

# DGE: Deep Generative Network Embedding Based on Commonality and Individuality

Sheng Zhou,<sup>1,2,3</sup> Xin Wang,<sup>4\*</sup> Jiajun Bu,<sup>1,2,3†</sup> Martin Ester,<sup>5</sup> Pinggang Yu,<sup>1,2</sup>  
Jiawei Chen,<sup>1</sup> Qihao Shi,<sup>1</sup> Can Wang<sup>1</sup>

<sup>1</sup>Zhejiang Provincial Key Laboratory of Service Robot, College of Computer Science, Zhejiang University, Hangzhou, China

<sup>2</sup>Alibaba-Zhejiang University Joint Institute of Frontier Technologies, Hangzhou, China

<sup>3</sup>Ningbo Research Institute, Zhejiang University, Ningbo 315100, China

<sup>4</sup>Tsinghua University, Beijing, China <sup>5</sup>Simon Fraser University, Canada

{zhousheng\_zju, bjj, pgyu, sleepyhunt, shiqihao321, wcan}@zju.edu.cn, xin\_wang@tsinghua.edu.cn, ester@cs.sfu.ca

## Abstract

Network embedding plays a crucial role in network analysis to provide effective representations for a variety of learning tasks. Existing attributed network embedding methods mainly focus on preserving the observed node attributes and network topology in the latent embedding space, with the assumption that nodes connected through edges will share similar attributes. However, our empirical analysis of real-world datasets shows that there exist both *commonality* and *individuality* between node attributes and network topology. On the one hand, similar nodes are expected to share similar attributes and have edges connecting them (commonality). On the other hand, each information source may maintain individual differences as well (individuality). Simultaneously capturing commonality and individuality is very challenging due to their exclusive nature and existing work fail to do so. In this paper, we propose a deep generative embedding (DGE) framework which simultaneously captures commonality and individuality between network topology and node attributes in a generative process. Stochastic gradient variational Bayesian (SGVB) optimization is employed to infer model parameters as well as the node embeddings. Extensive experiments on four real-world datasets show the superiority of our proposed DGE framework in various tasks including node classification and link prediction.

## Introduction

Networks are natural representations for a wide variety of real-life data, from social rating networks, author/paper citation networks to gene interaction networks. As more and more information becomes available, nodes in real-world networks are often associated with attributes, which is referred as attributed network (Wang et al. 2017b). Attributed network embedding is a powerful and increasingly popular approach to conduct comprehensive analysis on attributed networks. Tasks such as link prediction (Wang, Cui, and Zhu 2016) and node classification (Tang, Qu, and Mei 2015) all greatly benefit from effective latent node representations obtained by good network embedding approaches.

Most of existing attributed network embedding methods (Huang, Li, and Hu 2017b; 2017a; Liao et al. 2018; Li et al. 2017b) project the observed data into the latent embedding space while preserving properties including first/second order proximity (Tang et al. 2015), attribute based similarity (Liao et al. 2018) and global structural role (Zhang et al. 2018) etc. Although network topology and node attributes are explicitly considered in existing methods, they simply assume that adjacent nodes in a given network tend to share similar attributes which is referred as *commonality*, ignoring more complex relationships between network topology and node attributes. However, real-world network topology and node attributes may exhibit an opposite pattern where adjacent nodes have distinct attributes as well, which is referred as *individuality*. Consider a social network whose edges reflect the social connections between users and node attributes reflect the preferences of users, due to the difficulty of collecting comprehensive data or the fact that entities are not exposed to each other (Wang et al. 2016), users with similar preferences may be unknown to each other (i.e. unconnected). In bibliographic networks, similar phenomena can be observed. Although one paper will (intuitively) cite another paper with similar content, papers without citation relationships can also be similar due to limitations in the scale of the corpus and the number of citations.

The above observations demonstrate that there indeed exists individuality in addition to commonality in networks when depicting relationships among different nodes by network topology and node attributes. It is a challenging task to capture commonality and individuality simultaneously in latent embedding space since they are mutually exclusive and merging them by force may limit the capability of the learned representations. Some existing attributed network embedding methods take care of the commonality by concatenating the embeddings from both information sources (Li et al. 2017a) or directly preserving the properties of both information sources in the objective function (Huang, Li, and Hu 2017a), ignoring the individuality in each information source. Other methods (Kipf and Welling 2016) learn node embeddings by transforming the input attributes to preserve the network topology, which ignores the commonality between node attributes and network topology.

\*Corresponding Author

†Corresponding Author

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Recently, deep generative models (Tolstikhin et al. 2017; Kingma and Welling 2013; Goodfellow et al. 2014) have become an efficient way to analyze and understand networks through capturing the inner probabilistic distribution which can generate unobserved samples. Many variants of deep generative model have been extensively studied and applied in various representation learning tasks on image (Radford, Metz, and Chintala 2015) and text data (Zhong and Ghosh 2005). Though deep generative models have also been explored on network data, their applications in attributed network have received relatively less scrutiny.

