Detecting Multi-Label Out-of-Distribution Nodes on Graphs

Ruomeng Ding[1], **Xujiang Zhao**[2], **Chen Zhao**[3], **Minglai Shao**[1]*

[1] Tianjin University
[2] NEC Laboratories America
[3] University of Texas at Dallas
rmding@tju.edu.cn, xuzhao@nec-labs.com, charliezhaoyinpeng@gmail.com, shaoml@tju.edu.cn

Abstract

Research and applications involving Out-of-Distribution Detection (OOD) on graph-structured data are proving critical. Existing OOD detection methods on graphs do not apply to multi-label settings. There are other semi-supervised node classification methods that do not distinguish OOD nodes from those in distribution (ID). This paper proposes an Evidence-Based Out-of-Distribution Detection method for multi-label graphs based on Evidential Deep Learning (EDL). The evidence for multiple labels is predicted by Multi-Label Evidential Graph Neural Networks (ML-EGNNs) with Beta Loss. Multi-label opinions are fused using the Joint Belief by comultiplication. As an additional step, we introduce a kernel-based node positive evidence estimation (KNPE) method that is designed to reduce errors in estimating positive evidence. The results of our experiments show that our multi-label OOD detection model is both effective and efficient.

Introduction

Many real-world application scenarios can be represented by graph-structured data, ranging from natural networks to social networks. In graph scenarios, there are usually only a subset of nodes are labeled, and the inherent multi-label properties of nodes are inevitable. For example, in social networks, one user usually has more than one interest (Wang and Sukthankar 2013). In a Protein-Protein-Interaction (PPI) network, one protein can perform multiple functions (Wu et al. 2014). Further, if unlabeled nodes are ubiquitous, then the existence of unknown labels should be unavoidable, *i.e.*, some unlabeled nodes may be out-of-distribution (OOD). As shown in Fig 1, in a PPI network, Function 3 and Function 4 are unseen for Labeled Node A, B and C. A multi-class classification method classifies OOD Unlabeled Node H and F into one or more In-Distribution Functions(like Function 1 and Function 2). This leads to the model unable to detect the unknown functions. Hence, it is necessary to study the OOD detection problem on multi-label graphs.

Recently, some semi-supervised learning methods have been proposed for multi-label node classification on graphs (Song et al. 2021; Zhou et al. 2021; Akujuobi et al. 2019), with the purpose of predicting user interests in social networks or identifying functions of proteins in PPI networks. However, these methods cannot distinguish OOD nodes from in-distribution (ID) nodes. Due to the lack of uncertainty modeling, they will confidently tag an OOD node only with ID classes from training data without giving useful estimates of their predictive uncertainty (Ovadia et al. 2019).

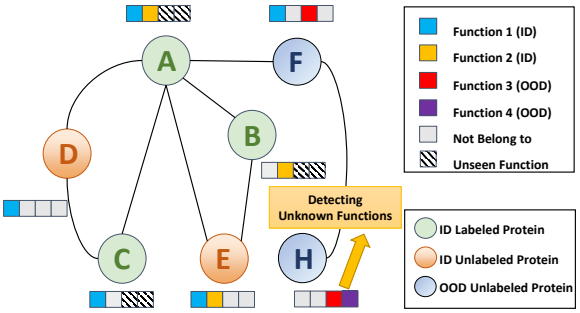

Figure 1: For a Protein-Protein-Interaction network, nodes represent proteins, edges connect pairs of interacting proteins, labels indicate different functions of proteins. There are three kinds of nodes: In-Distribution Labeled Protein A, B and C for training; In-Distribution Unlabeled Protein D and E; Out-of-Distribution Unlabeled Protein F and H. During the training process, Function 3 and 4 are unseen to model.

Some OOD detection methods (Rong et al. 2019; Hasanzadeh et al. 2020; Elinas, Bonilla, and Tiao 2020) based on uncertainty estimation (Gal and Ghahramani 2016; Lakshminarayanan, Pritzel, and Blundell 2017; Liu et al. 2020) are available for multi-class graphs with one class per node. Besides, there are some evidence-based methods (Zhao et al. 2020; Stadler et al. 2021) proposed for OOD detection on multi-class graphs with a Dirichlet distribution as conjugate prior (Sensoy, Kaplan, and Kandemir 2018). Nevertheless, such methods are not applicable for multi-label graphs. That is because classification probabilities in multi-label setting follow binomial distributions, not a categorical distribution, whose prior is the Beta distribution but not the Dirichlet distribution.

