Interaction Content Aware Network Embedding via Co-embedding of Nodes and Edges

Linchuan Xu[1], Xiaokai Wei[2⋆], Jiannong Cao[1], and Philip S. Yu[34]

[1] The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
{cslcxu, csjcao}@comp.polyu.edu.hk,

[2] Facebook Inc, 1 Hacker Way, Menlo Park, CA, USA
weixiaokai@gmail.com,

[3] University of Illinois at Chicago, Chicago, Illinois, USA
psyu@uic.edu,

[4] Institute for Data Science, Tsinghua University, Beijing, China

Abstract. Network embedding has been a hot topic as it can learn node representations that encode the network structure resulting from node interactions. In this paper, besides the network structure, the interaction content within which each interaction arises is also embedded because it reveals interaction preferences of the two nodes involved. Specifically, we propose interaction content aware network embedding (ICANE) via co-embedding of nodes and edges. The embedding of edges is to learn edge representations that preserve the interaction content, which then can be incorporated into node representations through edge representations. Experiments demonstrate ICANE outperforms five recent network embedding models in visualization, link prediction and classification.

1 Introduction

Network embedding has been a hot topic recently. Existing methods [13] [16] [7] [24] [23] basically embed the network structure in a Euclidean space of interest. In this way, however, they fair to consider the content within which node interactions arise. In practice, the content can be observed in various networks:

- In academic co-authorship networks as illustrated in Fig. 1, the particular paper is the interaction content associated with co-authorships.
- In gene co-expression networks where genes co-express functional gene products, such as protein, the functional products are the interaction content.
- In social interaction networks where users interact under social media, e.g., discussing under images and documents, the media is the interaction content.

Interaction content has been shown helpful in network analysis, such as community detection [14]. In the scenario of network embedding, we can see interaction content contains node interaction preferences. Specifically in the co-

⋆ The work was done when the author was a Ph.D. student at University of Illinois at Chicago.

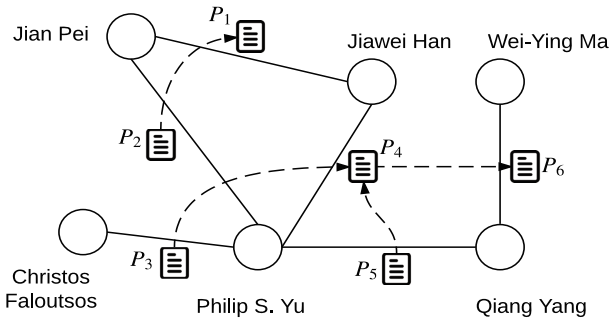

Fig. 1: A co-authorship network sampled from a DBLP dataset [17] where nodes denote researchers and rectangles with lines inside them denote papers that researchers co-authored. Some papers may be missing due to the sampling process.

authorship network, the interaction content indicates research interests. Similarly, the content in the social interaction networks reveals the events or activities that users are interested in. These two cases together indicate interaction preferences are specific for the social environment.

Moreover, some nodes may have multiple distinct interaction preferences, and each interaction may only arise within a single content. For example, a researcher may have interests in three research areas, such as Database, Machine Learning, and Data Mining. Different papers of the researcher and co-authors may belong to different areas. Not distinguishing different co-authorships in terms of the areas while embedding the co-authorship network, hence, is not appropriate.

To achieve this goal, the major challenge is that interaction content cannot be concatenated to node representations because it may not be affiliated to nodes. For example, in social networks where users interact under images or documents, the content may be belong to third parties not involved in the interactions.

In this paper, we propose interaction content aware network embedding (**ICANE**) via co-embedding of nodes and edges. Specifically, ICANE embeds the network structure in node representations, and embeds interaction content in edge representations. Moreover, ICANE incorporates interaction content into node representations via jointly learning representations for nodes and edges.

In some scenarios, interaction content can have relationships, e.g.,

- In co-authorship networks, the interaction content, i.e., papers, usually has citation relationships as illustrated in Fig. 1.
- In gene co-expression networks, the interaction content can be proteins, which can have protein-protein interactions in various biological processes.
- In social interaction networks, the interaction content is the media, such as documents, which can have references to each other.