Given the challenges in smoothly and losslessly merging the commonality and individuality into embedding space, we address this issue from an opposite perspective: instead of preserving node attributes and network topology in one single representation, we assume that node attributes and network topology can be generated from a latent representation. Thus the commonality and individuality can then be captured by a generative process. Our basic assumptions are as follows: (1) There exists a low-dimensional hidden space where each node has a density distribution. Both node attributes and network topology can be generated from the latent representations. Node with similar latent representations are expected to share similar attributes and have edges connecting them (commonality). (2) Node attributes and network topology are generated in an independent process with different probability distributions. The randomness in the probabilistic distributions and independent generation processes result in the existence of individual differences (individuality) within each of the two information sources.

We develop a stochastic gradient variational Bayesian optimization framework to obtain the model parameters efficiently. In general, the contribution of this paper can be summarized as follows:

1. We present commonality and individuality, two relationships between node attributes and network topology in attributed network embedding, and prove their existence through an empirical analysis.
2. We propose a deep generative embedding framework capable of simultaneously capturing commonality and individuality relations through modeling the generative processes of node attributes and network topology.
3. We conduct extensive experiments on four real-world datasets to demonstrate the effectiveness of our proposed DGE framework in several learning tasks.

## Related work

In this section, we will review the related studies in two areas, namely network embedding and deep generative models.

**Network Embedding** Network embedding aims at learning low-dimensional vector representation to facilitate a better understanding of semantic relationships among nodes. Among them, DeepWalk (Perozzi, Al-Rfou, and Skiena 2014) and Node2Vec (Grover and Leskovec 2016) employs truncated random walk to generate node sequences, which is treated as sentences in language models and fed into the skip-gram model to learn node embeddings. LINE

(Tang et al. 2015) is proposed for large scale network, which preserves both first-order and second-order proximities. GraRep (Cao, Lu, and Xu 2015) can be regarded as an extension of LINE which considers high-order proximity. SDNE (Wang, Cui, and Zhu 2016) incorporates graph structure into deep auto-encoder to preserve the highly non-linear first order and second order proximities.

All the above mentioned approaches, however, are limited to dealing with non-attributed networks. Attributed network analysis is then put forward due to the fact that networks are often associated with abundant content. For example, TADW (Yang et al. 2015) extends DeepWalk by assuming that each node is associated with rich texts and incorporate text features into the matrix factorization framework. AANE(Huang, Li, and Hu 2017a) is a distributed embedding approach that jointly learns node representations by decomposing attribute affinity matrix and penalizing the embedding difference between connected nodes with network lasso regularization. SNE (Yuan, Wu, and Xiang 2017) generates embeddings by leveraging an end-to-end neural network model to capture the complex interrelations between network structure and node attribute information. GraphSAGE (Hamilton, Ying, and Leskovec 2017) leverages node feature information and network topology to efficiently generate node embeddings for previously unseen data. PRRE (Zhou et al. 2018) considers the partial correlation between node attributes and network topology and preserved the re-ordered similarity to learn embedding. However, none of the above methods capture commonality and individuality between node attributes and network topology.

**Deep Generative Models** Recently, the deep generative models have attracted much attention and extended to the network embedding. Recent advances in Generative Adversarial Networks (GANs) (Goodfellow et al. 2014) have proven GANs as a powerful framework for learning complex data distributions. Some extensions have been proposed to apply GANs on network representation learning. In GraphGAN (Wang et al. 2017a), generative methods and discriminative methods are unified via adversarial training in a mini-max game. GANE (Hong, Li, and Wang 2018), ANE (Dai et al. 2017) and ARVGA (Pan et al. 2018) also explore GANs on network representation by preserving network structure with adversarial learning.

Another type of generative methods are the Variational Auto-encoder(VAE). In VGAE (Kipf and Welling 2016), a graph convolutional network encoder and inner product decoder is used to learn interpretable latent representations for undirected graphs, however, it only models the generation of network topology. In DVNE (Zhu et al. 2018), gaussian distributions are learned for each node to represent the uncertainty and similarity between distributions are measured in Wasserstein space so that transitivity of proximity in networks can be well preserved. In RINAE (Li et al. 2017a), the node attributes and network structure are concatenated as the feature representation of nodes. A variational auto-encoder is employed to model the generative process of the concatenated features. In CAN(Meng et al. 2019), two independent embedding are learned to model the generation of node attributes and network topology. However, none of ex-

isting generative method can capture the commonality and individuality in a unified framework.

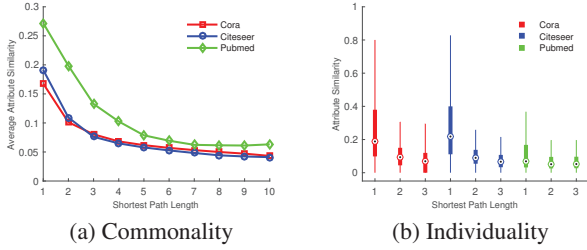


Figure 1: Commonality and individuality between node attributes and network topology.