To address aforementioned problems, we propose a

*Corresponding author

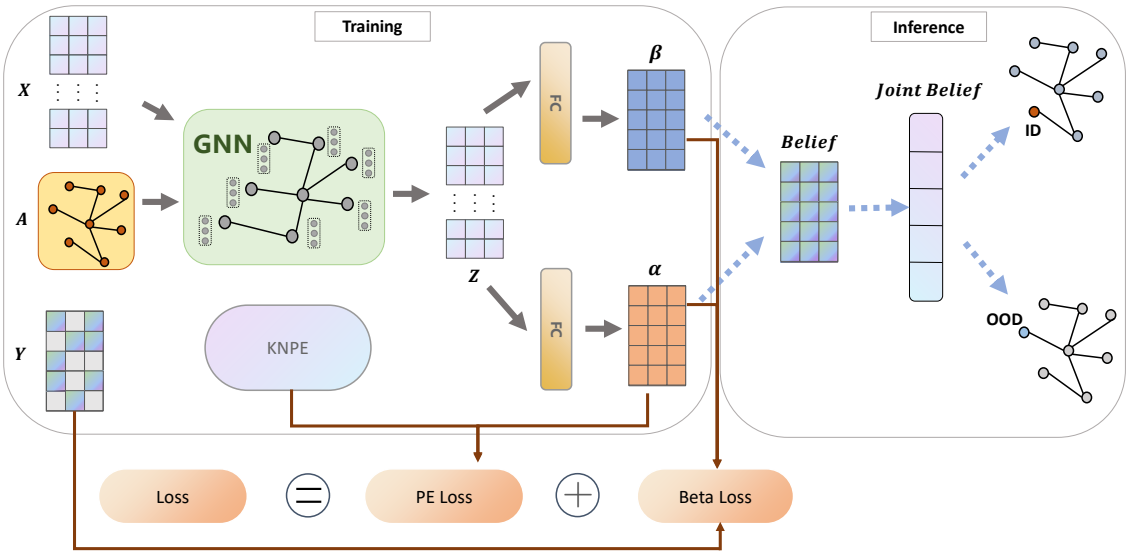

Figure 2: Overall framework of our proposed method ML-EGNNs for training and inference.

novel evidence based OOD detection method on multi-label graphs. Based on Subjective Logic (JSANG 2018), *Evidence* is the amount of support collected from data to suggest that a sample should (or should not) be classified into a specific class. Under multi-label setting, for each ID class, we define *positive evidence* as a measure of the confidence to classify a sample into this class. While, *negative evidence* is used to quantify the objections. In summary, the contribution of this paper is three-fold:

- We propose a novel problem of out-of-distribution (OOD) detection on the multi-label graph and develop a novel evidential method for node-level OOD detection. To the best of our knowledge, this is the first study to detect OOD nodes with multiple labels on graphs.
- We introduce Multi-Label Evidential Graph Neural Networks (ML-EGNNs) with Beta loss to predict uncertainty for multiple classes. Besides, we define *joint belief* for multi-label opinions fusion. Additionally, we develop a Kernel-based Node Positive Evidence Estimation (KNPE) method to reduce errors in quantifying positive evidence.
- Experimental results show both the effectiveness and efficiency of our model on multi-label OOD detection.

