Learning Robust Hypergraph Embeddings for Distribution-Free Uncertainty Quantification

Akash Choudhuri
University of Iowa
Department of Computer Science
Iowa City, IA, USA
akash-choudhuri@uiowa.edu

Bijaya Adhikari
University of Iowa
Department of Computer Science
Iowa City, IA, USA
bijaya-adhikari@uiowa.edu

Abstract

Hypergraph representation learning has gained immense popularity over the last few years due to its applications in real-world domains like social network analysis, recommendation systems, biological network modeling, and knowledge graphs. However, hypergraph neural networks (HGNNs) lack rigorous uncertainty estimates, which limits their deployment in critical applications where the reliability of predictions is crucial. To bridge this gap, we propose Contrastive Conformal HGNN (CCF-HGNN) that accounts for uncertainty in hypergraph-based models by explicitly regularizing on the hypergraph structure for guaranteed and robust uncertainty estimates. CCF-HGNN accounts for epistemic uncertainty in HGNN predictions by producing a prediction set that leverages the topological structure and provably contains the true label with a pre-defined coverage probability. It also accounts for aleatoric uncertainty by leveraging contrastive learning on the structure of the hypergraph. To enhance the power of the predictions, CCF-HGNN performs an additional auxiliary task of hyperedge degree prediction with an end-to-end differentiable sampling-based approach. Extensive experiments on real-world hypergraph datasets demonstrate the superiority of CCF-HGNN by improving the efficiency of prediction sets while maintaining valid coverage.

CCS Concepts

• **Computing methodologies** → **Neural networks**.

Keywords

Hypergraph Neural Networks, Uncertainty Quantification, Contrastive Learning

ACM Reference Format:
Akash Choudhuri and Bijaya Adhikari. 2026. Learning Robust Hypergraph Embeddings for Distribution-Free Uncertainty Quantification. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '26), August 09–13, 2026, Jeju Island, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3770855.3818095

Resource Availability:
The source code of this paper has been made publicly available at https://doi.org/10.5281/zenodo.20434513.

KDD '26, Jeju Island, Republic of Korea

ACM ISBN 979-8-4007-2259-2/2026/08
https://doi.org/10.1145/3770855.3818095

1 Introduction

Network-structured data underpins a broad spectrum of scientific and real-world applications, ranging from social interactions [38] and recommender systems [66] to biological networks [72] and knowledge graphs [39]. This has fueled the rapid growth of graph-based machine learning, where graph neural networks (GNNs) have emerged as a dominant paradigm for learning from relational data [24, 29, 55]. More recently, attention has shifted towards *hypergraph representation learning*, which extends beyond pairwise relations to model higher-order interactions, thereby offering a more faithful abstraction for many complex systems [8, 73]. The expressive power of hypergraphs has led to applications across diverse domains, including healthcare (e.g., multiple patients sharing a room) [12, 63, 64], social networks (e.g., users joining groups or channels) [32], bioinformatics [49], and cyber-security [35]. To exploit these structures, hypergraph neural networks (HGNNs) have been developed with specialized message-passing and aggregation mechanisms [5, 13, 19, 65], demonstrating superior performance when group-wise relations, rather than dyadic links, are essential.

The evolution of HGNNs has closely paralleled that of GNNs. Early work, such as HGNN [19], adapted the message-passing framework of GCN [29], while HCHA [5] extended the attention mechanism of GAT [55] to hypergraphs. More recent efforts have introduced advanced ideas, including multiset functions [10], network diffusion [57], energy-based formulations [60], and implicit modeling [13, 33]. Despite these innovations, a key limitation persists: *existing HGNNs provide no mechanism to quantify predictive uncertainty.* This omission is particularly problematic in high-stakes domains, where decisions require not only accuracy but also calibrated confidence.

Table 1: Comparison of the features of prior works. Note that the bands provided by conformal prediction methods have guaranteed coverage and do not hinge on any prior assumption about the distribution of the data. Our method provides robust UQ by accounting for structural aleatoric uncertainty.

Method	Coverage	Distribution Free	Robust
TS and VS [23]	✗	✗	✓
ETS [71]	✗	✗	✓
CF-GNN [26]	✓	✓	✗
Ours	✓	✓	✓

A principled solution is to construct *prediction sets* that guarantee high-probability coverage for each sample. While numerous uncertainty quantification methods have been proposed in the broader

machine learning literature [21, 23, 25, 31, 51, 70, 71], they generally lack frequentist verifiable marginal coverage guarantees (provable, verifiable, distribution-free marginal coverage), which limits their deployment in high-stakes deployment. The field of conformal prediction, pioneered by [56], provides a principled framework for constructing prediction sets with rigorous, finite-sample coverage guarantees under minimal distributional assumptions. By calibrating nonconformity scores on held-out data, conformal prediction methods ensure that the true label is included in the prediction set with a user-specified probability (e.g., $1-\alpha$), regardless of the underlying data distribution. This property has fueled widespread adoption in areas such as computer vision [4], natural language processing [30], and time-series forecasting [22, 45, 67]. Conformal prediction has gained immense popularity in graph representation learning, with frameworks aimed at quantifying uncertainty in inductive [15, 69] and transductive [26, 68] node classification and edge/link prediction [14, 36, 74].

While conformal prediction guarantees safety with uncertainty quantification, it does not guarantee *utility*. When a model faces high aleatoric uncertainty, such as structural noise in a hypergraph or ambiguous node features, standard conformal prediction methods maintain their validity guarantee by simply increasing the size of the prediction sets. If the underlying model is sensitive to noisy hyperedges, conformal prediction may be forced to output trivial prediction sets (containing nearly all possible classes) to satisfy the coverage requirement. For instance, in a medical diagnosis hypergraph (where hyperedges represent symptoms shared by patient groups), a valid but trivial prediction set might include every possible disease. While statistically 'safe' (the true disease is in the set), such a prediction is operationally useless for a practitioner. Thus, the challenge in hypergraph uncertainty quantification is not just to provide valid intervals, but to provide efficient (informative and tight) intervals in the presence of structural noise. Additionally, HGNNs often suffer from over-smoothing [13], where the addition of more layers makes the embeddings of connected nodes similar, making nodes belonging to different classes indistinguishable, making the model uncertain about all of them. Traditional CP reacts to this by outputting massive prediction sets. We propose that to reduce the size of prediction sets without sacrificing coverage, the encoder must learn representations that are invariant to these structural perturbations. By enforcing consistency across augmented views of the hypergraph, the model can filter out the aleatoric noise that typically inflates the conformal scores. Table 1 illustrates the differences between the prior works and highlights the advantages of our proposed method.

To address these challenges, we introduce *Contrastive Conformal Hypergraph Neural Network* **(CCF-HGNN)**, an end-to-end framework that jointly models predictive uncertainty in hypergraph representation learning. Our primary contributions are as follows:

- To the best of our knowledge, this is the first work that combines contrastive augmentation- aided learning with conformal prediction for any network-structured data.
- We additionally propose an efficient computational method to sample the important hyperedges based on the augmentation strategy and perform the hyperedge-degree prediction task only on those hyperedges. Predicting hyperedge degree forces the model to encode the cardinality of high-order interactions, which is the specific signal often lost during standard message passing.
- We provide theoretical evidence that guarantees that the joint modeling contrastive learning and conformal prediction is both efficient (ie, the predictive bands returned are shorter) and effective (empirical coverage provably exceeds the given confidence level).
- Extensive experiments on several real-world hypergraph datasets for uncertainty quantification in the node classification task demonstrate the overall utility of our method.

