Uncertainty Aware Semi-Supervised Learning on Graph Data

Xujiang Zhao[1], **Feng Chen**[1], **Shu Hu**[2], **Jin-Hee Cho**[3]
[1]The University of Texas at Dallas, {xujiang.zhao, feng,chen}@utdallas.edu
[2]University at Buffalo, SUNY, shuhu@buffalo.edu
[3]Virginia Tech, jicho@vt.edu

Abstract

Thanks to graph neural networks (GNNs), semi-supervised node classification has shown the state-of-the-art performance in graph data. However, GNNs have not considered different types of uncertainties associated with class probabilities to minimize risk of increasing misclassification under uncertainty in real life. In this work, we propose a multi-source uncertainty framework using a GNN that reflects various types of predictive uncertainties in both deep learning and belief/evidence theory domains for node classification predictions. By collecting evidence from the given labels of training nodes, the *Graph-based Kernel Dirichlet distribution Estimation* (GKDE) method is designed for accurately predicting node-level Dirichlet distributions and detecting out-of-distribution (OOD) nodes. We validated the outperformance of our proposed model compared to the state-of-the-art counterparts in terms of misclassification detection and OOD detection based on six real network datasets. We found that dissonance-based detection yielded the best results on misclassification detection while vacuity-based detection was the best for OOD detection. To clarify the reasons behind the results, we provided the theoretical proof that explains the relationships between different types of uncertainties considered in this work.

1 Introduction

Inherent uncertainties derived from different root causes have realized as serious hurdles to find effective solutions for real world problems. Critical safety concerns have been brought due to lack of considering diverse causes of uncertainties, resulting in high risk due to misinterpretation of uncertainties (e.g., misdetection or misclassification of an object by an autonomous vehicle). Graph neural networks (GNNs) [16, 30] have received tremendous attention in the data science community. Despite their superior performance in semi-supervised node classification and regression, they didn't consider various types of uncertainties in the their decision process. Predictive uncertainty estimation [14] using Bayesian NNs (BNNs) has been explored for classification prediction and regression in the computer vision applications, based on aleatoric uncertainty (AU) and epistemic uncertainty (EU). AU refers to data uncertainty from statistical randomness (e.g., inherent noises in observations) while EU indicates model uncertainty due to limited knowledge (e.g., ignorance) in collected data. In the belief or evidence theory domain, Subjective Logic (SL) [12] considered vacuity (or a lack of evidence or ignorance) as uncertainty in a subjective opinion. Recently other uncertainty types, such as dissonance, consonance, vagueness, and monosonance [12], have been discussed based on SL to measure them based on their different root causes.

We first considered multidimensional uncertainty types in both deep learning (DL) and belief and evidence theory domains for node-level classification, misclassification detection, and out-of-distribution (OOD) detection tasks. By leveraging the learning capability of GNNs and considering multi-dimensional uncertainties, we propose a uncertainty-aware estimation framework by quantifying

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

different uncertainty types associated with the predicted class probabilities. In this work, we made the following **key contributions**:

- **A multi-source uncertainty framework for GNNs**. Our proposed framework first provides the estimation of various types of uncertainty from both DL and evidence/belief theory domains, such as dissonance (derived from conflicting evidence) and vacuity (derived from lack of evidence). In addition, we designed a Graph-based Kernel Dirichlet distribution Estimation (GKDE) method to reduce errors in quantifying predictive uncertainties.
- **Theoretical analysis**: Our work is the first that provides a theoretical analysis about the relationships between different types of uncertainties considered in this work. We demonstrate via a theoretical analysis that an OOD node may have a high predictive uncertainty under GKDE.
- **Comprehensive experiments for validating the performance of our proposed framework**: Based on the six real graph datasets, we compared the performance of our proposed framework with that of other competitive counterparts. We found that the dissonance-based detection yielded the best results in misclassification detection while vacuity-based detection best performed in OOD detection.

Note that we use the term 'predictive uncertainty' in order to mean uncertainty estimated to solve prediction problems.

2 Related Work

DL research has mainly considered *aleatoric* uncertainty (AU) and *epistemic* uncertainty (EU) using BNNs for computer vision applications. AU consists of homoscedastic uncertainty (i.e., constant errors for different inputs) and heteroscedastic uncertainty (i.e., different errors for different inputs) [5]. A Bayesian DL framework was presented to simultaneously estimate both AU and EU in regression (e.g., depth regression) and classification (e.g., semantic segmentation) tasks [14]. Later, *distributional uncertainty* was defined based on distributional mismatch between testing and training data distributions [20]. *Dropout variational inference* [7] was used for an approximate inference in BNNs using epistemic uncertainty, similar to *DropEdge* [23]. Other algorithms have considered overall uncertainty in node classification [3, 18, 32]. However, no prior work has considered uncertainty decomposition in GNNs.

In the belief (or evidence) theory domain, uncertainty reasoning has been substantially explored, such as Fuzzy Logic [1], Dempster-Shafer Theory (DST) [27], or Subjective Logic (SL) [11]. Belief theory focuses on reasoning inherent uncertainty in information caused by unreliable, incomplete, deceptive, or conflicting evidence. SL considered predictive uncertainty in subjective opinions in terms of *vacuity* (i.e., a lack of evidence) and *vagueness* (i.e., failing in discriminating a belief state) [11]. Recently, other uncertainty types have been studied, such as *dissonance* caused by conflicting evidence[12]. In the deep NNs, [26] proposed evidential deep learning (EDL) model, using SL to train a deterministic NN for supervised classification in computer vision based on the sum of squared loss. However, EDL didn't consider a general method of estimating multidimensional uncertainty or graph structure.

3 Multidimensional Uncertainty and Subjective Logic

This section provides an overview of SL and discusses multiple types of uncertainties estimated based on SL, called *evidential uncertainty*, with the measures of *vacuity* and *dissonance*. In addition, we give a brief overview of *probabilistic uncertainty*, discussing the measures of *aleatoric* uncertainty and *epistemic* uncertainty.

3.1 Subjective Logic

A multinomial opinion of a random variable y is represented by $\omega = (\boldsymbol{b}, u, \boldsymbol{a})$ where a domain is $\mathbb{Y} \equiv \{1, \cdots, K\}$ and the additivity requirement of ω is given as $\sum_{k \in \mathbb{Y}} b_k + u = 1$. To be specific, each parameter indicates,

- $\boldsymbol{b}$: *belief mass distribution* over $\mathbb{Y}$ and $\boldsymbol{b} = [b_1, \ldots, b_K]^T$;
- u: *uncertainty mass* representing *vacuity of evidence*;
- $\boldsymbol{a}$: *base rate distribution* over $\mathbb{Y}$ and $\boldsymbol{a} = [a_1, \ldots, a_K]^T$.

The projected probability distribution of a multinomial opinion can be calculated as:

$$P(y = k) = b_k + a_k u, \quad \forall k \in \mathbb{Y}. \tag{1}$$

A multinomial opinion ω defined above can be equivalently represented by a K-dimensional Dirichlet probability density function (PDF), where the special case with $K = 2$ is the Beta PDF as a binomial opinion. Let $\boldsymbol{\alpha}$ be a strength vector over the singletons (or classes) in $\mathbb{Y}$ and $\mathbf{p} = [p_1, \cdots, p_K]^T$ be a probability distribution over $\mathbb{Y}$. The Dirichlet PDF with $\mathbf{p}$ as a random vector K-dimensional variables is defined by:

$$\text{Dir}(\boldsymbol{p}|\boldsymbol{\alpha}) = \frac{1}{B(\boldsymbol{\alpha})} \prod_{k \in \mathbb{Y}} p_k^{(\alpha_k - 1)}, \tag{2}$$

where $\frac{1}{B(\boldsymbol{\alpha})} = \frac{\Gamma(\sum_{k \in \mathbb{Y}} \alpha_k)}{\prod_{k \in \mathbb{Y}} (\alpha_k)}$, $\alpha_k \geq 0$, and $p_k \neq 0$, if $\alpha_k < 1$.

The term *evidence* is introduced as a measure of the amount of supporting observations collected from data that a sample should be classified into a certain class. Let e_k be the evidence derived for the class $k \in \mathbb{Y}$. The total strength α_k for the belief of each class $k \in \mathbb{Y}$ can be calculated as: $\alpha_k = e_k + a_k W$, where $e_k \geq 0, \forall k \in \mathbb{Y}$, and W refers to a non-informative weight representing the amount of uncertain evidence. Given the Dirichlet PDF as defined above, the expected probability distribution over $\mathbb{Y}$ can be calculated as:

$$\mathbb{E}[p_k] = \frac{\alpha_k}{\sum_{k=1}^{K} \alpha_k} = \frac{e_k + a_k W}{W + \sum_{k=1}^{K} e_k}. \tag{3}$$

The observed evidence in a Dirichlet PDF can be mapped to a multinomial opinion as follows:

$$b_k = \frac{e_k}{S},\ u = \frac{W}{S}, \tag{4}$$

where $S = \sum_{k=1}^{K} \alpha_k$ refers to the Dirichlet strength. Without loss of generality, we set $a_k = \frac{1}{K}$ and the non-informative prior weight (i.e., $W = K$), which indicates that $a_k \cdot W = 1$ for each $k \in \mathbb{Y}$.

3.2 Evidential Uncertainty

In [12], we discussed a number of multidimensional uncertainty dimensions of a subjective opinion based on the formalism of SL, such as singularity, vagueness, vacuity, dissonance, consonance, and monosonance. These uncertainty dimensions can be observed from binomial, multinomial, or hyper opinions depending on their characteristics (e.g., the vagueness uncertainty is only observed in hyper opinions to deal with composite beliefs). In this paper, we discuss two main uncertainty types that can be estimated in a multinomial opinion, which are *vacuity* and *dissonance*.

The main cause of vacuity is derived from a lack of evidence or knowledge, which corresponds to the uncertainty mass, u, of a multinomial opinion in SL as: $vac(\omega) \equiv u = K/S$, as estimated in Eq. (4). This uncertainty exists because the analyst may have insufficient information or knowledge to analyze the uncertainty. The *dissonance* of a multinomial opinion can be derived from the same amount of conflicting evidence and can be estimated based on the difference between singleton belief masses (e.g., class labels), which leads to 'inconclusiveness' in decision making applications. For example, a four-state multinomial opinion is given as $(b_1, b_2, b_3, b_4, u, a) = (0.25, 0.25, 0.25, 0.25, 0.0, a)$ based on Eq. (4), although the vacuity u is zero, a decision can not be made if there are the same amounts of beliefs supporting respective beliefs. Given a multinomial opinion with non-zero belief masses, the measure of dissonance can be calculated as:

$$diss(\omega) = \sum_{i=1}^{K} \Big(\frac{b_i \sum_{j \neq i} b_j \text{Bal}(b_j, b_i)}{\sum_{j \neq i} b_j} \Big), \tag{5}$$

where the relative mass balance between a pair of belief masses b_j and b_i is defined as $\text{Bal}(b_j, b_i) = 1 - |b_j - b_i|/(b_j + b_i)$. We note that the dissonance is measured only when the belief mass is non-zero. If all belief masses equal to zero with vacuity being 1 (i.e., $u = 1$), the dissonance will be set to zero.

3.3 Probabilistic Uncertainty

For classification, the estimation of the probabilistic uncertainty relies on the design of an appropriate Bayesian DL model with parameters $\boldsymbol{\theta}$. Given input x and dataset $\mathcal{G}$, we estimate a class probability by $P(y|x) = \int P(y|x;\boldsymbol{\theta}) P(\boldsymbol{\theta}|\mathcal{G}) d\boldsymbol{\theta}$, and obtain ***epistemic uncertainty*** estimated by mutual information [2, 20]:

$$\underbrace{I(y, \boldsymbol{\theta}|x, \mathcal{G})}_{\textit{Epistemic}} = \underbrace{\mathcal{H}\big[\mathbb{E}_{P(\boldsymbol{\theta}|\mathcal{G})}[P(y|x;\boldsymbol{\theta})]\big]}_{\textit{Entropy}} - \underbrace{\mathbb{E}_{P(\boldsymbol{\theta}|\mathcal{G})}\big[\mathcal{H}[P(y|x;\boldsymbol{\theta})]\big]}_{\textit{Aleatoric}}, \tag{6}$$

where $\mathcal{H}(\cdot)$ is Shannon's entropy of a probability distribution. The first term indicates ***entropy*** that represents the total uncertainty while the second term is ***aleatoric*** that indicates data uncertainty. By computing the difference between entropy and aleatoric uncertainties, we obtain epistemic uncertainty, which refers to uncertainty from model parameters.

4 Relationships Between Multiple Uncertainties

We use the shorthand notations u_v, u_{diss}, u_{alea}, u_{epis}, and u_{en} to represent vacuity, dissonance, aleatoric, epistemic, and entropy, respectively.

To interpret multiple types of uncertainty, we show three prediction scenarios of 3-class classification in Figure 1, in each of which the strength parameters $\alpha = [\alpha_1, \alpha_2, \alpha_3]$ are known. To make a prediction with high confidence, the subjective multinomial opinion, following a Dirichlet distribution, will yield a sharp distribution on one corner of the simplex (see Figure 1 (a)). For a prediction with conflicting evidence, called a conflicting prediction (CP), the multinomial opinion should yield a central distribution, representing confidence to predict a flat categorical distribution over class labels (see Figure 1 (b)). For an OOD scenario with $\alpha = [1, 1, 1]$, the multinomial opinion would yield a flat distribution over the simplex (Figure 1 (c)), indicating high uncertainty due to the lack of evidence. The first technical contribution of this work is as follows.

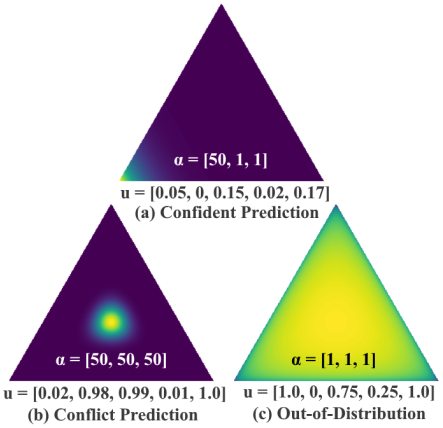


Figure 1: Multiple uncertainties of different prediction. Let $\mathbf{u} = [u_v, u_{diss}, u_{alea}, u_{epis}, u_{en}]$.