Because the interaction content is affiliated to edges, we name the network resulting from content relationships as an edge network. Hence, ICANE encodes interaction content into edge representations by embedding the edge network.

In other scenarios, the interaction content has text information, e.g., paper content in co-authorship networks. Collectively, there may exist both an edge network and text information in some scenarios.

It is worthy of noting that node representations may also benefit the learning of edge representations which explicitly preserve node interaction preferences. Since node representations encode the network structure, i.e., interactions between nodes, node representations implicitly preserve interaction preferences of nodes. Hence, node representations and edge representations actually preserve similar characteristics but from different views.

2 Related work

The development of recent network embedding starts with DeepWalk [13], which employs Skip-gram to present pairs of nodes reached in the same truncated random walks to be close in the embedding space. There are other Skip-gram based models, such as TADW [25] to embed both network structure and node attributes, and node2vec [7] to explore diverse neighborhoods in random walks.

There are also many methods not based on Skip-gram. LINE [16] is proposed to embed large-scale networks by directly presenting pairs of nodes with first-order or second order connections to be close. GraRep [4] models first-order up to a pre-defined k-order proximities into transition matrices. A recent study [26] concludes that modelling high-order proximities can improve the quality of node representations. Besides simply preserving the network structure, some methods also preserve network properties, such as HOPE [12] preserving asymmetric transitivities and M-NMF [21] preserving communities. Some methods [5] [15] [22] [6] even embed heterogeneous information networks. Deep learning has also been applied for network embedding [20]. Most methods above are unsupervised learning methods. Semi-supervised methods [19] [27] [8] have also been studied.

3 Preliminaries

DEFINITION 1. *$G_v(V_v, E_v, C)$ denotes a **network with interaction content**, where V_v is a set of nodes, E_v is a set of weighted or unweighted, directed or undirected edges, and C is a set of interaction content.*

DEFINITION 2. *$G_e(V_e, E_e)$ denotes an **edge network**. V_e is a set of nodes which are the concept of edges in $G_v(V_v, E_v, C)$, E_e is a set of weighted or unweighted, directed or undirected edges among the interaction content C.*

Note that $|V_e|$ corresponds to the number of interaction content, and it may not be equal to $|E_v|$ due to two reasons. Firstly, multiple nodes may interact within the same content, e.g., multi-author papers, which results in multiple edges. Secondly, a pair of nodes may interact under multiple content. Multiple interactions are treated as a weighted edge like existing embedding models do.

As an embedding method, ICANE presents nodes connected by edges to be close in an Euclidean space. The closeness of two nodes is quantified as follows:

DEFINITION 3. *The **closeness** of two nodes is quantified as the probability of an edge between them, where the probability is defined as follows:*

$$p(\boldsymbol{v}_i, \boldsymbol{v}_j) = \frac{1}{1 + exp\{-\boldsymbol{v}_i^\top \boldsymbol{v}_j\}}, \tag{1}$$

where $\boldsymbol{v}_i \in \mathbb{R}^D$ and $\boldsymbol{v}_j \in \mathbb{R}^D$ are column vectors of representations for nodes i and j, respectively, and D is the dimension of the Euclidean space of interest.

The closeness is reasonable as larger probabilities indicate larger inner product of two vectors, which is a measurement of closeness in Euclidean space.

4 Model Development

4.1 Node Representation Learning

To embed the network structure, ICANE not only presents pairs of nodes connected by edges to be close but also presents pairs of nodes not connected to be apart in the embedding space because non-linkage information is also an important part of network structure. Since the closeness is quantified as probability, the network structure preserving can be formulated into an optimization objective according to maximum likelihood estimation as follows:

$$\max_{\boldsymbol{V} \in \mathbb{R}^{|V_v| \times D}} \prod_{(i,j) \in E_v, (h,k) \notin E_v} p(\boldsymbol{v}_i, \boldsymbol{v}_j)(1 - p(\boldsymbol{v}_h, \boldsymbol{v}_k)), \tag{2}$$

which maximizes the probabilities of both linkage and non-linkage relationships.