2 Preliminaries

Let $H = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \mathcal{Y})$ be a hypergraph, where $\mathcal{V}$ is a set of nodes, $\mathcal{E}$ is a set of hyperedges, and $\mathcal{X} = \{\mathbf{x}_v\}_{v\in\mathcal{V}}$ is the set of node attributes, where $\mathbf{x}_v \in \mathbb{R}^d$ is a d-dimensional feature vector for node $v \in \mathcal{V}$. Let $\mathcal{Y} = \{y_v\}_{v\in\mathcal{V}}$ be the set of node labels. Our paper focuses on classification problems, but our theory and method naturally extend to regression problems. To perform point predictions, we are given a mean estimator $\hat{\mu}$ that predicts the node label $\hat{y}_v$ given the node embedding x_v.

2.1 Transductive Setting

We focus on the transductive node classification problem with a random data split akin to [26]. In this setting, we partition the node labels into three disjoint sets: $\mathcal{Y}_{\text{train}}$, $\mathcal{Y}_{\text{cal}}$, and $\mathcal{Y}_{\text{test}}$. This leads to the corresponding *training* data $D_{\text{train}} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \mathcal{Y}_{\text{train}})$, *calibration* data $D_{\text{cal}} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \mathcal{Y}_{\text{cal}})$, and *testing* data $D_{\text{test}} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \mathcal{Y}_{\text{test}})$. In particular, during training, the model can access $\mathcal{V}, \mathcal{E}, \mathcal{X}$, but only the training labels $\mathcal{Y}_{\text{train}}$ are revealed to the model. Abusing the notation, we use $\mathcal{V}_{\text{train}}$ to denote elements of $\mathcal{V}$ for which the node labels are in $\mathcal{Y}_{\text{train}}$. We follow the same notation throughout the paper. After training, the calibration data $\{y_v\}_{v\in\mathcal{V}_{\text{cal}}}$ is used to construct uncertainty estimates. Finally, we predict the uncertainty bands for the remaining nodes (i.e., $\mathcal{V}_{\text{test}}$).

2.2 Mean Estimator: Hypergraph Neural Network

Hypergraph Neural Networks (HGNNs) are powerful machine learning models that leverage the high-order network structure during message passing. Unlike traditional graph neural networks that only aggregate pairwise information, HGNNs can handle the complexity of hypergraphs, where relationships between nodes are generalized beyond pairwise connections. Like Graph Neural Networks (GNNs), HGNNs aggregate neighborhood information [5, 19] via a sequence of propagation layers where each layer consists of a Message Passing Step, and a Node Update Step. Further details about the propagation steps are provided in the Appendix A.

2.3 Conformal Prediction

In this work, we focus on split conformal prediction [56], which proceeds in four primary steps. Given a miscoverage rate $\alpha \in [0, 1]$, the steps are: **(1) Training:** Train the mean estimator $\hat{\mu}$ on the training data D_{train}. **(2) Calibration:** For each node v in $\mathcal{V}_{\text{cal}}$, compute the non-conformity scores (heuristic notion of how off the prediction is from the true label) $\{V(\mathbf{x}_v, y_v)\}_{v\in\mathcal{V}_{\text{cal}}}$ and create an

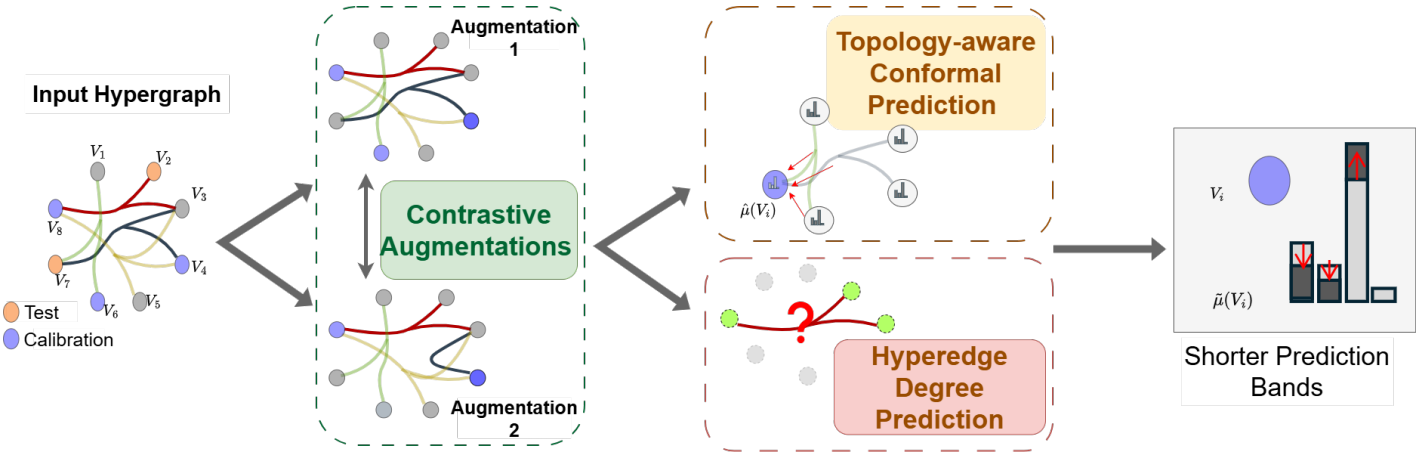


Figure 1: Contrastive Conformal Hypergraph Neural Network: The overall framework minimizes three losses: 1) Contrastive Loss: Structural alterations generate multiple views of the hypergraph, encouraging the model to learn invariant representations. 2) Conformal Inefficiency Loss: Topology-aware conformal loss ensures similarity in uncertainties of a node based on its local neighbors (nodes that share hyperedges). 3) Degree Loss: Predicting the hyperedge degree of a sample of hyperedges to guide the model to learn the structure. This leads to shorter and more confident prediction bands.

empirical distribution from the scores. **(3) Quantile Computation:** Compute the $(1-\alpha)^{\text{th}}$ quantile $\hat{Q}_{1-\alpha}$ of the distribution $\frac{1}{|\mathcal{V}_{cal}|+1}\sum_{v\in\mathcal{V}_{cal}}\delta_{V_v}+\delta_\infty$, where δ_a is Dirac Delta distribution at point a, and V_v is shorthand for $V(\mathbf{x}_v, y_v)$. **(4) Band Computation:** Given a test node v and corresponding feature $\mathbf{x}_v$, a prediction set/interval $\hat{C}(\mathbf{x}_v) = \{y \in \mathcal{Y} : V(\mathbf{x}_v, y) \leq \hat{Q}_{1-\alpha}\}$ is constructed. The notion of transferring the prediction bands computed on the calibration data to the points in test data relies on the following permutation invariance assumption [26, 69].