Theorem 1. *We consider a simplified scenario, where a multinomial random variable y follows a K-class categorical distribution: $y \sim Cal(\boldsymbol{p})$, the class probabilities $\boldsymbol{p}$ follow a Dirichlet distribution: $\boldsymbol{p} \sim Dir(\boldsymbol{\alpha})$, and $\boldsymbol{\alpha}$ refer to the Dirichlet parameters. Given a total Dirichlet strength $S = \sum_{i=1}^{K} \alpha_i$, for any opinion ω on a multinomial random variable y, we have*

1. *General relations on all prediction scenarios.*

   (a) $u_v + u_{diss} \leq 1$*; (b)* $u_v > u_{epis}$.

2. *Special relations on the OOD and the CP.*

   (a) For an OOD sample with a uniform prediction (i.e., $\alpha = [1, \ldots, 1]$), we have

   $$1 = u_v = u_{en} > u_{alea} > u_{epis} > u_{diss} = 0$$

   (b) For an in-distribution sample with a conflicting prediction (i.e., $\alpha = [\alpha_1, \ldots, \alpha_K]$ with $\alpha_1 = \alpha_2 = \cdots = \alpha_K$, if $S \to \infty$), we have

   $$u_{en} = 1, \lim_{S\to\infty} u_{diss} = \lim_{S\to\infty} u_{alea} = 1, \lim_{S\to\infty} u_v = \lim_{S\to\infty} u_{epis} = 0$$

   with $u_{en} > u_{alea} > u_{diss} > u_v > u_{epis}$.

The proof of Theorem 1 can be found in Appendix A.1. As demonstrated in Theorem 1 and Figure 1, entropy cannot distinguish OOD (see Figure 1 (c)) and conflicting predictions (see Figure 1 (b)) because entropy is high for both cases. Similarly, neither aleatoric uncertainty nor epistemic uncertainty can distinguish OOD from conflicting predictions. In both cases, aleatoric uncertainty is high while epistemic uncertainty is low. On the other hand, vacuity and dissonance can clearly distinguish OOD from a conflicting prediction. For example, OOD objects typically show high vacuity with low dissonance while conflicting predictions exhibit low vacuity with high dissonance. This observation is confirmed through the empirical validation via our extensive experiments in terms of misclassification and OOD detection tasks.

5 Uncertainty-Aware Semi-Supervised Learning

In this section, we describe our proposed uncertainty framework based on semi-supervised node classification problem. It is designed to predict the subjective opinions about the classification

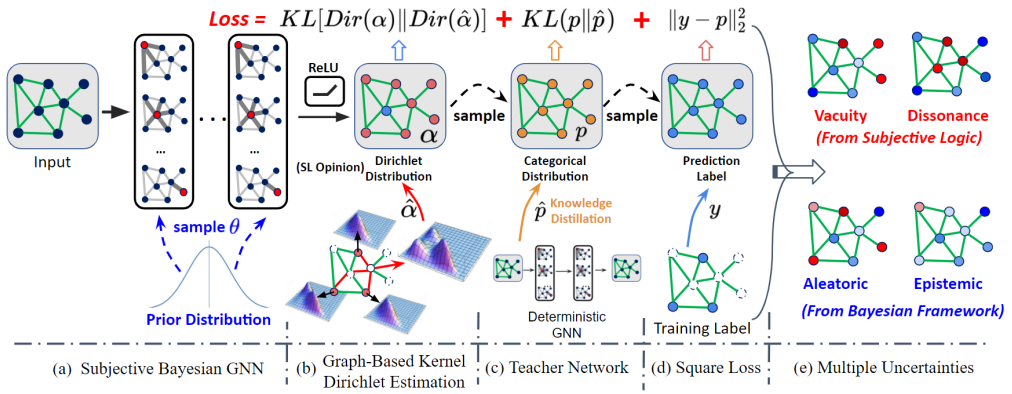


Figure 2: **Uncertainty Framework Overview.** Subjective Bayesian GNN (a) designed for estimating the different types of uncertainties. The loss function includes square error (d) to reduce bias, GKDE (b) to reduce errors in uncertainty estimation and teacher network (c) to refine class probability.

of testing nodes, such that a variety of uncertainty types, such as vacuity, dissonance, aleatoric uncertainty, and epistemic uncertainty, can be quantified based on the estimated subjective opinions and posterior of model parameters. As a subjective opinion can be equivalently represented by a Dirichlet distribution about the class probabilities, we proposed a way to predict the node-level subjective opinions in the form of node-level Dirichlet distributions. The overall description of the framework is shown in Figure 2.

5.1 Problem Definition

Given an input graph $\mathcal{G} = (\mathbb{V}, \mathbb{E}, \mathbf{r}, \mathbf{y}_{\mathbb{L}})$, where $\mathbb{V} = \{1, \ldots, N\}$ is a ground set of nodes, $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$ is a ground set of edges, $\mathbf{r} = [\mathbf{r}_1, \cdots, \mathbf{r}_N]^T \in \mathbb{R}^{N \times d}$ is a node-level feature matrix, $\mathbf{r}_i \in \mathbb{R}^d$ is the feature vector of node i, $\mathbf{y}_{\mathbb{L}} = \{y_i \mid i \in \mathbb{L}\}$ are the labels of the training nodes $\mathbb{L} \subset \mathbb{V}$, and $y_i \in \{1, \ldots, K\}$ is the class label of node i. **We aim to predict**: (1) the **class probabilities** of the testing nodes: $\mathbf{p}_{\mathbb{V}\setminus\mathbb{L}} = \{\mathbf{p}_i \in [0,1]^K \mid i \in \mathbb{V} \setminus \mathbb{L}\}$; and (2) the **associated multidimensional uncertainty estimates** introduced by different root causes: $\mathbf{u}_{\mathbb{V}\setminus\mathbb{L}} = \{\mathbf{u}_i \in [0,1]^m \mid i \in \mathbb{V} \setminus \mathbb{L}\}$, where $p_{i,k}$ is the probability that the class label $y_i = k$ and m is the total number of uncertainty types.

5.2 Proposed Uncertainty Framework

Learning evidential uncertainty. As discussed in Section 3.1, evidential uncertainty can be derived from multinomial opinions or equivalently Dirichlet distributions to model a probability distribution for the class probabilities. Therefore, we design a Subjective GNN (S-GNN) f to form their multinomial opinions for the node-level Dirichlet distribution $\text{Dir}(\mathbf{p}_i|\boldsymbol{\alpha}_i)$ of a given node i. Then, the conditional probability $P(\mathbf{p}|A, \mathbf{r}; \boldsymbol{\theta})$ can be obtained by:

$$P(\mathbf{p}|A, \mathbf{r}; \boldsymbol{\theta}) = \prod_{i=1}^{N} \text{Dir}(\mathbf{p}_i|\boldsymbol{\alpha}_i), \ \boldsymbol{\alpha}_i = f_i(A, \mathbf{r}; \boldsymbol{\theta}), \tag{7}$$

where f_i is the output of S-GNN for node i, $\boldsymbol{\theta}$ is the model parameters, and A is an adjacency matrix. The Dirichlet probability function $\text{Dir}(\mathbf{p}_i|\boldsymbol{\alpha}_i)$ is defined by Eq. (2).

Note that S-GNN is similar to classical GNN, except that we use an activation layer (e.g., *ReLU*) instead of the *softmax* layer (only outputs class probabilities). This ensures that S-GNN would output non-negative values, which are taken as the parameters for the predicted Dirichlet distribution.

Learning probabilistic uncertainty. Since probabilistic uncertainty relies on a Bayesian framework, we proposed a Subjective Bayesian GNN (S-BGNN) that adapts S-GNN to a Bayesian framework, with the model parameters $\boldsymbol{\theta}$ following a prior distribution. The joint class probability of $\mathbf{y}$ can be estimated by:

$$\begin{aligned} P(\mathbf{y}|A, \mathbf{r}; \mathcal{G}) &= \int\int P(\mathbf{y}|\mathbf{p}) P(\mathbf{p}|A, \mathbf{r}; \boldsymbol{\theta}) P(\boldsymbol{\theta}|\mathcal{G}) d\mathbf{p} d\boldsymbol{\theta} \\ &\approx \frac{1}{M} \sum_{m=1}^{M} \sum_{i=1}^{N} \int P(\mathbf{y}_i|\mathbf{p}_i) P(\mathbf{p}_i|A, \mathbf{r}; \boldsymbol{\theta}^{(m)}) d\mathbf{p}_i, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \end{aligned} \tag{8}$$

where $P(\boldsymbol{\theta}|\mathcal{G})$ is the posterior, estimated via dropout inference, that provides an approximate solution of posterior $q(\boldsymbol{\theta})$ and taking samples from the posterior distribution of models [7]. Thanks to the

benefit of dropout inference, training a DL model directly by minimizing the cross entropy (or square error) loss function can effectively minimize the KL-divergence between the approximated distribution and the full posterior (i.e., KL$[q(\boldsymbol{\theta})\|P(\theta|\mathcal{G})]$) in variational inference [7, 13]. For interested readers, please refer to more detail in Appendix B.8.

Therefore, training S-GNN with stochastic gradient descent enables learning of an approximated distribution of weights, which can provide good explainability of data and prevent overfitting. We use a *loss function* to compute its Bayes risk with respect to the sum of squares loss $\|\mathbf{y} - \mathbf{p}\|_2^2$ by:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum\nolimits_{i\in\mathbb{L}} \int \|\mathbf{y}_i - \mathbf{p}_i\|_2^2 \cdot P(\mathbf{p}_i|A, \mathbf{r}; \boldsymbol{\theta}) d\mathbf{p}_i = \sum\nolimits_{i\in\mathbb{L}} \sum\nolimits_{k=1}^{K} \left(y_{ik} - \mathbb{E}[p_{ik}]\right)^2 + \text{Var}(p_{ik}), \quad (9)$$

where $\mathbf{y}_i$ is an one-hot vector encoding the ground-truth class with $y_{ij} = 1$ and $y_{ik} \neq$ for all $k \neq j$ and j is a class label. Eq. (9) aims to minimize the prediction error and variance, leading to maximizing the classification accuracy of each training node by removing excessive misleading evidence.

5.3 Graph-based Kernel Dirichlet distribution Estimation (GKDE)

The loss function in Eq. (9) is designed to measure the sum of squared loss based on class labels of training nodes. However, it does not directly measure the quality of the predicted node-level Dirichlet distributions. To address this limitation, we proposed *Graph-based Kernel Dirichlet distribution Estimation* (GKDE) to better estimate node-level Dirichlet distributions by using graph structure information. The key idea of the GKDE is to estimate prior Dirichlet distribution parameters for each node based on the class labels of training nodes (see Figure 3). Then, we use the estimated prior Dirichlet distribution in the training process to learn the following patterns: (i) nodes with a high vacuity will be shown far from training nodes; and (ii) nodes with a high dissonance will be shown near the boundaries of classes.

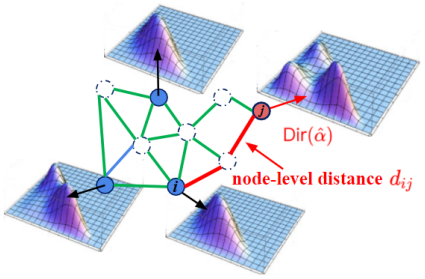


Figure 3: Illustration of GKDE. Estimate prior Dirichlet distribution Dir($\hat{\alpha}$) for node j (red) based on training nodes (blue) and graph structure information.

Based on SL, let each training node represent one evidence for its class label. Denote the contribution of evidence estimation for node j from training node i by $\mathbf{h}(y_i, d_{ij}) = [h_1, \ldots, h_k, \ldots, h_K] \in [0, 1]^K$, where $h_k(y_i, d_{ij})$ is obtained by:

$$h_k(y_i, d_{ij}) = \begin{cases} 0 & y_i \neq k \\ g(d_{ij}) & y_i = k, \end{cases} \quad (10)$$

$g(d_{ij}) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{d_{ij}^2}{2\sigma^2})$ is the Gaussian kernel function used to estimate the distribution effect between nodes i and j, and d_{ij} means the **node-level distance** (**a shortest path between nodes** i **and** j), and σ is the bandwidth parameter. The prior evidence is estimated based GKDE: $\hat{\boldsymbol{e}}_j = \sum_{i\in\mathbb{L}} \mathbf{h}(y_i, d_{ij})$, where $\mathbb{L}$ is a set of training nodes and the prior Dirichlet distribution $\hat{\boldsymbol{\alpha}}_j = \hat{\boldsymbol{e}}_j + \mathbf{1}$. During the training process, we minimize the KL-divergence between model predictions of Dirichlet distribution and prior distribution: $\min$ KL$[\text{Dir}(\boldsymbol{\alpha})\|\text{Dir}(\hat{\boldsymbol{\alpha}})]$. This process can prioritize the extent of data relevance based on the estimated evidential uncertainty, which is proven effective based on the proposition below.

Proposition 1. *Given L training nodes, for any testing nodes i and j, let $\boldsymbol{d}_i = [d_{i1}, \ldots, d_{iL}]$ be the vector of graph distances from nodes i to training nodes and $\boldsymbol{d}_j = [d_{j1}, \ldots, d_{jL}]$ be the graph distances from nodes j to training nodes, where d_{il} is the node-level distance between nodes i and l. If for all $l \in \{1, \ldots, L\}$, $d_{il} \geq d_{jl}$, then we have*

$$\hat{u}_{v_i} \geq \hat{u}_{v_j},$$

where $\hat{u}_{v_i}$ and $\hat{u}_{v_j}$ refer to vacuity uncertainties of nodes i and j estimated based on GKDE.

The proof for this proposition can be found in Appendix A.2. The above proposition shows that if a testing node is too far from training nodes, the vacuity will increase, implying that an OOD node is expected to have a high vacuity.

In addition, we designed a simple iterative knowledge distillation method [10] (i.e., Teacher Network) to refine the node-level classification probabilities. The key idea is to train our proposed model

(Student) to imitate the outputs of a pre-train a vanilla GNN (Teacher) by adding a regularization term of KL-divergence. This leads to solving the following optimization problem:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) + \lambda_1 \text{KL}[\text{Dir}(\boldsymbol{\alpha}) \| \text{Dir}(\hat{\boldsymbol{\alpha}})] + \lambda_2 \text{KL}[P(\mathbf{y} \mid A, \mathbf{r}; \mathcal{G}) \parallel P(\mathbf{y}|\hat{\mathbf{p}})], \tag{11}$$

where $\hat{\mathbf{p}}$ is the vanilla GNN's (Teacher) output and λ_1 and λ_2 are trade-off parameters.

6 Experiments

In this section, we conduct experiments on the tasks of misclassification and OOD detections to answer the following questions for semi-supervised node classification:

Q1. Misclassification Detection: What type of uncertainty is the most promising indicator of high confidence in node classification predictions?

