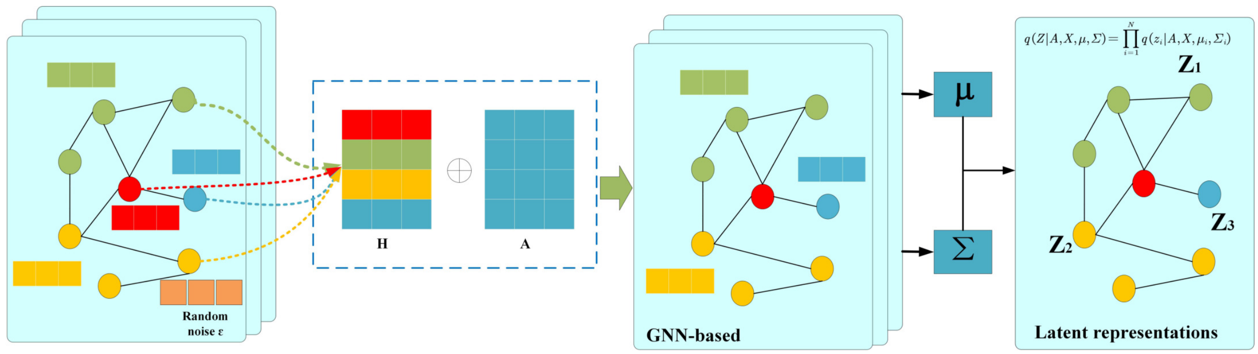


Fig. 2. The flowchart of our encoder. $\mathbf{H}$ and $\mathbf{A}$ denote the output of GNN and adjacency matrix respectively. In the input graph, we add random noise to every layer, which makes output $H$ has randomness. $\bigoplus$ denote CONCAT operation.

$$\mathbf{Z}^* = \mathbf{GNN_\Theta}(\tilde{\mathbf{A}}, [\mathbf{Z}|\mathbf{X}]) \tag{8}$$

Hence, the decoder is given as follows:

$$p_\Theta(\mathbf{A}|\mathbf{Z}, \mathbf{X}) = \prod_{i=1}^{n}\prod_{j=1}^{n} p_\Theta^{(i,j)}(\mathbf{A}_{i,j}|\mathbf{Z}^*) \tag{9}$$

In order to representation learning of large graph more scalable, the matrix right multiplications is adopted for inference. A simplified graph propagation rule is adopted: $\mathbf{H}^{(l)} \leftarrow \zeta_l(\tilde{\mathbf{A}}\mathbf{H}^{(l-1)})$. For finally embedding of nodes, we use $\mathbf{Z}^f = (1-\lambda)\mathbf{Z} + \lambda\mathbf{Z}^*$.

Towards to link prediction task, We can compute the probability that whether two edges are connected as follows:

$$p(\mathbf{A}_{i,j} = 1|z_i^f, z_j^f) = \delta(\mathbf{Z}^f {\mathbf{Z}^f}^T) \tag{10}$$

where $\lambda$ is hyper-parameter and $\delta$ is sigmoid function.

### C. Model Inference

The first term on the right side of equation (2) is the ELBO, which is used to inference the model. Following the SIVI [24], we construct a hierarchical function: $\mathbf{Z} \sim q(\mathbf{Z}|\mathbf{\Psi})$, where $\Psi \sim q_\Phi(\Psi|\mathbf{X}, \mathbf{A})$. i.e., $\Psi$ drawn from a distribution Therefore, we can rewritten the ELBO as follows:

$$\begin{aligned}\mathbb{L} &= \mathbb{E}_{q_\Phi(\mathbf{Z}|\mathbf{A},\mathbf{X})}\left[\log \frac{p_\Theta(\mathbf{A},\mathbf{Z}|\mathbf{X})}{q_\Phi(\mathbf{Z}|\mathbf{A},\mathbf{X})}\right] \\ &= -\mathbf{KL}(\mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})})\left[q(\mathbf{Z}|\mathbf{\Psi})||p(\mathbf{Z}))\right] \\ &+ \mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})}\left[\mathbb{E}_{\mathbf{Z}\sim q(\mathbf{Z}|\mathbf{\Psi})}\log p(\mathbf{A}|\mathbf{Z})\right].\end{aligned} \tag{11}$$

Based on the [24]'s first theorem, we have:

$$\begin{aligned}&\mathbf{KL}(\mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})})\left[q(\mathbf{Z}|\mathbf{\Psi})||p(\mathbf{Z}))\right] \\ &\leqslant \mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})}\mathbf{KL}(q(\mathbf{Z}|\mathbf{\Psi})||p(\mathbf{Z})).\end{aligned} \tag{12}$$

Hence, equation (11) can be written as follows:

$$\begin{aligned}\mathbb{L} &= -\mathbf{KL}(\mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})})\left[q(\mathbf{Z}|\mathbf{\Psi})||p(\mathbf{Z}))\right] \\ &+ \mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})}\left[\mathbb{E}_{\mathbf{Z}\sim q(\mathbf{Z}|\mathbf{\Psi})}\log p(\mathbf{A}|\mathbf{Z})\right] \\ &\geqslant \mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})}\left[\mathbf{KL}(q(\mathbf{Z}|\mathbf{\Psi})||p(\mathbf{Z}))\right] \\ &+ \mathbb{E}_{\Psi\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})}\left[\mathbb{E}_{\mathbf{Z}\sim q(\mathbf{Z}|\mathbf{\Psi})}\log p(\mathbf{A}|\mathbf{Z})\right] = \mathbb{L}^*\end{aligned} \tag{13}$$

