



Embedding Networks with Edge Attributes

Palash Goyal

USC Information Sciences Institute
Marina Del Rey, CA
goyal@isi.edu

Emilio Ferrara

USC Information Sciences Institute
Marina Del Rey, CA
ferrarae@isi.edu

Homa Hosseinmardi

USC Information Sciences Institute
Marina Del Rey, CA
homahoss@isi.edu

Aram Galstyan

USC Information Sciences Institute
Marina Del Rey, CA
galstyan@isi.edu

ABSTRACT

Predicting links in information networks requires deep understanding and careful modeling of network structure. Network embedding, which aims to learn low-dimensional representations of nodes, has been used successfully for the task of link prediction in the past few decades. Existing methods utilize the observed edges in the network to model the interactions between nodes and learn representations which explain the behavior. In addition to the presence of edges, networks often have information which can be used to improve the embedding. For example, in author collaboration networks, the bag of words representing the abstract of co-authored paper can be used as edge attributes. In this paper, we propose a novel approach, which uses the edges and their associated labels to learn node embeddings. Our model jointly optimizes higher order node neighborhood, social roles and edge attributes reconstruction error using deep architecture which can model highly non-linear interactions. We demonstrate the efficacy of our model over existing state-of-the-art methods on two real world data sets. We observe that such attributes can improve the quality of embedding and yield better performance in link prediction.

KEYWORDS

Graph Embedding, Deep Learning, Network Representation

ACM Reference Format:

Palash Goyal, Homa Hosseinmardi, Emilio Ferrara, and Aram Galstyan. 2018. Embedding Networks with Edge Attributes. In *HT '18: 29th ACM Conference on Hypertext and Social Media, July 9–12, 2018, Baltimore, MD, USA*. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3209542.3209571>

1 INTRODUCTION

Many real world problems can be formulated in terms of networks. For example, friendship in online community can be modeled as a friendship network [7] in which users are nodes and strength

of friendship defines the weights of the links. Similarly, in biology, interactions of proteins can be modeled as a network and the connections in the network can be used to identify genes [28]. Other types of network include author collaboration network [8], router network [3] and language networks [14]. By nature, information networks are dynamic and constantly evolve. Understanding their evolution and predicting unforeseen links is a widely studied problem in this domain [19]. Over the past few decades, several techniques have been proposed to solve the link prediction problem. Common approaches include computing similarity between pairs of disconnected nodes using existing links [1, 15]. Some other methods use Bayesian models to learn inherent structure of the network [20]. Another class of models involve compute low-dimensional representation of each node and computing similarity between nodes in this space. Such methods, also referred as network embedding, have gained popularity in recent years [2, 5, 10, 22, 24, 27].

Network embedding methods studied so far learn latent representation from the links between the nodes [2, 5, 9, 22, 24, 27]. They define an objective function to preserve structure of the network characterized by certain properties and optimize it to obtain the embedding. Although many methods have been proposed to learn network embeddings from observed links, little attention has been given to utilize the metadata available with the network. For example, networks often have node and edge attributes which can be useful in predicting future links. Recently, some approaches have been proposed to incorporate node attributes in learning the representation [6, 12, 13]. However, these approaches do not consider edge attributes which can provide additional information about the network. For example, in collaboration networks in which authors are the nodes and edges represent presence of co-authored papers, the content of a co-authored paper can be used to characterize the interaction between the author nodes.

This work aims to bridge this gap and learn node representations capturing network structure and edge attributes in a unified manner. To this aim, we introduce *Edge Label Aware Network Embedding (ELAINE)* which uses a coupled deep autoencoder along with an edge attribute decoder to jointly optimize the reconstruction loss of network structure and edge attribute reconstruction. Our model is thus capable to learning non-linear manifold from the network. Furthermore, we input node similarity using random walk simulations and social roles metrics thus preserving higher order interactions between nodes and network roles. We evaluate our model on the task of link prediction. We show through our experiments that our

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

HT '18, July 9–12, 2018, Baltimore, MD, USA

© 2018 Copyright held by the owner/author(s). Publication rights licensed to Association for Computing Machinery.