The multiplication maximization is usually transformed to an equivalent minimization by taking negative natural logarithm, which is denoted as follows:

$$\min_{\boldsymbol{V}} - \left[\sum_{(i,j) \in E_v} (w_v)_{ij} \log p(\boldsymbol{v}_i, \boldsymbol{v}_j) + \sum_{(h,k) \notin E_v} \log(1 - p(\boldsymbol{v}_h, \boldsymbol{v}_k)) \right], \tag{3}$$

where $(w_v)_{ij} \in \mathbb{R}$ is the weight of edge (i, j) added to reflect to relationship strength. The loss function is referred to as $\mathcal{L}_v$ in the rest of the paper.

4.2 Edge Representation Learning

For interaction content that can produce an edge network, edge representations can be learned by embedding the edge network structure, which can be performed in the same way as embedding the node network structure. Hence, the loss function is referred to as $\mathcal{L}_e$, which is the same as $\mathcal{L}_v$ except that node representations are replaced with edge representations.

For interaction content with text information, the content can be embedded into edge representations via regularizing edge representations to accord with the text information. The regularization is reasonable because the text information is the ground truth about the interaction preferences, e.g., paper content

denotes research topics. The regularization can be performed by projecting the representations to corresponding content, which is formulated as follows:

$$\min_{\boldsymbol{M}\in\mathbb{R}^{D\times Q}} ||\boldsymbol{E}\boldsymbol{M}-\boldsymbol{A}||_F^2, \tag{4}$$

where $\boldsymbol{M}$ is a projection matrix to be estimated, Q is the number of terms in text, $\boldsymbol{E}\in\mathbb{R}^{|V_e|\times D}$, $\boldsymbol{A}\in\mathbb{R}^{|V_e|\times Q}$ is a term-frequency matrix extracted from text, and $||\cdot||_F^2$ is Frobenius norm. The intuition behind Eq. (4) is that the content is well represented by edge representations through the projection matrix.

4.3 Joint Learning

The key to joint learning is how to relate edge representations to node representations so that interaction content can be incorporated into node representations. As mentioned in the introduction, node representations encoding the network structure implicitly preserve node interaction preferences and edge representations explicitly preserve interaction preferences. Hence, node representations should be similar to representations of their incident edges. To make the problem simple, nodes are presented to be close to their incident edges, which can be achieved in a similar way to encode linkage relationships among nodes.

Hence, the overall loss function for joint learning can be obtained as follows:

$$\begin{aligned}\mathcal{L}(\boldsymbol{V},\boldsymbol{E},\boldsymbol{M}) =& \mathcal{L}_v+\mathcal{L}_e-\left[\sum_{v_i\to e_m}\log p(\boldsymbol{v}_i,\boldsymbol{e}_m)+\sum_{v_i\mapsto e_l}\log(1-p(\boldsymbol{v}_i,\boldsymbol{e}_l))\right]\\ &+||\boldsymbol{E}\boldsymbol{M}-\boldsymbol{A}||_F^2+\lambda(||\boldsymbol{V}||_F^2+||\boldsymbol{E}||_F^2+||\boldsymbol{M}||_F^2),\end{aligned} \tag{5}$$

which directly adds loss functions for node representation learning, edge representation learning and joint learning. More sophisticated ways for the combination is left as future work. $v_i\to e_m$ denotes e_m is an incident edge of v_i while $v_i\mapsto e_l$ denotes the opposite. $p(\boldsymbol{v}_i,\boldsymbol{e}_m)$ is the closeness measurement between a node and an edge, which is defined similarly to the closeness among nodes as mentioned above. Specifically, $p(\boldsymbol{v}_i,\boldsymbol{e}_m)$ is quantified as follows:

$$p(\boldsymbol{v}_i,\boldsymbol{e}_m)=\frac{1}{1+\exp\{-\boldsymbol{v}_i^\top\boldsymbol{e}_m\}}, \tag{6}$$

Eq. (5) assumes that there exist both an edge network and text information. In some cases where there may be only one type of content information, we can safely remove the corresponding component from Eq. (5). Hence, for cases where there is only an edge network, we name the model as INCAE(E) while for cases where there is only text information, we name the model as ICANE(A).