Directly optimizing $\mathbb{L}^*$ could engender our method degenerates to the vanilla VGAE, duo to could lead to a point mass density as $q_\Phi(\Psi|\mathbf{X}, \mathbf{A})$. Therefore, in order to prevent this degeneracy, add a regularization term to $\mathbb{L}^*$. Assume that $S$ samples are derived from $q_\Phi(\Psi|\mathbf{X}, \mathbf{A})$, which denotes $\{\Psi^{(i)}\}_{i=1}^S$. Hence, we can define $\mathcal{L}_S$ as follows:

$$\mathcal{L}_S = \mathbb{E}_{\Psi,\Psi^{(1)},\ldots,\Psi^{(S)}\sim q_\Phi(\Psi|\mathbf{A},\mathbf{X})}[\mathbf{KL}(\mathbf{q}(\mathbf{A}|\mathbf{\Psi})||\hat{\mathbf{h}}_\mathbf{S}(\mathbf{Z})] \tag{14}$$

where,

$$\hat{h}_S(\mathbf{Z}) = \frac{\mathbf{q_\Phi}(\mathbf{\Psi}|\mathbf{A},\mathbf{X}) + \sum_{\mathbf{s=1}}^{\mathbf{S}} \mathbf{q_\Phi}(\mathbf{\Psi^{(s)}}|\mathbf{A},\mathbf{X})}{\mathbf{S}+\mathbf{1}}$$

Finally, the ELBO can be described as follows:

$$\mathcal{L} = \mathbb{L}^* + \mathcal{L}_S \tag{15}$$

Our method is described in Algorithm 1.

**Algorithm 1** The algorithm of our method.

INPUT: $\mathbf{A}, \mathbf{X}$.
Initializing $\Theta$ and $\Phi$.
Sample $\epsilon_p \sim q_p(\epsilon)$ where $q_p(\epsilon)$ is Bernoulli distribution.

**for** iteration $t = 1, 2, \ldots, n$ do **do**
Computing $\mathbf{h_P}$ according to equation (4).
Computing $\mu$ and $\sigma$ according to equation (5a) and (5b) respectively.
Computing $q(\mathbf{Z}|\mathbf{X}, \mathbf{A})$ according to equation (6).
Computing $p(\mathbf{A}|\mathbf{X}, \mathbf{Z})$ according to equation (9).
Update $\Theta$ and $\Phi$ by maximizing $\mathcal{L}$ in (14).

**return** $\Theta, \Phi$.
**end for**

OUTPUT: Model Parameter $\Theta$.

For link prediction task, computing $\mathbf{Z}^*$ according to equation (8), then, $\mathbf{Z}^f = (1-\lambda)\mathbf{Z} + \lambda\mathbf{Z}^*$ is used to computing $p(\mathbf{A}_{i,j} = 1|z_i^f, z_j^f) = \delta(\mathbf{Z}^f {\mathbf{Z}^f}^T)$.

## IV. Experiments

### A. Datasets

We evaluate our method on 3 citation networks, ie., Cora, Citeseer, and Pubmed with paper as nodes and citations as edges [53]. Furthermore, four different datasets without features are used to validate our method, *i.e.,* NS: a collaborative