### Case Study of Commonality and Individuality

In this section, we report the results of an empirical analysis of commonality and individuality between node attributes and network topology, using a Cora, a CiteSeer and a PubMed dataset.

Figure 1 shows the results of our empirical analysis. The statistics of the three datasets are presented in table 1. The X axis denotes the graph distance of the node pairs, the Y axis denotes the average attribute based similarity of the node pairs. In figure 2 (a), we can observe that with increasing graph distance (shortest-path length), the attribute based similarity decreases. In other words, nodes that are close in the network have higher average feature similarity than nodes far from each other in the network. This shows the commonality between two information source in describing the relationship between nodes. In figure 2 (b), each box represents the distribution of the attribute similarity of a group of node pairs with the given graph distance (shortest-path length). The central mark is the median, the edges of the box are the 25th and 75th percentiles, two lines (the whiskers) extend to the most extreme data points not considered outliers. We observe that there is a considerable number of pairs that are close to each other in the network but have relatively low attribute similarity. Comparing the adjacent bars in the figure, it can be seen that there are many node pairs that have smaller graph distance but smaller attribute similarity than other node pairs. This demonstrates the individuality between the two information sources in describing the relationships between nodes. To conclude, both node attributes and network topology are describing the relationships between nodes, there exists commonality and individuality between them.

### Deep Generative Network Embedding

In this section, we describe the proposed deep generative network embedding (DGE ) framework.

#### Notations and problem definition

An attributed network is defined as  $\mathbf{G} = \{\mathbf{V}, \mathbf{E}, \mathbf{X}\}$ , where  $\mathbf{V} = \{v_1, v_2, \dots, v_N\}$  denotes a set of nodes and  $N$  is the number of the nodes.  $\mathbf{E}$  is the set of edges between the

nodes,  $\mathbf{E}_{ij} = 1$  if there exists an edge between node  $v_i, v_j$ ,  $M = |\mathbf{E}|$  is the number of edges.  $\mathbf{X} \in R^{N \times D}$  is the node attributes matrix where  $x_i$  denotes the attributes of node  $v_i$ , and  $D$  is the dimension of node attributes.

**Definition 1 Generative Attributed Network Embedding**  
 Given an attributed network  $\mathbf{G} = \{\mathbf{V}, \mathbf{E}, \mathbf{X}\}$ , generative attributed network embedding aims at modeling the generation of node attributes and edges in the attributed network and represent each node  $v_i \in \mathbf{V}$  as a lower-dimensional vector  $h_i \in R^L$  so that the relationship between nodes can be represented in the vector space.  $L$  is the embedding dimension which satisfies  $L \ll N$  and  $L \ll D$ .

### The generation process

The generative framework is expected to uniformly model two aspects of the attributed network: (1) The observation including node attributes and network topology, (2) The commonality and individuality between node attributes and network topology.

In DGE framework, we consider both binary and continuous attributes. The attributes are assumed to be generated from Bernoulli distribution for binary attributes and Gaussian distribution for continuous attributes, the parameters of the probabilistic distribution are generated from node embedding with deep neural networks. We assume that edge is generated from a Bernoulli distribution parameterized by the similarity between embedding of the connected nodes. It is worth noting that the DGE framework can be expanded to weighted network and directed network by using different probabilistic distributions. We leave it in the future work. Above all, an observed attributed network  $\mathbf{G} = \{\mathbf{V}, \mathbf{E}, \mathbf{X}\}$  is assumed to be generated from the following process:

1. For each node  $v_i \in V$ , draw vector  $\mathbf{z}_i \sim \mathcal{N}(\mu_0, \sigma_0^2 \mathbf{I})$
2. For node with binary attributes
  - (a) Calculate expectation vector  $\mu_i$ 

$$\mu_i = f(\mathbf{z}_i; \theta) \tag{1}$$
  - (b) Draw attributes  $\mathbf{x}_i \sim Ber(\mu_i)$
3. For node with continuous attributes
  - (a) Calculate mean and variance
$$[\mu_i, \sigma_i^2] = f(\mathbf{z}_i; \theta) \tag{2}$$
  - (b) Draw attributes  $\mathbf{x}_i \sim \mathcal{N}(\mu_i, \sigma_i^2 \mathbf{I})$
4. For node pair  $v_i, v_j$ , draw the edge  $\mathbf{E}_{ij} \sim Ber(\sigma(\mathbf{z}_i^T \mathbf{z}_j))$

where  $\mathbf{z}_i$  is the embedding of node  $v_i$ ,  $\mathcal{N}(\mu, \sigma^2 \mathbf{I})$  is multivariate Gaussian distribution parameterized by  $\mu, \sigma^2$ ,  $\mathbf{I}$  is an identity matrix.  $f(\mathbf{z}; \theta)$  is a mapping function with non-linear activation parameterized by  $\theta$ .  $Ber(\cdot)$  is multivariate Bernoulli distribution,  $\sigma$  is the sigmoid function, respectively.