ACM ISBN 978-1-4503-5427-1/18/07...\$15.00

<https://doi.org/10.1145/3209542.3209571>

model, ELAINE, effectively uses edge attributes and outperforms state-of-the-art embedding approaches. We show our results on two real world networks - an author collaboration network and an email communication network. We use meta data from publication abstracts and email content to inform our model.

Overall, we make three contributions. We propose ELAINE, a deep learning model which captures network structure and edge attributes. We show that edge attributes can help improve accuracy for the task of link prediction. Finally, we extend the deep architecture to preserve higher order proximity and social roles.

The rest of the paper is organized as follows. In Section 2, we provide the definitions required to understand the problem and models discussed next. We introduce our model in Section 3, and then describe our experimental setup and obtained results (Sections 4 and 5). Finally, in Section 6 we draw our conclusions and discuss potential applications and future research directions.

2 PROBLEM STATEMENT

We denote a weighted graph as $G(V, E)$ where V is the vertex set and E is the edge set. The weighted adjacency matrix of G is denoted by A . If $(i, j) \in E$, we have $A_{ij} > 0$ denoting the weight of edge (i, j) ; otherwise we have $A_{ij} = 0$. We use $\mathbf{a}_i = [A_{i,1}, \dots, A_{i,n}]$ to denote the i -th row of the adjacency matrix. We use $E^a \in \mathbb{R}^{m \times p}$ to denote the edge attribute matrix and $\mathbf{e}_{ij}^a = [e_{ij1}^a, \dots, e_{ijp}^a]$ to denote the attributes of edge (i, j) , where p is the number of edge attributes.

We define our problem as follows: Given a graph $G = (V, E)$ and associated edge attributes E^a , we aim to represent each node u in a low-dimensional vector space \mathbf{y}_u by learning a mapping $f: \{V, E^a\} \rightarrow \mathbb{R}^d$, namely $\mathbf{y}_v = f(v, E^a) \forall v \in V$. We require that $d \ll n$ and the function f preserves some proximity measure defined on the graph G . Intuitively, if two nodes u and v are "similar" in graph G , their embedding \mathbf{y}_u and \mathbf{y}_v should be close to each other in the embedding space. We use the notation $f(G) \in \mathbb{R}^{n \times d}$ for the embedding matrix of all nodes in the graph G . Note that the embedding of an edge (u, v) is defined as $g(u, v) = [\mathbf{y}_u, \mathbf{y}_v]$, i.e. the concatenation of embeddings of nodes u and v . It can be written as $g: E \rightarrow \mathbb{R}^{2d}$. We use $g(u, v)$ to reconstruct the edge label \mathbf{e}_{uv}^a . This enables us to infer the missing edge labels by using the adjacency of the incident nodes.

3 ELAINE

We propose an edge label aware information network embedding method - ELAINE, which models l^{th} -order proximity, social role features and edge labels using a deep variational autoencoder. The core component of the model is based on a deep autoencoder which can be used to learn the network embedding by minimizing the following loss function:

$$L = \sum_{i=1}^n \|(\hat{\mathbf{a}}_i - \mathbf{a}_i) \odot \boldsymbol{\beta}_i\|_2^2 = \|(\hat{A} - A) \odot \mathcal{B}\|_F^2 \quad (1)$$

The objective function penalizes inaccurate reconstruction of node neighborhood. As many legitimate links are not observed in the networks, a weight $\boldsymbol{\beta}_i$ is traditionally used to impose more penalty on reconstruction of observed edges [29].

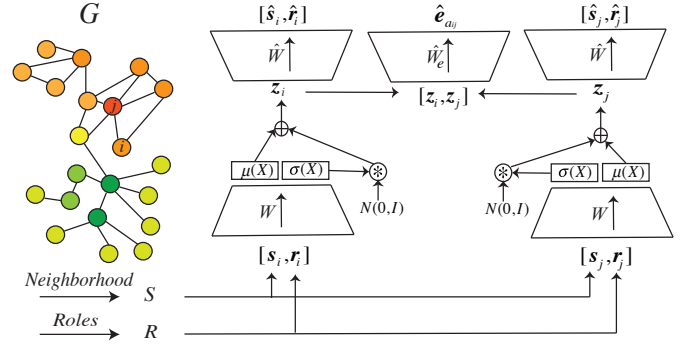


Figure 1: Edge label aware embedding model. ELAINE extracts higher order relations between nodes using random walks and social role based features. The coupled autoencoder jointly optimizes these features and edge attributes to obtain a unified representation.

