TAXOGAN: Hierarchical Network Representation Learning via Taxonomy Guided Generative Adversarial Networks (Extended Abstract)[*]

Carl Yang[1][†] , **Jieyu Zhang**[2] and **Jiawei Han**[3]

[1]Emory University
[2]University of Washington
[3]University of Illinois, Urbana Champaign
j.carlyang@emory.edu, jieyuz2@cs.washington.edu, hanj@illinois.edu

Abstract

Network representation learning aims at transferring node proximity in networks into distributed vectors, which can be leveraged in various downstream applications. Recent research has shown that nodes in a network can often be organized in latent hierarchical structures, but without a particular underlying taxonomy, the learned node embedding is less useful nor interpretable. In this work, we aim to improve network embedding by modeling the *conditional node proximity* in networks indicated by node labels residing in real taxonomies. In the meantime, we also aim to model the *hierarchical label proximity* in the given taxonomies, which is too coarse by solely looking at the hierarchical topologies. Comprehensive experiments and case studies demonstrate the utility of TAXOGAN.

1 Introduction

Representation learning has become the backbone of various tasks in artificial intelligence [Bengio *et al.*, 2013; Yang *et al.*, 2020b]. Unsupervised learning is often the default setting due to the desired generalizability. However, many recent works in various fields have demonstrated the profit of leveraging limited label data to learn representations that are not only powerful for the corresponding predictive objectives, but also transferrable to other related tasks. Among them, hierarchical labels residing in given taxonomies have been widely used for natural language processing and bioinformatics, which are especially useful for the tasks of hypernym modeling and hierarchical classification [Wehrmann *et al.*, 2018; Peng *et al.*, 2018; Alsuhaibani *et al.*, 2018; Nguyen *et al.*, 2017; Vulić and Mrkšić, 2018]. In their essence, these methods jointly learn the representations of objects and labels in a shared latent space. The objects they model often have rich features, but they do not directly interact with each other.

As for representation learning on networks of interconnected objects (nodes), intensive research has been done on the modeling of both plain networks without node features [Yang *et al.*, 2018; Perozzi *et al.*, 2014; Tang *et al.*, 2015b; Wang *et al.*, 2018; Dai *et al.*, 2019; Gao *et al.*, 2019] and content-rich networks with node attributes and/or labels [Yang *et al.*, 2017; Meng *et al.*, 2019; Huang *et al.*, 2017a; Kipf and Welling, 2017; Hamilton *et al.*, 2017; Yang *et al.*, 2020a]. Recently, the notion of taxonomy has been explored by pioneering research [Nickel and Kiela, 2017; Ma *et al.*, 2018], which assume and seek for the latent hierarchical structure underlying the seemingly flatly connected nodes. However, without proper reference to a particular underlying taxonomy, the learned network embedding is still limited to global network mining tasks and uninterpretable without further analysis [Liu *et al.*, 2018].

Thanks to the vast effort in taxonomy construction from both the research community [Zhang *et al.*, 2018; Park *et al.*, 2017; Wang *et al.*, 2015; Downey *et al.*, 2015; Yang *et al.*, 2019] and industry[1][2][3], increasing amount of network data nowadays can be readily associated with existing taxonomies, which provides great opportunities for enhancing network embedding and enabling novel network mining tasks. Meanwhile, the rich relational data in networks may also help in better modeling and interpreting the existing taxonomies.

Consider a toy example in Figure 1, which consists of an author network and a research topic taxonomy. Author-author links can be generated *w.r.t.* co-authorships, while author-label links can be generated by keyword matching between the topic names in the taxonomy and the published papers of the authors. In this work, we stress the importance of two novelly observed properties, *i.e.*, *conditional node proximity* and *hierarchical label proximity*.