According to the generation process and the independent assumption, the joint probability  $p(X, E, Z)$  can be factorized as:

$$p(X, E, Z) = p(X|Z)p(E|Z)p(Z) \tag{3}$$

$$p(Z) = \mathcal{N}(\mu_0, \sigma_0^2 \mathbf{I}) \tag{4}$$

$$p(X|Z) = Ber(X|\mu_i) \text{ or } \mathcal{N}(X|\mu_i, \sigma_\mu^2 \mathbf{I}) \tag{5}$$

$$p(E|Z) = Ber(E|\sigma(\mathbf{Z}^T \mathbf{Z})) \tag{6}$$

## Model inference

Following the variational principle (Wainwright, Jordan, and others 2008), we derive a lower bound on the marginal likelihood of the model which forms our objective function. Our purpose is to maximize the log-likelihood of the given network topology and node attributes, which can be written as:

$$\begin{aligned} \log p(X, E) &\geq E_{q_\phi}[\log p(X, E, Z) - \log q_\phi(Z|X, E)] \\ &= E_{q_\phi}[\log p_\theta(X, E|Z) - KL(q_\phi(Z|X, E)||p_\theta(Z))] \\ &= \mathcal{L}_{ELBO}(X, E) \end{aligned} \quad (7)$$

where  $\mathcal{L}_{ELBO}(X, E)$  is the evidence lower bound (ELBO). In DGE model, we use deep neural network with non-linear function to model the variational posterior  $q_\phi(Z|X, E)$  and infer the parameter  $\theta, \phi$  to approximate  $\mathcal{L}_{ELBO}(X, E)$ . The true posterior is assumed to take on an approximate Gaussian form with an approximately diagonal covariance. The variational posterior  $q_\phi(Z|X, E)$  is assumed to be multivariate Gaussian distribution with a diagonal covariance structure.

$$\log q_\phi(Z|X, E) = \sum_i \log \mathcal{N}(\mathbf{z}_i; \tilde{\mu}_i, \tilde{\sigma}^2 \mathbf{I}) \quad (8)$$

where  $\tilde{\mu}_i$  and  $\tilde{\sigma}^2$  are the mean and variance of the variational posterior. The second term of ELBO can be rewrote with simply Gaussian transformation as:

$$\begin{aligned} &KL(q_\phi(Z|X, E)||p_\theta(Z)) \\ &= \frac{1}{2} \sum_i \sum_{j=1}^L (1 + \log(\tilde{\sigma}_i^j)^2 - (\tilde{\mu}_i^j)^2 - (\tilde{\sigma}_i^j)^2) \end{aligned} \quad (9)$$

Computation of the exact posterior distribution is intractable due to the nonlinear, non-conjugate dependencies between the random variables. We apply *reparameterization trick* with Monte-Carlo sampling to infer the parameters. As we have discussed, DGE can be applied on both binary and continuous attributes. In the binary attribute situation, the first term can be written as:

$$\begin{aligned} &E_{q(Z|X, E)}[\log p(X, E|Z)] \\ &\simeq \frac{1}{S} \sum_{s=1}^S (\sum_i \log p(x_i|z_i^s) + \sum_{ij} \log p(E_{ij}|z_i^s, z_j^s)) \\ &= \frac{1}{S} \sum_{s=1}^S (\sum_i \sum_{d=1}^D x_{id} \log \mu_{id} + (1 - x_{id}) \log(1 - \mu_{id}) \\ &+ \sum_{ij} E_{ij} \cdot \log \sigma(z_i^{sT} z_j^s) + (1 - E_{ij}) \log(1 - \sigma(z_i^{sT} z_j^s))) \\ &\text{where } z_i^s = \tilde{\mu}_i + \tilde{\sigma}_i \odot \epsilon^s \text{ and } \epsilon^s \sim \mathcal{N}(0, \mathbf{I}) \end{aligned} \quad (10)$$

where  $S$  is the number of Monte-Carlo sampling,  $D$  is the dimension of node attributes. When the attribute is continuous, the first term of ELBO in Eq. (7) can be written as

$$\begin{aligned} &E_{q(Z|X, E)}[\log p(X, E|Z)] \\ &\simeq \frac{1}{S} \sum_{s=1}^S (\sum_i \sum_{d=1}^D -\log(\sqrt{2\pi}\sigma_{i,d}) + \frac{(x_{i,d} - \mu_{i,d})^2}{2\sigma_{i,d}^2} \\ &+ \sum_{ij} E_{ij} \cdot \log \sigma(z_i^{sT} z_j^s) + (1 - E_{ij}) \log(1 - \sigma(z_i^{sT} z_j^s))) \end{aligned} \quad (11)$$

For binary attributes, the loss function can be written as:

$$\begin{aligned} \mathcal{J}_r &= -\frac{1}{S} \sum_{s=1}^S (\sum_i \sum_{d=1}^D x_{id} \log \mu_{id} + (1 - x_{id}) \log(1 - \\ &\mu_{id}) + \sum_{ij} E_{ij} \log \sigma(z_i^{sT} z_j^s) + (1 - E_{ij}) \log(1 - \sigma(z_i^{sT} \\ &z_j^s))) - \frac{1}{2} \sum_i \sum_{j=1}^L (1 + \log(\tilde{\sigma}_i^j)^2 - (\tilde{\mu}_i^j)^2 - (\tilde{\sigma}_i^j)^2) \\ &\text{where } z_i^s = \tilde{\mu}_i + \tilde{\sigma}_i \odot \epsilon^s \text{ and } \epsilon^s \sim \mathcal{N}(0, \mathbf{I}) \end{aligned} \quad (12)$$

Loss function for continuous attributes can be derived by replacing the corresponding term in Equation 12 with Equation 11. The encoder and decoder are designed with simple multi-layered perceptrons (MLPs) and the detailed setting can be found in our open-sourced code.