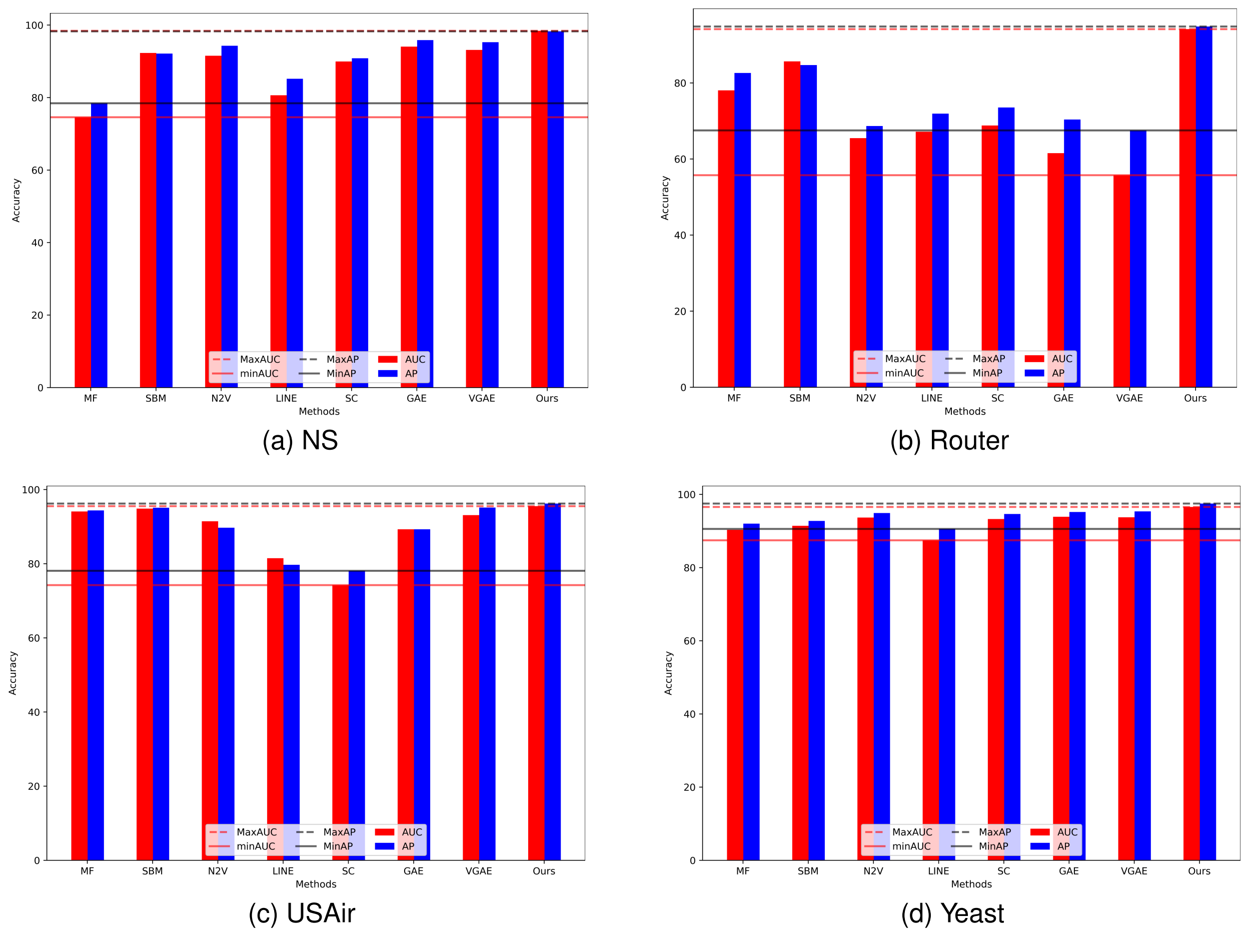


Fig. 3. AUC and AP of link prediction on the four different datasets. (a) NS: Collaboration Network. (b) Router: Internet Network. (c) USAir: Transportation Network. (d) Yeast: Protein Network. The red dashed and solid lines represent the maximum and minimum values of AUC. The black dashed and solid lines represent the maximum and minimum values of AP.

TABLE I
DESCRIPTION OF THE DATASETS DETAILS. '-' INDICATES THAT THE CHARACTERISTIC IS NOT AVAILABLE IN THE CORRESPONDING DATASET

| | Nodes | Edges | Node feature | Labels |
|---|---|---|---|---|
| Cora | 2708 | 5429 | 1433 | 7 |
| Citeseer | 3327 | 4732 | 3703 | 6 |
| Pubmed | 19717 | 44338 | 500 | 3 |
| NS | 1589 | 2742 | - | - |
| USAir | 332 | 2126 | - | - |
| Router | 5022 | 6258 | - | - |
| Yeast | 2375 | 11693 | - | - |

network of network science researchers, including 1589 nodes and 2742 edges [55], Router: a router internet with 5022 nodes and 6258 edges [56], USAir: US Airlines network, including 332 nodes and 2126 edges [57], Yeast: a protein-protein interaction (PPI) network in yeast, which has 2375 nodes and 11693 edges [58]. The detailed characteristics of the datasets are summarized in TABLE I.

### B. Training Configurations

We evaluate the performance of our approach for the link prediction task. All GNN models in the experiments were implemented using GCN [10]. The link prediction task is to predict whether an edge exists between two nodes [21]. We experiment on two different types of datasets: with and without features. The datasets was split into 5%, 10% and remaining for validation, testing and training respectively as done in [22]. We run our model for 3500 epochs with a learning rate of 0.0005 and training using Adam optimizer. The latent space dimensional is 16. The dimensional of Bernoulli noise $\epsilon$ is 64 and 32 for features and featureless respectively.

We organized the experiments on an experimental machine with Intel Xeon(R) CPU E7-8867 v4 @2.00GHz*80, GPU NVIDIA GTX 2080Ti, MEMORY 47.0GiB, Disk 698.4GB. Experimenting with implementation on the GPU version using TensorFlow [54].

### C. Baselines And Metric

We compared our approach with some similar methods: Spectral Clustering (SC) [59], DeepWalk (DW) [60], Graphite [52], VGAE [22], GAE [22]. The model was evaluated by Area Under the ROC Curve (AUC) and Average Precision (AP). Particular, SC and DW do not provide the ability to merge node features when learning embeddings, hence they both validate on featureless datasets.