Q2. OOD Detection: What type of uncertainty is a key indicator of accurate detection of OOD nodes?

Q3. GKDE with Uncertainty Estimates: How can GKDE help enhance prediction tasks with what types of uncertainty estimates?

Through extensive experiments, we found the following answers for the above questions:

A1. Dissonance (i.e., uncertainty due to conflicting evidence) is more effective than other uncertainty estimates in misclassification detection.

A2. Vacuity (i.e., uncertainty due to lack of confidence) is more effective than other uncertainty estimates in OOD detection.

A3. GKDE can indeed help improve the estimation quality of node-level Dirichlet distributions, resulting in a higher OOD detection.

6.1 Experiment Setup

Datasets: We used six datasets, including three citation network datasets [25] (i.e., Cora, Citeseer, Pubmed) and three new datasets [28] (i.e., Coauthor Physics, Amazon Computer, and Amazon Photo). We summarized the description and experimental setup of the used datasets in Appendix B.2[1].

Comparing Schemes: We conducted the extensive comparative performance analysis based on our proposed models and several state-of-the-art competitive counterparts. We implemented all models based on the most popular GNN model, GCN [16]. We compared our model (S-BGCN-T-K) against: (1) Softmax-based GCN [16] with uncertainty measured based on entropy; and (2) Drop-GCN that adapts the Monte-Carlo Dropout [7, 24] into the GCN model to learn probabilistic uncertainty; (3) EDL-GCN that adapts the EDL model [26] with GCN to estimate evidential uncertainty; (4) DPN-GCN that adapts the DPN [20] method with GCN to estimate probabilistic uncertainty. We evaluated the performance of all models considered using the area under the ROC (AUROC) curve and area under the Precision-Recall (AUPR) curve in both experiments [9].

6.2 Results

Misclassification Detection. The misclassification detection experiment involves detecting whether a given prediction is incorrect using an uncertainty estimate. Table 1 shows that S-BGCN-T-K outperforms all baseline models under the AUROC and AUPR for misclassification detection. The outperformance of dissonance-based detection is fairly impressive. This confirms that low dissonance (a small amount of conflicting evidence) is the key to maximize the accuracy of node classification prediction. We observe the following performance order: `Dissonance` $>$ `Entropy` $\approx$ `Aleatoric` $>$ `Vacuity` $\approx$ `Epistemic`, which is aligned with our conjecture: higher dissonance with conflicting prediction leads to higher misclassification detection. We also conducted experiments on additional three datasets and observed similar trends of the results, as demonstrated in Appendix C.

OOD Detection. This experiment involves detecting whether an input example is out-of-distribution (OOD) given an estimate of uncertainty. For semi-supervised node classification, we randomly selected one to four categories as OOD categories and trained the models based on training nodes of the other categories. Due to the space constraint, the experimental setup for the OOD detection is detailed in Appendix B.3.

In Table 2, across six network datasets, our vacuity-based detection significantly outperformed the other competitive methods, exceeding the performance of the epistemic uncertainty and other type of

[1]The source code and datasets are accessible at https://github.com/zxj32/uncertainty-GNN

Table 1: AUROC and AUPR for the Misclassification Detection.

Data	Model	AUROC					AUPR					Acc
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.	
Cora	S-BGCN-T-K	70.6	**82.4**	75.3	68.8	77.7	90.3	**95.4**	92.4	87.8	93.4	**82.0**
	EDL-GCN	70.2	81.5	-	-	76.9	90.0	94.6	-	-	93.6	81.5
	DPN-GCN	-	-	78.3	75.5	77.3	-	-	92.4	92.0	92.4	80.8
	Drop-GCN	-	-	73.9	66.7	76.9	-	-	92.7	90.0	93.6	81.3
	GCN	-	-	-	-	79.6	-	-	-	-	94.1	81.5
Citeseer	S-BGCN-T-K	65.4	**74.0**	67.2	60.7	70.0	79.8	**85.6**	82.2	75.2	83.5	**71.0**
	EDL-GCN	64.9	73.6	-	-	69.6	79.2	84.6	-	-	82.9	70.2
	DPN-GCN	-	-	66.0	64.9	65.5	-	-	78.7	77.6	78.1	68.1
	Drop-GCN	-	-	66.4	60.8	69.8	-	-	82.3	77.8	83.7	70.9
	GCN	-	-	-	-	71.4	-	-	-	-	83.2	70.3
Pubmed	S-BGCN-T-K	64.1	**73.3**	69.3	64.2	70.7	85.6	**90.8**	88.8	86.1	89.2	**79.3**
	EDL-GCN	62.6	69.0	-	-	67.2	84.6	88.9	-	-	81.7	79.0
	DPN-GCN	-	-	72.7	69.2	72.5	-	-	87.8	86.8	87.7	77.1
	Drop-GCN	-	-	67.3	66.1	67.2	-	-	88.6	85.6	89.0	79.0
	GCN	-	-	-	-	68.5	-	-	-	-	89.2	79.0

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, En.: Entropy

Table 2: AUROC and AUPR for the OOD Detection.

Data	Model	AUROC					AUPR				
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.
Cora	S-BGCN-T-K	**87.6**	75.5	85.5	70.8	84.8	**78.4**	49.0	75.3	44.5	73.1
	EDL-GCN	84.5	81.0		-	83.3	74.2	53.2	-	-	71.4
	DPN-GCN	-	-	77.3	78.9	78.3	-	-	58.5	62.8	63.0
	Drop-GCN	-	-	81.9	70.5	80.9	-	-	69.7	44.2	67.2
	GCN	-	-	-	-	80.7	-	-	-	-	66.9
Citeseer	S-BGCN-T-K	**84.8**	55.2	78.4	55.1	74.0	**86.8**	54.1	80.8	55.8	74.0
	EDL-GCN	78.4	59.4	-	-	69.1	79.8	57.3	-	-	69.0
	DPN-GCN	-	-	68.3	72.2	69.5	-	-	68.5	72.1	70.3
	Drop-GCN	-	-	72.3	61.4	70.6	-	-	73.5	60.8	70.0
	GCN	-	-	-	-	70.8	-	-	-	-	70.2
Pubmed	S-BGCN-T-K	**74.6**	67.9	71.8	59.2	72.2	**69.6**	52.9	63.6	44.0	56.5
	EDL-GCN	71.5	68.2	-	-	70.5	65.3	53.1	-	-	55.0
	DPN-GCN	-	-	63.5	63.7	63.5	-	-	50.7	53.9	51.1
	Drop-GCN	-	-	68.7	60.8	66.7	-	-	59.7	46.7	54.8
	GCN	-	-	-	-	68.3	-	-	-	-	55.3
Amazon Photo	S-BGCN-T-K	**93.4**	76.4	91.4	32.2	91.4	**94.8**	68.0	92.3	42.3	92.5
	EDL-GCN	63.4	78.1	-	-	79.2	66.2	74.8	-	-	81.2
	DPN-GCN	-	-	83.6	83.6	83.6	-	-	82.6	82.4	82.5
	Drop-GCN	-	-	84.5	58.7	84.3	-	-	87.0	57.7	86.9
	GCN	-	-	-	-	84.4	-	-	-	-	87.0
Amazon Computer	S-BGCN-T-K	**82.3**	76.6	80.9	55.4	80.9	**70.5**	52.8	60.9	35.9	60.6
	EDL-GCN	53.2	70.1	-	-	70.0	33.2	43.9	-	-	45.7
	DPN-GCN	-	-	77.6	77.7	77.7	-	-	50.8	51.2	51.0
	Drop-GCN	-	-	74.4	70.5	74.3	-	-	50.0	46.7	49.8
	GCN	-	-	-	-	74.0	-	-	-	-	48.7
Coauthor Physics	S-BGCN-T-K	**91.3**	87.6	89.7	61.8	89.8	**72.2**	56.6	68.1	25.9	67.9
	EDL-GCN	88.2	85.8	-	-	87.6	67.1	51.2	-	-	62.1
	DPN-GCN	-	-	85.5	85.6	85.5	-	-	59.8	60.2	59.8
	Drop-GCN	-	-	89.2	78.4	89.3	-	-	66.6	37.1	66.5
	GCN	-	-	-	-	89.1	-	-	-	-	64.0

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, En.: Entropy

uncertainties. This demonstrates that vacuity-based model is more effective than other uncertainty estimates-based counterparts in increasing OOD detection. We observed the following performance order: $\texttt{Vacuity} > \texttt{Entropy} \approx \texttt{Aleatoric} > \texttt{Epistemic} \approx \texttt{Dissonance}$, which is consistent with the theoretical results as shown in Theorem 1.

Ablation Study. We conducted additional experiments (see Table 3) in order to demonstrate the contributions of the key technical components, including GKDE, Teacher Network, and subjective Bayesian framework. The key findings obtained from this experiment are: (1) GKDE can enhance the OOD detection (i.e., 30% increase with vacuity), which is consistent with our theoretical proof about the outperformance of GKDE in uncertainty estimation, i.e., OOD nodes have a higher vacuity than other nodes; and (2) the Teacher Network can further improve the node classification accuracy.

6.3 Why is Epistemic Uncertainty Less Effective than Vacuity?

Although epistemic uncertainty is known to be effective to improve OOD detection [7, 14] in computer vision applications, our results demonstrate it is less effective than our vacuity-based approach. The first potential reason is that epistemic uncertainty is always smaller than vacuity (From Theorem 1), which potentially indicates that epistemic may capture less information related to OOD. Another potential reason is that the previous success of epistemic uncertainty for OOD detection is limited to supervised learning in computer vision applications, but its effectiveness for OOD detection was not

sufficiently validated in semi-supervised learning tasks. Recall that epistemic uncertainty (i.e., model uncertainty) is calculated based on mutual information (see Eq. (6)). In a semi-supervised setting, the features of unlabeled nodes are also fed to a model for training process to provide the model with a high confidence on its output. For example, the model output $P(\mathbf{y}|A, \mathbf{r}; \theta)$ would not change too much even with differently sampled parameters $\boldsymbol{\theta}$, i.e., $P(\mathbf{y}|A, \mathbf{r}; \theta^{(i)}) \approx P(\mathbf{y}|A, \mathbf{r}; \theta^{(j)})$, which result in a low epistemic uncertainty. We also designed a semi-supervised learning experiment for image classification and observed a consistent pattern with the results demonstrated in Appendix C.6.

Table 3: Ablation study of our proposed models: (1) `S-GCN`: Subjective GCN with vacuity and dissonance estimation; (2) `S-BGCN`: S-GCN with Bayesian framework; (3) `S-BGCN-T`: S-BGCN with a Teacher Network; (4) `S-BGCN-T-K`: S-BGCN-T with GKDE to improve uncertainty estimation.

Data	Model	AUROC (Misclassification Detection)					AUPR (Misclassification Detection)					Acc
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.	
Cora	S-BGCN-T-K	70.6	82.4	75.3	68.8	77.7	90.3	**95.4**	92.4	87.8	93.4	82.0
	S-BGCN-T	70.8	**82.5**	75.3	68.9	77.8	90.4	**95.4**	92.6	88.0	93.4	**82.2**
	S-BGCN	69.8	81.4	73.9	66.7	76.9	89.4	94.3	92.3	88.0	93.1	81.2
	S-GCN	70.2	81.5	-	-	76.9	90.0	94.6	-	-	93.6	81.5
		AUROC (OOD Detection)					AUPR (OOD Detection)					
Amazon Photo	S-BGCN-T-K	**93.4**	76.4	91.4	32.2	91.4	**94.8**	68.0	92.3	42.3	92.5	-
	S-BGCN-T	64.0	77.5	79.9	52.6	79.8	67.0	75.3	82.0	53.7	81.9	-
	S-BGCN	63.0	76.6	79.8	52.7	79.7	66.5	75.1	82.1	53.9	81.7	-
	S-GCN	64.0	77.1	-	-	79.6	67.0	74.9	-	-	81.6	-

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, En.: Entropy

7 Conclusion

In this work, we proposed a multi-source uncertainty framework of GNNs for semi-supervised node classification. Our proposed framework provides an effective way of predicting node classification and out-of-distribution detection considering multiple types of uncertainty. We leveraged various types of uncertainty estimates from both DL and evidence/belief theory domains. Through our extensive experiments, we found that dissonance-based detection yielded the best performance on misclassification detection while vacuity-based detection performed the best for OOD detection, compared to other competitive counterparts. In particular, it was noticeable that applying GKDE and the Teacher network further enhanced the accuracy in node classification and uncertainty estimates.

Acknowledgments

We would like to thank Yuzhe Ou for providing proof suggestions. This work is supported by the National Science Foundation (NSF) under Grant No #1815696 and #1750911.

Broader Impact

In this paper, we propose a uncertainty-aware semi-supervised learning framework of GNN for predicting multi-dimensional uncertainties for the task of semi-supervised node classification. Our proposed framework can be applied to a wide range of applications, including computer vision, natural language processing, recommendation systems, traffic prediction, generative models and many more [33]. Our proposed framework can be applied to predict multiple uncertainties of different roots for GNNs in these applications, improving the understanding of individual decisions, as well as the underlying models. While there will be important impacts resulting from the use of GNNs in general, our focus in this work is on investigating the impact of using our method to predict multi-source uncertainties for such systems. The additional benefits of this method include improvement of safety and transparency in decision-critical applications to avoid overconfident prediction, which can easily lead to misclassification.

We see promising research opportunities that can adopt our uncertainty framework, such as investigating whether this uncertainty framework can further enhance misclassification detection or OOD detection. To mitigate the risk from different types of uncertainties, we encourage future research to understand the impacts of this proposed uncertainty framework to solve other real world problems.

References

[1] C. W. De Silva. *Intelligent control: fuzzy logic applications*. CRC press, 1995.

[2] S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, pages 1184–1193. PMLR, 2018.

[3] D. Eswaran, S. Günnemann, and C. Faloutsos. The power of certainty: A dirichlet-multinomial model for belief propagation. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 144–152. SIAM, 2017.

[4] T. Fawcett. An introduction to roc analysis. *Pattern recognition letters*, pages 861–874, 2006.

[5] Y. Gal. Uncertainty in deep learning. *University of Cambridge*, 2016.

[6] Y. Gal and Z. Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. *arXiv preprint arXiv:1506.02158*, 2015.

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

[8] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[9] D. Hendrycks and K. Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[10] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[11] A. Josang. *Subjective logic*. Springer, 2016.

[12] A. Josang, J.-H. Cho, and F. Chen. Uncertainty characteristics of subjective opinions. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 1998–2005. IEEE, 2018.