## Explanation of commonality and individuality captured by DGE model

According to the generation process, both node attributes and network topology are generated from latent vector  $\mathbf{z}$ , the commonality is easy to understand as nodes with similar embedding has higher probability to form edges and share similar attributes. The individuality can be explained from the following aspects: First, node attributes and edges are generated from probability distribution, which means for nodes with similar or even same latent embedding, they may also have different attributes or do not have edges between them. Second, the generation of node attributes and network topology are independent, which brings more inconsistency between the generated results. We use the sum of the variance in the generation process to denote the individuality between node attributes and network topology.

Fig 2 illustrates two cases that denotes the individuality between node attributes and network topology. Each curve denotes the Gaussian probability of observing continuous attributes and dots on the curve represents the attribute of an observed node, the mean of distribution is generated from the embedding. For nodes with similar mean and embedding, the probabilistic of generating edge is higher which denotes the commonality. In the first case(subfigure (a)), the mean of two distribution are different which means there may be no edge between the nodes, however, the attributes may be similar such as the blue and red dot. In the second case(subfigure (b)), the mean of two distribution are close which means there may exist edge with high possibility,



however, the attributed may be dissimilar such as the blue and red dot.

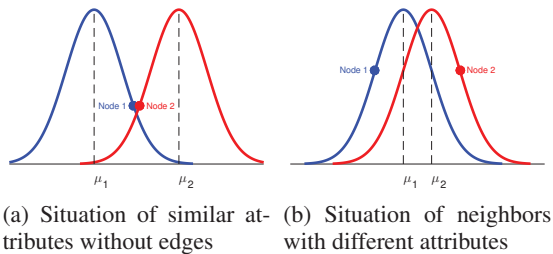


Figure 2: Example of individuality between node attributes and network topology.

---

**Algorithm 1** Framework of deep generative network embedding.

---

**Input:** Attributed network  $\mathbf{G} = \{\mathbf{V}, \mathbf{E}, \mathbf{X}\}$ , embedding dimension  $L$ ;

**Output:** Node embedding  $\mathbf{H} \in R^{N \times L}$

- 1: Initialize the deep neural network parameters
  - 2: **while** DGE not converged **do**
  - 3:   Sample batch nodes for minibatch and extract their attributes and edges
  - 4:   Encode:  $z \in q_\phi(Z|X, E)$  and reparameterize
  - 5:   Decode: Generate node attributes and edges
  - 6:   Update  $\theta, \phi, p_\theta(X, E|Z), q_\phi(Z|X, E)$  via gradient descent on the ELBO in Eq. (7)
  - 7: **end while**
  - 8: **return**  $\mathbf{H}$
- 

### Complexity Analysis

The overall algorithmic complexity of a single joint update of the parameters  $\{\theta, \phi\}$  is  $C = MSC_{MLP}$  where  $M$  is the minibatch size,  $S$  is the number of samples of the random variate  $\epsilon$ , and  $C_{MLP}$  is the cost of an evaluation of the MLPs in the conditional distributions  $p_\theta(X, E|Z)$  and  $q_\phi(Z|X, E)$ . The cost  $C_{MLP}$  is of the form  $O(KD^2)$  where  $K$  is the total number of layers and  $D$  is the average dimension of the layers of the MLPs in the model. These complexities make this approach extremely appealing, since they are no more expensive than alternative approaches based on auto-encoder or neural models.

### Experimental Evaluation

In this section, we conduct extensive experiments on real-world datasets to demonstrate the effectiveness of proposed DGE model on tasks including node classification and link prediction. We also prove DGE model can capture the individuality between node attributes and network topology.

#### Datasets

We conduct experiments on three paper citation networks and one social network with different scale of nodes. Table 1 illustrates the details of datasets used in our experiment.

**Citation datasets** We select three datasets of bibliographic network namely Citeseer, Cora and Pubmed. Each dataset contains bibliographic network which included scientific publications classified into corresponding research area. Links represent citations among publications. Each publication in Cora and Citeseer dataset is described by a 0/1-valued word vector indicating the absence/presence of the corresponding word from a dictionary with unique words. The attributes in Pubmed dataset are bag-of-words representation of a document which is continuous valued.

**Social network** BlogCatalog is a social network for users to publish blogs where users are classified based on blog category, edges represent the friendship connection. Each blog is associated with keywords for brief description which is treated as node attributes.

Table 1: Summary of the datasets used in our experiments.