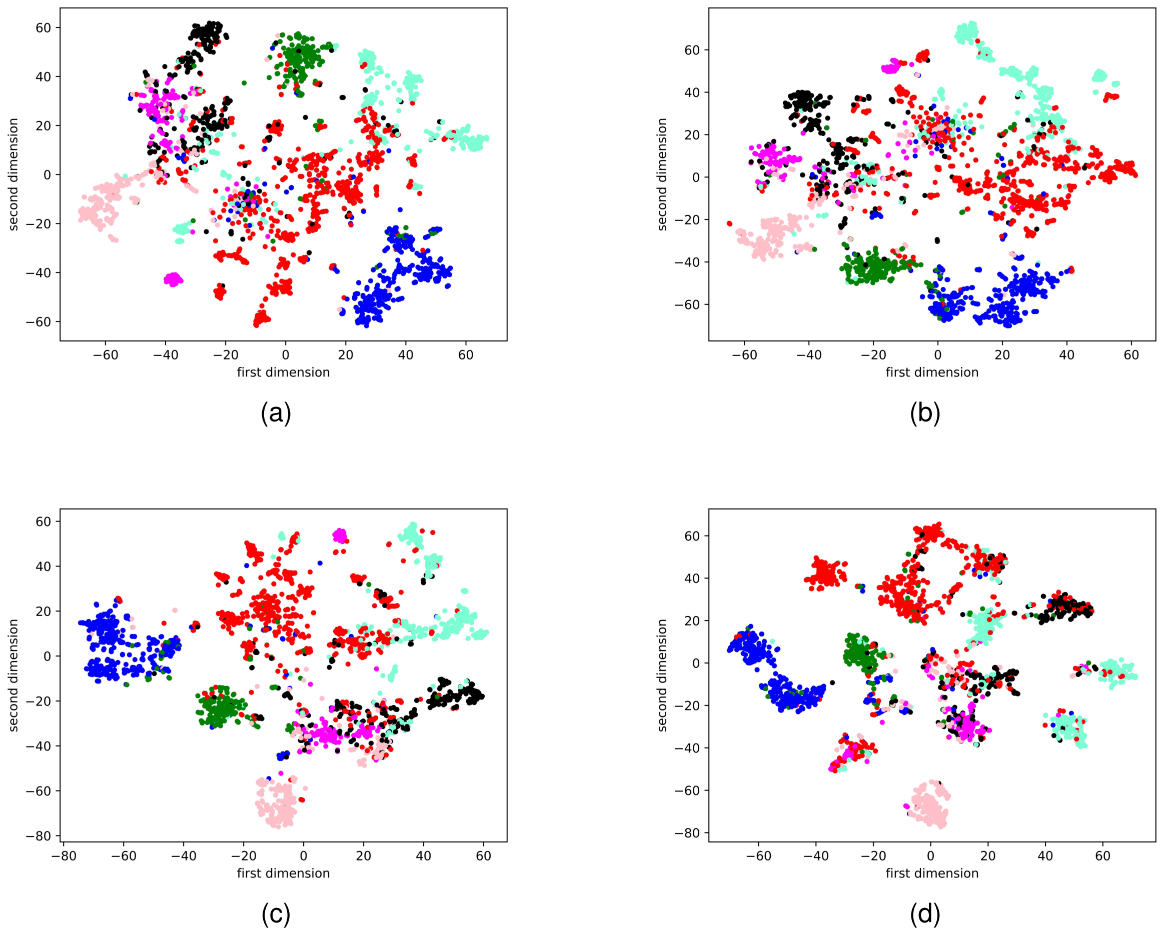


Fig. 4. TSNE embeddings of the latent feature vectors for the Cora dataset. Colors denote labels. (a) VAE. (b) VGAE. (c) Graphite. (d) Ours.

TABLE II
AREA UNDER THE ROC CURVE (AUC) FOR LINK PREDICTION WITH NODE FEATURES. THE BEST RESULTS ARE BOLDED. '-' DENOTE NO NODE FEATURES

| | Cora | Citeseer | Pubmed |
|---|---|---|---|
| SC | - | - | - |
| DW | - | - | - |
| GAE | 91.05 | 89.99 | 92.33 |
| VGAE | 91.37 | 90.05 | 84.64 |
| Graphite | 94.50 | 97.63 | 95.96 |
| Ours | **94.84** | **98.00** | **96.89** |

TABLE III
AVERAGE PRECISION (AP) FOR LINK PREDICTION WITH FEATURES. THE BEST RESULTS ARE BOLDED. '-' DENOTE NO NODE FEATURES.

| | Cora | Citeseer | Pubmed |
|---|---|---|---|
| SC | - | - | - |
| DW | - | - | - |
| GAE | 92.62 | 91.00 | 92.61 |
| VGAE | 92.26 | 91.00 | 86.02 |
| Graphite | **95.37** | 97.41 | 96.15 |
| Ours | **95.37** | **97.80** | **97.00** |

### *D. Results*

*1)* **With nodes features:** In the first place, we conduct our method on 3 datasets with node features. The results are illustrated in TABLE II and TABLE III. We can discover at the table, our approach outperforms others. Compared with VGAE, which uses simple Gaussian assumptions, our method is more efficient than it in Citeseer and Pubmed datasets with a large margin, which demonstrates our model has powerful expression ability for the complex graph. In particular, our method outperforms VGAE by 10% on the Pubmed dataset. The reason behind this is our method can capture the complex posterior and propagate uncertainty between nodes and their neighbourhoods, enhancing expressive ability.