[13] A. Kendall, V. Badrinarayanan, and R. Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680*, 2015.

[14] A. Kendall and Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pages 5574–5584, 2017.

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[17] D.-H. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, 2013.

[18] Z.-Y. Liu, S.-Y. Li, S. Chen, Y. Hu, and S.-J. Huang. Uncertainty aware graph gaussian process for semi-supervised learning. In *AAAI*, pages 4957–4964, 2020.

[19] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, pages 2579–2605, 2008.

[20] A. Malinin and M. Gales. Predictive uncertainty estimation via prior networks. In *Advances in Neural Information Processing Systems*, pages 7047–7058, 2018.

[21] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52, 2015.

[22] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, pages 1979–1993, 2018.

[23] Y. Rong, W. Huang, T. Xu, and J. Huang. Dropedge: Towards deep graph convolutional networks on node classification. In *International Conference on Learning Representations*, 2019.

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

[25] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad. Collective classification in network data. *AI magazine*, pages 93–93, 2008.

[26] M. Sensoy, L. Kaplan, and M. Kandemir. Evidential deep learning to quantify classification uncertainty. In *Advances in Neural Information Processing Systems*, pages 3179–3189, 2018.

[27] K. Sentz, S. Ferson, et al. *Combination of evidence in Dempster-Shafer theory*, volume 4015. Sandia National Laboratories Albuquerque, 2002.

[28] O. Shchur, M. Mumme, A. Bojchevski, and S. Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

[29] A. Tarvainen and H. Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017.

[30] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018.

[31] Z. Yang, W. Cohen, and R. Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*, pages 40–48. PMLR, 2016.

[32] Y. Zhang, S. Pal, M. Coates, and D. Ustebay. Bayesian graph convolutional neural networks for semi-supervised classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5829–5836, 2019.

[33] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.

Appendix

A Proofs

A.1 Theorem 1's Proof

Theorem 1. *We consider a simplified scenario, where a multinomial random variable y follows a K-class categorical distribution: $y \sim Cal(\boldsymbol{p})$, the class probabilities $\boldsymbol{p}$ follow a Dirichlet distribution: $\boldsymbol{p} \sim Dir(\boldsymbol{\alpha})$, and $\boldsymbol{\alpha}$ refer to the Dirichlet parameters. Given a total Dirichlet strength $S = \sum_{i=1}^{K} \alpha_i$, for any opinion ω on a multinomial random variable y, we have*

1. *General relations on all prediction scenarios.*

   (a) $u_v + u_{diss} \leq 1$*; (b)* $u_v > u_{epis}$.

2. *Special relations on the OOD and the CP.*

   (a) *For an OOD sample with a uniform prediction (i.e.,* $\alpha = [1, \ldots, 1]$*), we have*

   $$1 = u_v = u_{en} > u_{alea} > u_{epis} > u_{diss} = 0$$

   (b) *For an in-distribution sample with a conflicting prediction (i.e.,* $\alpha = [\alpha_1, \ldots, \alpha_K]$ *with* $\alpha_1 = \alpha_2 = \cdots = \alpha_K$*, if* $S \to \infty$*), we have*

   $$u_{en} = 1, \lim_{S\to\infty} u_{diss} = \lim_{S\to\infty} u_{alea} = 1, \lim_{S\to\infty} u_v = \lim_{S\to\infty} u_{epis} = 0$$

   with $u_{en} > u_{alea} > u_{diss} > u_v > u_{epis}$.

Interpretation. **Theorem 1.1 (a)** implies that increases in both uncertainty types may not happen at the same time. A higher vacuity leads to a lower dissonance, and vice versa (a higher dissonance leads to a lower vacuity). This indicates that a high dissonance only occurs only when a large amount of evidence is available and the vacuity is low. **Theorem 1.1 (b)** shows relationships between vacuity and epistemic uncertainty in which vacuity is an upper bound of epistemic uncertainty. Although some existing approaches [11, 26] treat epistemic uncertainty the same as vacuity, it is not necessarily true except for an extreme case where a sufficiently large amount of evidence available, making vacuity close to zero. **Theorem 1.2 (a) and (b)** explain how entropy differs from vacuity and/or dissonance. We observe that entropy is 1 when either vacuity or dissonance is 0. This implies that entropy cannot distinguish different types of uncertainty due to different root causes. For example, a high entropy is observed when an example is an either OOD or misclassified example. Similarly, a high aleatoric uncertainty value and a low epistemic uncertainty value are observed under both cases. However, vacuity and dissonance can capture different causes of uncertainty due to lack of information and knowledge and to conflicting evidence, respectively. For example, an OOD objects typically show a high vacuity value and a low dissonance value while a conflicting prediction exhibits a low vacuity and a high dissonance.

Proof. 1. (a) Let the opinion $\omega = [b_1, \ldots, b_K, u_v]$, where K is the number of classes, b_i is the belief for class i, u_v is the uncertainty mass (vacuity), and $\sum_{i=1}^{K} b_i + u_v = 1$. Dissonance has a upper bound with

$$\begin{aligned} u_{diss} &= \sum_{i=1}^{K} \Big(\frac{b_i \sum_{j=1, j\neq i}^{K} b_j \text{Bal}(b_i, b_j)}{\sum_{j=1, j\neq i}^{K} b_j} \Big) \\ &\leq \sum_{i=1}^{K} \Big(\frac{b_i \sum_{j=1, j\neq i}^{K} b_j}{\sum_{j=1, j\neq i}^{K} b_j} \Big), \quad (\text{since } 0 \leq \text{Bal}(b_i, b_j) \leq 1) \\ &= \sum_{i=1}^{K} b_i, \end{aligned} \tag{12}$$

where $\text{Bal}(b_i, b_j)$ is the relative mass balance, then we have

$$u_v + u_{diss} \leq \sum_{i=1}^{K} b_i + u_v = 1. \tag{13}$$

1. (b) For the multinomial random variable y, we have

$$y \sim \text{Cal}(\mathbf{p}), \quad \mathbf{p} \sim \text{Dir}(\boldsymbol{\alpha}), \tag{14}$$

where $\text{Cal}(\mathbf{p})$ is the categorical distribution and $\text{Dir}(\boldsymbol{\alpha})$ is Dirichlet distribution. Then we have

$$\text{Prob}(y|\boldsymbol{\alpha}) = \int \text{Prob}(y|\mathbf{p}) \text{Prob}(\mathbf{p}|\boldsymbol{\alpha}) d\mathbf{p}, \tag{15}$$

and the epistemic uncertainty is estimated by mutual information,

$$\mathcal{I}[y, \mathbf{p}|\boldsymbol{\alpha}] = \mathcal{H}\Big[\mathbb{E}_{\text{Prob}(\mathbf{p}|\boldsymbol{\alpha})}[P(y|\mathbf{p})]\Big] - \mathbb{E}_{\text{Prob}(\mathbf{p}|\boldsymbol{\alpha})}\Big[\mathcal{H}[P(y|\mathbf{p})]\Big]. \tag{16}$$

Now we consider another measure of ensemble diversity: *Expected Pairwise KL-Divergence* between each model in the ensemble. Here the expected pairwise KL-Divergence between two independent distributions, including $P(y|\mathbf{p}_1)$ and $P(y|\mathbf{p}_2)$, where $\mathbf{p}_1$ and $\mathbf{p}_2$ are two independent samples from $\text{Prob}(\mathbf{p}|\boldsymbol{\alpha})$, can be computed,

$$\begin{aligned}
\mathcal{K}[y, \mathbf{p}|\boldsymbol{\alpha}] &= \mathbb{E}_{\text{Prob}(\mathbf{p}_1|\boldsymbol{\alpha}\text{Prob}(\mathbf{p}_2|\boldsymbol{\alpha})}\Big[KL[P(y|\mathbf{p}_1)\|P(y|\mathbf{p}_2)]\Big] \\
&= -\sum_{i=1}^{K} \mathbb{E}_{\text{Prob}(\mathbf{p}_1|\boldsymbol{\alpha})}[P(y|\mathbf{p}_1)]\mathbb{E}_{\text{Prob}(\mathbf{p}_2|\boldsymbol{\alpha})}[\ln P(y|\mathbf{p}_2)] - \mathbb{E}_{\text{Prob}(\mathbf{p}|\boldsymbol{\alpha})}\Big[\mathcal{H}[P(y|\mathbf{p})]\Big] \\
&\geq \mathcal{I}[y, \mathbf{p}|\boldsymbol{\alpha}],
\end{aligned} \tag{17}$$

where $\mathcal{I}[y, \mathbf{p}_1|\boldsymbol{\alpha}] = \mathcal{I}[y, \mathbf{p}_2|\boldsymbol{\alpha}]$. We consider Dirichlet ensemble, the *Expected Pairwise KL Divergence*,

$$\begin{aligned}
\mathcal{K}[y, \mathbf{p}|\boldsymbol{\alpha}] &= -\sum_{i=1}^{K} \frac{\alpha_i}{S}\Big(\psi(\alpha_i) - \psi(S)\Big) - \sum_{i=1}^{K} -\frac{\alpha_i}{S}\Big(\psi(\alpha_i + 1) - \psi(S + 1)\Big) \\
&= \frac{K-1}{S},
\end{aligned} \tag{18}$$

where $S = \sum_{i=1}^{K} \alpha_i$ and $\psi(\cdot)$ is the *digamma Function*, which is the derivative of the natural logarithm of the gamma function. Now we obtain the relations between vacuity and epistemic,

$$\underbrace{\frac{K}{S}}_{\text{Vacuity}} > \mathcal{K}[y, \mathbf{p}|\boldsymbol{\alpha}] = \frac{K-1}{S} \geq \underbrace{\mathcal{I}[y, \mathbf{p}|\boldsymbol{\alpha}]}_{\text{Epistemic}}. \tag{19}$$

2. (a) For an out-of-distribution sample, $\alpha = [1, \ldots, 1]$, the vacuity can be calculated as

$$u_v = \frac{K}{\sum_{i=1}^{K} \alpha_i} = \frac{K}{K} = 1, \tag{20}$$

and the belief mass $b_i = (\alpha_i - 1)/\sum_{i=1}^{K} \alpha_i = 0$, we estimate dissonance,

$$u_{diss} = \sum_{i=1}^{K} \Big(\frac{b_i \sum_{j=1, j\neq i}^{K} b_j \text{Bal}(b_i, b_j)}{\sum_{j=1, j\neq i}^{K} b_j}\Big) = 0. \tag{21}$$

Given the expected probability $\hat{p} = [1/K, \ldots, 1/K]^\top$, the entropy is calculated based on $\log_K$,

$$u_{en} = \mathcal{H}[\hat{p}] = -\sum_{i=1}^{K} \hat{p}_i \log_K \hat{p}_i = -\sum_{i=1}^{K} \frac{1}{K} \log_K \frac{1}{K} = \log_K \frac{1}{K}^{-1} = \log_K K = 1, \tag{22}$$

where $\mathcal{H}(\cdot)$ is the entropy. Based on Dirichlet distribution, the aleatoric uncertainty refers to the expected entropy,

$$
\begin{aligned}
u_{alea} &= \mathbb{E}_{p\sim \text{Dir}(\alpha)}[\mathcal{H}[p]] \\
&= -\sum_{i=1}^{K} \frac{\Gamma(S)}{\prod_{i=1}^{K}\Gamma(\alpha_i)} \int_{S_K} p_i \log_K p_i \prod_{i=1}^{K} p_i^{\alpha_i - 1} d\boldsymbol{p} \\
&= -\frac{1}{\ln K}\sum_{i=1}^{K} \frac{\Gamma(S)}{\prod_{i=1}^{K}\Gamma(\alpha_i)} \int_{S_K} p_i \ln p_i \prod_{i=1}^{K} p_i^{\alpha_i - 1} d\boldsymbol{p} \\
&= -\frac{1}{\ln K}\sum_{i=1}^{K} \frac{\alpha_i}{S}\frac{\Gamma(S+1)}{\Gamma(\alpha_i+1)\prod_{i'=1,\neq i}^{K}\Gamma(\alpha_{i'})} \int_{S_K} p_i^{\alpha_i} \ln p_i \prod_{i'=1,\neq i}^{K} p_{i'}^{\alpha_{i'} - 1} d\boldsymbol{p} \\
&= \frac{1}{\ln K}\sum_{i=1}^{K} \frac{\alpha_i}{S}\big(\psi(S+1) - \psi(\alpha_i+1)\big) \\
&= \frac{1}{\ln K}\sum_{i=1}^{K} \frac{1}{K}(\psi(K+1) - \psi(2)) \\
&= \frac{1}{\ln K}(\psi(K+1) - \psi(2)) \\
&= \frac{1}{\ln K}(\psi(2) + \sum_{k=2}^{K}\frac{1}{k} - \psi(2)) \\
&= \frac{1}{\ln K}\sum_{k=2}^{K}\frac{1}{k} < \frac{1}{\ln K}\ln K = 1,
\end{aligned}
\tag{23}
$$

where $S = \sum_{i=1}^{K}\alpha_i$, $\boldsymbol{p} = [p_1, \ldots, p_K]^\top$, and $K \geq 2$ is the number of category. The epistemic uncertainty can be calculated via the mutual information,

$$
\begin{aligned}
u_{epis} &= \mathcal{H}[\mathbb{E}_{p\sim \text{Dir}(\alpha)}[p]] - \mathbb{E}_{p\sim \text{Dir}(\alpha)}[\mathcal{H}[p]] \\
&= \mathcal{H}[\hat{p}] - u_{alea} \\
&= 1 - \frac{1}{\ln K}\sum_{k=2}^{K}\frac{1}{k} < 1.
\end{aligned}
\tag{24}
$$

To compare aleatoric uncertainty with epistemic uncertainty, we first prove that aleatoric uncertainty (Eq. (24)) is monotonically increasing and converging to 1 as K increases. Based on *Lemma 1*, we have

$$
\begin{aligned}
&\Big(\ln(K+1) - \ln K\Big)\sum_{k=2}^{K}\frac{1}{k} < \frac{\ln K}{K+1} \\
&\Rightarrow \ln(K+1)\sum_{k=2}^{K}\frac{1}{k} < \ln K\Big(\sum_{k=2}^{K}\frac{1}{k} + \frac{1}{K+1}\Big) = \ln K\sum_{k=2}^{K+1}\frac{1}{k} \\
&\Rightarrow \frac{1}{\ln K}\sum_{k=2}^{K}\frac{1}{k} < \frac{1}{\ln(K+1)}\sum_{k=2}^{K+1}\frac{1}{k}.
\end{aligned}
\tag{25}
$$

Based on Eq. (25) and Eq. (24), we prove that aleatoric uncertainty is monotonically increasing with respect to K. So the minimum aleatoric can be shown to be $\frac{1}{\ln 2}\frac{1}{2}$, when $K = 2$.