Dataset	#Nodes	#Features	#Classes
Citeseer	3327	3703	6
Cora	2708	1433	7
BlogCatalog	5196	8189	6
Pubmed	19717	500	3

#### Baselines

We compare the DGE framework with several state-of-the-art (attributed) network embedding methods, which can be divided into the following groups.

**Attribute Only.** This group of algorithms only consider node attributes and transform the attribute vectors to low dimensional representations. We select SVM (Hearst et al. 1998) and AutoEncoder in this group.

**Network Only.** This group of algorithms leverage network topology information while ignores node attributes. Representative works include DeepWalk (Perozzi, Al-Rfou, and Skiena 2014) and LINE (Tang et al. 2015), which use truncated random walks to generate node sequences and employ skip-gram model for node representation learning.

**Attribute and Network.** This group of algorithms consider both node attributes and network topology together in learning node embedding. We select AANE (Yang et al. 2015), SNE (Yuan, Wu, and Xiang 2017), GraphSAGE (Hamilton, Ying, and Leskovec 2017), TADW (Yang et al. 2015), PRRE(Zhou et al. 2018), CAN(Meng et al. 2019), VGAE(Kipf and Welling 2016) and RIANE(Li et al. 2017a).

The source code and detailed settings of DGE model can be found in <https://github.com/zhoushengisnoob/DGE>

#### Node classification

Node classification has been widely used in literature to evaluate network embedding. In this subsection, we report the experimental results of node classification on labeled datasets. Following the experimental procedure of existing works (Perozzi, Al-Rfou, and Skiena 2014), we first train models on the training sets to obtain node representations. Then we randomly sample 30% labeled nodes to train a SVM classifier and the rest of the nodes are used to test the

Table 2: Experimental results of node classification on four datasets.

Group	Algorithm	Citeseer		Cora		Blogcatalog		Pubmed	
		Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Attribute Only	SVM	0.670	0.635	0.700	0.673	0.865	0.863	0.807	0.798
	AutoEncoder	0.651	0.616	0.659	0.630	0.859	0.856	0.787	0.776
Network Only	DeepWalk	0.556	0.511	0.775	0.761	0.613	0.606	0.743	0.716
	LINE	0.560	0.531	0.777	0.770	0.667	0.663	0.727	0.703
Attribute + Network	AANE	0.579	0.541	0.751	0.715	0.856	0.855	0.820	0.813
	SNE	0.631	0.596	0.802	0.783	0.814	0.812	0.822	0.802
	GraphSAGE	0.622	0.587	0.792	0.778	0.829	0.820	0.814	0.804
	TADW	0.720	0.656	0.830	0.816	0.887	0.884	0.833	0.819
	PRRE	0.725	0.665	0.852	0.828	0.918	0.917	0.840	0.823
	CAN	0.691	0.640	0.839	0.830	0.742	0.718	0.821	0.818
	RINAE	0.562	0.516	0.581	0.532	0.771	0.765	0.753	0.752
	VGAE	0.705	0.661	0.854	0.851	0.792	0.788	0.818	0.810
DGE variants	DGE -A	0.712	0.606	0.739	0.702	0.809	0.802	0.806	0.796
	DGE -N	0.555	0.498	0.693	0.682	0.649	0.644	0.767	0.759
	DGE	0.727	0.668	0.858	0.853	0.931	0.929	0.846	0.846

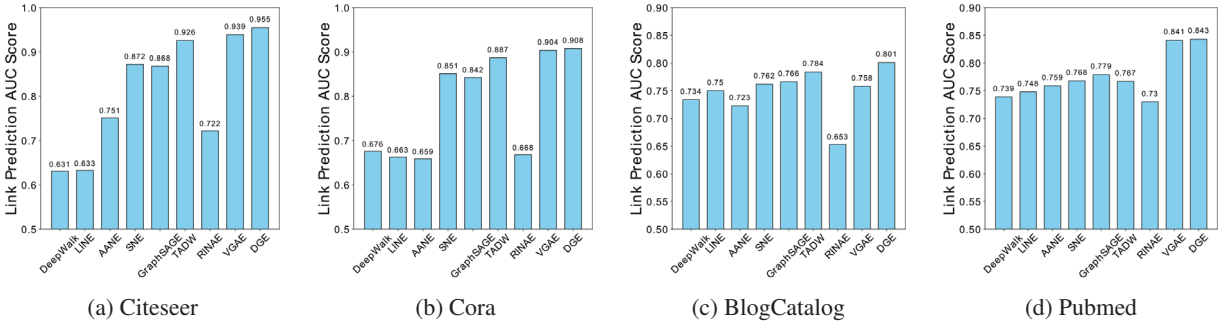


Figure 3: Link prediction results on four real world datasets.

model. We repeat this process 10 times and report the average performance in terms of both Micro-F1 and Macro-F1 score. The detailed results are shown in Table 2. To summarize, we have the following observations:

(1) The basic observation is that our proposed DGE framework achieves better results on four datasets compared with baseline methods and variants. This shows the effectiveness of our proposed DGE framework in modeling node attributes and network topology. By comparing DGE with the variants, we can further infer the advantage of capturing the commonality and individuality between node attributes and network topology.