5 The Optimization

$\mathcal{L}(\boldsymbol{V},\boldsymbol{E},\boldsymbol{M})$ is not jointly convex over the three variables. We thus solve it by an alternating algorithm [3] which replaces a complex optimization problem with a sequence of easier sub-problems, and then solves the sub-problems

Algorithm 1: The optimization algorithm

Input : $G_v(V_v, E_v, C)$, D, λ, and negative ratio
Output: $\boldsymbol{V}$ and $\boldsymbol{E}$

Pre-training $\boldsymbol{V}$ and $\boldsymbol{E}$ with gradient descent;
while *(not converge)* **do**
  Fix $\boldsymbol{V}$ and $\boldsymbol{E}$, find the optimal $\boldsymbol{M}$ with the Eq. (10);
  Fix other variable(s), find the optimal $\boldsymbol{E}$ with gradient descent;
  Fix other variable(s), find the optimal $\boldsymbol{V}$ with gradient descent;
return $\boldsymbol{V}$ and $\boldsymbol{E}$

alternatingly. In our case, the sub-problems w.r.t $\boldsymbol{v}_i$ and $\boldsymbol{e}_i$ can be solved by gradient-based algorithms, e.g., steepest descent or L-BFGS. The derivative for minimizing $\mathcal{L}(\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{M})$ with respect to $\boldsymbol{v}_i$ is computed as follows:

$$\begin{aligned}\frac{\partial \mathcal{L}(\boldsymbol{V}, \boldsymbol{E})}{\partial \boldsymbol{v}_i} = &- \sum_{(i,j)\in E_v} \left[\frac{(w_v)_{ij} \exp\{-\boldsymbol{v}_i^\top \boldsymbol{v}_j\}}{1+\exp\{-\boldsymbol{v}_i^\top \boldsymbol{v}_j\}} \boldsymbol{v}_j \right] + \sum_{(i,k)\notin E_v} \left[\frac{\boldsymbol{v}_k}{1+\exp\{-\boldsymbol{v}_i^\top \boldsymbol{v}_k\}} \right] \\ &- \sum_{v_i \to e_m} \left[\frac{\exp\{-\boldsymbol{v}_i^\top \boldsymbol{e}_m\}}{1+\exp\{-\boldsymbol{v}_i^\top \boldsymbol{e}_m\}} \boldsymbol{e}_m \right] + \sum_{v_i \mapsto e_l} \left[\frac{\boldsymbol{e}_l}{1+\exp\{-\boldsymbol{v}_i^\top \boldsymbol{e}_l\}} \right] + 2\lambda(\boldsymbol{v}_i),\end{aligned} \tag{7}$$

The derivative with respect to $\boldsymbol{e}_m$ is computed as follows: $\frac{\partial \mathcal{L}(\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{M})}{\partial \boldsymbol{e}_m} =$

$$\begin{aligned}&- \sum_{(m,n)\in E_e} \left[\frac{(w_e)_{mn} \exp\{-\boldsymbol{e}_m^\top \boldsymbol{e}_n\}}{1+\exp\{-\boldsymbol{e}_m^\top \boldsymbol{e}_n\}} \boldsymbol{e}_n \right] + \sum_{(m,l)\notin E_e} \left[\frac{\boldsymbol{e}_l}{1+\exp\{-\boldsymbol{e}_m^\top \boldsymbol{e}_l\}} \right] \\ &- \sum_{v_i \to e_m} \frac{\exp\{-\boldsymbol{e}_m^\top \boldsymbol{v}_i\}}{1+\exp\{-\boldsymbol{e}_m^\top \boldsymbol{v}_i\}} \boldsymbol{v}_i + \sum_{v_k \mapsto e_m} \frac{\boldsymbol{v}_k}{1+\exp\{-\boldsymbol{e}_m^\top \boldsymbol{v}_k\}} + 2(\boldsymbol{e}_m^T \boldsymbol{M} - \boldsymbol{a}_m^T)\boldsymbol{M}^T + 2\lambda(\boldsymbol{e}_m^d),\end{aligned} \tag{8}$$

To minimize $\mathcal{L}(\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{M})$ with respect to $\boldsymbol{M}$, the optimization objective actually turns into solving the following problem:

$$\min_{\boldsymbol{M}} ||\boldsymbol{E}\boldsymbol{M} - \boldsymbol{A}||_2^2 + \lambda ||\boldsymbol{M}||_2^2. \tag{9}$$