Similarly, for epistemic uncertainty, which is monotonically decreasing as K increases based on *Lemma 1*, the maximum epistemic can be shown to be $1 - \frac{1}{\ln 2}\frac{1}{2}$ when $K = 2$. Then we have,

$$
u_{alea} \geq \frac{1}{\ln 2}\frac{1}{2} > 1 - \frac{1}{2\ln 2} \geq u_{epis} \tag{26}
$$

Therefore, we prove that $1 = u_v = u_{en} > u_{alea} > u_{epis} > u_{diss} = 0$.

2. (b) For a conflicting prediction, i.e., $\alpha = [\alpha_1, \ldots, \alpha_K]$, with $\alpha_1 = \alpha_2 = \cdots = \alpha_K = C$, and $S = \sum_{i=1}^{K}\alpha_i = CK$, the expected probability $\hat{p} = [1/K, \ldots, 1/K]^\top$, the belief mass $b_i = (\alpha_i - 1)/S$, and the vacuity can be calculated as

$$
u_v = \frac{K}{S} \xrightarrow{S\to\infty} 0, \tag{27}
$$

and the dissonance can be calculated as

$$\begin{aligned} u_{diss} &= \sum_{i=1}^{K} \Big(\frac{b_i \sum_{j=1, j\neq i}^{K} b_j \text{Bal}(b_i, b_j)}{\sum_{j=1, j\neq i}^{K} b_j} \Big) = \sum_{i=1}^{K} b_i \\ &= \sum_{i=1}^{K} \left(\frac{a_i - 1}{\sum_{i=1}^{K} a_i} \right) \\ &= \frac{\sum_{i=1}^{K} a_i - k}{\sum_{i=1}^{K} a_i} \\ &= 1 - \frac{K}{S} \xrightarrow{S\to\infty} 1. \end{aligned} \tag{28}$$

Given the expected probability $\hat{p} = [1/K, \ldots, 1/K]^\top$, the entropy can be calculated based on Dirichlet distribution,

$$u_{en} = \mathcal{H}[\hat{p}] = \sum_{i=1}^{K} \hat{p}_i \log_K \hat{p}_i = 1, \tag{29}$$

and the aleatoric uncertainty is estimated as the expected entropy,

$$\begin{aligned} u_{alea} &= \mathbb{E}_{p\sim \text{Dir}(\alpha)}[\mathcal{H}[p]] \\ &= -\sum_{i=1}^{K} \frac{\Gamma(S)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \int_{S_K} p_i \log_K p_i \prod_{i=1}^{K} p_i^{\alpha_i - 1} d\boldsymbol{p} \\ &= -\frac{1}{\ln K} \sum_{i=1}^{K} \frac{\Gamma(S)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \int_{S_K} p_i \ln p_i \prod_{i=1}^{K} p_i^{\alpha_i - 1} d\boldsymbol{p} \\ &= -\frac{1}{\ln K} \sum_{i=1}^{K} \frac{\alpha_i}{S} \frac{\Gamma(S+1)}{\Gamma(\alpha_i + 1) \prod_{i'=1, \neq i}^{K} \Gamma(\alpha_{i'})} \int_{S_K} p_i^{\alpha_i} \ln p_i \prod_{i'=1, \neq i}^{K} p_{i'}^{\alpha_{i'} - 1} d\boldsymbol{p} \\ &= \frac{1}{\ln K} \sum_{i=1}^{K} \frac{\alpha_i}{S} \big(\psi(S+1) - \psi(\alpha_i + 1) \big) \\ &= \frac{1}{\ln K} \sum_{i=1}^{K} \frac{1}{K} (\psi(S+1) - \psi(C+1)) \\ &= \frac{1}{\ln K} (\psi(S+1) - \psi(C+1)) \\ &= \frac{1}{\ln K} (\psi(C+1) + \sum_{k=C+1}^{S} \frac{1}{k} - \psi(C+1)) \\ &= \frac{1}{\ln K} \sum_{k=C+1}^{S} \frac{1}{k} \xrightarrow{S\to\infty} 1. \end{aligned} \tag{30}$$

The epistemic uncertainty can be calculated via mutual information,

$$\begin{aligned} u_{epis} &= \mathcal{H}[\mathbb{E}_{p\sim \text{Dir}(\alpha)}[p]] - \mathbb{E}_{p\sim \text{Dir}(\alpha)}[\mathcal{H}[p]] \\ &= \mathcal{H}[\hat{p}] - u_{alea} \\ &= 1 - \frac{1}{\ln K} \sum_{k=C+1}^{S} \frac{1}{k} \xrightarrow{S\to\infty} 0. \end{aligned} \tag{31}$$

Now we compare aleatoric uncertainty with vacuity,

$$\begin{aligned} u_{alea} &= \frac{1}{\ln K}\sum_{k=C+1}^{S}\frac{1}{k} \\ &= \frac{1}{\ln K}\sum_{k=C+1}^{CK}\frac{1}{k} \\ &= \frac{\ln(CK+1)-\ln(C+1)}{\ln K} \\ &= \frac{\ln(K-\frac{K-1}{C+1})}{\ln K} \\ &> \frac{\ln(K-\frac{K-1}{2})}{\ln K} \\ &= \frac{\ln(4/K+4/K+1/2)}{\ln K} \\ &\geq \frac{\ln[3(4/K+4/K+1/2)^{\frac{1}{3}}]}{\ln K} \\ &= \frac{\ln 3+\frac{1}{3}\ln(\frac{K^2}{32})}{\ln K} \\ &= \frac{\ln 3+\frac{2}{3}\ln K-\frac{1}{3}\ln 32}{\ln K} > \frac{2}{3}. \end{aligned} \tag{32}$$

Based on Eq. (33), when $C > \frac{3}{2}$, we have

$$u_{alea} > \frac{2}{3} > \frac{1}{C} = u_v \tag{33}$$

We have already proved that $u_v > u_{epis}$, when $u_{en} = 1$, we have $u_{alea} > u_{diss}$ Therefore, we prove that $u_{en} > u_{alea} > u_{diss} > u_v > u_{epis}$ with $u_{en} = 1, u_{diss} \to 1, u_{alea} \to 1, u_v \to 0, u_{epis} \to 0$ □

Lemma 1. *For all integer $N \geq 2$, we have $\sum_{n=2}^{N}\frac{1}{n} < \frac{\ln N}{(N+1)\ln(\frac{N+1}{N})}$.*

Proof. We will prove by induction that, for all integer $N \geq 2$,

$$\sum_{n=2}^{N}\frac{1}{n} < \frac{\ln N}{(N+1)\ln(\frac{N+1}{N})}. \tag{34}$$

Base case: When $N = 2$, we have $\frac{1}{2} < \frac{\ln 2}{3\ln\frac{3}{2}}$ and Eq. (34) is true for $N = 2$.

Induction step: Let the integer $K \geq 2$ is given and suppose Eq. (34) is true for $N = K$, then

$$\sum_{k=2}^{K+1}\frac{1}{k} = \frac{1}{K+1}+\sum_{k=2}^{K}\frac{1}{k} < \frac{1}{K+1}+\frac{\ln K}{(K+1)\ln(\frac{K+1}{K})} = \frac{\ln(K+1)}{(K+1)\ln(\frac{K+1}{K})}. \tag{35}$$

Denote that $g(x) = (x+1)\ln(\frac{x+1}{x})$ with $x > 2$. We get its derivative, $g'(x) = \ln(1+\frac{1}{x}) - \frac{1}{x} < 0$, such that $g(x)$ is monotonically decreasing, which results in $g(K) > g(K+1)$. Based on Eq. (35) we have,

$$\sum_{k=2}^{K+1}\frac{1}{k} < \frac{\ln(K+1)}{g(K)} < \frac{\ln(K+1)}{g(K+1)} = \frac{\ln(K+1)}{(K+2)\ln(\frac{K+2}{K+1})}. \tag{36}$$

Thus, Eq. (34) holds for $N = K+1$, and the proof of the induction step is complete.

Conclusion: By the principle of induction, Eq. (34) is true for all integer $N \geq 2$. □

A.2 Proposition 1's Proof

Proposition 1. *Given L training nodes, for any testing nodes i and j, let $\boldsymbol{d}_i = [d_{i1}, \ldots, d_{iL}]$ be the vector of graph distances from nodes i to training nodes and $\boldsymbol{d}_j = [d_{j1}, \ldots, d_{jL}]$ be the graph distances from nodes j to training nodes, where d_{il} is the node-level distance between nodes i and l. If for all $l \in \{1, \ldots, L\}$, $d_{il} \geq d_{jl}$, then we have*

$$\hat{u}_{v_i} \geq \hat{u}_{v_j},$$

where $\hat{u}_{v_i}$ and $\hat{u}_{v_j}$ refer to vacuity uncertainties of nodes i and j estimated based on GKDE.

Interpretation. From the above proposition, if a testing node is too distant (far away) from training nodes, the vacuity increases, indicating that an OOD node is expected to have a high vacuity value.

Proof. Let $\boldsymbol{y} = [y_1, \ldots, y_L]$ be the label vector for training nodes. Based on GKDE, the evidence contribution for the node i and a training node $l \in \{1, \ldots, L|\}$ is $\boldsymbol{h}(y_l, d_{il}) = [h_1(y_l, d_{il}), \ldots, h_K(y_l, d_{il})]$, where

$$h_k(y_l, d_{il}) = \begin{cases} 0 & y_l \neq k \\ g(d_{il}) = \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{{d_{il}}^2}{2\sigma^2}) & y_l = k \end{cases}, \tag{37}$$

and the prior evidence can be estimated based GKDE:

$$\hat{\boldsymbol{e}}_i = \sum_{m=1}^{L} \sum_{k=1}^{K} h_k(y_l, d_{il}), \tag{38}$$

where $\hat{\boldsymbol{e}}_i = [e_{i1}, ..., e_{iK}]$. Since each training node only contributes the same evidence based on its label based on Eq. (37), the total evidence is estimated by all the contributing evidence as

$$\sum_{k=1}^{K} e_{ik} = \sum_{m=1}^{L} \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{{d_{il}}^2}{2\sigma^2}), \quad \sum_{k=1}^{K} e_{jk} = \sum_{m=1}^{L} \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{{d_{jl}}^2}{2\sigma^2}), \tag{39}$$

where the vacuity values for node i and node j based on GKDE are,

$$\hat{u}_{v_i} = \frac{K}{\sum_{k=1}^{K} e_{ik} + K}, \quad \hat{u}_{v_j} = \frac{K}{\sum_{k=1}^{K} e_{jk} + K}. \tag{40}$$

Now, we prove Eq. (40) above. If $d_{il} \geq d_{jl}$ for $\forall l \in \{1, \ldots, L\}$, we have

$$\begin{aligned} \sum_{k=1}^{K} e_{ik} &= \sum_{m=1}^{L} \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{{d_{il}}^2}{2\sigma^2}) \\ &\leq \sum_{m=1}^{L} \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{{d_{jl}}^2}{2\sigma^2}) \\ &= \sum_{k=1}^{K} e_{jk}, \end{aligned} \tag{41}$$

such that

$$\hat{u}_{v_i} = \frac{K}{\sum_{k=1}^{K} e_{ik} + K} \geq \frac{K}{\sum_{k=1}^{K} e_{jk} + K} = \hat{u}_{v_j}. \tag{42}$$

□

B Additional Experimental Details

B.1 Source code

The source code and datasets are accessible at https://github.com/zxj32/uncertainty-GNN

B.2 Description of Datasets

Table 4: Description of datasets and their experimental setup for the node classification prediction.

	Cora	**Citeseer**	**Pubmed**	**Co. Physics**	**Ama.Computer**	**Ama.Photo**
#Nodes	2,708	3,327	19,717	34, 493	13, 381	7, 487
#Edges	5,429	4,732	44,338	282, 455	259, 159	126, 530
#Classes	7	6	3	5	10	8
#Features	1,433	3,703	500	8,415	767	745
#Training nodes	140	120	60	100	200	160
#Validation nodes	500	500	500	500	500	500
#Test nodes	1,000	1,000	1,000	1000	1,000	1000

Cora, Citeseer, and Pubmed [25]: These are citation network datasets, where each network is a directed network in which a node represents a document and an edge is a citation link, meaning that there exists an

edge when A document cites B document, or vice-versa with a direction. Each node's feature vector contains a bag-of-words representation of a document. For simplicity, we don't discriminate the direction of links and treat citation links as undirected edges and construct a binary, symmetric adjacency matrix $\mathbf{A}$. Each node is labeled with the class to which it belongs.

Coauthor Physics, Amazon Computers, and Amazon Photo [28]: Coauthor Physics is the dataset for co-authorship graphs based on the Microsoft Academic Graph from the KDD Cup 2016 Challenge[2]. In the graphs, a node is an author and an edge exists when two authors co-author a paper. A node's features represent the keywords of its papers and the node's class label indicates its most active field of study. Amazon Computers and Amazon Photo are the segments of an Amazon co-purchase graph [21], where a node is a good (i.e., product), an edge exists when two goods are frequently bought together. A node's features are bag-of-words representation of product reviews and the node's class label is the product category.

For all the used datasets, we deal with undirected graphs with 20 training nodes for each category. We chose the same dataset splits as in [31] with an additional validation node set of 500 labeled examples for the hyperparameter obtained from the citation datasets, and followed the same dataset splits in [28] for Coauthor Physics, Amazon Computer, and Amazon Photo datasets, for the fair comparison[3].

Metric: We used the following metrics for our experiments:

- *Area Under Receiver Operating Characteristics (AUROC)*: AUROC shows the area under the curve where FPR (false positive rate) is in x-axis and TPR (true positive rate) is in y-axis. It can be interpreted as the probability that a positive example is assigned a higher detection score than a negative example[4]. A perfect detector corresponds to an AUROC score of 100%.
- *Area Under Precision-Prediction Curve (AUPR)*: The PR curve is a graph showing the precision=TP/(TP+FP) and recall=TP/(TP+FN) against each other,and AUPR denotes the area under the precision-recall curve. The ideal case is when Precision is 1 and Recall is 1.