Although the above model can learn network representations which can reconstruct the graph well, it suffers from four challenges. Firstly, as the model reconstructs the observed neighborhood of each vertex, it only preserves second order proximity of nodes. Wang et. al. [29] extend the model to preserve first order proximity but their model fails to capture higher order proximities. Concretely, if two nodes have disjoint neighborhoods the model will keep them apart regardless of the similarity of their neighborhoods. Secondly, the model is prone to overfitting leading to a satisfactory reconstruction performance but sub-par performance in tasks like link prediction and node classification. Wang et. al. [29] use l_1 and l_2 regularizers to address this issue but we show that using variational autoencoders can achieve better performance. Thirdly, the model does not explicitly capture social role information. Real world networks often have a role based structure understanding which can help with various prediction tasks. Lastly, the model does not consider edge labels. We show that incorporating edge label reconstruction leads to improved performance in various tasks.

To address the above challenges, we propose a random walk based deep variational autoencoder model with an objective to jointly optimize the higher order neighborhood, role based features and edge label reconstruction.

As we aim to find a low-dimensional manifold the original graph lies in, we want to learn a representation which is maximally informative of observed edges and edge labels. At the same time, as the autoencoder penalizes reconstruction error, it encourages perfect reconstruction at the cost of overfitting to the training data. This is in particular problematic for learning representations for graphs as networks are constructed from interactions which may be incomplete or noisy. We want to find embeddings which are robust to such noise and can help us in tasks such as link prediction and node classification. We propose to use variational autoencoder for graphs and illustrate in Section 5 that it can improve performance in different tasks. Variational autoencoders (VAEs) look at autoencoders from a generative network perspective and minimize the sum of two terms: (1) reconstruction loss and (2) KL-divergence of latent variable distribution and unit Gaussian, using backpropagation. The reconstruction variance controls the model generalization, which can be considered as the coefficient of KL-divergence loss.

3.1 Random Walks

Nodes in a network are related to each other via many degrees of connection. Some nodes have direct connections while others are connected through paths of varying lengths. A good embedding should preserve such higher order relations. Naively using node adjacency as the input, an autoencoder cannot achieve this. This is because the model then only considers the neighborhoods of nodes and does not explicitly capture the relations between these neighborhoods. To preserve higher order proximities, we obtain global distance based similarities of each node with the rest of the nodes. One way to obtain such a set of vectors is to use metrics such as Katz Index [16], Adamic Adar [1] and Common Neighbors [21]. Although such metrics capture global proximities accurately, their computation is inefficient and the time complexity is up to $O(n^3)$. We overcome the inefficiency by approximating them using random walks [4]. For each node i , we simulate k random walks each of length l . Each random walk, $\{v_{i,1}, v_{i,2} \dots v_{i,l}\}$, from node i generates a node j with probability:

$$P(v_{i,j}|v_{i,j-1}) = \begin{cases} \frac{1}{d_{j-1}} & \text{if } (v_{i,j-1}, v_{i,j}) \in E \\ 0 & \text{otherwise} \end{cases}$$

where d_k is the degree of node k . Note that since a random walk of length l from node i is equivalent to a random walk of length $l-1$ for node $v_{i,1}$, generating k random walks of length l only requires $O(k)$ time each node.

3.2 Role preserving features

Social roles in a network are characterized by various local and global statistics. For example, high degree can be reflective of social importance. Broadly, we classify role discriminating features into two categories: (a) statistical features, and (b) edge attributes. We consider the following statistical features which have been shown to correlate with social roles[11]: (i) node's degree, (ii) weighted degree, (iii) clustering coefficient, (iv) eccentricity, (v) structural hole and (vi) local gatekeeper. We append these features with node's neighborhood as input to our model. Having such statistical features helps obtain an embedding which preserves social roles. Conversely, a node can take different roles with different neighbors (henceforth referred as interactive roles) which cannot be captured by such statistical features. For example, in a collaboration network, author i may take the role of Professor with his student j and colleague with another professor k . Identifying such distribution of roles can help model the network more accurately. For this we use the edge attributes which can be reflective of such interactions. Concretely, we consider the topics of conversation between nodes and jointly optimize their reconstruction of node neighborhood reconstruction.