Methodology

Problem Formulation

Given a multi-label graph $\mathcal{G} = (V, E, \mathbf{A}, \mathbf{X}, \mathbf{Y}_L)$ consisting of a set of nodes $V = \{1, ..., N\}$ and a set of edges $E \subset V \times V$, where the connections in $\mathcal{G}$ can be represented by the adjacency matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$. $\mathbf{X} = [\mathbf{x}_1^T, \mathbf{x}_2^T, ..., \mathbf{x}_N^T]$ is the node feature matrix. $\mathbf{Y}_L = \{\mathbf{y_i} | i \in L\}$ are the labels of the training nodes $L \subset V$. $\mathbf{y}_i = [0, 1]^K$ is the class label of node i, where K is the number of in-distribution classes. Following the semi-supervised learning pattern, among all the nodes, L are labeled nodes while the remaining $U = V \backslash L$ are unlabeled. $U = U_{ID} + U_{OOD}$, where U_{ID} denotes unlabeled ID nodes and U_{OOD} denotes unlabeled OOD nodes. Here we only consider U_{OOD} as nodes which do not have any labels in K known classes. **We aim to predict:** (1) the **class probabilities** of U: $\mathbf{p}_U = \{\mathbf{p}_i \in [0, 1]^K | i \in U\}$; (2) **the belief estimates**: the joint belief of U: $\mathbf{b}_U = \{\mathbf{b}_i \in [0, 1] | i \in U\}$, where $\mathbf{b}_i$ indicates the confidence in dividing node i into ID samples.

Multi-Label Evidential Graph Neural Networks (ML-EGNNs)

Multi-Label Evidence Estimation. Compared with classical neural networks, Evidential Neural Networks (ENNs) (Sensoy, Kaplan, and Kandemir 2018) (Hu et al. 2021) do not have a softmax layer, but use an activation layer (e.g., ReLU) to make sure that the output is non-negative. To be specific, as shown in Fig.2, Multi-Label Evidential Graph Neural Networks (ML-EGNNs) are built by stacking graph convolutional layers and two fully connected layers (FCs) and ReLU layers, which are taken as the positive and negative evidence vectors for Beta distribution respectively. Given sample i, let $f_{pos}(\mathbf{X}, \mathbf{A}|\theta)$ and $f_{neg}(\mathbf{X}, \mathbf{A}|\theta)$ represent the positive and negative evidence vectors predicted by ML-EGNNs, where θ represents the network parameters. Then, the two parameters $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ of Beta distribution for node i:

$$\begin{aligned} \boldsymbol{\alpha}_i &= f_{pos}(\mathbf{X}, \mathbf{A}|\theta) + \mathbf{1}, \\ \boldsymbol{\beta}_i &= f_{neg}(\mathbf{X}, \mathbf{A}|\theta) + \mathbf{1}. \end{aligned} \tag{1}$$

where k indicates the k-th class of total K classes.

Training Loss. With N training samples and K different classes, a multi-label evidential neural network is trained by minimizing the Beta loss:

$$\mathcal{L}_{Beta} = \sum_{i=1}^{N} \sum_{k=1}^{K} \int \left[\mathbf{BCE}\left(y_{ik}, p_{ik}\right)\right] B\left(\alpha_{ik}, \beta_{ik}\right) dp_{ik}, \tag{2}$$

where $B(\alpha_{ik}, \beta_{ik})$ is a 2-dimensional Beta function. $\mathbf{BCE}(\cdot)$ denotes the Binary Cross Entropy Loss. Besides, as the belief and disbelief of label k for sample i, we have:

$$b_{ik} = \frac{\alpha_{ik} - 1}{\alpha_{ik} + \beta_{ik}}, \qquad d_{ik} \quad = \frac{\beta_{ik} - 1}{\alpha_{ik} + \beta_{ik}}. \tag{3}$$

So far, for in-distribution multi-label classification, we set the positive belief as the probability of class i for sample j, *i.e.*, $\frac{\alpha_{ik}-1}{\alpha_{ik}+\beta_{ik}}$, without additional time consuming.

Kernel-based Node Positive Evidence Estimation (KNPE)

We focus on the estimation the prior information of multi-label evidence. For each pair of training nodes i and j, calculate the node-level distance d_{ij}, *i.e.*, the shortest path between nodes i and j. Then the Gaussian kernel function is used to estimate the positive distribution effect between nodes i and j:

$$g\left(d_{ij}\right) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right), \tag{4}$$

where σ is the bandwidth parameter. The contribution of positive evidence estimation for node i from labeled node j is $\boldsymbol{h_{ij}}(\mathbf{y}_i, d_{ij}) = [h_{ij}^1, h_{ij}^2, ..., h_{ij}^k, ..., h_{ij}^K]$. And h_{ij}^k is obtained by:

$$h_{ij}^k = \begin{cases} 0 & y_{jk} = 0, \\ g\left(d_{ij}\right) & y_{jk} = 1, \end{cases} \tag{5}$$