(2) Compared with the attributed only group methods, we can find that DGE-A achieves similar performance. However, compared with the network only group methods, we find that the variant DGE-N achieves worse performance. This is explainable since only modeling the first-order neighborhood will limit the information captured by the model and ignore the relationship with node attributes.

(3) Compared with VGAE, CAN and RINAE, we can find that DGE achieves better performance. Although they all use variational auto-encoder structure, VGAE only preserves the network topology while ignores the reconstruction of node attributes, it can not capture either commonal-

ity or individuality. CAN learns two independent embedding for each node and can not capture the commonality. RINAE directly concatenates the attributes and adjacent vectors as feature representation of nodes. Such concatenation ignore the fact that node attributes and network topology are heterogeneous information. The improvement over these variational auto-encoder based methods indicates the advantage of modeling both network topology and node attributes in a unified framework while capturing the commonality and individuality.

### Network Reconstruction

The most primal objective for network embedding is to reconstruct the given network, as a good network embedding method should ensure that the learned embedding can preserve the original network structure (Zhu et al. 2018). Following the same experimental procedure in many existing works (Grover and Leskovec 2016; Yang et al. 2015), we randomly hold out 50% of the existing links as positive instances in test set and randomly sample the same amount of non-existing links as negative instances. The residual network is used to train the network embedding models. After obtaining the latent representations of each node, we evaluate link prediction in the labeled edge dataset. Specifically,

we rank both positive and negative instances according to the cosine similarity function. To judge the ranking quality, we employ the Area Under Curve (AUC) score to evaluate the ranking list and a higher value indicates a better performance. Figure 3 illustrates the results of link prediction on four real-world datasets. Based on the experimental results, we have the following observations:

(1) The basic observation is that DGE model achieve better performance compared with baseline methods in four real world datasets. This further indicates the superior of DGE framework in capturing the similarity between nodes by embedding nodes into low dimensional space.

(2) In citeseer and cora dataset, algorithms in network only group achieve poor performance compared with methods considering both node attributes and network topology. In Blogcatalog and pubmed dataset, the two group of algorithms achieve similar performance. This is explainable since the network of citeseer and cora is more sparse than BlogCatalog and Pubmed. This further indicates the necessary of combining node attributes and network topology when network is sparse.

(3) Among the four datasets, the RINAE method gets poor performance. We analysis the results and find that in large scale network, the adjacent vector is usually sparse, directly concatenating node attributes and adjacent may miss the individuality of each information source.

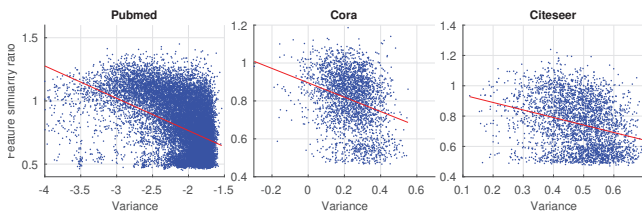


Figure 4: Relationship between the feature similarity ratio and variance.

### Commonality and individuality captured by DGE

In this section, we conduct experiments to prove that DGE model can actually capture the commonality and individuality. We calculate two average feature similarity: between nodes and its neighborhoods as well as between nodes and all the other nodes in the network. Higher ratio between two average feature similarity denotes nodes close in network are more similar in attributes than other nodes, which refers to higher commonality and lower individuality. As we have discussed, DGE model captures the individuality by the variance of probabilistic distribution in the generative process. We use the logarithmic form of variance to denote how much individuality DGE model captured.

Figure 4 illustrates the results of feature similarity ratio w.r.t. variance in DGE model. Each blue dot denotes a node and the red line is the straight line fitting of the blue dots. We can observe obvious inverse relationship between variance and feature similarity ratio, nodes with lower ratio (higher individuality) will have larger variance. This shows the vari-

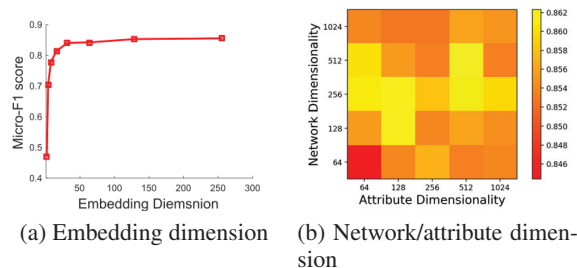


Figure 5: Parameter tuning of DGE in Cora dataset.

ance in DGE model can actually capture the individuality between node attributes and network topology.

**Hyper Parameter Tuning** For network embedding methods, an important parameter to tune is the dimension of learned embedding. Also, in DGE model, another important parameter is the dimension of embedding based on single input feature and adjacent vectors. We vary the dimension from 2 to 256 and examine how the performance changes with respect to the dimensions. We present the tuning of dimension of network embedding in Figure 5-(a) and dimension of single input based embedding in Figure 5-(b).