B.3 Experimental Setup for Out-of-Distribution (OOD) Detection

For OOD detection on semi-supervised node classification, we randomly selected 1-4 categories as OOD categories and trained the models only based on training nodes of the other categories. In this setting, we still trained a model for semi-supervised node classification task, but only part of node categories were not used for training. Hence, we suppose that our model only outputs partial categories (as we don't know the OOD category), see Table 5. For example, Cora dataset, we trained the model with 80 nodes (20 nodes for each category) with the predictions of 4 categories. Positive ratio is the ratio of out-of-distribution nodes among on all test nodes.

Table 5: Description of datasets and their experimental setup for the OOD detection.

Dataset	**Cora**	**Citeseer**	**Pubmed**	**Co.Physics**	**Ama.Computer**	**Ama.Photo**
Number of training categories	4	3	2	3	5	4
Training nodes	80	60	40	60	100	80
Test nodes	1000	1000	1000	1000	1000	1000
Positive ratio	38%	55%	40.4%	45.1%	48.1%	51.1%

B.4 Baseline Setting

In experiment part, we considered 4 baselines. For GCN, we used the same hyper-parameters as [16]. For EDL-GCN, we used the same hyper-parameters as GCN, and replaced softmax layer to activation layer (Relu) with squares loss [26]. For DPN-GCN, we used the same hyper-parameters as GCN, and changed the softmax layer to activation layer (exponential). Note that as we can not generate OOD node, we only used in-distribution loss of (see Eq.12 in [20]) and ignored the OOD part loss. For Drop-GCN, we used the same hyper-parameters as GCN, and set Monte Carlo sampling times $M = 100$, dropout rate equal to 0.5.

B.5 Time Complexity Analysis

S-BGCN has a similar time complexity with GCN while S-BGCN-T has the double complexity of GCN. For a given network where $|\mathbb{V}|$ is the number of nodes, $|\mathbb{E}|$ is the number of edges, C is the number of dimensions of the input feature vector for every node, F is the number of features for the output layer, and M is Monte Carlo sampling times.

[2]KDD Cup 2016 Dataset: Online Available at `https://kddcup2016.azurewebsites.net/`

[3]The source code and datasets are accessible at https://github.com/zxj32/uncertainty-GNN

Table 6: Big-O time complexity of our method and baseline GCN.

Dataset	GCN	S-GCN	S-BGCN	S-BGCN-T	S-BGCN-T-K
Time Complexity (Train)	$O(\|\mathbb{E}\|CF)$	$O(\|\mathbb{E}\|CF)$	$O(2\|\mathbb{E}\|CF)$	$O(2\|\mathbb{E}\|CF)$	$O(2\|\mathbb{E}\|CF)$
Time Complexity (Test)	$O(\|\mathbb{E}\|CF)$	$O(\|\mathbb{E}\|CF)$	$O(M\|\mathbb{E}\|CF)$	$O(M\|\mathbb{E}\|CF)$	$O(M\|\mathbb{E}\|CF)$

B.6 Model Setups for semi-supervised node classification

Our models were initialized using Glorot initialization [8] and trained to minimize loss using the Adam SGD optimizer [15]. For the S-BGCN-T-K model, we used the *early stopping strategy* [28] on Coauthor Physics, Amazon Computer and Amazon Photo datasets while *non-early stopping strategy* was used in citation datasets (i.e., Cora, Citeseer and Pubmed). We set bandwidth $\sigma = 1$ for all datasets in GKDE, and set trade off parameters $\lambda_1 = 0.001$ for misclassification detection, $\lambda_1 = 0.1$ for OOD detection and $\lambda_2 = \min(1, t/200)$ (where t is the index of a current training epoch) for both task; other hyperparameter configurations are summarized in Table 7.

For semi-supervised node classification, we used 50 random weight initialization for our models on Citation network datasets. For Coauthor Physics, Amazon Computer and Amazon Photo datasets, we reported the result based on 10 random train/validation/test splits. In both effect of uncertainty on misclassification and the OOD detection, we reported the AUPR and AUROC results in percent averaged over 50 times of randomly chosen 1000 test nodes in all of test sets (except training or validation set) for all models tested on the citation datasets. For S-BGCN-T-K model in these tasks, we used the same hyperparameter configurations as in Table 7, except S-BGCN-T-K Epistemic using 10,000 epochs to obtain the best result.

Table 7: Hyperparameter configurations of S-BGCN-T-K model

	Cora	Citeseer	Pubmed	Co.Physics	Ama.Computer	Ama.Photo
Hidden units	16	16	16	64	64	64
Learning rate	0.01	0.01	0.01	0.01	0.01	0.01
Dropout	0.5	0.5	0.5	0.1	0.2	0.2
L_2 **reg.strength**	0.0005	0.0005	0.0005	0.001	0.0001	0.0001
Monte-Carlo samples	100	100	100	100	100	100
Max epoch	200	200	200	100000	100000	100000

B.7 Pseudo code for Our Algorithms

Algorithm 1: S-BGCN-T-K

Input: $\mathbb{G} = (\mathbb{V}, \mathbb{E}, \mathbf{r})$ and $\mathbf{y}_{\mathbb{L}}$
Output: $\mathbf{p}_{\mathbb{V}\setminus\mathbb{L}}$, $\mathbf{u}_{\mathbb{V}\setminus\mathbb{L}}$

$\ell = 0$;
Set hyper-parameters $\eta, \lambda_1, \lambda_2$;
Initialize the parameters γ, β;
Calculate the prior Dirichlet distribution Dir($\hat{\alpha}$);
Pretrain the teacher network to get Prob($\mathbf{y}|\hat{\mathbf{p}}$);
repeat
  Forward pass to compute $\boldsymbol{\alpha}$, Prob($\mathbf{p}_i|A, \mathbf{r}; \mathcal{G}$) for $i \in \mathbb{V}$;
  Compute joint probability Prob($\mathbf{y}|A, \mathbf{r}; \mathcal{G}$);
  Backward pass via the chain-rule the calculate the sub-gradient gradient: $g^{(\ell)} = \nabla_{\Theta}\mathcal{L}(\Theta)$
  Update parameters using step size η via $\Theta^{(\ell+1)} = \Theta^{(\ell)} - \eta \cdot g^{(\ell)}$
  $\ell = \ell + 1$;
until *convergence*
Calculate $\mathbf{p}_{\mathbb{V}\setminus\mathbb{L}}$, $\mathbf{u}_{\mathbb{V}\setminus\mathbb{L}}$
return $\boldsymbol{p}_{\mathbb{V}\setminus\mathbb{L}}$, $\boldsymbol{u}_{\mathbb{V}\setminus\mathbb{L}}$

B.8 Bayesian Inference with Dropout

The marginalization in Eq.(8) (in main paper) is generally intractable. A dropout technique is used to obtain an approximate solution and use samples from the posterior distribution of models [7]. Hence, we adopted a dropout technique in [6] for variational inference in Bayesian convolutional neural networks where Bernoulli distributions are assumed over the network's weights. This dropout technique allows us to perform probabilistic

inference over our Bayesian DL framework using GNNs. For Bayesian inference, we identified a posterior distribution over the network's weights, given the input graph $\mathcal{D}$ and observed labels $\mathbf{y}_{\mathbb{L}}$ by Prob($\boldsymbol{\theta}|\mathcal{D}$), where $\boldsymbol{\theta} = \{\mathbf{W}_1, \ldots, \mathbf{W}_L, b_1, ..., b_L\}$, L is the total number of layers and W_i refers to the GNN's weight matrices of dimensions $D_i \times D_{i-1}$, and b_i is a bias vector of dimensions D_i for layer $i = 1, \cdots, L$.

Since the posterior distribution is intractable, we used a **variational inference** to learn $q(\boldsymbol{\theta})$, a distribution over matrices whose columns are randomly set to zero, approximating the intractable posterior by minimizing the Kullback-Leibler (KL)-divergence between this approximated distribution and the full posterior, which is given by:

$$\text{KL}(q(\boldsymbol{\theta})\|\text{Prob}(\boldsymbol{\theta}|\mathcal{D})) \tag{43}$$

We define $\mathbf{W}_i$ in $q(\boldsymbol{\theta})$ by:

$$\mathbf{W}_i = \mathbf{M}_i \text{diag}([z_{ij}]_{j=1}^{D_i}), \quad z_{ij} \sim \text{Bernoulli}(d_i) \text{ for } i = 1, \ldots, L, j = 1, \ldots, D_{i-1} \tag{44}$$

where $\boldsymbol{\gamma} = \{\mathbf{M}_1, \ldots, \mathbf{M}_L, \mathbf{m}_1, \ldots, \mathbf{m}_L\}$ are the variational parameters, $\mathbf{M}_i \in \mathbb{R}^{D_i \times D_{i-1}}$, $\mathbf{m}_i \in \mathbb{R}^{D_i}$, and $\mathbf{d} = \{d_1, \ldots, d_L\}$ is the dropout probabilities with z_{ij} of Bernoulli distributed random variables. The binary variable $z_{ij} = 0$ corresponds to unit j in layer $i-1$ being dropped out as an input to layer i. We can obtain the approximate model of the Gaussian process from [6]. The dropout probabilities, d_i's, can be optimized or fixed [13]. For simplicity, we fixed d_i's in our experiments, as it is beyond the scope of our study. In [6], the minimization of the cross entropy (or square error) loss function is proven to minimize the KL-divergence (see Eq. (43)). Therefore, training the GNN model with stochastic gradient descent enables learning of an approximated distribution of weights, which provides good explainability of data and prevents overfitting.

For the dropout inference, we performed training on a DL model with dropout before every weight layer and dropout at a test time to sample from the approximate posterior (i.e., stochastic forward passes, a.k.a. Monte Carlo dropout; see Eq. (45)). At the test stage, we infer the joint probability by:

$$\begin{aligned} p(\mathbf{y}|A, \mathbf{r}; \mathcal{D}) &= \int\int \text{Prob}(\mathbf{y}|\mathbf{p})\text{Prob}(\mathbf{p}|A, \mathbf{r}; \boldsymbol{\theta})\text{Prob}(\boldsymbol{\theta}|\mathcal{D})d\mathbf{p}d\boldsymbol{\theta} \\ &\approx \frac{1}{M}\sum\nolimits_{m=1}^{M} \int \text{Prob}(\mathbf{y}|\mathbf{p})\text{Prob}(\mathbf{p}|A, \mathbf{r}; \boldsymbol{\theta}^{(m)})d\mathbf{p}, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}), \end{aligned} \tag{45}$$

where M is Monte Carlo sampling times. We can also infer the Dirichlet parameters $\boldsymbol{\alpha}$ as:

$$\boldsymbol{\alpha} \approx \frac{1}{M}\sum\nolimits_{m=1}^{M} f(A, \mathbf{r}, \boldsymbol{\theta}^{(m)}), \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}). \tag{46}$$

C Additional Experimental Results

In addition to the uncertainty analysis in Section 5, we also conducted additional experiments. **First**, we conducted an ablation experiment for each component (such as GKDE, Teacher network, Subjective framework and Bayesian framework) we proposed. **Second**, we provide additional uncertainty visualization results in network node classifications for Citeseer dataset. To clearly understand the effect of different types of uncertainty in classification accuracy and OOD, we used the AUROC and AUPR curves for all types of models considered in this work.

C.1 Ablation Experiments

We conducted an additional experiments in order to clearly demonstrate the contributions of the key technical components, including a teacher Network, Graph kernel Dirichlet Estimation (GKDE) and subjective Bayesian framework. The key findings obtained from this experiment are: (1) The teacher Network can further improve node classification accuracy (i.e., 0.2% - 1.5% increase, as shown in Table 8); and (2) GKDE (Graph-Based Kernel Dirichlet Distribution Estimation) using the uncertainty estimates can enhance OOD detection (i.e., 4% - 30% increase, as shown in Table 9).

C.2 Experiment based on GAT model

We also conducted the semi-supervised node classification based on GAT model [30]).Model setup: The S-BGAT-T-K model has two dropout probabilities, which are a dropout on features and a dropout on attention coefficients, as shown in Table 10. We changed the dropout on attention coefficients to 0.4 at the test stage and set trade off parameters $\lambda = \min(1, t/50)$, using the same early stopping strategy [30]. The result are shown in Table 11.

C.3 Misclassification Detection

For Amazon Photo, Amazon Computer and Coauthor Physics dataset, the misclassification detection results are shown in Tabel 12.

Table 8: Ablation experiment on AUROC and AUPR for the Misclassification Detection.

Data	Model	AUROC					AUPR					
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.	Acc
Cora	S-BGCN-T-K	70.6	82.4	75.3	68.8	77.7	90.3	**95.4**	92.4	87.8	93.4	82.0
	S-BGCN-T	70.8	**82.5**	75.3	68.9	77.8	90.4	**95.4**	92.6	88.0	93.4	**82.2**
	S-BGCN	69.8	81.4	73.9	66.7	76.9	89.4	94.3	92.3	88.0	93.1	81.2
	S-GCN	70.2	81.5	-	-	76.9	90.0	94.6	-	-	93.6	81.5
Citeseer	S-BGCN-T-K	65.4	**74.0**	67.2	60.7	70.0	79.8	**85.6**	82.2	75.2	83.5	71.0
	S-BGCN-T	65.4	73.9	67.1	60.7	70.1	79.6	85.5	82.1	75.2	83.5	**71.3**
	S-BGCN	63.9	72.1	66.1	58.9	69.2	78.4	83.8	80.6	75.6	82.3	70.6
	S-GCN	64.9	71.9	-	-	69.4	79.5	84.2	-	-	82.5	71.0
Pubmed	S-BGCN-T-K	63.1	**69.9**	66.5	65.3	68.1	85.6	90.8	88.8	86.1	89.2	**79.3**
	S-BGCN-T	63.2	**69.9**	66.6	65.3	64.8	85.6	**90.9**	88.9	86.0	89.3	79.2
	S-BGCN	62.7	68.1	66.1	64.4	68.0	85.4	90.5	88.6	85.6	89.2	78.8
	S-GCN	62.9	69.5	-	-	68.0	85.3	90.4	-	-	89.2	79.1
Amazon Photo	S-BGCN-T-K	66.0	89.3	83.0	83.4	83.2	95.4	**98.9**	98.4	98.1	98.4	92.0
	S-BGCN-T	66.1	89.3	83.1	83.5	83.3	95.6	99.0	98.4	98.2	98.4	**92.3**
	S-BGCN	68.6	**93.6**	90.6	83.6	90.6	90.4	98.1	97.3	95.8	97.3	81.0
	S-GCN	-	-	-	-	86.7	-	-	-	-	-	98.4
Amazon Computer	S-BGCN-T-K	65.0	87.8	83.3	79.6	83.6	89.4	96.3	95.0	94.2	95.0	84.0
	S-BGCN-T	65.2	88.0	83.4	79.7	83.6	89.4	**96.5**	95.0	94.5	95.1	**84.1**
	S-BGCN	63.7	**89.1**	84.3	76.1	84.4	84.9	95.7	93.9	91.4	93.9	76.1
	S-GCN	-	-	-	-	81.5	-	-	-	-	-	95.2
Coauthor Physics	S-BGCN-T-K	80.2	91.4	87.5	81.7	87.6	98.3	**99.4**	99.0	98.4	98.9	93.0
	S-BGCN-T	80.4	**91.5**	87.6	81.7	87.6	98.3	**99.4**	99.0	98.6	99.0	**93.2**
	S-BGCN	79.6	90.5	86.3	81.2	86.4	98.0	99.2	98.8	98.3	98.8	92.9
	S-GCN	89.1	89.0	-	-	89.2	99.0	99.0	-	-	99.0	92.9

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, En.: Entropy

Table 9: Ablation experiment on AUROC and AUPR for the OOD Detection.