It is easy to see that the optimal $\boldsymbol{M}$ can be obtained by setting the derivative of Eq. (9) w.r.t $\boldsymbol{M}$ to zero. Hence, the optimal $\boldsymbol{M}$ is obtained as follows:

$$\boldsymbol{M} = (\boldsymbol{E}^T \boldsymbol{E} + \lambda \boldsymbol{I})^{-1} \boldsymbol{E}^T \boldsymbol{A}, \tag{10}$$

where $\boldsymbol{I} \in \mathbb{R}^{D \times D}$ is an identity matrix.

The pseudo-codes of the alternating optimization algorithm are presented in Algorithm 1. Negative ratio is the ratio of the number positive edges to that of negative edges as used in LINE [16]. With the negative ration, the scalability to large-scale networks can be guaranteed. ***Pre-training*** is performed to initialize the model to a point in parameter space that renders the learning process more effective [2]. The pre-training on $\boldsymbol{V}$ or $\boldsymbol{E}$ is performed by solely preserving the network structure of $G_v(V_v, E_v, C)$ or $G_e(V_e, E_e)$, i.e., minimizing $\mathcal{L}_v$ or $\mathcal{L}_e$ by

Table 1: Network statistics

Network	Co-authorship	Paper Citation	User Interaction	Photo(group)
# Nodes	12407	8208	5342	2613
# Edges	27714	10532	230123	38841
# Attributes	6934		4070	

gradient descent. The learning rates of gradient descent are obtained by back-tracking line search [1]. If there is no $G_e(V_e, E_e)$, the pre-training of $\boldsymbol{E}$ can be performed by factorizing the term-frequency matrix $\boldsymbol{A}$ using SVD [4].

Algorithm 1 is essentially a block-wise coordinate descent algorithm [18] whose convergence can be guaranteed.

6 Empirical Evaluation

6.1 Datasets

- DBLP [17]: A co-authorship network in Table 1 is sampled with papers as the interaction content. Papers are selected from conferences of four fields, which are SIGMOD, VLDB, ICDE, EDBT, and PODS for Database, KDD, ICDM, SDM, and PAKDD for Data Mining, ICML, NIPS, AAAI, IJCAI and ECML for Machine Learning, SIGIR, WSDM, WWW, CIKM, and ECIR for Information Retrieval. Publication time span is set as 17 years from 1990 to 2006.
- CLEF [11]: From CLEF, we sample a user interaction network where interactions are established between users commenting on the same photo. Hence, the photos are the interaction content. Photos can be categorized into different groups, such as scenery, explore, etc. The groups can be used to construct a photo network where edges are established between two photos belonging to the same group. We refer to this photo network as photo(group) network.

6.2 Experiment Settings

Five recent network embedding models, DeepWalk [13], LINE [16], TADW [25], node2vec [7], and EOE [22] are used as baselines. Both TADW and EOE embed networks with node content. They can be applied to the DBLP co-authorship network because paper content can also be used as node content. However, the tags of images of the Flickr user interaction network cannot be used as node content in that the tags belong to a third party. For the implementation of Algorithm 1, we set the embedding dimension as 128, which is used in all the baselines, negative ratio as 5, which is used in LINE.

6.3 Representation Visualization

This section visually presents how effectively the representations encode the network structure. The DBLP data is used as the illustration. t-SNE [10] is employed to visualize the author representations in Fig. 2. From Fig. 2(1) through

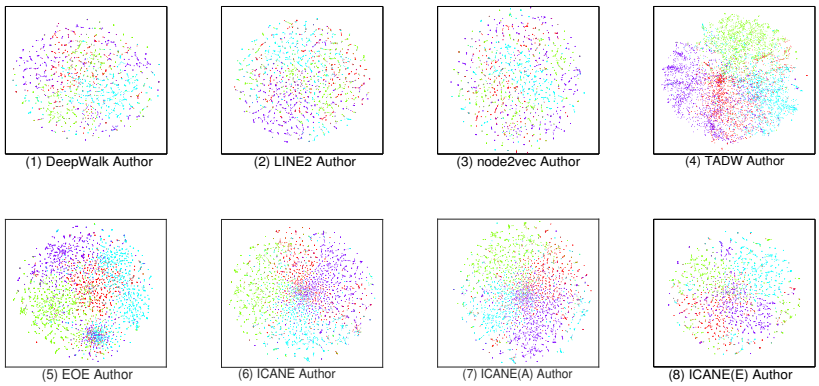

Fig. 2: Visualization of representations for the DBLP dataset, where green points are for authors from DB, light blue for IR, dark blue for DM, and red for ML. The filed of an author is chosen as the one where he/she published the most papers.