Conditional node proximity. While existing works on network embedding mostly consider network proximity within the same set of nodes, we argue that node proximity should be conditionally measured within the proper context. For example, on the left side of Figure 1, consider the proximity between C. Faloutsos and J. Kleinberg (particularly, in comparison to that between C. Faloutsos and J. Han). When working on Graph Mining (Graph) problems, C. Falout-
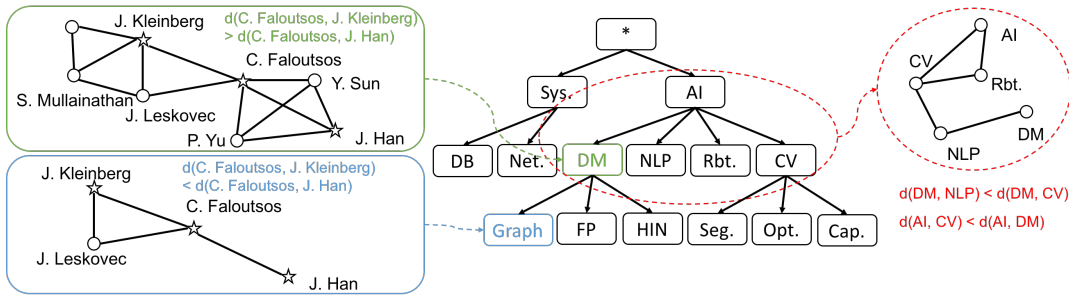

Figure 1: **Toy example of TAXOGAN: Authors in a publication network are naturally connected to a research topic taxonomy. Through proper modeling of** *conditional node proximity* **(on the left side) and** *hierarchical label proximity* **(on the right side), we aim to leverage author node proximity in the network to capture topic label proximity in the taxonomy, which in turn can benefit the learning of both author and topic representations in a closed loop.**

sos and J. Kleinberg share more important coauthors like J. Leskovec, thus resulting in a smaller distance. However, when working on broader problems in Data Mining (DM), they find their own coauthors like S. Mullainathan and J. Han from different fields, hence resulting in a larger distance. As such, under different conditions, node proximity can be rather different and even contradictory.

Hierarchical label proximity. Although we assume the existence of given taxonomies for particular networks, where node labels are organized in tree-structured hierarchies, the actual distribution and relative distance of these labels in the embedding space is unknown. For example, consider the four labels CV, NLP, Rbt. and DM on the right side of Figure 1. Although they are all child labels of the parent label AI, the distances among these siblings as well as their distances to AI might be rather different, which is impossible to understand by solely looking at the taxonomy structure itself. In this work, we propose to leverage the rich relational information from the networks to model the fine-grained proximity among the hierarchical labels. Continue with our example. Since authors working on Rbt. may overlap or collaborate more with those working on CV than DM, the distance between Rbt. and CV should be smaller than that between Rbt. and DM. Moreover, compared with authors working on DM, authors working on CV might more often study the core problems of AI. As a consequence, the distance between AI and CV should be smaller than that between AI and DM.

Present work. We propose TAXOGAN to co-embed network nodes and hierarchical labels, which leverages stacked generative adversarial nets to model the conditional node proximity and hierarchical label proximity in networks guided by label taxonomies. Specifically, TAXOGAN models a hierarchical network generation process, where a network generator is devised at each parent label in the taxonomy to model the children network induced by the corresponding child labels and labeled nodes in the original network. Moreover, a learnable network encoder is devised at each child label to enable the learning of proximity transfer from the embedding spaces of children to parents in a fine-to-abstract manner along the actual label paths in the taxonomy. Finally, we device hierarchical adversarial learning to achieve efficient and robust model inference.

2 TAXOGAN

We propose TAXOGAN to co-embed network nodes and hierarchical labels through a hierarchical network generation process, where a network generator is devised at each parent label in the taxonomy to model the subnetwork of nodes and child labels, and a network encoder is devised at each child label to learn the transferrable proximity across levels in the taxonomy. The generator and encoder are jointly trained through efficient and robust hierarchical adversarial learning, where a network discriminator is devised in each embedding space to enforce correct node-node and node-label proximity. In the following, we motivate and describe each component of TAXOGAN in details.

Label-wise subnetwork generator: jointly model node and label proximities in conditional subnetworks. To properly model conditional node proximity and respect the label hierarchy, we propose to generate a specific node-label network under each parent (non-leaf) label in the taxonomy. Let l_p denote an arbitrary parent label in $\mathcal{T}$, and $\mathcal{L}_p$ denote the set of all immediate child labels of l_p. Then $\mathcal{V}_p$ is the subset of $\mathcal{V}$ consisting of all nodes with label l_p or labels in $\mathcal{L}_p$. A *conditional subnetwork* $\mathcal{B}_p$ is constructed from $\mathcal{V}_p$, $\mathcal{L}_p$ as well as the node-node links $\mathcal{E}_p$ among nodes $\mathcal{V}_p$ and node-label links $\mathcal{Y}_p$ between nodes $\mathcal{V}_p$ and labels $\mathcal{L}_p$.