Assumption 1. For any permutation π on the calibration and test nodes, the non-conformity score V obeys

$$V(\mathbf{x}_v, y_v; \{y_a\}_{a\in\mathcal{V}_{\text{train}\cup\text{cal}}}, \mathcal{X}, \mathcal{V}, \mathcal{E}) = V(\mathbf{x}_v, y_v; \{y_a\}_{a\in\mathcal{V}_{\text{train}\cup\text{cal}}}, \mathcal{X}, \mathcal{V}_\pi, \mathcal{E})$$

This means that the non-conformity scores of nodes in a hypergraph H are exchangeable.

Assumption 1 imposes the permutation invariance condition for the HGNN training to later compute the non-conformity scores for node prediction, which means that the model output/non-conformity score is invariant to permuting the order of the calibration and test nodes on the hypergraph. HGNNs do not rely on the ordering of the nodes, hence they typically satisfy the assumption.

Lemma 2. *(Coverage Guarantee for Conformal Inference) [50, 56] Under Assumption 1, for any $\alpha > 0$, the confidence band returned by the conformal inference algorithm satisfies:*

$$\mathbb{P}(y_v \in \hat{C}_{1-\alpha}(\mathbf{x}_v)) \geq 1-\alpha \tag{1}$$

where the probability is taken over the calibration fold D_{cal} and the testing point $(\mathbf{x}_v, y_v)$.

Here, $\mathbb{P}(y_v \in \hat{C}_{1-\alpha}(\mathbf{x}_v))$ denotes the **coverage**, i.e., the probability that the true label y_v lies in the predictive band.

3 Our Method

In this section, we propose our method, Contrastive Conformal Hypergraph Neural Network (CCF-HGNN), which aims to reduce the size of the predictive band length while maintaining coverage for hypergraph neural networks. The main idea is to boost the APS and RAPS scores (see section 4.1) with the help of local topological information and account for data-noise in the form of contrastive augmentations.

3.1 Computing Differentiable Inefficiency Loss

Instead of using pairwise local topological information as done by [26], our work uses high-order local topological information that goes beyond homophily or other aggregation mechanisms (like mean, sum, etc.). To implement this idea, we use a separate HGNN learner $\tilde{\mu}$ parameterized by the weights ϑ for the same hypergraph network H with node features initialized by $\hat{\mu}(\mathcal{X})$. Here $\hat{\mu}(\cdot)$ denotes the mean estimator that has been used during the training process. Given $\tilde{\mu}(\mathcal{X}) = \text{HGNN}_\vartheta(\hat{\mu}(\mathcal{X}), H)$, and a target miscoverage rate α, we partition the calibration data D_{cal} into $D_{\text{corr-cal}}$ (correction subset) and $D_{\text{cal-test}}$ (testing subset) compute a differentiable loss in the following steps: 1) **Differentiable Quantile Computation:** Compute the smooth differentiable quantile $\hat{\eta} = \text{DiffQuantile}(\{V(\mathbf{x}_i, y_i) \mid i \in \mathcal{D}_{\text{corr-cal}}\})$ on $D_{\text{corr-cal}}$. 2) **Inefficiency Proxies Computation:** Construct a differentiable proxy of the miscoverage on $D_{\text{cal-test}}$ by using $D_{\text{corr-cal}}$ as calibration data. For class k and node i in $D_{\text{cal-test}}$, the non-conformity score is given as $V(\mathbf{x}_i, k)$ (as per APS and RAPS scores). The inefficiency proxy will thus be $c_i = \sigma\left(\frac{V(\mathbf{x}_i,k)-\hat{\eta}}{\tau_1}\right)$, where $\sigma(\cdot)$ denotes the sigmoid function and τ_1 denotes the temperature hyperparameter [46]. Note that τ_1 controls the "hardness" of the approximation. A lower temperature makes the sigmoid approach a step function, while a higher temperature smooths the gradients, facilitating more stable optimization during the early training phases. 3) **Overall Loss Computation:**

Compute the overall inefficiency loss as an average of the inefficiency proxies $\mathcal{L}_{\text{Ineff}} = \frac{1}{m} \sum_{i \in \mathcal{D}_{\text{cal-test}}} \frac{1}{|\mathcal{Y}|} \sum_{k \in \mathcal{Y}} c_i$.

The proof that the inefficiency loss is exchangeable follows the proof of the same theorem in [26], since our setup operates in the transductive setting and hypergraphs can be represented as graphs via clique/star expansions [1]. Note that while the number of edges changes due to these expansions, the number of nodes remains the same, which is why the proof holds. Additionally, the clique expansion is deterministic and symmetric. Thus, permuting the input nodes results in an equivalent permutation of the expanded graph, preserving the exchangeability required for conformal validity.

3.2 Using Contrastive Augmentations

Minimizing the inefficiency loss in isolation exposes the framework to overfitting on the often noisy topology of the input hypergraph. Real-world hypergraphs frequently contain spurious connections or miss critical high-order links. Since HGNNs rely heavily on recursive aggregation across these structures, they are highly sensitive to such topological noise. If the UQ model overfits to such structural noise, the resulting node embeddings would capture them, propagating to unstable non-conformity scores and artificially inflated prediction sets on unseen data, which would lead to inferior performance in downstream tasks. This motivates us to regularize the learning process by enforcing *structural invariance*, ensuring that the predictive bands remain tight and stable even in the presence of topological perturbations.

To execute this motivation, we utilize contrastive augmentations to boost the power of node embeddings in a self-supervised manner [61]. We design contrastive structural augmentations by constructing augmentations $\mathcal{H}_1 = \hat{f}(H, A_1)$ $\mathcal{H}_2 = \hat{f}(H, A_2)$ and corresponding node embeddings where $\hat{f}(\cdot, \cdot)$ is a function that perturbs the structure of a hypergraph given a perturbation schema A. Hence, A_1 and A_2 are two instantiations of the perturbation schema. Finally, we can obtain the node embeddings of the augmented hypergraphs as $Z^1 = \tilde{\mu}(\mathcal{H}_1, X)$ $Z^2 = \tilde{\mu}(\mathcal{H}_2, X)$ and minimizing the contrastive loss as follows:

$$\mathcal{L}_{\text{Contra}} = \text{InfoNCE}(Z^1, Z^2, \tau_2) = -\sum_{i=1}^{|\mathcal{V}|} \log \frac{\exp\left(\frac{\text{sim}(\mathbf{z}_i^1, \mathbf{z}_i^2)}{\tau_2}\right)}{\sum_{j=1}^{|\mathcal{V}|} \exp\left(\frac{\text{sim}(\mathbf{z}_i^1, \mathbf{z}_j^2)}{\tau_2}\right)}, \tag{2}$$

Here τ_2 is a temperature hyperparameter to the popular InfoNCE loss [9] and $\text{sim}(\cdot)$ denotes a similarity function like cosine similarity. The contrastive loss is also exchangeable as the loss depends on the embeddings, which are thus dependent on the mean estimator (HGNN in this case). As HGNN is permutation invariant, the contrastive loss is also exchangeable.