Fig.2 (3) of baselines (LINE(1st) is omitted due to space limitation because it performs worse than LINE(2nd)), we see that a considerably large number of authors from different fields are mixed up. This may be because the selected four fields, i.e., DB, DM, ML, and IR, are closely related, and there are many cross-field co-authorships. Hence, the network structure along is not enough to distinguish authors from one research field to another.

TADW, EOE and the proposed ICANE work better by utilizing the paper content as illustrated in Fig. 2(4) though Fig. 2(6) where data points of the same color are distributed together. This is because research focus of each field is distinct, which is reflected on paper content. To make fair comparison with TADW and EOE, we visualize representations learned by ICANE(A) in Fig. 2(7). We see that ICANE(A) is also comparable with TADW and EOE. Moreover, we visualize representations learned by ICANE(E) in Fig. 2(8). We can see Fig. 2(8) performs better than the baselines only embedding the co-authorship network.

It might not be easy to visually tell which one of TADW, EOE, and ICANE performs better, but ICANE can jointly learn author representations and paper representations while all the baselines can only learn author representations. Learning paper representations can lend strengths to learn author representations, and vice versa because paper representations capture research interests of authors. As a result, the data mining applications with respect to either nodes or edges may benefit from each other. The advantage of the joint learning is demonstrated in the following link prediction and classification.

6.4 Link Prediction

Link prediction is usually performed by measuring similarities between two nodes [9]. Here, the inner product of two node representations normalized by sigmoid function is employed as the similarity measurement. We first perform Flickr

Table 2: AUC scores(100%) for Flickr interaction prediction when different ratios of interactions are used in the training phase.

Model	10%	20%	30%	40%	50%	60%	70%	80%	90%
DeepWalk	79.30	86.74	89.52	90.59	91.30	91.49	91.73	91.75	91.88
LINE(1st)	74.26	85.45	89.62	91.87	93.17	93.96	94.61	95.00	95.36
LINE(2nd)	78.22	84.03	86.75	88.47	89.34	90.00	90.19	89.81	89.22
node2vec	77.07	80.57	81.67	81.79	81.17	81.64	82.01	81.46	81.73
ICANE(E)	90.25	93.21	94.39	**95.95**	**96.38**	**96.50**	**97.02**	97.25	**97.42**
ICANE(A)	**93.82**	**94.93**	**95.73**	95.22	96.22	96.29	96.96	97.10	97.29
ICANE	92.40	93.58	94.27	94.78	95.64	95.06	96.89	**97.32**	97.39

Table 3: AUC scores(100%) for DBLP co-authorship prediction, where (E) and (A) denote ICANE(E) and ICANE(A), respectively.

Model	DeepWalk	LINE(1st)	**LINE(2nd)**	node2vec	TADW	EOE	(E)	(A)	ICANE
AUC	76.62	72.38	**83.05**	76.68	73.82	80.95	78.02	81.30	82.83

user interaction prediction, and conduct 9 runs of experiments where training interactions range from 10% to 90% of the total interactions and the rest are used as test interactions. Moreover, for each experiment, the same number of negative interactions are randomly sampled for the evaluation purpose. AUC is employed as the evaluation metric, and the results are presented in Table 2.

Table 2 shows ICANE consistently outperform all the baselines no matter what kind of content information is utilized (TADW and EOE are not applicable since photos belong to a third party instead of the nodes of the interaction network). Moreover, ICANE still work well given very limited training interactions, such as 10% and 20%. The superior performance of ICANE benefits from the extra information provided by photo tags and the photo(group) network where photos are the interaction content within which users have interactions.