*2)* **Without nodes features:** We execute our method on node featureless datasets. The results are shown in TABLE IV and TABLE V. As we can see in the table, our method achieved excellent results. Compared with GAE and VGAE, our method improves up to nearly 10% with AUC for Citeseer and Pubmed datasets. Besides, we demonstrate our approach on the four different datasets: NS, Router, USAir, Yeast. The results can be shown in Fig. 3. From Fig. 3 we can discover,

TABLE IV
AREA UNDER THE ROC CURVE (AUC) FOR LINK PREDICTION WITHOUT FEATURES. THE BEST RESULTS ARE BOLDED.

| | Cora | Citeseer | Pubmed |
|---|---|---|---|
| SC | 84.6 | 80.5 | 84.2 |
| DW | 83.1 | 80.5 | 84.4 |
| GAE | 85.92 | 78.34 | 85.36 |
| VGAE | 84.65 | 79.04 | 85.13 |
| Graphite | 88.58 | 85.57 | **95.36** |
| Ours | **91.14** | **91.30** | 95.01 |

TABLE V
AVERAGE PRECISION (AP) FOR LINK PREDICTION WITHOUT FEATURES. THE BEST RESULTS ARE BOLDED.

| | Cora | Citeseer | Pubmed |
|---|---|---|---|
| SC | 88.5 | 85.0 | 87.8 |
| DW | 85.0 | 83.6 | 84.1 |
| GAE | 88.74 | 83.4 | 88.65 |
| VGAE | 87.37 | 83.12 | 88.49 |
| Graphite | 90.09 | 85.14 | 94.32 |
| Ours | **92.84** | **92.89** | **96.00** |

our method obtained state-of-the-art results in four datasets. Especially, our method is more effective than LINE and SC in USAir dataset, 20% higher in the AUC and AP. For the Yeast dataset, our approach procures the best result, compared with LINE, our method is nearly seven percentage points higher than the line in AP, ten percentage points higher than the line in AUC. For NS, Router, and Yeast datasets, reference the dash and solid lines, compared to other methods, our method greatly improves the results in terms of both AUC and AP. The results prove that our method is more effective for both sparse and dense graphs.

*3)* **Visualization Display:** We visualize the embeddings learned by our method using 2-D TSNE for the Cora dataset. As is shown in Fig. 4. As we can see, Our method has a better clustering effect with more compactness between nodes. For example, for the red label, compared with other methods, our method makes clustering more compact. And the division between each class is more obvious.

### *E. Complexity Analysis*

*1)* **Computational Complexity Analysis***:* In our method, the highest complexity is the operation of inner products of potentially dense matrices $\mathbf{Z}$ in Equation (7) (i.e. $\mathbf{Z}\mathbf{Z}^T$). In order to computationally efficient, in our method, we apply a simplified graph propagation rule: $\mathbf{H}^{(l)} \leftarrow \zeta_l(\tilde{\mathbf{A}}\mathbf{H}^{(\mathbf{l-1})})$. Instead of directly compute inner product of $\mathbf{Z}$, we use associativity property of matrix multiplications. If $d_l$ and $d_{l-1}$ denote dimensional of layers of $\mathbf{H}^{(l)}$ and $\mathbf{H}^{(l-1)}$ respectively, the computational complexity is given by $\mathcal{O}(nkd_{l-1} + nd_{l-1}d_l)$, where $k$ is the dimension of the per-node latent vectors $\mathbf{Z}_i$ used to define $\tilde{\mathbf{A}}$, $n$ is the number of nodes.

*2)* **Time Cost***:* For the analysis of the real-world graph dataset Cora on a single NVIDIA GTX 2080Ti GPU node, it took 12.8, 20.8 , and 30.5 seconds for Graphite, VGAE, and Our method with 100 epochs, respectively. For the analysis of the small real-world graph dataset USAir on a same GPU node, it took 4.7, 4.8 , and 15.8 seconds for Graphite, VGAE, and Our method with 100 epochs, respectively.

## V. CONCLUSION

In this paper, we proposed a novel GNN model based on semi-implicit variational inference. Our method uses a hierarchical frame to construct the model, which can obtain a tractable posterior inference. Specifically, in the encoder step, differs from the traditional method, which assumes the posterior as Gaussian distribution, which restrict expressiveness of posterior, hence, in this paper, we design a hierarchical semi-implicit variational posterior to approximate the true posterior. Contributed to this variational posterior and the hierarchical architecture between GNN layers, our method can capture complex posterior and propagate uncertainly between nodes, which is very essential for information aggregation. We prove our method procure outperformance results on citation networks and four different scenarios datasets. In the future, we will be using more simple approaches to approximate complex posterior, such as normalization flow.

## REFERENCES


[1] Zhang, Ying, et al. "Towards end-to-end speech recognition with deep convolutional neural networks". *arXiv preprint arXiv*:1701.02720, 2017.

[2] Di Wu, Chao Wang, Yong Wu, Qi-Cong Wang and D.S.Huang, "Attention deep model with multi-scale deep supervision for person re-identification", *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 1, pp. 70-78. 2021.

[3] Wu, Yong, et al. "Person re-identification by multi-scale feature representation learning with random batch feature mask". *IEEE Transactions on Cognitive and Developmental Systems* (2020).

[4] Zhang, Qinhu, Lin Zhu, and De-Shuang Huang. "High-order convolutional neural network architecture for predicting DNA-protein binding sites". *IEEE/ACM transactions on computational biology and bioinformatics* 16.4, 2018: 1184-1192.