Data	Model	AUROC					AUPR				
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.
Cora	S-BGCN-T-K	**87.6**	75.5	85.5	70.8	84.8	**78.4**	49.0	75.3	44.5	73.1
	S-BGCN-T	84.5	81.2	83.5	71.8	83.5	74.4	53.4	75.8	46.8	71.7
	S-BGCN	76.3	79.3	81.5	70.5	80.6	61.3	55.8	68.9	44.2	65.3
	S-GCN	75.0	78.2	-	-	79.4	60.1	54.5	-	-	65.3
Citeseer	S-BGCN-T-K	**84.8**	55.2	78.4	55.1	74.0	**86.8**	54.1	80.8	55.8	74.0
	S-BGCN-T	78.6	59.6	73.9	56.1	69.3	79.8	57.4	76.4	57.8	69.3
	S-BGCN	72.7	63.9	72.4	61.4	70.5	73.0	62.7	74.5	60.8	71.6
	SGCN	72.0	62.8	-	-	70.0	71.4	61.3	-	-	70.5
Pubmed	S-BGCN-T-K	**74.6**	67.9	71.8	59.2	72.2	**69.6**	52.9	63.6	44.0	56.5
	S-BGCN-T	71.8	68.6	70.0	60.1	70.8	65.7	53.9	61.8	46.0	55.1
	S-BGCN	70.8	68.2	70.3	60.8	68.0	65.4	53.2	62.8	46.7	55.4
	S-GCN	71.4	68.8	-	-	69.7	66.3	54.9	-	-	57.5
Amazon Photo	S-BGCN-T-K	**93.4**	76.4	91.4	32.2	91.4	**94.8**	68.0	92.3	42.3	92.5
	S-BGCN-T	64.0	77.5	79.9	52.6	79.8	67.0	75.3	82.0	53.7	81.9
	S-BGCN	63.0	76.6	79.8	52.7	79.7	66.5	75.1	82.1	53.9	81.7
	S-GCN	64.0	77.1	-	-	79.6	67.0	74.9	-	-	81.6
Amazon Computer	S-BGCN-T-K	**82.3**	76.6	80.9	55.4	80.9	**70.5**	52.8	60.9	35.9	60.6
	S-BGCN-T	53.7	70.5	70.4	69.9	70.1	33.6	43.9	46.0	46.8	45.9
	S-BGCN	56.9	75.3	74.1	73.7	74.1	33.7	46.2	48.3	45.6	48.3
	S-GCN	56.9	75.3	-	-	74.2	33.7	46.2	-	-	48.3
Coauthor Physics	S-BGCN-T-K	**91.3**	87.6	89.7	61.8	89.8	**72.2**	56.6	68.1	25.9	67.9
	S-BGCN-T	88.7	86.0	87.9	70.2	87.8	67.4	51.9	64.6	29.4	62.4
	S-BGCN	89.1	87.1	89.5	78.3	89.5	66.1	49.2	64.6	35.6	64.3
	S-GCN	89.1	87.0	-	-	89.4	-66.2	49.2	-	-	64.3

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, D.En.: Differential Entropy, En.: Entropy

C.4 Graph Embedding Representations of Different Uncertainty Types

To better understand different uncertainty types, we used t-SNE (t-Distributed Stochastic Neighbor Embedding [19]) to represent the computed feature representations of a pre-trained BGCN-T model's first hidden layer on the Cora dataset and the Citeseer dataset.

Seven Classes on Cora Dataset: In Figure 4, (a) shows the representation of seven different classes, (b) shows our model prediction and (c)-(f) present the extent of uncertainty for respective uncertainty types, including vacuity, dissonance, and aleatoric uncertainty, respectively.

Six Classes on Citeseer Dataset: In Figure 5 (a), a node's color denotes a class on the Citeseer dataset where 6 different classes are shown in different colors. Figure 5 (b) is our prediction result.

Table 10: Hyper-parameters of S-BGAT-T-K model

	Cora	**Citeseer**	**Pubmed**
Hidden units	64	64	64
Learning rate	0.01	0.01	0.01
Dropout	0.6/0.6	0.6/0.6	0.6/0.6
L_2 **reg.strength**	0.0005	0.0005	0.001
Monte-Carlo samples	100	100	100
Max epoch	100000	100000	100000

Table 11: Semi-supervised node classification accuracy based on GAT

	Cora	**Citeseer**	**Pubmed**
GAT	83.0 ± 0.7	72.5 ± 0.7	79.0 ± 0.3
GAT-Drop	82.8 ± 0.8	72.6 ± 0.7	79.0 ± 0.3
S-GAT	83.0 ± 0.7	72.6 ± 0.6	79.0 ± 0.3
S-BGAT	82.9 ± 0.7	72.4 ± 0.7	78.9 ± 0.3
S-BGAT-T	83.7 ± 0.6	$\mathbf{73.2 \pm 0.5}$	79.1 ± 0.2
S-BGAT-T-K	$\mathbf{83.8 \pm 0.7}$	73.0 ± 0.7	$\mathbf{79.1 \pm 0.2}$

Table 12: AUROC and AUPR for the Misclassification Detection.

Data	Model	AUROC					AUPR					Acc
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.	
Amazon Photo	S-BGCN-T-K	66.0	**89.3**	83.0	83.4	83.2	95.4	**98.9**	98.4	98.1	98.4	**92.0**
	EDL-GCN	65.1	88.5	-	-	82.2	94.6	98.1	-	-	98.0	91.2
	DPN-GCN	-	-	81.8	80.8	81.3	-	-	98.1	98.0	98.0	**92.0**
	Drop-GCN	-	-	84.5	84.4	84.6	-	-	98.2	98.1	98.2	91.3
	GCN	-	-	-	-	86.8	-	-	-	-	98.5	91.2
Amazon Computer	S-BGCN-T-K	65.0	**87.8**	83.3	79.6	83.6	89.4	**96.3**	95.0	94.2	95.0	84.0
	EDL-GCN	64.1	86.5	-	-	82.2	93.6	97.1	-	-	97.0	79.7
	DPN-GCN	-	-	76.8	76.0	76.3	-	-	94.5	94.3	94.4	**84.8**
	Drop-GCN	-	-	79.1	75.9	79.2	-	-	95.1	94.5	95.1	79.6
	GCN	-	-	-	-	81.7	-	-	-	-	95.4	82.6
Coauthor Physics	S-BGCN-T-K	80.2	**91.4**	87.5	81.7	87.6	98.3	**99.4**	99.0	98.4	98.9	**93.0**
	EDL-GCN	78.8	89.5	-	-	86.2	96.6	97.2	-	-	97.0	92.7
	DPN-GCN	-	-	87.0	86.4	86.8	-	-	99.1	99.0	99.0	92.5
	Drop-GCN	-	-	87.6	84.1	87.7	-	-	98.9	98.6	98.9	93.0
	GCN	-	-	-	-	88.7	-	-	-	-	99.0	92.8

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, En.: Entropy

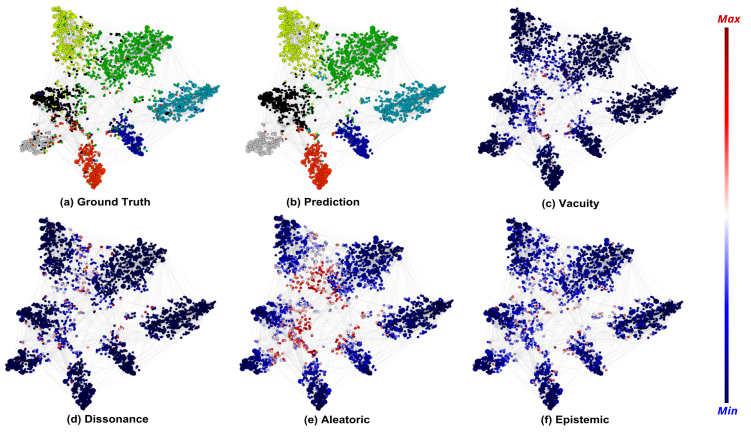


Figure 4: Graph embedding representations of the Cora dataset for classes and the extent of uncertainty: (a) shows the representation of seven different classes; (b) shows our model prediction; and (c)-(f) present the extent of uncertainty for respective uncertainty types, including vacuity, dissonance, aleatoric, epistemic.

Eight Classes on Amazon Photo Dataset: In Figure 6, a node's color denotes vacuity uncertainty value, and the big node represent training node. These results are based on OOD detection experiment. Compare Figure 6 (a) and (b), we found that GKDE can indeed improve the OOD detection.

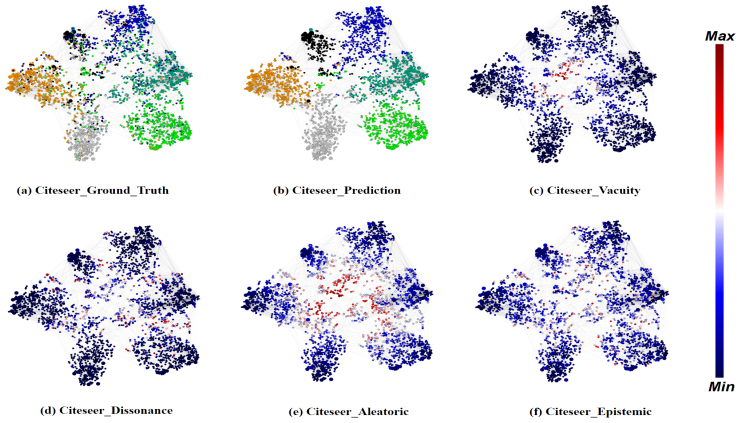


Figure 5: Graph embedding representations of the Citeseer dataset for classes and the extent of uncertainty: (a) shows the representation of seven different classes, (b) shows our model prediction and (c)-(f) present the extent of uncertainty for respective uncertainty types, including vacuity, dissonance, and aleatoric uncertainty, respectively.

For Figures 5 (c)-(f), the extent of uncertainty is presented where a blue color refers to the lowest uncertainty (i.e., minimum uncertainty) while a red color indicates the highest uncertainty (i.e., maximum uncertainty) based on the presented color bar. To examine the trends of the extent of uncertainty depending on either training nodes or test nodes, we draw training nodes as bigger circles than test nodes. Overall we notice that most training nodes (shown as bigger circles) have low uncertainty (i.e., blue), which is reasonable because the training nodes are the ones that are already observed. Now we discuss the extent of uncertainty under each uncertainty type.

Vacuity: In Figure 6 (b), most training nodes show low uncertainty, we observe majority of OOD nodes in the button cluster show high uncertainty as appeared in red.

Dissonance: In Figure 5 (d), similar to vacuity, training nodes have low uncertainty. But unlike vacuity, test nodes are much less uncertain. Recall that dissonance represents the degree of conflicting evidence (i.e., discrepancy between each class probability). However, in this dataset, we observe a fairly low level of dissonance and the obvious outperformance of Dissonance in node classification prediction.

Aleatoric uncertainty: In Figure 5 (e), a lot of nodes show high uncertainty with larger than 0.5 except a small amount of training nodes with low uncertainty.

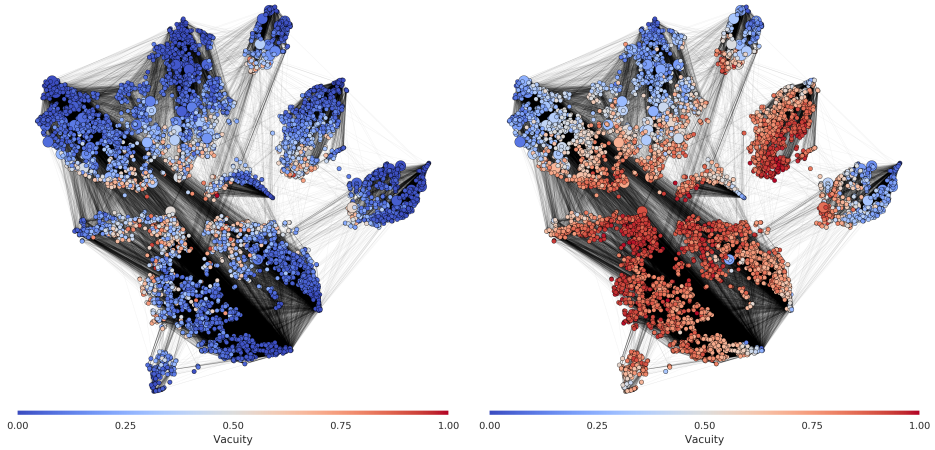
Epistemic uncertainty: In Figure 5 (f), most nodes show very low epistemic uncertainty values because uncertainty derived from model parameters can disappear as they are trained well.

C.5 PR and ROC Curves

AUPR for the OOD Detection: Figure 8 shows the AUPRC for the OOD detection when S-BGCN-T-K is used to detect OOD in which test nodes are considered based on their high uncertainty level, given a different uncertainty type, such as vacuity, dissonance, aleatoric, epistemic, or entropy (or total uncertainty). Also to check the performance of the proposed models with a baseline model, we added S-BGCN-T-K with test nodes randomly selected (i.e., Random).