$\mathcal{B}_p$ acts as a bridge between node proximity and label proximity under the condition of l_p. In the corresponding embedding space $\mathcal{S}_p$, $\mathcal{V}_p$ and $\mathcal{L}_p$ can then be arranged in a flat way. To learn the node embedding $\mathbf{U}^p$ and label embedding $\mathbf{Q}^p$ in the space of $\mathcal{S}_p$, we devise a subnetwork generator $\mathcal{G}$ to enforce $\mathcal{E}_p$ and $\mathcal{Y}_p$ based on the softmax function as follows

$$\mathcal{G}(v_j, v_i | l_p) = \frac{\exp(\mathbf{u}_j^{pT} \cdot \mathbf{u}_i^p)}{\sum_{v_k \in \mathcal{V}_p} \exp(\mathbf{u}_k^{pT} \cdot \mathbf{u}_i^p)}, \qquad (1)$$

$$\mathcal{G}(l_s, v_i | l_p) = \frac{\exp(\mathbf{q}_s^{pT} \cdot \mathbf{u}_i^p)}{\sum_{l_k \in \mathcal{L}_p} \exp(\mathbf{q}_k^{pT} \cdot \mathbf{u}_i^p)}. \qquad (2)$$

Following LINE [Tang *et al.*, 2015b], we can use negative sampling to compute the softmax in Eq. 1, since the

number of nodes $|\mathcal{V}_p|$ can be quite large even in the subnetworks. However, since the number of child labels $|\mathcal{L}_p|$ is often quite small, we can directly compute the softmax in Eq. 2 for better label accuracy. Note that, in each conditional subnetwork, there exist no direct links among labels. Thus, the fine-grained relative distances among child labels under each parent label are learned based on the corresponding network structure, which cannot be inferred from the taxonomy structure itself.

Cross-level learnable encoder: proximity transfer and parameter sharing in the taxonomy. The generator $\mathcal{G}$, without the consideration of label correlations and transferrable information in the taxonomy, can either model all conditional subnetworks essentially in a single embedding space or separately in independent spaces. The key difference lies in the computation of $\mathbf{U}^p$ and $\mathbf{Q}^p$. Since in each conditional subnetwork $\mathcal{B}_p$, we co-embed nodes $\mathcal{V}_p$ and labels $\mathcal{L}_p$ in the space $\mathcal{S}_p$, $\mathbf{U}^p$ and $\mathbf{Q}^p$ can be computed from $\mathbf{U}$ and $\mathbf{Q}$ in the same way. Without loss of generality, we will focus our discussion on the computation of $\mathbf{U}^p$.

Particularly, if $\mathbf{U}^p = \mathbf{U}$, which is shared across all conditional subnetworks, all nodes and labels are essentially flatly arranged in a single embedding space of $\mathbf{U}$, which violates the label hierarchy, resulting in clutter embedding space and underfitting. Otherwise, if we compute a completely different $\mathbf{U}^p$ for each conditional subnetwork, the subnetworks are modeled in independent spaces, which ignores label correlations, leading to large parameter redundancy and overfitting.

As a remedy to this trap, we propose to compute each $\mathbf{U}^p$ as an *encoded version* of $\mathbf{U}$, *i.e.*, $\mathbf{U}^p = \mathcal{A}(\mathbf{U}, l_p)$, so as to essentially transfer proximities captured by different subnetwork generators in the taxonomy. However, since the semantic information in taxonomies is coarse, it is hard to decide how to exactly transfer the proximities. For example, consider the sibling labels of NLP and CV under parent AI. Since NLP communities might be *tighter* than CV as including less diverse subtopics, it should transfer stronger proximity signals. That is, in the subspace of AI, authors close in the subspace of NLP should be closer than those close in the subspace of CV. To capture such subtle semantics in the taxonomy, we require the encoder $\mathcal{A}$ to be *learnable* and *label-dependent*. To this end, we leverage the simple but powerful nonlinear fully connected feedforward neural network (FNN) to model $\mathbf{U}^p$ as

$$\mathbf{U}^p = \mathcal{A}(\mathbf{U}, l_p) = \text{ReLU}(\mathbf{A}_p\mathbf{U}) + \mathbf{b}_p, \quad (3)$$

where $\mathbf{A}_p$ and $\mathbf{b}_p$ are the learnable parameters in the encoder at l_p.