where $\mathbf{y}_j = [y_{j1}, ..., y_{jk}, ..., y_{jK}] = [0, 1]^K$ represents the ID labels of training node j. The prior positive parameter is estimated as:

$$\hat{\boldsymbol{\alpha}}_{\boldsymbol{i}} = \sum_{j \in L} \boldsymbol{h_{ij}}(\mathbf{y}_j, d_{ij}) + \mathbf{1}, \tag{6}$$

where L is the set of labeled nodes. During the training process, we minimize $\mathcal{L}_{PE} = \sum_{i=1}^{N} \hat{\boldsymbol{\alpha}}_{\boldsymbol{i}} \log \frac{\hat{\boldsymbol{\alpha}}_{\boldsymbol{i}}}{\boldsymbol{\alpha}_{\boldsymbol{i}}}$. The total loss function we use to optimize the model is:

$$\mathcal{L}_{total} = \mathcal{L}_{Beta} + \lambda \cdot \mathcal{L}_{PE}, \tag{7}$$

where λ denotes a trade-off parameter with $\mathcal{L}_{PE}$.

Multi-Label Opinions Fusion

After obtaining separate beliefs of multiple labels, we need to combine these opinions and quantify a integrate opinion, *i.e.*, Opinions Fusion. Note that, if a sample belongs to any label we already know, then it is an ID sample. In other words, only samples that do not belong to any known category should be classified as OOD samples. Hence, naive operations like summing up all the beliefs are inapplicable for multi-label setting.

Multi-Label Joint Belief. Inspired by the multiplication in Subjective Logic (JSANG 2018), a multi-label opinion $\boldsymbol{\Omega} = \omega_1 \vee \omega_2 \vee \cdots \vee \omega_K$. Based on that, the multi-label joint belief over all classes is defined as:

$$\mathbf{b} = b_1 \vee b_2 \vee \cdots \vee b_K. \tag{8}$$

To be specific, we can formulate their joint belief $b_{m \vee n} = b_m + b_n - b_m b_n$.

Experiments

Datasets

The data used to validate our model are required to be graph-structured and multi-labeled. We collect 5 public available benchmark datasets to perform our experiments including DBLP (Akujuobi et al. 2019), Facebook (Zhou et al. 2021), BlogCatalog (Chen et al. 2018), Flickr (Tang and Liu 2009), and Yeast (Cheng et al. 2002). The major details of the datasets are listed in Table1. $|\mathbf{V}|$, $|\mathbf{E}|$ and $|\mathbf{Y}|$ represent the number of nodes, the number of edges, and the number of labels, respectively. $|\mathbf{X}|$ denote the dimensions of node features. $|\mathbf{Y}_{id}|$ and $|\mathbf{Y}_{ood}|$ denote the number of ID classes and OOD classes, respectively. $|\mathbf{N}_{id}|$ and $|\mathbf{N}_{ood}|$ denote the number of ID nodes and OOD nodes, respectively.

Baselines and Configurations

The effectiveness of our method is validated using 3 well-known graph neural network models as backbone: GCN (Kipf and Welling 2016), GAT (Velickovic et al. 2017) and GraphSAGE (Hamilton, Ying, and Leskovec 2017). Since they are the most representative models according to the types of aggregators. Besides, we compare our method with three state-of-the-art multi-label classification methods, MLGW (Akujuobi et al. 2019), LANC (Zhou et al. 2021) and MLGD (Song et al. 2021). Also, two OOD detection methods, MC-Dropout (Dropout) (Gal and Ghahramani 2016) (Ryu, Kwon, and Kim 2019) and Deep Ensembles (Ensemble) (Lakshminarayanan, Pritzel, and Blundell 2017), which can be applied on graphs are compared with our method. For the part of our method, ML-EGNNs use 2 fully connected layers and ReLU layers to obtain the positive evidence and negative evidence, respectively.

Multi-Label OOD Detection

For multi-label OOD detection, TABLE2 shows the performance of each comparing method (mean ± std) in terms of AUC, respectively. For each backbone, the top-1 model is bolded. The results show that our method improve the performance of multi-label OOD detection over all 3 backbones. That is because all the backbones are optimized by BCE loss with softmax layers forehead. Without the constraint of Beta prior and ReLU layers to output evidence, it is difficult to distinguish OOD nodes effectively only according to the prediction probability. For the multi-label classification methods, they are not designed for OOD setting with a lack of evaluating uncertainty. Therefore, the performance of these classification methods in multi-lable OOD detection is basically the same as that of backbones.