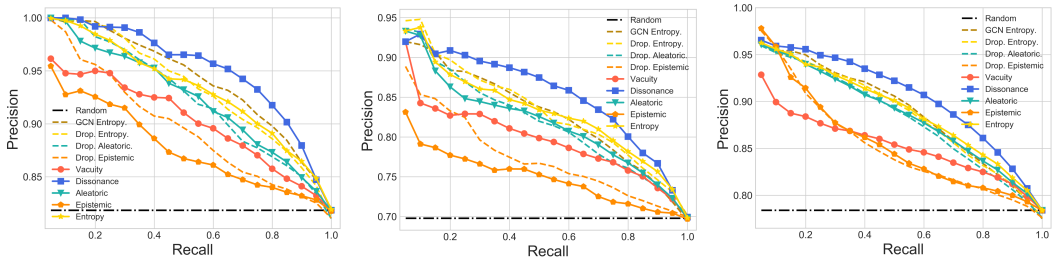
Obviously, in Random baseline, precision was not sensitive to increasing recall while in S-BGCN-T-K (with test nodes being selected based on high uncertainty) precision decreases as recall increases. But although most S-BGCN-T-K models with various uncertainty types used to select test nodes shows sensitive precision to increasing recall (i.e., proving uncertainty being an indicator of improving OOD detection). In addition, unlike AUPR in misclassification detection, which showed the best performance in S-BGCN-T-K Dissonance (see Figure 7), S-BGCN-T-K Dissonance showed the second worst performance among the proposed S-BGCN-T-K models with other uncertainty types. This means that less conflicting information does not help OOD detection. On the other hand, overall we observed Vacuity performs the best among all. From this finding, we can claim that to improve OOD detection, less information with a high vacuity value can help boost the accuracy of the OOD detection.

AUROC for the OOD Detection: First, we investigated the performance of our proposed S-BGCN-T-K models when test nodes are selected based on seven different criteria (i.e., uncertainty measures). For AUROC in Figure 9, we observed much better performance in most S-BGCN-T-K models with all uncertainty types except epistemic uncertainty.



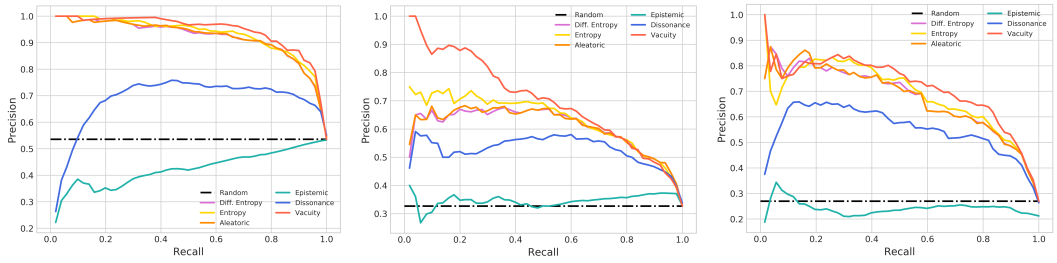
(a) S-BGCN-T (b) S-BGCN-T-K

Figure 6: Graph embedding representations of the Amazon Photo dataset for the extent of vacuity uncertainty based on OOD detection experiment.



(a) PR curves on Cora (b) PR curves on Citeseer (c) PR curves on Pubmed

Figure 7: PR curves of misclassification detection for S-BGCN-T-K and other baselines, GCN-Drop and GCN.



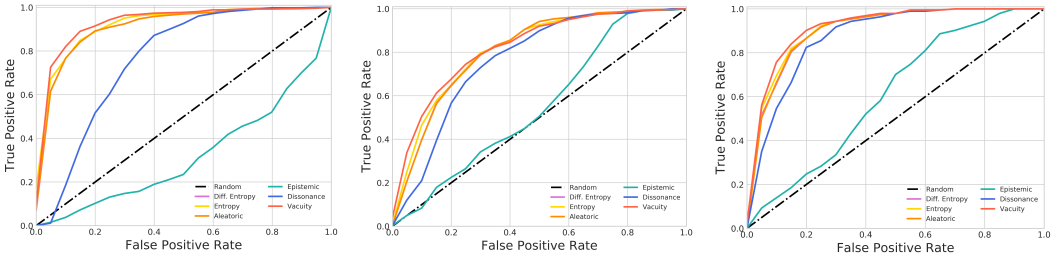
(a) Amazon Photo (b) Amazon Computers (c) Coauthor Physics

Figure 8: PR cuves of OOD detection for S-BGCN-T-K with uncertainties.

C.6 Analysis for Epistemic Uncertainty in OOD Detection

Although epistemic uncertainty is known to be effective to improve OOD detection [7, 14] in computer vision applications, our results demonstrate it is less effective than our vacuity-based approach. One potential reason is that the previous success of epistemic in computer vision applications are only applied in supervised learning, but they are not sufficiently validated in semi-supervised learning.

To back up our conclusion, designe a image classification experiment based on MC-Drop[7] method to do the following experiment: 1) supervised learning on MNIST dataset with 50 labeled images; 2) semi-supervised learning (SSL) on MNIST dataset with 50 labeled images and 49950 unlabeled images, while there are 50% OOD images (24975 FashionMNIST images) in unlabeled set. For both experiment, we test the epistemic uncertainty on 49950 unlabeled set (50% In-distribution (ID) images and 50% OOD images). We conduct the experiment the



(a) Amazon Photo (b) Amazon Computers (c) Coauthor Physics

Figure 9: ROC curves of OOD detection for S-BGCN-T-K with uncertainties.

experiment based on three popular SSL methods, VAT [22], Mean Teacher [29] and pseudo label [17]. Table 13

Table 13: Epistemic uncertainty for semi-supervised image classification.

Epistemic	**Supervised**	**VAT**	**Mean Teacher**	**Pseudo Label**
In-Distribution	0.140	**0.116**	**0.105**	**0.041**
Out-of-Distribution	**0.249**	0.049	0.076	0.020

shows the average epistemic uncertainty value for in-distribution samples and OOD samples. The result shows the same pattern with [14, 13] in a supervised setting, but an opposite pattern in a semi-supervised setting that low epistemic of OOD samples, which is less effective top detect OOD. Note that the SSL setting is similar to our semi-supervised node classification setting, which feed the unlabeled sample to train the model.

C.7 Compare with Bayesian GCN baseline

Compare with a (Bayesian) GCN baseline, Dropout+DropEdge [23]. As shown in the table 14 below, our proposed method performed better than Dropout+DropEdge on the Cora and Citeer datasets for misclassificaiton detection. A similar trend was observed for OOD detection.

Table 14: Compare with DropEdge on Misclassification Detection .

Dataset	Model	AUROC					AUPR				
		Va.	Dis.	Al.	Ep.	En.	Va.	Dis.	Al.	Ep.	En.
Cora	S-BGCN-T-K	70.6	**82.4**	75.3	68.8	77.7	90.3	**95.4**	92.4	87.8	93.4
	DropEdge	-	-	76.6	56.1	76.6	-	-	93.2	85.4	93.2
Citeseer	S-BGCN-T-K	65.4	**74.0**	67.2	60.7	70.0	79.8	**85.6**	82.2	75.2	83.5
	DropEdge	-	-	71.1	51.2	71.1	-	-	84.0	70.3	84.0

Va.: Vacuity, Dis.: Dissonance, Al.: Aleatoric, Ep.: Epistemic, En.: Entropy

D Derivations for Joint Probability and KL Divergence

D.1 Joint Probability

At the test stage, we infer the joint probability by:

$$
\begin{aligned}
p(\mathbf{y}|A,\mathbf{r};\mathcal{G}) &= \int\int \text{Prob}(\mathbf{y}|\mathbf{p})\text{Prob}(\mathbf{p}|A,\mathbf{r};\boldsymbol{\theta})\text{Prob}(\boldsymbol{\theta}|\mathcal{G})d\mathbf{p}d\boldsymbol{\theta} \\
&\approx \int\int \text{Prob}(\mathbf{y}|\mathbf{p})\text{Prob}(\mathbf{p}|A,\mathbf{r};\boldsymbol{\theta})q(\boldsymbol{\theta})d\mathbf{p}d\theta \\
&\approx \frac{1}{M}\sum_{m=1}^{M}\int \text{Prob}(\mathbf{y}|\mathbf{p})\text{Prob}(\mathbf{p}|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \\
&\approx \frac{1}{M}\sum_{m=1}^{M}\int \sum_{i=1}^{N}\text{Prob}(\mathbf{y}_i|\mathbf{p}_i)\text{Prob}(\mathbf{p}_i|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}_i, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \\
&\approx \frac{1}{M}\sum_{m=1}^{M}\sum_{i=1}^{N}\int \text{Prob}(\mathbf{y}_i|\mathbf{p}_i)\text{Prob}(\mathbf{p}_i|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}_i, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \\
&\approx \frac{1}{M}\sum_{m=1}^{M}\prod_{i=1}^{N}\int \text{Prob}(\mathbf{y}_i|\mathbf{p}_i)\text{Dir}(\mathbf{p}_i|\boldsymbol{\alpha}_i^{(m)})d\mathbf{p}_i, \quad \boldsymbol{\alpha}^{(m)} = f(A,\mathbf{r},\boldsymbol{\theta}^{(m)}), q \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}),
\end{aligned}
$$

where the posterior over class label p will be given by the mean of the Dirichlet:

$$
\text{Prob}(y_i = p|\boldsymbol{\theta}^{(m)}) = \int \text{Prob}(y_i = p|\mathbf{p}_i)\text{Prob}(\mathbf{p}_i|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}_i = \frac{\alpha_{ip}^{(m)}}{\sum_{k=1}^{K}\alpha_{ik}^{(m)}}.
$$

The probabilistic form for a specific node i by using marginal probability,

$$
\begin{aligned}
\text{Prob}(\mathbf{y}_i|A,\mathbf{r};\mathcal{G}) &= \sum_{y\setminus y_i}\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G}) \\
&= \sum_{y\setminus y_i}\int\int\prod_{j=1}^{N}\text{Prob}(\mathbf{y}_j|\mathbf{p}_j)\text{Prob}(\mathbf{p}_j|A,\mathbf{r};\boldsymbol{\theta})\text{Prob}(\boldsymbol{\theta}|\mathcal{G})d\mathbf{p}d\boldsymbol{\theta} \\
&\approx \sum_{y\setminus y_i}\int\int\prod_{j=1}^{N}\text{Prob}(\mathbf{y}_j|\mathbf{p}_j)\text{Prob}(\mathbf{p}_j|A,\mathbf{r};\boldsymbol{\theta})q(\boldsymbol{\theta})d\mathbf{p}d\boldsymbol{\theta} \\
&\approx \sum_{m=1}^{M}\sum_{y\setminus y_i}\int\prod_{j=1}^{N}\text{Prob}(\mathbf{y}_j|\mathbf{p}_j)\text{Prob}(\mathbf{p}_j|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \\
&\approx \sum_{m=1}^{M}\Big[\sum_{y\setminus y_i}\int\prod_{j=1}^{N}\text{Prob}(\mathbf{y}_j|\mathbf{p}_j)\text{Prob}(\mathbf{p}_j|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}_j\Big], \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \\
&\approx \sum_{m=1}^{M}\Big[\sum_{y\setminus y_i}\prod_{j=1,j\neq i}^{N}\text{Prob}(\mathbf{y}_j|A,\mathbf{r}_j;\boldsymbol{\theta}^{(m)})\Big]\text{Prob}(\mathbf{y}_i|A,\mathbf{r};\boldsymbol{\theta}^{(m)}), \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}) \\
&\approx \sum_{m=1}^{M}\int\text{Prob}(\mathbf{y}_i|\mathbf{p}_i)\text{Prob}(\mathbf{p}_i|A,\mathbf{r};\boldsymbol{\theta}^{(m)})d\mathbf{p}_i, \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}).
\end{aligned}
$$

To be specific, the probability of label p is,

$$
\text{Prob}(y_i = p|A,\mathbf{r};\mathcal{G}) \approx \frac{1}{M}\sum_{m=1}^{M}\frac{\alpha_{ip}^{(m)}}{\sum_{k=1}^{K}\alpha_{ik}^{(m)}}, \quad \boldsymbol{\alpha}^{(m)} = f(A,\mathbf{r},\boldsymbol{\theta}^{(m)}), \quad \boldsymbol{\theta}^{(m)} \sim q(\boldsymbol{\theta}).
$$

D.2 KL-Divergence

KL-divergence between $\text{Prob}(\mathbf{y}|\mathbf{r};\boldsymbol{\gamma},\mathcal{G})$ and $\text{Prob}(\mathbf{y}|\hat{\mathbf{p}})$ is given by

$$\begin{aligned}
\text{KL}[\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G})||\text{Prob}(\mathbf{y}|\hat{\mathbf{p}}))] &= \mathbb{E}_{\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G})}\Big[\log\frac{\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G})}{\text{Prob}(\mathbf{y}|\hat{\mathbf{p}})}\Big] \\
&\approx \mathbb{E}_{\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G})}\Big[\log\frac{\prod_{i=1}^{N}\text{Prob}(\mathbf{y}_i|A,\mathbf{r};\mathcal{G})}{\prod_{i=1}^{N}\text{Prob}(\mathbf{y}|\hat{\mathbf{p}})}\Big] \\
&\approx \mathbb{E}_{\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G})}\Big[\sum_{i=1}^{N}\log\frac{\text{Prob}(\mathbf{y}_i|A,\mathbf{r};\mathcal{G})}{\text{Prob}(\mathbf{y}|\hat{\mathbf{p}})}\Big] \\
&\approx \sum_{i=1}^{N}\mathbb{E}_{\text{Prob}(\mathbf{y}|A,\mathbf{r};\mathcal{G})}\Big[\log\frac{\text{Prob}(\mathbf{y}_i|A,\mathbf{r};\mathcal{G})}{\text{Prob}(\mathbf{y}|\hat{\mathbf{p}})}\Big] \\
&\approx \sum_{i=1}^{N}\sum_{j=1}^{K}\text{Prob}(y_i=j|A,\mathbf{r};\mathcal{G})\Big(\log\frac{\text{Prob}(y_i=j|A,\mathbf{r};\mathcal{G})}{\text{Prob}(y_i=j|\hat{\mathbf{p}})}\Big)
\end{aligned}$$

The KL divergence between two Dirichlet distributions $\text{Dir}(\alpha)$ and $\text{Dir}(\hat{\alpha})$ can be obtained in closed form as,

$$\text{KL}[\text{Dir}(\alpha)\|\text{Dir}(\hat{\alpha})] = \ln\Gamma(S) - \ln\Gamma(\hat{S}) + \sum_{c=1}^{K}\big(\ln\Gamma(\hat{\alpha}_c) - \ln\Gamma(\alpha_c)\big) + \sum_{c=1}^{K}(\alpha_c-\hat{\alpha}_c)(\psi(\alpha_c)-\psi(S)),$$

where $S=\sum_{c=1}^{K}\alpha_c$ and $\hat{S}=\sum_{c=1}^{K}\hat{\alpha}_c$.