3.3 Incorporating edge labels

Autoencoder defined above takes node neighborhood and statistical role preserving features as input and aims to reconstruct them. One possible approach to incorporate edge attributes is to aggregate them for each node and append them with other node features. The drawback of this approach is that information loss can incur following aggregation. Such aggregation cannot preserve interactive roles between nodes.

We propose to overcome this problem by coupling copies of autoencoders for nodes i and j . The model is composed of a coupled autoencoder and an edge attribute decoder, Figure 1. The intuition is to force the embeddings of nodes i and j to capture information pertaining to the attributes of the edge between them. This is ensured by adding the edge attribute reconstruction loss to the objective function. Thus, we learn model parameters by minimizing a loss function with the following terms:

Network structure reconstruction: The l th-order neighborhood of each node along with the social role preserving statistical features:

$$L_n = \|([\hat{S}, \hat{R}] - [S, R]) \odot \mathcal{B}\|_F^2,$$

where each row of $S \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{n \times r}$ compose of neighborhood similarity and role statistics respectively. Henceforth, we will refer to $[\hat{S}, \hat{R}]$ by $\hat{F} \in \mathbb{R}^{n \times n+r}$ and $[S, R]$ by $F \in \mathbb{R}^{n \times n+r}$.

Edge label/attributes reconstruction: For each pair of nodes, we reconstruct the attributes of the edge between them, $L_e = \|\hat{E}^a - E^a\|_F^2$, where each row i of $E^a \in \mathbb{R}^{m \times p}$ is the vector of attributes of the i th edge. The overall objective function thus becomes

$$L = L_n + \alpha_1 L_e + L_{reg}, \quad (2)$$

where L_{reg} is the regularization on the weights. For our model, we use KL-divergence, Lasso and Ridge regularization.

To get the optimal parameters for the model defined above, we minimize the loss function L using stochastic gradient descent (SGD) [25] with Adaptive Moment Estimation (Adam)[17].

4 EXPERIMENTS

Here, we describe the data sets used and then discuss the baselines we use to compare our model and the evaluation metrics.

4.1 Datasets

Table 1: Dataset Statistics

| Name | n | m | Avg. degree | # of edge attributes |
|--------|-------|--------|-------------|----------------------|
| Hep-th | 7,980 | 21,036 | 5.27 | 100 |
| Enron | 145 | 912 | 12.58 | 10 |

We conduct experiments on the following real-world datasets to evaluate our proposed algorithm (summarized in Table 1):

Hep-th [8]: The original data set contains abstracts of papers in High Energy Physics Theory conference in the period from January 1993 to April 2003. We create a collaboration network for the first five years. We get the node labels using the Google Scholar API¹ to obtain university labels for each author. We apply NMF [26] on the set of abstracts to get topic distribution for each abstract. We aggregate the topic distribution of all the coauthored papers between two authors to get the edge attributes.

Enron [18]: This dataset contains emails among about 150 users, mostly senior management at Enron. We connect two users if they exchanged email(s). Edge attribute between node i and j is the extracted topics from each set of emails between them using NMF.

¹<https://pypi.python.org/pypi/scholarly/0.2>

4.2 Baselines

We compare our model with the following state-of-the-art methods:

- *Graph Factorization* (GF) [2]: It factorizes the adjacency matrix with regularization.
- *Structural Deep Network Embedding* (SDNE) [29]: It uses deep autoencoder along with Laplacian Eigenmaps objective to preserve first and second order proximities.
- *Higher Order Proximity Preserving* [22] (HOPE): It factorizes the higher order similarity matrix between nodes using generalized singular value decomposition [23].
- *node2vec* [10]: It preserves higher order proximity by maximizing the probability of occurrence of subsequent nodes in fixed length biased random walks. They use shallow neural networks to obtain the embeddings. *DeepWalk* is a special case of *node2vec* with the random walk bias set to 0.