Moreover, compared to Dropout and Ensemble, our method has better and more stable performance, though it is slightly inferior on Facebook and BlogCatalog with GAT and GraphSAGE as backbones. We think this is acceptable due to the characteristics of different datasets and the stable performance of our method on the whole. Dropout and Ensemble are widely used for OOD detection. Though they can be applied on graphs, they still have the defect of being unable to model multi-label problems. Generally, our method works better on multiple datasets and different backbones

Dataset	$\|\mathbf{V}\|$	$\|\mathbf{E}\|$	$\|\mathbf{Y}\|$	$\|\mathbf{X}\|$	$\|\mathbf{Y}_{id}\|$	$\|\mathbf{Y}_{ood}\|$	$\|\mathbf{N}_{id}\|$	$\|\mathbf{N}_{ood}\|$
DBLP	28,702	68,335	4	300	3	1	21, 553	4,539
Facebook	792	14,024	17	319	14	3	524	243
BlogCatalog	10,312	333,983	39	128	25	14	8,513	1,037
Flickr	80,513	5,899,882	195	128	150	45	57,185	14,775
Yeast	681	910	13	200	5	8	138	13

Table 1: Details of 5 benchmark multi-label graph-structured datasets.

Backbone	Method	AUC				
		DBLP	**Facebook**	**BlogCatalog**	**Flickr**	**Yeast**
GCN	**Backbone**	0.518 ±0.006	0.823 ± 0.012	0.423 ± 0.013	0.450 ± 0.006	0.698 ± 0.021
	Dropout	0.634 ±0.002	0.503 ± 0.009	0.536 ± 0.010	0.500 ± 0.007	0.530 ± 0.018
	Ensemble	0.643 ±0.002	0.507 ± 0.006	0.504 ± 0.004	0.500 ± 0.007	0.583 ± 0.033
	Ours	**0.655 ± 0.004**	**0.846 ± 0.048**	**0.612 ± 0.021**	**0.552 ± 0.010**	**0.746 ± 0.021**
GAT	**Backbone**	0.422 ± 0.002	0.425 ± 0.003	0.464 ± 0.001	0.497 ± 0.004	0.646 ± 0.016
	Dropout	0.759 ± 0.001	0.913 ± 0.021	**0.612 ± 0.027**	0.484 ± 0.008	0.542 ± 0.061
	Ensemble	0.757 ± 0.003	0.920 ± 0.008	0.577 ± 0.002	0.486 ± 0.003	0.588 ± 0.073
	Ours	**0.811 ± 0.008**	**0.922 ± 0.028**	0.565 ± 0.028	**0.512 ± 0.002**	**0.763 ± 0.005**
GraphSAGE	**Backbone**	0.489 ± 0.006	0.326 ± 0.041	0.501 ± 0.001	0.500 ± 0.006	0.641 ± 0.023
	Dropout	0.768 ± 0.001	**0.957 ± 0.007**	**0.698 ± 0.001**	0.492 ± 0.008	0.637 ± 0.065
	Ensemble	0.762 ± 00013	0.956 ± 0.005	0.697 ± 0.006	0.492 ± 0.005	0.612 ± 0.020
	Ours	**0.796 ± 0.001**	0.937 ± 0.028	0.615 ± 0.021	**0.528 ± 0.008**	**0.741 ± 0.003**
-	**MLGW**	0.566 ± 0.004	0.497 ± 0.031	0.502 ± 0.002	0.495 ± 0.010	0.538 ± 0.042
	LANC	0.494 ± 0.049	0.681 ± 0.008	0.478 ± 0.009	0.507 ± 0.009	0.568 ± 0.014
	MLGD	0.512 ± 0.003	0.689 ± 0.007	0.508 ± 0.024	0.511 ± 0.011	0.615 ± 0.014

Table 2: The performance for multi-label OOD detection in terms of AUC (mean ± std).

which proves the effectiveness and the generalization ability of our model on different benchmarks.