Learning a separate encoder function at each child label does not really leverage the hierarchical structure of $\mathcal{T}$ and still leads to large parameter spaces. To this end, we get motivated by the idea of hierarchical image representation learning [Huang *et al.*, 2017b], which leverages stacked encoders to guide the generation of image representations from high (abstract) to low (detailed) levels. In our scenario, since nodes in the network are connected with labels in the taxonomy, they can also be described by representations at multiple granularities [Ma *et al.*, 2018]. Therefore, we propose to parameterize $\mathcal{A}$ as *nested embedding transformations* following

the hierarchy paths along the taxonomy. For any label l_p, let $l_p \rightarrow \ldots \rightarrow l_j \rightarrow l_i$ denote the path from l_p to a certain leaf label l_i. We have

$$\mathbf{U}^p = \mathcal{A}(\mathbf{U}, l_p) = \mathcal{A}_p(\cdots \mathcal{A}_j(\mathbf{U}, l_j) \cdots, l_p). \quad (4)$$

Note that, the number of parameters in $\mathcal{A}$ grows linearly with the number of labels $|\mathcal{L}|$ in the taxonomy. However, since the main purpose for using $\mathcal{A}$ is to compute multi-granularity node embeddings and separate labels on different levels, it is reasonable to share the parameters of $\mathcal{A}$ among all labels on the same levels of the taxonomy, which reduces the model complexity of $\mathcal{A}$ to $\log |\mathcal{L}|$, and further alleviates possible overfitting due to sparse data in certain subspaces.

Adversarial network discriminator: enable efficient and robust learning. Through subnetwork generation and learnable encoding, we essentially manage to partition the whole network and taxonomy into a set of conditional subnetworks with proper proximity transfer functions. Following the classic heterogeneous network embedding framework of PTE [Tang *et al.*, 2015a], we formulate the overall objective of TAXOGAN into

$$J_{\text{TAXOGAN}} = J_{vl} + \lambda_1 J_{vv} + \lambda_2 J_{ll}, \quad (5)$$

where each of J_{vv}, J_{vl} and J_{ll} is parameterized by the conditional generators and embedding encoders defined in Eq. 1-4.

In practice, we find the joint training of generator networks $\mathcal{G}$ and encoder networks $\mathcal{A}$ to be often inefficient and unstable. Inspired by recent advances in adversarial learning [Gui *et al.*, 2020], we propose to improve the efficiency and robustness of model inference, by designing a novel hierarchical adversarial network discriminator $\mathcal{D}$. Specifically, each of J_{vv}, J_{vl} and J_{ll} can be optimized through a two-player minimax game between $\mathcal{G}$ and $\mathcal{D}$ as defined in [Wang *et al.*, 2018], with the corresponding designs of $\mathcal{G}$ and $\mathcal{A}$ defined in Eqs. 1-4 and $\mathcal{D}$ defined as follows, which measure the log-probability of node-node and node-label links.

$$\mathcal{D}(v_j, v_i | l_p) = \frac{1}{1 + \exp(-\mathbf{u}_j^{pT}\mathbf{u}_i^p)}, \quad (6)$$

$$\mathcal{D}(l_s, v_i | l_p) = \frac{1}{1 + \exp(-\mathbf{q}_s^{pT}\mathbf{u}_i^p)}. \quad (7)$$

3 Experiments

3.1 Datasets

We construct four datasets of real-world networks with explicit taxonomies.

- **DBLP:** We collect the author network[4] with the research topic taxonomy[5]. Undirected uniform links in the network are generated based on coauthorships. A label in the taxonomy is assigned to an author if her/his papers mentions the keyword.