4.3 Metrics and Hyperparameters

In our experiments, we evaluate our model on link prediction. We use Mean Average Precision (MAP) as our metric. MAP averages the precision over all nodes. It can be written as $MAP = \frac{\sum_i AP(i)}{|V|}$, where $AP(i) = \frac{\sum_k precision@k(i) \cdot \mathbb{I}\{E_{pred_i}(k) \in E_{gt_i}\}}{|\{k: E_{pred_i}(k) \in E_{gt_i}\}|}$ and $precision@k(i) = \frac{|E_{pred_i}(1:k) \cap E_{gt_i}|}{k}$.

For our model, we use two hidden layers for feature encoder and decoder with size [500, 300]. For the edge attribute decoder, we experiment with a single hidden layer with 1000 neurons and without any hidden layer. Optimal values of other hyperparameters are obtained using grid search over $[10^{-5}, 10^3]$ in factors of 10.

5 RESULTS AND ANALYSIS

In this section, we present results of our model on link prediction, and provide a comparison with baselines.

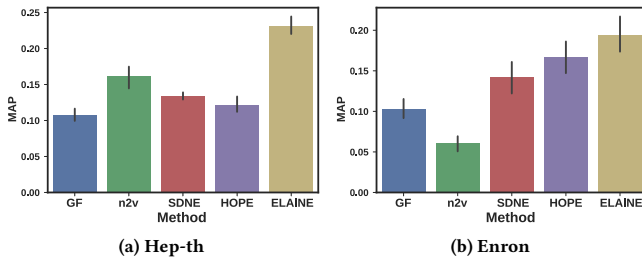


Figure 2: MAP of link prediction for different data sets.

Information networks are meant to capture the interactions in real world. This translation of interactions can be noisy and inaccurate. Predicting missing links in the constructed networks and links likely to occur in the future is an important and difficult task. We test our model on this link prediction task to understand the generalizability of our model. For each network, we randomly hide 20% of the network edges. We use the rest of the network to learn the embeddings of nodes and sort the likelihood of each unobserved edge to predict the missing links. As number of node pairs for a network of size N is $N(N-1)/2$, we randomly sample 1024 nodes

for evaluation (cf. [9]). We get 5 samples for each data set and report the mean and standard deviation of precision and MAP values.

Figures 2 illustrates the link prediction MAP values for the methods on data sets. We observe that our model significantly outperforms baselines on Hep-th. This implies that using the topic distribution of abstracts can help us understand the relation between authors. On Enron, we observe that gain in MAP isn't as significant as Hep-th but still improves over the baselines. This can be attributed to the characteristic of emails which tend to be more unstructured and noisy compared to abstracts of publications and hence more challenging to model. Furthermore, we observe that the performance of other models is not consistent over these two data sets. *node2vec* achieves best performance among the rest of the methods on Hep-th but performs poorly on Enron. Similarly, HOPE has high MAP value for Enron but is outperformed by SDNE and *node2vec* on Hep-th. Also, as our model extends deep autoencoders for edge attributes, to understand the importance edge attributes, we compare against SDNE which has similar architecture. We see that our model achieves significant gains over SDNE on both the data sets showing that our model can utilize edge attributes in different domains and improve link prediction performance.

6 CONCLUSION

In this work, we presented a novel method, ELAINE, which preserves network structure and edge attributes to obtain unified node representations. The model captures higher order proximities and social roles and can model non-linear network interactions. It uses a coupled deep autoencoder along with an edge attribute decoder to jointly optimize the reconstruction loss of network structure and edge attribute reconstruction. Our experiments demonstrate the efficacy of our model over two real world data sets including a collaboration network and an email network on the task of link prediction. In the future, we plan to extend our model to predict missing edge labels along with presence of edges. Furthermore, we believe that understanding dynamics of networks over time is an important tasks due to the nature of information spread. We thus want to extend our model to networks for which the edge attributes as well as the connections evolve. As our model uses a deep architecture, we see that the performance vastly depends on hyper parameters. We plan to study their behavior and develop a model which can automatically identify their optimal values. Another key direction is the choice of regularization. We observed improvement using a variational autoencoder and would like to understand the effect of regularization functions on the performance.