[5] Tang L, Liu H. "Relational learning via latent social dimensions"[C] *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2009: 817-826.

[6] Ying R, He R, Chen K, et al. "Graph convolutional neural networks for web-scale recommender systems"[C] *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018: 974-983.

[7] Yu W, Zheng C, Cheng W, et al. "Learning deep network representations with adversarially regularized autoencoders"[C] *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018: 2663-2671.

[8] Wu, Zonghan, et al. "A comprehensive survey on graph neural networks". *IEEE transactions on neural networks and learning systems* 32.1 (2020): 4-24.

[9] Li G, Li N, Chang F, et al. "Adaptive graph convolutional network with adversarial learning for skeleton-based action prediction"[J]. *IEEE Transactions on Cognitive and Developmental Systems*, 2021.

[10] Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks". *arXiv preprint arXiv*:1609.02907. 2016.

[11] Hamilton, William L., Rex Ying, and Jure Leskovec. "Inductive representation learning on large graphs". *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017.

[12] Veličković, Petar, et al. "Graph attention networks". *arXiv preprint arXiv*:1710.10903 (2017).

[13] Li, Zhi-Peng et al., "Hierarchical Graph Pooling with Self-Adaptive Cluster Aggregation", in *IEEE Transactions on Cognitive and Developmental Systems*, doi: 10.1109/TCDS.2021.3100883.

[14] Liao Y, Kodagoda S, Wang Y, et al. "Place classification with a graph regularized deep neural network"[J]. *IEEE Transactions on Cognitive and Developmental Systems*, 2016, 9(4): 304-315.

[15] Liu Y, Zhang X, Lin Y, et al. "Facial expression recognition via deep action units graph network based on psychological mechanism"[J]. *IEEE Transactions on Cognitive and Developmental Systems*, 2019, 12(2): 311-322.

[16] Zhang, Yingxue, et al. "Bayesian graph convolutional neural networks for semi-supervised classification". *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 33. No. 01. 2019.

[17] Pal, Soumyasundar, Florence Regol, and Mark Coates. "Bayesian graph convolutional neural networks using non-parametric graph learning". *arXiv preprint arXiv*:1910.12132 (2019).

[18] Hasanzadeh, Arman, et al. "Bayesian graph neural networks with adaptive connection sampling". *International conference on machine learning*. PMLR, 2020.

[19] Chandra, Rohitash, et al. "Bayesian graph convolutional neural networks via tempered MCMC". *arXiv preprint arXiv*:2104.08438 (2021).

[20] Ng, Yin Cheng, Nicolò Colombo, and Ricardo Silva. "Bayesian semi-supervised learning with graph gaussian processes". *arXiv preprint arXiv*:1809.04379 (2018).

[21] Chen, Gilad. "Latent Variable Models: An Introduction to Factor, Path, and Structural Equation Analysis". *Organizational Research Methods* 7.4 (2004): 475.

[22] Kipf, Thomas N., and Max Welling. "Variational graph auto-encoders". *arXiv preprint arXiv*:1611.07308 (2016).

[23] Li J, Ji C, Yan G, et al. "An ensemble net of convolutional auto-encoder and graph auto-encoder for auto-diagnosis"[J]. *IEEE Transactions on Cognitive and Developmental Systems*, 2020, 13(1): 189-199.

[24] Yin, Mingzhang, and Mingyuan Zhou. "Semi-implicit variational inference". *International Conference on Machine Learning. PMLR*, 2018.

[25] Kingma, Durk P., et al. "Improved variational inference with inverse autoregressive flow". *Advances in neural information processing systems* 29 (2016): 4743-4751.

[26] Papamakarios, George, Theo Pavlakou, and Iain Murray. "Masked autoregressive flow for density estimation". *arXiv preprint arXiv*:1705.07057 (2017).

[27] Rezende, Danilo, and Shakir Mohamed. "Variational inference with normalizing flows".*International conference on machine learning. PMLR*, 2015.

[28] Hasanzadeh, Arman, et al. "Semi-implicit graph variational auto-encoders". *arXiv preprint arXiv*:1908.07078 (2019).

[29] Grover, Aditya, Aaron Zweig, and Stefano Ermon. "Graphite: Iterative generative modeling of graphs". *International conference on machine learning. PMLR*, 2019.

[30] Sperduti, Alessandro, and Antonina Starita. "Supervised neural networks for the classification of structures". *IEEE Transactions on Neural Networks* 8.3 (1997): 714-735.

[31] Gori, Marco, Gabriele Monfardini, and Franco Scarselli. "A new model for learning in graph domains". Proceedings. 2005 *IEEE International Joint Conference on Neural Networks*, 2005.. Vol. 2. IEEE, 2005.

[32] Scarselli, Franco, et al. "The graph neural network model". *IEEE transactions on neural networks* 20.1 (2008): 61-80.

[33] Gallicchio, Claudio, and Alessio Micheli. "Graph echo state networks". *The 2010 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2010.

[34] Li, Yujia, et al. "Gated graph sequence neural networks". *arXiv preprint arXiv*:1511.05493 (2015).

[35] Dai, Hanjun, et al. "Learning steady-states of iterative algorithms over graphs". *International conference on machine learning. PMLR*, 2018.