- **Yelp:** We collect the business network[6] with the category taxonomy[7]. Undirected uniform links in the network are

---

[4]https://dblp.uni-trier.de/xml/
[5]https://dl.acm.org/ccs/ccs_flat.cfm
[6]https://www.yelp.com/dataset
[7]https://www.yelp.com/developers/documentation/v3/all_category_list

Model	Micro F1				Macro F1			
	DBLP	Yelp	FreeBase	PubMed	DBLP	Yelp	FreeBase	PubMed
DeepWalk	11.07 ± 0.61	26.24 ± 0.84	26.41 ± 1.12	10.94 ± 1.06	13.11 ± 0.81	4.54 ± 0.97	28.11 ± 1.06	39.37 ± 0.36
GraphGAN	16.10 ± 0.55	26.40 ± 1.21	25.97 ± 0.85	13.68 ± 1.28	16.19 ± 0.71	4.90 ± 1.04	26.65 ± 0.43	40.35 ± 0.44
PTE	16.42 ± 0.47	33.73 ± 0.93	50.27 ± 1.40	12.71 ± 1.64	18.61 ± 0.67	5.47 ± 0.39	28.19 ± 0.31	40.74 ± 0.87
GraphSage	18.72 ± 1.18	29.06 ± 0.29	45.77 ± 0.60	12.05 ± 1.17	16.65 ± 0.72	9.43 ± 1.03	24.06 ± 0.90	36.39 ± 1.09
Poincare	13.87 ± 0.51	29.02 ± 1.12	30.43 ± 1.29	12.73 ± 1.90	18.49 ± 0.51	4.25 ± 1.08	28.57 ± 0.37	40.09 ± 0.33
Nethiex	10.06 ± 0.56	19.44 ± 1.53	35.39 ± 1.37	12.22 ± 1.31	13.86 ± 0.54	4.06 ± 1.03	24.75 ± 0.52	40.83 ± 0.78
TAXOGAN-sin	20.56 ± 0.25	34.88 ± 0.42	65.36 ± 0.59	11.81 ± 1.13	30.44 ± 0.63	13.33 ± 0.39	32.21 ± 0.32	39.86 ± 0.63
TAXOGAN-sep	25.80 ± 1.01	28.47 ± 1.04	63.46 ± 0.46	11.98 ± 0.42	33.37 ± 0.45	11.63 ± 0.95	29.41 ± 0.98	39.86 ± 0.74
TAXOGAN-noadv	29.52 ± 0.79	39.83 ± 1.09	65.79 ± 1.07	16.31 ± 0.22	30.13 ± 0.62	12.74 ± 0.93	31.55 ± 0.62	40.05 ± 0.98
TAXOGAN	$\mathbf{31.97 \pm 1.44}$	$\mathbf{41.37 \pm 0.58}$	$\mathbf{65.98 \pm 0.98}$	$\mathbf{20.11 \pm 1.41}$	$\mathbf{36.42 \pm 0.57}$	$\mathbf{15.19 \pm 0.72}$	$\mathbf{36.62 \pm 0.95}$	$\mathbf{40.89 \pm 0.63}$

Table 1: Performance of all compared algorithms on hierarchical node classification.

generated based on common customers who posted reviews for both businesses. Label assignments are given in the original dataset.

- **FreeBase:** We collect the entity network[8] with the type taxonomy[9]. Undirected uniform links in the network are generated if two entities appear together in any triplet of facts. Labels are assigned by retrieving the nested entity types.

- **PubMed:** We collect the protein network[10] with the disease taxonomy[11]. Undirected uniform links in the network are generated if mentions of two proteins appear in any same sentence. Labels are assigned by surface name matching.

3.2 Performance Evaluations

Table 1 presents the performance of compared algorithms on hierarchical node classification. The improvements of TAXOGAN over the second runners all passed the significance t-test with p-value 0.01. Since the classification at each level in the label taxonomy is multi-class, and deeper labels are harder to be correctly predicted (if any precedent label is predicted wrong, the label path can never reach the correct label), the absolute F1 values are all pretty low. Dataset like Yelp has a lot of deep but narrow labels, which are hard to correctly predict, and the mistakes largely impact the macro F1, whereas dataset like PubMed has a lot of shallow but wide labels, and the mistakes largely impact the micro F1. Thus the suite of datasets and metrics provides a comprehensive evaluation towards the compared algorithms.