Acknowledgements. This work has been partly funded by the Defense Advanced Research Projects Agency (DARPA #W911NF-17-C-0094 and #D16AP00115), the Office of the Director of National Intelligence (ODNI), and the Intelligence Advanced Research Projects Activity (IARPA). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of DARPA, ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation. The U.S. Government had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

REFERENCES

- [1] Lada A Adamic and Eytan Adar. 2003. Friends and neighbors on the web. *Social networks* 25, 3 (2003), 211–230.
- [2] Amr Ahmed, Nino Shervashidze, Shravan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. 2013. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 37–48.
- [3] Réka Albert, Hawoong Jeong, and Albert-László Barabási. 2000. Error and attack tolerance of complex networks. *nature* 406, 6794 (2000), 378–382.
- [4] Lars Backstrom and Jure Leskovec. 2011. Supervised random walks: predicting and recommending links in social networks. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 635–644.
- [5] Shaosheng Cao, Wei Lu, and Qiongkai Xu. 2015. Grarep: Learning graph representations with global structural information. 891–900.
- [6] Shiyu Chang, Wei Han, Jiliang Tang, Guo-Jun Qi, Charu C Aggarwal, and Thomas S Huang. 2015. Heterogeneous network embedding via deep architectures. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 119–128.
- [7] Linton C Freeman. 2000. Visualizing social networks. *Journal of social structure* 1, 1 (2000), 4.
- [8] Johannes Gehrke, Paul Ginsparg, and Jon Kleinberg. 2003. Overview of the 2003 KDD Cup. *ACM SIGKDD Explorations* 5, 2 (2003).
- [9] Palash Goyal and Emilio Ferrara. 2018. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems* (2018). <https://doi.org/10.1016/j.knsys.2018.03.022>
- [10] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining*. ACM, 855–864.
- [11] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. 2012. Rolx: structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1231–1239.
- [12] Xiao Huang, Jundong Li, and Xia Hu. 2017. Accelerated attributed network embedding. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 633–641.
- [13] Xiao Huang, Jundong Li, and Xia Hu. 2017. Label informed attributed network embedding. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 731–739.
- [14] Ramon Ferrer i Cancho and Richard V Solé. 2001. The small world of human language. *Proceedings of the Royal Society of London B: Biological Sciences* 268, 1482 (2001), 2261–2265.
- [15] Paul Jaccard. 1901. *Etude comparative de la distribution florale dans une portion des Alpes et du Jura*. Impr. Corbaz.
- [16] Leo Katz. 1953. A new status index derived from sociometric analysis. *Psychometrika* 18, 1 (1953), 39–43.
- [17] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [18] Bryan Klimt and Yiming Yang. 2004. The enron corpus: A new dataset for email classification research. In *European Conference on Machine Learning*. 217–226.
- [19] David Liben-Nowell and Jon Kleinberg. 2007. The link-prediction problem for social networks. *Journal of the Association for Information Science and Technology* 58, 7 (2007), 1019–1031.
- [20] Kurt Miller, Michael I Jordan, and Thomas L Griffiths. 2009. Nonparametric latent feature models for link prediction. In *Advances in neural information processing systems*. 1276–1284.
- [21] Mark EJ Newman. 2001. Clustering and preferential attachment in growing networks. *Physical review E* 64, 2 (2001), 025102.
- [22] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. 2016. Asymmetric transitivity preserving graph embedding. In *Proc. of ACM SIGKDD*. 1105–1114.
- [23] Christopher C Paige and Michael A Saunders. 1981. Towards a generalized singular value decomposition. *SIAM J. Numer. Anal.* 18, 3 (1981), 398–405.
- [24] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings 20th international conference on Knowledge discovery and data mining*. 701–710.
- [25] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1988. Neurocomputing: Foundations of research. *J A Anderson and E. Rosenfeld, Eds* (1988), 696–699.
- [26] Suvrit Sra and Inderjit S Dhillon. 2006. Generalized nonnegative matrix approximations with Bregman divergences. In *Advances in neural information processing systems*. 283–290.
- [27] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings 24th International Conference on World Wide Web*. 1067–1077.
- [28] Athanasios Theodoridis, Stijn Van Dongen, Anton Enright, and Tom Freeman. 2009. Network visualization and analysis of gene expression data using BioLayout Express3D. *Nature protocols* 4 (2009), 1535–1550.
- [29] Daixin Wang, Peng Cui, and Wenwu Zhu. 2016. Structural deep network embedding. In *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining*. ACM, 1225–1234.