[36] Bruna, Joan, et al. "Spectral networks and locally connected networks on graphs". *arXiv preprint arXiv*:1312.6203 (2013).

[37] Defferrard, Michaël, Xavier Bresson, and Pierre Vandergheynst. "Convolutional neural networks on graphs with fast localized spectral filtering". *Advances in neural information processing systems* 29 (2016): 3844-3852.

[38] Henaff, Mikael, Joan Bruna, and Yann LeCun. "Deep convolutional networks on graph-structured data". *arXiv preprint arXiv*:1506.05163 (2015).

[39] Weisfeiler, Boris, and Andrei Leman. "The reduction of a graph to canonical form and the algebra which appears therein". *NTI, Series* 2.9 (1968): 12-16.

[40] Morris, Christopher, et al. "Weisfeiler and leman go neural: Higher-order graph neural networks". *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. No. 01. 2019.

[41] Grohe, Martin, Pascal Schweitzer, and Daniel Wiebking. "Deep weisfeiler leman". *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA). Society for Industrial and Applied Mathematics*, 2021.

[42] Kingma, Diederik P., and Max Welling. "Auto-encoding variational bayes". *arXiv preprint arXiv*:1312.6114 (2013).

[43] Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra. "Stochastic backpropagation and approximate inference in deep generative models". *International con-*

This article has been accepted for publication in IEEE Transactions on Cognitive and Developmental Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCDS.2022.3193398



*ference on machine learning. PMLR*, 2014.

[44] Simonovsky, Martin, and Nikos Komodakis. “Graphvae: Towards generation of small graphs using variational autoencoders”. *International conference on artificial neural networks*. Springer, Cham, 2018.

[45] Salha, Guillaume, Romain Hennequin, and Michalis Vazirgiannis. “Keep it simple: Graph autoencoders without graph convolutional networks”. *arXiv preprint arXiv*:1910.00942 (2019).

[46] Liu, Qi, et al. “Constrained graph variational autoencoders for molecule design”. *arXiv preprint arXiv*:1805.09076 (2018).

[47] Ranganath, Rajesh, Dustin Tran, and David Blei. “Hierarchical variational models”. *International Conference on Machine Learning. PMLR*, 2016.

[48] Bishop, Christopher M., and Michael Tipping. “Variational relevance vector machines”. *arXiv preprint arXiv*:1301.3838 (2013).

[49] Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe. “Variational inference: A review for statisticians”. *Journal of the American statistical Association* 112.518 (2017): 859-877.

[50] Jordan, Michael I., et al. “An introduction to variational methods for graphical models”. *Machine learning* 37.2 (1999): 183-233.

[51] Wainwright, Martin J., and Michael Irwin Jordan. “Graphical models, exponential families, and variational inference”. *Now Publishers Inc*, 2008.

[52] Grover, Aditya, Aaron Zweig, and Stefano Ermon. “Graphite: Iterative generative modeling of graphs”. *International conference on machine learning. PMLR*, 2019.

[53] Sen, Prithviraj, et al. “Collective classification in network data”. *AI magazine* 29.3 (2008): 93-93.

[54] Abadi, Martín, et al. “Tensorflow: A system for large-scale machine learning”. *12th USENIX symposium on operating systems design and implementation* (OSDI 16). 2016.

[55] Newman, Mark EJ. “Finding community structure in networks using the eigenvectors of matrices”. *Physical review* E 74.3 (2006): 036104.

[56] Spring, Neil, et al. “Measuring ISP topologies with Rocketfuel”. *IEEE/ACM Transactions on networking* 12.1 (2004): 2-16.

[57] Vladimir Batagelj and Andrej Mrvar. http://vlado.fmf.uni-lj.si/pub/networks/data/, 2006.

[58] Von Mering, Christian, et al. “Comparative assessment of large-scale data sets of protein–protein interactions”. *Nature* 417.6887 (2002): 399-403.

[59] Tang, Lei, and Huan Liu. “Leveraging social media networks for classification”. *Data Mining and Knowledge Discovery* 23.3 (2011): 447-478.

[60] Perozzi, Bryan, Rami Al-Rfou, and Steven Skiena. “Deepwalk: Online learning of social representations”. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014.

**Hai-Long Su** is now a Ph.D. candidate with the Institute of Machine Learning and Systems Biology, School of Electronic and Information Engineering, Tongji University, Shanghai, China. He received his master’s degree from School of Computer, Electronics and Information, Guangxi University, China in 2019. And he received the B.S. degree from School of Computer Science and Engineering, SouthWest Minzu University, China, in 2016. His research interests are graph neural networks and computer vision.

**Zhi-Peng Li** is now a Ph.D. candidate with the Institute of Machine Learning and Systems Biology, School of Electronics and Information Engineering, Tongji University, China. He received the B.S. degree from Qilu University of Technology (Shandong Academy of Sciences), China, in 2017, and the M.S. degree from Qilu University of Technology (Shandong Academy of Sciences), China, in 2019. His research focuses on graph neural networks and computer vision.

**Xiao-Bo Zhu** is now a Ph.D. candidate with the Institute of Machine Learning and Systems Biology, School of Electronics and Information Engineering, Tongji University, China. He received the B.S. degree from Liaoning University of Technology, China, in 2016, and the M.S. degree from Ningxia University, China, in 2019. His research focuses on graph neural networks.