3.3 Boosting Contrastive Augmentation with Auxiliary Hyperedge Degree Prediction

To appropriately guide the calibration model $\tilde{\mu}(\cdot)$ with the structure of the hypergraph by addressing cardinality erasure under HGNN aggregation, where nodes in large hyperedges should be more uncertain, we propose jointly training the hypergraph augmentations with the task of predicting the original hyperedge degrees. However, as the number of hyperedges in real-world hypergraphs is much greater than the number of nodes, we propose an efficient augmentation strategy to sample the most important hyperedges to perform the auxiliary hyperedge degree prediction task.

Let the hyperedge-Laplacian matrix of the hypergraph be $\boldsymbol{L} \in \mathbb{R}^{m \times n}$, where $m = |\mathcal{E}|$ is the number of hyperedges and $n = |\mathcal{V}|$ is the number of nodes. The hyperedge-Laplacian can be computed as $\boldsymbol{L} = \mathbf{D}_e^{-\frac{1}{2}} \mathbf{H}^T \mathbf{D}_v^{-\frac{1}{2}}$ [19], where $\mathbf{H}$ denotes the incidence matrix. We apply self-attention mechanism [54] over the hyperedge Laplacian to get attention weights a_j=Self-Attention($\boldsymbol{L}_{:,j}$) for each hyperedge index j. The self-attention mechanism on the hyperedge Laplacian allows the model to capture global dependencies between hyperedges (eg, identifying hyperedges that frequently co-occur or share bridge nodes), which is crucial for identifying which hyperedges are informative to guide the downstream predictive task.

To sample the k most important hyperedges in a fully differentiable manner, we use the Gumbel-Softmax trick [28] as $\boldsymbol{s} = \text{GumbelSoftmax}(\boldsymbol{a}, k, \tau_3)$, where $\boldsymbol{s} \in \mathbb{R}^n$ is a soft selection mask, k is the desired number of hyperedges, and τ_3 is the temperature parameter. Using the Gumbel-Softmax trick allows the gradients to flow from the degree prediction loss back into the attention weights and the calibration learner. The auxiliary hyperedge degree prediction task is then $\hat{d}_j = h(\boldsymbol{L}_{:,j})$, where $h(\cdot)$ is a learnable predictor and $\hat{d}_j$ is the predicted degree of hyperedge e_j. Given the true degree d_j for the hyperedge in the augmented hypergraph, the loss for the degree prediction task is

$$\mathcal{L}_{\text{deg}} = \sum_{j=1}^{n} s_j \cdot \ell(\hat{d}_j, d_j) \tag{3}$$

where $\ell(\cdot, \cdot)$ is a regression loss, e.g., mean squared error. The degree prediction loss is also exchangeable as it does not relate to node labels in the transductive setting.

The overall training algorithm of our method is given in Algorithm 1. This joint training encourages the model to learn representations sensitive to the structure of the most informative hyperedges while maintaining differentiability for end-to-end optimization.

3.4 Theoretical Guarantee

This section provides theoretical guarantees for our proposed method, in terms of shorter uncertainty band length (compared to the naive extension of the graph counterpart [26] to hypergraphs). We will first define some notations that form the foundation of our theoretical results.

Notations: Assume an encoder-decoder architecture of the conformal corrector $\tilde{\mu}(\cdot)$, where the encoder maps the input node features to latent embeddings and the decoder maps those embeddings to predictions. Consider two models: (1) **CF-HGNN:** $\mathbf{Z}_0 = h_0(X)$ and $\hat{Y} = g_0(\mathbf{Z}_0)$ where $h_0(\cdot)$ and $g_0(\cdot)$ is the encoder and decoder, and $\mathbf{Z}_0$ is the latent representation. Its prediction set has expected size $C_0(\mathbf{x})$ given the node embedding $\mathbf{x}$. This is the naive extension of [26] to hypergraphs. (2) **CCF-HGNN:** $\mathbf{Z}_1 = h_1(\mathcal{X}, A)$ and $\hat{Y}_1 = g_1(\mathbf{Z}_1)$ where $h_1(\cdot)$ and $g_1(\cdot)$ is encoder and decoder, and $\mathbf{Z}_1$ is the latent representation under contrastive augmentation A. Its prediction set has expected size $C_1(\mathbf{x})$ given the node embedding $\mathbf{x}$. Recall, this is our proposed approach.

Table 2: Statistics of the selected datasets. Here, DBLP is a homophilic dataset while the others are heterophilic.

Property	DBLP	Congress	House-Bills	Walmart-Trips	High-School	Trivago-Clicks
# nodes	41,302	1,718	1,494	88,860	327	170,994
# hyperedges	22,363	83,105	60,987	69,906	7818	232,013
# classes	6	2	2	11	9	80
avg. $\lvert e\rvert$	4.452	8.656	20.500	6.589	2.300	3.116

Algorithm 1 Contrastive Hypergraph Conformal Prediction (CCF-HGNN)

Input: Hypergraph $H = \{\mathcal{V}, \mathcal{E}\}$, feature matrix $\boldsymbol{X}$, label set $\mathcal{Y}$, Incidence Matrix H
HGNN train Model $\hat{\mu}(\cdot)$, calibration model $\tilde{\mu}(\cdot)$ with weights ϑ, non-conformity score function
$V(\cdot,\cdot)$, Calibration dataset $\mathcal{D}_{\text{cal}}$ partitioned into $\mathcal{D}_{\text{corr-cal}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_{\text{corr-cal}}}$,
and $\mathcal{D}_{\text{cal-test}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n_{\text{cal-test}}}$, significance level α, Hypergraph incidence, node and hyperedge degree matrices $\mathbf{H}, \mathbf{D}_e, \mathbf{D}_v$

Train HGNN model $\hat{\mu}(H, \boldsymbol{X})$ on prediction task.
while Not converged **do**
  Obtain augmentations $\mathcal{H}_1$ and $\mathcal{H}_2$ of H.
  Compute the hyperedge Laplacians $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$, corresponding attention weights $\boldsymbol{a}^1$ and $\boldsymbol{a}^2$.
  Select the k important hyperedges using the Gumbel-Softmax trick.
  Compute the overall degree loss $\mathcal{L}_{\text{deg}} = \mathcal{L}^1_{\text{deg}} + \mathcal{L}^2_{\text{deg}}$.
  Get embeddings $\mathbf{Z}^1 = \tilde{\mu}(\mathcal{H}_1, f(\boldsymbol{X}))$, $\mathbf{Z}^2 = \tilde{\mu}(\mathcal{H}_2, f(\boldsymbol{X}))$.
  Get calibration predictions $\mathbf{Z}^1_{\text{cal}}, \mathbf{Z}^2_{\text{cal}}$ from $\mathbf{Z}^1, \mathbf{Z}^2$.
  Compute $\mathbf{Z}_{\text{cal}} = \frac{\mathbf{Z}^1_{\text{cal}} + \mathbf{Z}^2_{\text{cal}}}{2}$.
  Get test predictions $\mathbf{Z}^1_{\text{test}}, \mathbf{Z}^2_{\text{test}}$ from $\mathbf{Z}^1, \mathbf{Z}^2$.
  Compute $\mathbf{Z}_{\text{test}} = \frac{\mathbf{Z}^1_{\text{test}} + \mathbf{Z}^2_{\text{test}}}{2}$.
  Compute $\hat{\alpha} = \frac{1}{n+1} \cdot \alpha$.
  $\hat{\eta} = \text{DiffQuantile}(\{V(\mathbf{Z}_i, y_i) \mid i \in \mathcal{D}_{\text{cal}}\})$.
  $\mathcal{L}_{\text{Ineff}} = \frac{1}{m} \sum_{i \in \mathcal{D}_{\text{cal-test}}} \frac{1}{|\mathcal{Y}|} \sum_{k \in \mathcal{Y}} \sigma\left(\frac{V(\mathbf{z}_i, k) - \hat{\eta}}{\tau_1}\right)$.
  $\mathcal{L}_{\text{Contra}} = \text{INFONCE}(\mathbf{Z}^1, \mathbf{Z}^2, \tau_2)$
  $\mathcal{L}_{\text{Total}} = \gamma \mathcal{L}_{\text{Ineff}} + (1-\gamma)\mathcal{L}_{\text{Contra}} + \mathcal{L}_{\text{deg}}$.
  $\vartheta = \vartheta - \nabla_\vartheta \mathcal{L}_{\text{Total}}$.
end while