Figure 5(b) illustrates the performance w.r.t. the dimensionality  $L_A$  of attribute based embedding and  $L_N$  of network based embedding. The basic observation is that the performance is not sensitive to the dimensionality as the F1-score changes a little with the change of dimensionality of both network topology based embedding and node attribute based embedding.

## Conclusion

In this paper, we have proposed the DGE framework to utilize the deep generative framework to model the commonality and individuality between node attributes and network structure. Based on the statistic analysis on real world attributed network, we use a deep generative model to capture such relationships and learn node embedding. Meanwhile, extensive experiments are conducted on real-world datasets which shows the superior of the proposed DGE framework.

## Acknowledgement

This work is supported by Alibaba-Zhejiang University Joint Institute of Frontier Technologies, The National Key R&D Program of China (No. 2018YFC2002603, 2018YFB1403202), Zhejiang Provincial Natural Science Foundation of China (No. LZ13F020001), the National Natural Science Foundation of China (No. 61972349, 61173185, 61173186) and the National Key Technology R&D Program of China (No. 2012BAI34B01, 2014BAK15B02). National Natural Science Foundation of China (Grant No: U1866602) and National Key Research and Development Project (Grant No: 2018AAA0101503). China Postdoctoral Science Foundation No. BX201700136.

## References

- Cao, S.; Lu, W.; and Xu, Q. 2015. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM international conference on information and knowledge management*.
- Dai, Q.; Li, Q.; Tang, J.; and Wang, D. 2017. Adversarial network embedding. *arXiv preprint arXiv:1711.07838*.
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*.
- Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*.
- Hearst, M. A.; Dumais, S. T.; Osuna, E.; Platt, J.; and Scholkopf, B. 1998. Support vector machines. *IEEE Intelligent Systems and their applications*.
- Hong, H.; Li, X.; and Wang, M. 2018. Gane: A generative adversarial network embedding. *arXiv preprint arXiv:1805.07324*.
- Huang, X.; Li, J.; and Hu, X. 2017a. Accelerated attributed network embedding. In *Proceedings of the 2017 SIAM International Conference on Data Mining*.
- Huang, X.; Li, J.; and Hu, X. 2017b. Label informed attributed network embedding. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*.
- Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.
- Kipf, T. N., and Welling, M. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*.
- Li, H.; Wang, H.; Yang, Z.; and Liu, H. 2017a. Effective representing of information network by variational autoencoder. In *IJCAI*.
- Li, J.; Dani, H.; Hu, X.; Tang, J.; Chang, Y.; and Liu, H. 2017b. Attributed network embedding for learning in a dynamic environment. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*.
- Liao, L.; He, X.; Zhang, H.; and Chua, T.-S. 2018. Attributed social network embedding. *IEEE Transactions on Knowledge and Data Engineering*.
- Meng, Z.; Liang, S.; Bao, H.; and Zhang, X. 2019. Co-embedding attributed networks. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*.
- Pan, S.; Hu, R.; Long, G.; Jiang, J.; Yao, L.; and Zhang, C. 2018. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407*.
- Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.
- Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*.
- Tang, J.; Qu, M.; and Mei, Q. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *SIGKDD*.
- Tolstikhin, I.; Bousquet, O.; Gelly, S.; and Schoelkopf, B. 2017. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*.
- Wainwright, M. J.; Jordan, M. I.; et al. 2008. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*.
- Wang, X.; Lu, W.; Ester, M.; Wang, C.; and Chen, C. 2016. Social recommendation with strong and weak ties. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*.
- Wang, H.; Wang, J.; Wang, J.; Zhao, M.; Zhang, W.; Zhang, F.; Xie, X.; and Guo, M. 2017a. Graphgan: Graph representation learning with generative adversarial nets. *arXiv preprint arXiv:1711.08267*.
- Wang, S.; Aggarwal, C.; Tang, J.; and Liu, H. 2017b. Attributed signed network embedding. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*.
- Wang, D.; Cui, P.; and Zhu, W. 2016. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Yang, C.; Liu, Z.; Zhao, D.; Sun, M.; and Chang, E. Y. 2015. Network representation learning with rich text information. In *IJCAI*.
- Yuan, S.; Wu, X.; and Xiang, Y. 2017. Sne: signed network embedding. In *Pacific-Asia conference on knowledge discovery and data mining*.
- Zhang, Z.; Cui, P.; Wang, X.; Pei, J.; Yao, X.; and Zhu, W. 2018. Arbitrary-order proximity preserved network embedding. In *SIGKDD*.
- Zhong, S., and Ghosh, J. 2005. Generative model-based document clustering: a comparative study. *Knowledge and Information Systems*.
- Zhou, S.; Yang, H.; Wang, X.; Bu, J.; Ester, M.; Yu, P.; Zhang, J.; and Wang, C. 2018. Prre: Personalized relation ranking embedding for attributed networks. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*.
- Zhu, D.; Cui, P.; Wang, D.; and Zhu, W. 2018. Deep variational network embedding in wasserstein space. In *SIGKDD*.