The baselines have varying performance across different datasets, while PTE and GraphSage often perform better due to the leverage of labeled data during training. By considering latent hierarchies, Poincare and Nethiex perform better than DeepWalk and GraphGAN in many cases, but their learned latent hierarchies do not always perfectly match the reality and even lead to worse performance in some cases like on DBLP.

Overall, TAXOGAN constantly outperforms all compared algorithms in all cases, with significant margins over the best baseline ranging from 11% to 70%, and the scores all passed

---

t-test with p-value 0.05, demonstrating its superior effectiveness and generalizability. In particular, the improvements of TAXOGAN are more significant when the numbers of labels are larger and the hierarchies of labels are deeper, like with DBLP and Yelp, which supports the appropriate design of our model to leverage the explicit hierarchical structure of associative labels. Note that, while the unsupervised baselines (DeepWalk, GraphGAN, Poincare and Nethiex) do not have access to the node labels in the taxonomy, PTE and GraphSage use the exact same labels as TAXOGAN. This shows TAXOGAN to be effective in modeling hierarchical label spaces, as we will further demonstrate in the ablation study.

For ablation study, our TAXOGAN-sin model has close performance towards the best baselines like PTE, because they are indeed similar only by the difference in adversarial training; our TAXOGAN-sep model does not always outperform TAXOGAN-sin, indicating that even if the evaluation protocol of level-by-level classification may favor multiple embeddings, simply using separate embeddings is not good enough and can harm the performance due to problems like subnetwork sparsity and overfitting, and TAXOGAN-sep is extremely hard to train due to redundant parameters and large memory cost; our TAXOGAN-noadv model is the nested space model without adversarial training, which outperforms TAXOGAN-sep with significant margins, corroborating the effectiveness of our model design with connected subspaces through base and transformed embeddings; our TAXOGAN model further outperforms TAXOGAN-noadv, directly showing the advantage of our novel hierarchical adversarial training technique.

For more experiment and case study results, please refer to our full paper published in ICDM 2020 [Yang *et al.*, 2020c].

4 Conclusion

To the best of our knowledge, we are the first to jointly model networks and taxonomies. By stressing the important properties of conditional node proximity and hierarchical label proximity, we develop TAXOGAN, which computes high-quality network embedding under the guidance of hierarchical labels, while in turn produce fine-grained label embedding. Extensive experimental results and interpretable case studies demonstrate the advantages of TAXOGAN in both traditional network mining tasks and unique novel applications.

References

[Alsuhaibani *et al.*, 2018] Mohammed Alsuhaibani, Takanori Maehara, and Danushka Bollegala. Joint learning of hierarchical word embeddings from a corpus and a taxonomy. In *AKBC*, 2018.

[Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *TPAMI*, 35(8):1798–1828, 2013.

[Dai *et al.*, 2019] Quanyu Dai, Xiao Shen, Liang Zhang, Qiang Li, and Dan Wang. Adversarial training methods for network embedding. In *WWW*, 2019.

[Downey *et al.*, 2015] Doug Downey, Chandra Bhagavatula, and Yi Yang. Efficient methods for inferring large sparse topic hierarchies. In *ACL*, 2015.

[Gao *et al.*, 2019] Hongchang Gao, Jian Pei, and Heng Huang. Progan: Network embedding via proximity generative adversarial network. In *KDD*, 2019.

[Gui *et al.*, 2020] Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, and Jieping Ye. A review on generative adversarial networks: Algorithms, theory, and applications. *arXiv preprint arXiv:2001.06937*, 2020.

[Hamilton *et al.*, 2017] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NIPS*, 2017.

[Huang *et al.*, 2017a] Xiao Huang, Jundong Li, and Xia Hu. Accelerated attributed network embedding. In *SDM*, 2017.

[Huang *et al.*, 2017b] Xun Huang, Yixuan Li, Omid Poursaeed, John Hopcroft, and Serge Belongie. Stacked generative adversarial networks. In *CVPR*, 2017.