Proposition 3. *Let $I(Y;\mathbf{Z}_1)$ and $I(Y;\mathbf{Z}_0)$ denote the mutual information between the labels and latent embeddings for CCF-HGNN and CF-HGNN, respectively, and $\Delta \in \mathbf{R}^+$ then,*

$$I(Y;\mathbf{Z}_1) \geq I(Y;\mathbf{Z}_0) + \Delta. \tag{4}$$

The proof is provided in Appendix C. Using the results from Lemma 2, we can prove the following theorem on the expected band length produced by CCF-HGNN and CF-HGNN.

Theorem 4. *Under the assumptions:*

(1) ***Bounded coverage:*** *Contrastive augmentations do not reduce conformal coverage (marginal coverage $\geq 1-\alpha$ is preserved on average).*
(2) ***Large Mutual Information gap:*** *$I(Y;\mathbf{Z}_1) - I(Y;\mathbf{Z}_0)$ is sufficiently large (Lemma 6).*

Then, the expected conformal prediction set size under CCF-HGNN is smaller than under CF-HGNN:

$$\mathbb{E}\big[|C_1(\mathbf{x})|\big] \leq \mathbb{E}\big[|C_0(\mathbf{x})|\big]. \tag{5}$$

The proof is provided in Appendix D. Note that the assumption of bounded coverage is reasonable because conformal calibration is performed after or concurrently with representation learning. As long as the calibration set is exchangeable with the test set, the marginal coverage property holds regardless of the quality of the embeddings, provided the scores are handled symmetrically. Some extreme examples of violation of this assumption are discussed in the Appendix B.

Theorem 5. *If the calibration model $\tilde{\mu}(\cdot)$ produces stable predictions $\hat{p}(y_i|\mathcal{X}_i)$ as the number of calibration samples $n_{cal} \to \infty$, the expected prediction set size $\mathbb{E}[|C(\mathbf{x})|]$ for a test point converges in probability to a fixed value:*

$$\mathbb{E}[|C(\mathbf{x})|] \to \sum_{y \in \mathcal{Y}} \mathbb{P}(\hat{p}(y|\mathbf{x}) \geq 1 - q^*), \tag{6}$$

where $q^* = F^{-1}(1-\alpha)$ is the $(1-\alpha)^{th}$-quantile of the true non-conformity score distribution.

The proof of the band-length convergence guarantee for CCF-HGNN in the Appendix E.

4 Experiments

Following the theoretical guarantees discussed earlier, we next demonstrate the empirical superiority of our proposed framework. Specifically, we evaluate the performance of our model and compare its performance against several non-trivial baselines on real-world datasets. We will first provide details about the experimental setup and then proceed to describe the evaluation metrics and experimental protocols, followed by the results.

4.1 Setup

We conducted all experiments on AMD EPYC 7763 64-Core Processor with 1.08 TB memory and 8 NVIDIA A40 GPUs with CUDA version 13.0.

4.1.1 Datasets: We evaluated the performance of our proposed framework on four real-world datasets used in prior works [10, 57]. The datasets include co-authorship datasets like DBLP [65], co-purchases large dataset like Walmart-Trips [2] and Trivago-Clicks [11], and co-voting datasets like House-Bills [11], and Congress [20], and co-association datasets like High-School [11, 37]. Summary statistics of the datasets are provided in Table 2. The prediction task is uncertainty quantification for classification.

4.1.2 Baseline Methods: As there are no prior works tailored to quantify uncertainty for hypergraphs, we use traditional uncertainty quantification methods (which do not provide statistical

Table 3: Empirical Marginal Coverage (%) of different models for the task of node classification on four datasets with $\alpha = 0.05$. The result takes the average and standard deviation across 20 independent runs.

Model	Walmart-Trips	House-Bills	Congress	DBLP	High-School	Trivago-Clicks	Covered?
TS	92.26 ± 0.31 ✗	91.21 ± 0.24 ✗	89.04 ± 0.48 ✗	87.34 ± 0.25 ✗	90.13 ± 0.82 ✗	93.77 ± 0.82 ✗	✗
VS	92.20 ± 0.18 ✗	91.18 ± 0.24 ✗	88.99 ± 0.46 ✗	87.33 ± 0.29 ✗	89.73 ± 0.17 ✗	93.04 ± 0.71 ✗	✗
ETS	92.20 ± 0.26 ✗	92.93 ± 1.77 ✗	89.23 ± 0.44 ✗	88.29 ± 0.65 ✗	92.48 ± 0.61 ✗	94.28 ± 0.35 ✗	✗
CP-APS	95.17 ± 0.00 ✓	99.83 ± 0.09 ✓	99.61 ± 0.02 ✓	95.04 ± 0.04 ✓	97.86 ± 0.45 ✓	95.08 ± 0.04 ✓	✓
CP-RAPS	95.11 ± 0.06 ✓	95.20 ± 0.04 ✓	95.17 ± 0.04 ✓	95.13 ± 0.03 ✓	96.14 ± 0.00 ✓	95.11 ± 0.04 ✓	✓
CF-HGNN-APS	95.05 ± 0.01 ✓	99.97 ± 0.00 ✓	99.94 ± 0.01 ✓	97.31 ± 2.58 ✓	97.33 ± 0.01 ✓	95.03 ± 0.00 ✓	✓
CF-HGNN-RAPS	95.01 ± 0.01 ✓	95.18 ± 0.10 ✓	95.14 ± 0.07 ✓	95.07 ± 0.01 ✓	95.98 ± 0.06 ✓	95.01 ± 0.03 ✓	✓
Ours-APS	95.06 ± 0.32 ✓	99.68 ± 0.00 ✓	99.79 ± 0.12 ✓	99.49 ± 0.39 ✓	97.79 ± 0.00✓	95.11 ± 0.04 ✓	✓
Ours-RAPS	95.06 ± 0.00 ✓	95.33 ± 0.03 ✓	95.34 ± 0.34 ✓	95.06 ± 0.04 ✓	96.13 ± 0.00 ✓	95.11 ± 0.00 ✓	✓

coverage guarantees) as baseline methods. These include Temperature Scaling (TS) [23], Vector Scaling (VS) [23], and Ensemble Temperature Scaling (ETS) [71]. Additionally, we adapt traditional conformal prediction methods by adopting an HGNN mean estimator to obtain point predictions on hypergraphs (CP). Finally, we adapted the SOTA conformal prediction method for GNNs [26] to aggregate information and perform conformal prediction in hypergraphs (CF-HGNN). Detailed descriptions of the baselines are provided in Appendix F.