For DBLP co-authorship prediction, the co-authorships arise from 2007 to 2013 are used as test links, and experiment results are presented in Table 3. ICANE outperforms all the baselines except for LINE(2nd). As we examine the node similarities of links computed on the representations learned by LINE(2nd), all the positive and negative links have node similarities close to 1.0, which is not that interpretable. The reason behind the phenomenon may be that LINE(2nd) presents nodes with second-order link to be close, which is not consistent with the fist-order link prediction. Hence, LINE(2nd) is omit in the following discussion.

ICANE(E) outperforms baselines that only embed the network structure but underperforms EOE which also embeds node content. This may be because the paper citation network brings less useful information than paper content which has explicit information about node interaction preferences. The useful information brought by the paper citation network can be seen in the better performance of ICANE than that of ICANE(A). Moreover, ICANE(A) and ICANE outperform EOE. Recall that ICANE(A) and ICANE use the paper content as interaction content while EOE uses it to construct an author-word coupled network.

Table 4: Micro-F1(100%)&Macro-F1(100%) for multi-label classification

Flickr	DeepWalk	LINE(1st)	LINE(2nd)	node2vec	TADW	EOE	(E)	(A)	**ICANE**
Micro-F1	65.1	58.6	57.2	72.5	69.5	70.3	73.2	73.1	**73.7**
Macro-F1	65.3	58.5	57.2	72.3	69.1	70.0	73.0	72.8	**73.2**
DBLP	DeepWalk	LINE(1st)	LINE(2nd)	node2vec	TADW	EOE	(E)	(A)	**ICANE**
Micro-F1	60.2	59.9	59.5	62.9	79.6	79.9	67.3	79.7	**80.8**
Macro-F1	37.6	35.8	36.2	43.7	74.5	74.6	56.4	74.8	**75.3**

Table 5: Accuracy on DBLP paper multi-class classification

Model	DeepWalk	LINE(1st)	LINE(2nd)	node2vec	TADW	EOE	(E)	(A)	**ICANE**
Accuracy	68.62	61.40	55.36	70.84	71.06	69.03	72.83	73.56	**73.66**

Hence, the interaction content is explicitly embedded into edge representations, and then is incorporated into node representations in ICANE. In EOE, paper content is fragmentarily embedded in word representations by embedding the word network. We can see the mechanism for incorporating paper content into node representations of ICANE is more effective than that of EOE.

TADW performs even worse than DeepWalk that only embeds the network structure. It is worthy of noting that TADW concatenates node representations learned from node content and node representations learned from the network structure. As a result, the node similarities of links are largely determined by their node content, which indicates research interests of researchers. It is intuitive that it is not necessary for researchers with similar interests to collaborate.

6.5 Multi-label Classification

For DBLP, a research field as a label is assigned to authors if they published papers in this field. For Flickr, there are 99 labels for the photos in the dataset. It is worthy of noting that for Flickr, the photo representations are edge representations in the proposed models while they are node representations learned by baselines from the photo(group) network. The photo tags are the node content of the photo(group) network. We employ Micro-F1 and Macro-F1 as the performance metrics, and results of ten-fold cross validation are presented in Table 4, which is produced by binary-relevance SVM with polynomial kernel. For Flickr, ICANE performs better than all the baselines because ICANE can utilize not only the photo tags and the photo(group) network, but also the user interaction network. The user interaction network results from the photos so that it can provide auxiliary information to the photo representations.

For DBLP, all the models utilizing both the co-authorship network and paper content perform significantly than those only utilizing the co-authorship network. ICANE(A) obtains similar performance as TADW and EOE, but ICANE performs better than TADW and EOE because it can even utilize the paper citation network. The benefits brought by the paper citation network can be seen in the superior performance of ICANE(E) to that of DeepWalk, LINE and node2vec.

6.6 Multi-class Classification

The research field as a label is assigned to each paper. We employ SVM with polynomial kernel as the classifier, and present the accuracy obtained by 10-fold cross validation in Table 5. Similarly, ICANE outperforms all the baselines. To this point, we have demonstrated not only the interaction content can help improve node representations, but also the nodes can in turn help improve content representations. Particularly in this case, the authors largely determine the fields of their papers because they have particular expertise.