**Li-Na Yang** received the B.S. degree in computer engineering from Shijiazhuang Railway University, Shijiazhuang, China, in 2005; the M.Eng. degree in computer science from the University of Malaya, Kuala Lumpur, Malaysia, in 2011; and the Ph.D. degree from the University of Macau, Macau, China, in 2015. She is currently a Lecturer with the School of Computer, Electronics and Information, Guangxi University, Nanning, China, the ”academic backbone” of high-level talents from abroad. Her research interests focus on pattern recognition, machine learning, image processing and artificial intelligence.

**Valeriya Gribova** is with the Institute of Automation and Control Processes, Far Eastern Branch of Russian Academy of Sciences, Russia. She is an expert of Analytic Center in Government of Russian Federation, the Vice-President of Russian Association of Artificial Intelligence, the member of ITHEA, the member of the Expert Council of Russian Foundation for Basic Research and the expert of Russian Science Foundation. She has received the ITHEA award for Outstanding Achievement in the Field of Information Theory and Application (2009), the commendation certificate of the Far East Branch of the Russian Academy of Sciences (2001, 2006, 2012), and the commendation certificate of the Ministry of Education and Science of Primorsky Krai (2011). Her research interests include artificial intelligence and decision making, user interface, multiagent systems, program models and systems, specialized program models and systems.

**Vladimir F. Filaretov** was born in 1948. In 1966 he finished school with an honors (gold) medal and in 1973 graduated from Moscow State Technical University named after Bauman with honors with the specialty "Automatic systems". In 1976 Mr. Filaretov was awarded the degree of candidate of sciences (engineering) and in 1990 he was awarded the degree of doctor of sciences in the field of automatic control. In 1992 Mr. Filaretov was confirmed in professor's degree. In 1995 he was elected the member of an Russian and in 1996 the member of an International Engineering Academy. At present he is head of Department of Automation and Control of Far Eastern Federal University and Head of Robotic Laboratory of the Institute of Automatics and Control Process of Russian Academy of Sciences, President of Far Eastern Branch Russian Engineering Academy and Vice-president of Russian Engineering Academy. His researches are mainly directed at creation both industrial and underwater robots and manipulators and also other dynamic systems, allowing to automate technical devices and technological processes.

**Anthony Cohn** is Professor of Automated Reasoning in the School of Computing, at the University of Leeds. His current research interests range from theoretical work on spatial calculi (receiving a KR test-of-time classic paper award in 2020) and spatial ontologies, to cognitive vision, grounding language to vision, robotics, modelling spatial information in the hippocampus, and Decision Support Systems, particularly for the built environment. He is Editor-in-Chief Spatial Cognition and Computation and was previously Editor-in-chief of the AI journal. He is the recipient of the 2015 IJCAI Donald E Walker Distinguished Service Award which honours senior scientists in AI for contributions and service to the field during their careers, as well as the 2012 AAAI Distinguished Service Award. He is a Fellow of the Royal Academy of Engineering, the Alan Turing Institute in the UK, and is also a Fellow of AAAI, AISB, EurAI (formerly ECCAI; Founding Fellow), the BCS, and the IET. He is a Distinguished Visiting Professor at Tongji University and Qingdao University of Science and Technology, and an Adjunct Professor at Shandong University.

**De-Shuang Huang** received the B.Sc., M.Sc. and Ph.D. degrees all in electronic engineering from Institute of Electronic Engineering, Hefei, China, National Defense University of Science and Technology, Changsha, China and Xidian University, Xian, China, in 1986, 1989 and 1993, respectively. During 1993-1997 period, he was a postdoctoral research fellow respectively in Beijing Institute of Technology and in National Key Laboratory of Pattern Recognition, Chinese Academy of Sciences, Beijing, China. In Sept, 2000, he joined the Institute of Intelligent Machines, Chinese Academy of Sciences as the Recipient of "Hundred Talents Program of CAS". In September 2011, he entered into Tongji University as Chaired Professor. From Sept 2000 to Mar 2001, he worked as Research Associate in Hong Kong Polytechnic University. From Aug. to Sept. 2003, he visited the George Washington University as visiting professor, WashingtonDC, USA. From July to Dec 2004, he worked as the University Fellow in Hong Kong Baptist University. From March, 2005 to March, 2006, he worked as Research Fellow in Chinese University of Hong Kong. From March to July, 2006, he worked as visiting professor in Queen's University of Belfast, UK. In 2007, 2008, 2009, he worked as visiting professor in Inha University, Korea, respectively. At present, he is the director of Institute of Machines Learning and Systems Biology, Tongji University. Dr. Huang is currently Fellow of International Association of Pattern Recognition (IAPR Fellow), IEEE Fellow ,senior members of the IEEE and International Neural Networks Society. He has published over 180 journal papers. Also, in 1996, he published a book entitled "Systematic Theory of Neural Networks for Pattern Recognition" (in Chinese), which won the Second-Class Prize of the 8th Excellent High Technology Books of China, and in 2001 & 2009 another two books entitled "Intelligent Signal Processing Technique for High Resolution Radars" (in Chinese) and "The Study of Data Mining Methods for Gene Expression Profiles" (in Chinese), respectively. His current research interest includes bioinformatics, pattern recognition and machine learning.