4.1.3 Non-Conformity Score Functions: We evaluate two popular conformal prediction scores.

(1) APS (Adaptive Prediction Sets) [43]: For a model outputting class probabilities $\hat{p}(y \mid \mathbf{x})$, let $\pi(\mathbf{x})$ denote the ordering of labels sorted by decreasing probability. The APS score for class y is defined as $V_{\text{APS}}(\mathbf{x}, y) = \sum_{j:\pi_j(\mathbf{x}) \prec y} \hat{p}(\pi_j(\mathbf{x}) \mid \mathbf{x}) + U \cdot \hat{p}(y \mid \mathbf{x})$, where $U \sim \text{Unif}(0, 1)$ and $\pi_j(\mathbf{x}) \prec y$ means label $\pi_j(\mathbf{x})$ is ranked higher than y. APS adaptively constructs prediction sets by accumulating probabilities until the threshold calibrated by conformal prediction is reached.

(2) RAPS (Regularized Adaptive Prediction Sets) [3]: RAPS extends APS by adding a regularization term that penalizes large set sizes. For class y, the score is $V_{\text{RAPS}}(\mathbf{x}, y) = S_{\text{APS}}(\mathbf{x}, y) + \lambda \cdot \left|\{j : \pi_j(\mathbf{x}) \prec y\}\right|^{\gamma}$, where $\lambda \geq 0$ controls the strength of the penalty and $\gamma \geq 1$ controls its growth rate. This modification encourages tighter prediction sets while preserving coverage guarantees.

4.1.4 Evaluation Metrics: We randomly split data into train, validation, calibration-test folds with a 20:30:50 split ratio. We adopt the following metrics to evaluate the empirical performance:

(1) Marginal Coverage: For a predictive confidence band $C(\mathbf{x})$ and test point $(\mathbf{x}, y)$, the marginal coverage is defined as $\Pr\big(y \in C(\mathbf{x})\big)$. A valid inference procedure should ensure that the empirical coverage satisfies $\Pr\big(y \in C(\mathbf{x})\big) \geq 1 - \alpha$, where α is the target miscoverage rate.

(2) Band Length: Given that the empirical coverage exceeds $1 - \alpha$, the efficiency of the method is quantified by the expected length of the confidence band, $\mathbb{E}\big[\text{ length}(C(\mathbf{x}))\big]$. Comparisons of band length are only meaningful under the regime $\Pr(y \in C(\mathbf{x})) \geq 1 - \alpha$, since trivially $C(\mathbf{x}) = \emptyset$ yields zero length but violates the coverage constraint.

4.2 Results

We will now provide empirical performances of all the baselines and our proposed framework to quantify uncertainty for classification tasks on the four datasets. The important conclusions derived from the experiments are listed below.

All Conformal Frameworks Achieve the Desired Empirical Marginal Coverage while Traditional UQ Methods do not: We report the marginal coverage of various UQ methods with target coverage at 95% in Table 3. There are two primary takeaways. Firstly, none of the traditional UQ methods (VS, TS, and ETS) achieves the target coverage for all datasets, while the conformal prediction methods (CP, CF-HGNN, and CCF-HGNN) do, highlighting the need for models with statistical guarantees when deployed in high-stakes environments. Secondly, these empirical results of all the conformal methods align with the theoretical coverage guarantee given in Lemma 2. Henceforth, we will only report the performance of models that obtain the desired coverage levels.

Our Proposed Framework (CCF-HGNN) achieves the shortest Band Length in Most Datasets: We report the empirical band length for 4 datasets in Table 4. The key observations are as follows. First, compared to standard conformal baselines (CP-APS, CP-RAPS). Our proposed approach CCF-HGNN-RAPS, produces tighter bands across all but one dataset, while maintaining an impressive overall rank of 1.33 (the closest baselines get to 2.83). Second, while CF-HGNN offers improvements over GNN-based conformal methods, it is consistently outperformed by the proposed CCF-HGNN on hypergraph datasets. These results validate that incorporating contrastive learning with conformal prediction is crucial for boosting efficiency without compromising validity.

As observed in Table 3, APS-based conformal methods often produce empirical coverage well above the target level (close to 99%). This behavior arises because APS adaptively accumulates class probabilities until the calibration cutoff is exceeded, which in practice tends to overshoot the nominal threshold. While this conservativeness ensures validity, it also leads to overly large prediction sets. Consequently, APS methods trade efficiency for coverage, resulting in inflated band lengths (Table 4). By contrast, RAPS introduces an explicit penalty on the set size, thereby reducing redundancy in the prediction sets while still maintaining the desired coverage guarantees. However, Walmart-Trips, High-School, and Trivago-Clicks are exceptions, as the difference between APS and RAPS is less pronounced, with APS achieving competitive band lengths relative

Table 4: Empirical Predictive Band Length of Different Models (that have the desired coverage level) on Four Datasets with $\alpha = 0.05$. The result takes the average and standard deviation across 20 independent runs. Lower is better.

Model	Walmart-Trips	House-Bills	Congress	DBLP	High-School	Trivago-Clicks	Rank
CP-APS	9.198 ± 0.048	1.958 ± 0.005	1.961 ± 0.007	3.479 ± 0.127	7.81 ± 0.07	56.23 ± 2.55	5.33
CP-RAPS	9.053 ± 0.008	1.261 ± 0.054	1.317 ± 0.007	**1.509 ± 0.038**	7.66 ± 0.04	54.79 ± 1.41	3.33
CF-HGNN-APS	8.541 ± 0.023	1.993 ± 0.013	1.989 ± 0.008	4.346 ± 0.387	7.44 ± 0.06	52.09 ± 1.04	4.67
CF-HGNN-RAPS	8.595 ± 0.400	1.646 ± 0.191	1.619 ± 0.129	1.977 ± 0.184	7.28 ± 0.05	51.38 ± 2.01	2.83
Ours-APS	**8.481 ± 0.007**	1.953 ± 0.010	1.949 ± 0.008	4.354 ± 1.014	7.30 ± 0.32	51.47 ± 1.44	3.50
Ours-RAPS	8.528 ± 0.162	**1.189 ± 0.027**	**1.213 ± 0.043**	1.541 ± 0.060	**7.13 ± 0.73**	**50.89 ± 1.66**	**1.33**

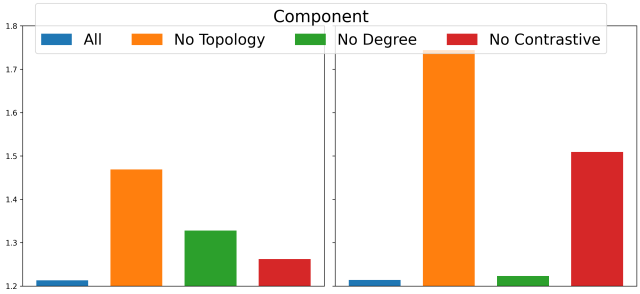
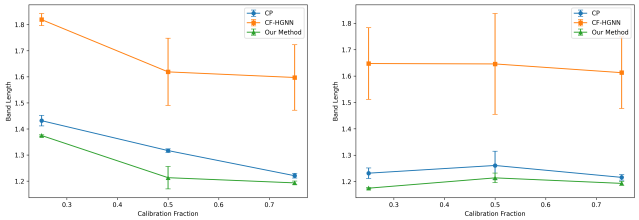


Figure 2: Ablation Study: Variation band length (right) for RAPS on CCF-HGNN on Congress (left) and House-Bills (right) dataset due to removal of individual components for $\alpha = 0.05$. Smaller is better.