[Kipf and Welling, 2017] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[Liu *et al.*, 2018] Ninghao Liu, Xiao Huang, Jundong Li, and Xia Hu. On interpretation of network embedding via taxonomy induction. In *KDD*, 2018.

[Ma *et al.*, 2018] Jianxin Ma, Peng Cui, Xiao Wang, and Wenwu Zhu. Hierarchical taxonomy aware network embedding. In *KDD*, 2018.

[Meng *et al.*, 2019] Zaiqiao Meng, Shangsong Liang, Hongyan Bao, and Xiangliang Zhang. Co-embedding attributed networks. In *WSDM*, 2019.

[Nguyen *et al.*, 2017] Kim Anh Nguyen, Maximilian Köper, Sabine Schulte im Walde, and Ngoc Thang Vu. Hierarchical embeddings for hypernymy detection and directionality. In *EMNLP*, 2017.

[Nickel and Kiela, 2017] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *NIPS*, 2017.

[Park *et al.*, 2017] Jisoo Park, Benjamin J. Hescott, and Donna K. Slonim. Towards a more molecular taxonomy of disease. *Journal of Biomedical Semantics*, 8(1):25, 2017.

[Peng *et al.*, 2018] Hao Peng, Jianxin Li, Yu He, Yaopeng Liu, Mengjiao Bao, Lihong Wang, Yangqiu Song, and Qiang Yang. Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In *WWW*, 2018.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *KDD*, 2014.

[Tang *et al.*, 2015a] Jian Tang, Meng Qu, and Qiaozhu Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *KDD*, 2015.

[Tang *et al.*, 2015b] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *WWW*, 2015.

[Vulić and Mrkšić, 2018] Ivan Vulić and Nikola Mrkšić. Specialising word vectors for lexical entailment. In *ACL*, 2018.

[Wang *et al.*, 2015] Chi Wang, Jialu Liu, Nihit Desai, Marina Danilevsky, and Jiawei Han. Constructing topical hierarchies in heterogeneous information networks. *Knowledge and Information Systems*, 44(3):529–558, 2015.

[Wang *et al.*, 2018] Hongwei Wang, Jia Wang, Jialin Wang, Miao Zhao, Weinan Zhang, Fuzheng Zhang, Xing Xie, and Minyi Guo. Graphgan: Graph representation learning with generative adversarial nets. In *AAAI*, 2018.

[Wehrmann *et al.*, 2018] Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. Hierarchical multi-label classification networks. In *ICML*, 2018.

[Yang *et al.*, 2017] Carl Yang, Lin Zhong, Li-Jia Li, and Luo Jie. Bi-directional joint inference for user links and attributes on large social graphs. In *WWW*, 2017.

[Yang *et al.*, 2018] Carl Yang, Mengxiong Liu, Vincent W Zheng, and Jiawei Han. Node, motif and subgraph: Leveraging network functional blocks through structural convolution. In *ASONAM*, 2018.

[Yang *et al.*, 2019] Carl Yang, Peiye Zhuang, Wenhan Shi, Alan Luu, and Pan Li. Conditional structure generation through graph variational generative adversarial nets. In *NIPS*, 2019.

[Yang *et al.*, 2020a] Carl Yang, Aditya Pal, Andrew Zhai, Nikil Pancha, , Jiawei Han, Chuck Rosenberg, and Jure Leskovec. Multisage: Empowering graphsage with contextualized multi-embedding on web-scale multipartite networks. In *KDD*, 2020.

[Yang *et al.*, 2020b] Carl Yang, Yuxin Xiao, Yu Zhang, Yizhou Sun, and Jiawei Han. Heterogeneous network representation learning: A unified framework with survey and benchmark. In *TKDE*, 2020.

[Yang *et al.*, 2020c] Carl Yang, Jieyu Zhang, and Jiawei Han. Co-embedding network nodes and hierarchical labels with taxonomy based generative adversarial nets. In *ICDM*, 2020.

[Zhang *et al.*, 2018] Chao Zhang, Fangbo Tao, Xiusi Chen, Jiaming Shen, Meng Jiang, Brian Sadler, Michelle Vanni, and Jiawei Han. Taxogen: Unsupervised topic taxonomy construction by adaptive term embedding and clustering. In *KDD*, 2018.