Conclusion

In this work, we first propose and formulate the multi-label OOD detection problem on graphs. To address this problem, we introduce a novel evidential method, Multi-Label Evidential Graph Neural Networks (ML-EGNNs) , to predict uncertainty for multiple classes. Our interpretation of joint belief combining multiple classes incorporates the idea of multiplication in Subjective Logic. Besides, a Kernel-based Node Positive Evidence Estimation (KNPE) method is applied for estimating prior evidence. Experimental results prove both the effectiveness and efficiency of our method. Our study considers OOD nodes which only contain OOD labels. In the future, we will leverage detection on nodes that contain both ID labels and OOD labels under multi-label setting, which is a more challenging and untouched issue.

References

Akujuobi, U.; Yufei, H.; Zhang, Q.; and Zhang, X. 2019. Collaborative graph walk for semi-supervised multi-label node classification. In *2019 IEEE International Conference on Data Mining (ICDM)*, 1–10. IEEE.

Chen, H.; Perozzi, B.; Hu, Y.; and Skiena, S. 2018. Harp: Hierarchical representation learning for networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Cheng, J.; Hatzis, C.; Hayashi, H.; Krogel, M.-A.; Morishita, S.; Page, D.; and Sese, J. 2002. KDD Cup 2001 report. *ACM SIGKDD Explorations Newsletter*, 3(2): 47–64.

Elinas, P.; Bonilla, E. V.; and Tiao, L. 2020. Variational inference for graph convolutional networks in the absence of graph data and adversarial settings. *Advances in Neural Information Processing Systems*, 33: 18648–18660.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, 1050–1059. PMLR.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Hasanzadeh, A.; Hajiramezanali, E.; Boluki, S.; Zhou, M.; Duffield, N.; Narayanan, K.; and Qian, X. 2020. Bayesian graph neural networks with adaptive connection sampling. In *International conference on machine learning*, 4094–4104. PMLR.

Hu, Y.; Ou, Y.; Zhao, X.; Cho, J.-H.; and Chen, F. 2021. Multidimensional uncertainty-aware evidential neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 7815–7822.

JSANG, A. 2018. *Subjective Logic: A formalism for reasoning under uncertainty*. Springer.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30.

Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 33: 21464–21475.

Ovadia, Y.; Fertig, E.; Ren, J.; Nado, Z.; Sculley, D.; Nowozin, S.; Dillon, J.; Lakshminarayanan, B.; and Snoek, J. 2019. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32.

Rong, Y.; Huang, W.; Xu, T.; and Huang, J. 2019. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903*.

Ryu, S.; Kwon, Y.; and Kim, W. Y. 2019. Uncertainty quantification of molecular property prediction with Bayesian neural networks. *arXiv preprint arXiv:1903.08375*.

Sensoy, M.; Kaplan, L.; and Kandemir, M. 2018. Evidential deep learning to quantify classification uncertainty. *Advances in neural information processing systems*, 31.

Song, Z.; Meng, Z.; Zhang, Y.; and King, I. 2021. Semi-supervised Multi-label Learning for Graph-structured Data. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 1723–1733.

Stadler, M.; Charpentier, B.; Geisler, S.; Zügner, D.; and Günnemann, S. 2021. Graph posterior network: Bayesian predictive uncertainty for node classification. *Advances in Neural Information Processing Systems*, 34: 18033–18048.

Tang, L.; and Liu, H. 2009. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 817–826.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *stat*, 1050: 20.

Wang, X.; and Sukthankar, G. 2013. Multi-label relational neighbor classification using social context features. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 464–472.

Wu, Q.; Ye, Y.; Ho, S.-S.; and Zhou, S. 2014. Semi-supervised multi-label collective classification ensemble for functional genomics. *BMC genomics*, 15(9): 1–14.

Zhao, X.; Chen, F.; Hu, S.; and Cho, J.-H. 2020. Uncertainty aware semi-supervised learning on graph data. *Advances in Neural Information Processing Systems*, 33: 12827–12836.

Zhou, C.; Chen, H.; Zhang, J.; Li, Q.; Hu, D.; and Sheng, V. S. 2021. Multi-label graph node classification with label attentive neighborhood convolution. *Expert Systems with Applications*, 180: 115063.