(a) (b)

Figure 3: Sensitivity study on varying the calibration set fraction (3a and 3b) for Congress and House-Bills datasets, respectively.

to RAPS. This can be attributed to the nature of the label categories in these datasets, which have a relatively large number of classes but moderate class imbalance. In such settings, APS's conservative accumulation of probabilities does not inflate the prediction sets as severely as in smaller-class datasets, since the distribution of probabilities is already more spread out across labels. As a result, while RAPS still improves efficiency, the margin of improvement over APS is narrower on Walmart-Trips compared to the other datasets.

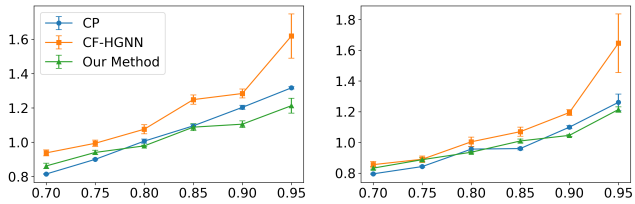
4.3 Ablation Studies

4.3.1 Removal of Model Components: We analyze the effect of removing three key components—the *topological-aware conformal loss*, *auxillary degree prediction loss*, and *contrastive loss*—on the **Congress** and **House-Bills** datasets on at a time. Figure 2 reports coverage and band length under RAPS with $\alpha = 0.05$. Our key observations are: (1) *Topology-aware conformal prediction loss is crucial*: removing it inflates RAPS length substantially, showing that structural information yields tighter sets. (2) *Minimizing the auxillary loss helps*: excluding degree modestly increases lengths. (3) *Contrastive learning improves efficiency*: dropping it slightly lengthens sets. Overall, each component contributes to efficiency, with topology offering the largest gains. The complete model yields the tightest bands while maintaining the desired coverage guarantees. It is expected that topology loss affects band length as it directly optimizes the conformal objective. But in isolation, it can overfit to the hypergraph structure. The contrastive loss ensures efficiency gains are stable. The smaller effect of the degree loss means that cardinality confusion is not the dominant uncertainty driver. However, it is the only component that addresses the scale of interaction uncertainty to improve efficiency

Table 5: Average Marginal Coverage and Band Length for all conformal methods using DAPS non-conformity score across 20 runs. $\alpha = 0.05$

Dataset	Method	Coverage	Band Length
Walmart-Trips	CP-APS	95.16 ± 0.02	8.973 ± 0.003
	CP-RAPS	95.08 ± 0.01	8.901 ± 0.053
	CF-HGNN-APS	95.05 ± 0.02	8.588 ± 0.003
	CF-HGNN-RAPS	95.01 ± 0.00	8.611 ± 0.001
	Our-APS	95.07 ± 0.03	**8.518 ± 0.001**
	Our-RAPS	95.05 ± 0.01	8.592 ± 0.003
DBLP	CP-APS	97.05 ± 0.00	3.470 ± 0.065
	CP-RAPS	95.07 ± 0.00	1.600 ± 0.011
	CF-HGNN-APS	97.11 ± 0.01	3.920 ± 0.011
	CF-HGNN-RAPS	95.08 ± 0.00	1.703 ± 0.002
	Our-APS	98.79 ± 0.01	3.734 ± 0.067
	Our-RAPS	95.13 ± 0.00	**1.508 ± 0.067**

4.3.2 Optimizing other Non-Conformity Scores: While our main experimental results were based on optimizing APS, we performed an additional experiment by using DAPS [68] as the non-conformity score function. The neighbour diffused scores of DAPS is given by $\hat{\mathbf{H}} = (1-\lambda)\mathbf{H} + \mathbf{D}^{-1}\mathbf{A}\mathbf{H}$, where $\mathbf{D}$ denotes the node degree matrix,



(a) (b)

Figure 4: Sensitivity study on varying α for Congress (4a) and House-Bills (4b).

A denotes the node adjacency matrix and **H** denotes the node-wise score matrix. We experimented on **Walmart-Trips** and **DBLP** datasets for a target coverage of 95%. The results of our experiments are presented in Table 5.

We notice that for **Walmart-Trips**, using DAPS does not improve performance for our method compared to optimizing the APS score in Table 4. However, the performance of our method improves when evaluated on the **DBLP dataset**. This is primarily due to the fact that the non-conformity score of DAPS induces homophily and thus does not improve performance in a heterophilous hypergraph like **Walmart-Trips**. However, for a homophilous hypergraph like **DBLP**, the performance of all methods improves significantly when using an appropriate non-conformity score.

4.4 Sensitivity Studies

4.4.1 Dependence on Confidence Level: We further study the sensitivity of our method to two key parameters: the miscoverage rate α (i.e., target confidence level) and the calibration set size. Figure 4 shows the results of this experiment for **Congress** and **House-Bills** datasets. Figure 4a and Figure 4b show the change in predictive band length as the confidence level increases from 0.7 to 0.95. Across both datasets, the band length grows monotonically with confidence, as expected. While all methods follow this trend, our method consistently achieves shorter band lengths compared to `CP` and `CF-HGNN`, especially at higher confidence levels (e.g., $\alpha = 0.05$). This demonstrates that our contrastive framework yields more informative uncertainty estimates without sacrificing coverage.

4.4.2 Size of Calibration Set: We also evaluate the effect of calibration set fraction (25%, 50%, 75%). Results in Figure 3a and Figure 3b show that our method remains stable with minimal fluctuation in band length as calibration data decreases. In contrast, CF-HGNN exhibits higher variance and inflated intervals, especially at smaller calibration fractions. This stability highlights the robustness of our approach under limited calibration resources, which is important in real-world healthcare applications where labeled calibration data may be scarce.

4.4.3 Effect of Mean Estimator: The mean estimator used for all experiments in the main text was HCHA [5]. As the conformal methods quantify uncertainty estimates on top of the point predictions made by the mean estimator, altering the mean estimator will cause fluctuations in performance. To illustrate this fact, we used a more recent backbone model, ED-HNN [57], that had slightly lower validation accuracy than HCHA **Congress** and slightly higher validation accuracy on **House-Bills** datasets. The results of our experiment are shown in Table 6.