Acknowledgement

The work described in this paper was partially supported by the funding for Project of Strategic Importance provided by The Hong Kong Polytechnic University (Project Code: 1-ZE26), RGC General Research Fund under Grant PolyU 152199/17E, NSFC Key Grant with Project No. 61332004, NSF through grants IIS-1526499, and CNS-1626432, and NSFC 61672313.

References

1. Armijo, L.: Minimization of functions having lipschitz continuous first partial derivatives. Pacific Journal of mathematics **16**(1), 1–3 (1966)
2. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H., et al.: Greedy layer-wise training of deep networks. Advances in neural information processing systems **19**, 153 (2007)
3. Bezdek, J.C., Hathaway, R.J.: Some notes on alternating optimization. In: AFSS International Conference on Fuzzy Systems. pp. 288–300. Springer (2002)
4. Cao, S., Lu, W., Xu, Q.: Grarep: Learning graph representations with global structural information. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 891–900. ACM (2015)
5. Chang, S., Han, W., Tang, J., Qi, G.J., Aggarwal, C.C., Huang, T.S.: Heterogeneous network embedding via deep architectures. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 119–128. ACM (2015)
6. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 135–144. ACM (2017)
7. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016)
8. Li, J., Zhu, J., Zhang, B.: Discriminative deep random walk for network classification. In: ACL (1) (2016)
9. Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. Journal of the American society for information science and technology **58**(7), 1019–1031 (2007)
10. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(2579-2605), 85 (2008)

11. McAuley, J., Leskovec, J.: Image labeling on a network: using social-network metadata for image classification. In: European Conference on Computer Vision. pp. 828–841. Springer (2012)
12. Ou, M., Cui, P., Pei, J., Zhang, Z., Zhu, W.: Asymmetric transitivity preserving graph embedding. In: KDD. pp. 1105–1114 (2016)
13. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 701–710. ACM (2014)
14. Qi, G.J., Aggarwal, C.C., Huang, T.: Community detection with edge content in social media networks. In: 2012 IEEE 28th International Conference on Data Engineering. pp. 534–545. IEEE (2012)
15. Tang, J., Qu, M., Mei, Q.: Pte: Predictive text embedding through large-scale heterogeneous text networks. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1165–1174. ACM (2015)
16. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web. pp. 1067–1077. International World Wide Web Conferences Steering Committee (2015)
17. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: Arnetminer: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 990–998. ACM (2008)
18. Tseng, P.: Convergence of a block coordinate descent method for nondifferentiable minimization. Journal of optimization theory and applications **109**(3), 475–494 (2001)
19. Tu, C., Zhang, W., Liu, Z., Sun, M.: Max-margin deepwalk: Discriminative learning of network representation. In: IJCAI. pp. 3889–3895 (2016)
20. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1225–1234. ACM (2016)
21. Wang, X., Cui, P., Wang, J., Pei, J., Zhu, W., Yang, S.: Community preserving network embedding. In: AAAI. pp. 203–209 (2017)
22. Xu, L., Wei, X., Cao, J., Yu, P.S.: Embedding of embedding (eoe): Joint embedding for coupled heterogeneous networks. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. pp. 741–749. ACM (2017)
23. Xu, L., Wei, X., Cao, J., Yu, P.S.: Multi-task network embedding. In: Proceedings of the Fourth IEEE International Conference on Data Science and Advanced Analytics. pp. 571–580. IEEE (2017)
24. Xu, L., Wei, X., Cao, J., Yu, P.S.: Multiple social role embedding. In: Proceedings of the Fourth IEEE International Conference on Data Science and Advanced Analytics. pp. 581–589. IEEE (2017)
25. Yang, C., Liu, Z., Zhao, D., Sun, M., Chang, E.Y.: Network representation learning with rich text information. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina. pp. 2111–2117 (2015)
26. Yang, C., Sun, M., Liu, Z., Tu, C.: Fast network embedding enhancement via high order proximity approximation. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI. pp. 19–25 (2017)
27. Yang, Z., Cohen, W.W., Salakhutdinov, R.: Revisiting semi-supervised learning with graph embeddings. arXiv preprint arXiv:1603.08861 (2016)