Table 6: Performance of Models using ED-HNN averaged across 20 runs. $\alpha = 0.05$

Dataset	Model	RAPS Coverage	RAPS Length
Congress	CP	95.29 ± 0.00	1.78 ± 0.12
	Ours	95.28 ± 0.00	1.40 ± 0.05
House-Bills	CP	95.25 ± 0.00	1.24 ± 0.07
	Ours	95.19 ± 0.00	1.15 ± 0.15

The experimental results show that using ED-HNN instead of HCHA produces wider uncertainty estimates for the **Congress** dataset. On the other hand, for the **House-Bills** dataset, we observe slightly shorter predictive bands. This empirically validates the correlation between the predictive performance of the mean estimator and the size of the uncertainty bands. The higher the predictive accuracy of the mean estimator, the shorter the size of the uncertainty bands, and vice versa.

We also performed an additional sensitivity study to evaluate the impact of the different structural augmentation strategies on downstream UQ. We exploited contrastive augmentations by perturbing the hypergraph structure. We compare two strategies: (i) *random hyperedge drop*, which removes entire hyperedges, and (ii) *random edge drop*, which removes individual edges in the bipartite node–hyperedge graph. Table 7 summarizes the results. Across datasets, both strategies achieve the target coverage, but their impact on efficiency differs. On **Congress** and **House-Bills**, edge drop consistently yields shorter RAPS sets (e.g., 1.213 vs. 1.309 on Congress), indicating that fine-grained perturbations help the model learn more stable and discriminative representations. In contrast, **DBLP** benefits slightly more from hyperedge drop, where APS sets are tighter (3.688 vs. 4.354), suggesting that larger-scale perturbations are useful in high-homophily graphs with many small hyperedges. For **Walmart-Trips**, the differences between the two strategies are marginal, likely due to its large number of classes and moderate imbalance, where both perturbations introduce comparable variability. Overall, edge drop is generally more effective for heterophilic co-voting datasets, while hyperedge drop can be advantageous for homophilic graphs like DBLP. This demonstrates the importance of tailoring contrastive augmentation strategies to the structural properties of the underlying hypergraph.

5 Related Works

We discuss here related works that are closest to the ideas in CCF-HGNN in this section.

- **Uncertainty Quantification in deep learning and GNNs:** Several approaches address model-agnostic risk estimation for Graph Neural Networks (GNNs) in both classification and regression tasks [41, 44, 71]. Other studies leverage structural properties of graphs to explore calibration challenges, particularly the tendency of GNNs to be underconfident [25, 59]. A

Table 7: Effect of different contrastive strategies (mean ± std dev across 20 runs.) for $\alpha = 0.05$.

Dataset	Technique	APS Coverage	APS Length	RAPS Coverage	RAPS Length
Congress	Hyperedge Drop	99.83 ± 0.13	1.997 ± 0.011	95.27 ± 0.19	1.309 ± 0.046
	Edge Drop	99.79 ± 0.12	1.949 ± 0.008	95.34 ± 0.034	1.213 ± 0.043
DBLP	Hyperedge Drop	99.39 ± 0.10	3.688 ± 0.384	95.08 ± 0.16	1.641 ± 0.132
	Edge Drop	99.49 ± 0.39	4.354 ± 1.014	95.06 ± 0.04	1.541 ± 0.060
House-Bills	Hyperedge Drop	99.68 ± 0.00	1.953 ± 0.010	95.33 ± 0.03	1.189 ± 0.027
	Edge Drop	99.64 ± 0.03	1.955 ± 0.006	95.19 ± 0.00	1.214 ± 0.018
Walmart-Trips	Hyperedge Drop	95.05 ± 0.05	8.506 ± 0.136	95.05 ± 0.00	8.571 ± 0.102
	Edge Drop	95.06 ± 0.32	8.481 ± 0.007	95.06 ± 0.00	8.528 ± 0.162

foundational perspective is provided by [21], who interpret dropout training in deep neural networks as approximate Bayesian inference in deep Gaussian Processes. Complementary work investigates factors such as network depth, width, weight decay, batch normalization, and temperature scaling for improving calibration [23, 31]. More recently, stochastic centering has been proposed and applied as an effective calibration technique for GNNs [51, 52].

- **Conformal Prediction:** Conformal inference provides distribution free uncertainty quantification with rigorous coverage guarantees, enabling applications across diverse domains such as model calibration [47], passenger booking systems [62], computer vision [3, 7], and time-series forecasting [22, 34]. Given a user-specified miscoverage rate $\alpha \in (0, 1)$, the framework uses a calibration dataset to construct prediction sets or intervals that contain the true outcome with probability at least $1 - \alpha$. A variety of nonconformity scores have been proposed to improve performance in classification settings [27, 42, 43], with recent work introducing scores in the latent feature space [48]. While the classical framework relies on exchangeability, several extensions relax this assumption to handle label shift, covariate shift, or dependent data [6, 22, 34, 50].
- **Conformal Prediction for GNNs:** The use of conformal inference for network-structured data has recently gained traction. The first application in the inductive setting [15] demonstrated that nonconformity scores in this context are not exchangeable. In contrast, subsequent works [26, 36, 68] study the transductive setting, where nonconformity scores retain exchangeability. These approaches exploit the local neighborhood structure of graphs to improve effectiveness while maintaining computational efficiency. More recently, [69] introduced the notions of node-exchangeability and edge-exchangeability in growing graphs for the inductive setting, and proposed nonconformity scores defined on the evolving graph structure at each step. Recent works also include conformalized link prediction [74], weighted edge prediction [14, 36], dynamic GNNs [17, 58] and adversasial attack detection [18].

6 Conclusion

In this work, we extend the notion of UQ on hypergraphs by jointly accounting for both aleatoric and epistemic sources of uncertainty and proposing a hypergraph-based conformal prediction framework that leads to improved band lengths. While this is a promising direction, potential directions of future work include the evaluation of the performance of other HGNN models like Allset [10], and accounting for other sources of aleatoric uncertainty. On the side of conformal prediction, possible future directions include evaluation in the inductive setting [15, 69] where the assumption of exchangeability is not maintained.

7 Acknowledgments

This project was funded by the National Science Foundation (NSF) under Career Award IIS 2442159. The UIowa Post Comprehensive Research Fellowship funded A. Choudhuri during the completion of this project.

References

[1] Sameer Agarwal, Kristin Branson, and Serge Belongie. 2006. Higher order learning with graphs. In *Proceedings of the 23rd international conference on Machine learning*. 17–24.

[2] Ilya Amburg, Nate Veldt, and Austin Benson. 2020. Clustering in graphs and hypergraphs with categorical edge labels. In *Proceedings of the web conference 2020*. 706–717.

[3] Anastasios Angelopoulos, Stephen Bates, Jitendra Malik, and Michael I Jordan. 2020. Uncertainty sets for image classifiers using conformal prediction. *arXiv preprint arXiv:2009.14193* (2020).

[4] Anastasios N Angelopoulos and Stephen Bates. 2021. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511* (2021).

[5] Song Bai, Feihu Zhang, and Philip HS Torr. 2021. Hypergraph convolution and hypergraph attention. *Pattern Recognition* 110 (2021), 107637.

[6] Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. 2023. Conformal prediction beyond exchangeability. *The Annals of Statistics* 51, 2 (2023), 816–845.

[7] Stephen Bates, Anastasios Angelopoulos, Lihua Lei, Jitendra Malik, and Michael Jordan. 2021. Distribution-free, risk-controlling prediction sets. *Journal of the ACM (JACM)* 68, 6 (2021), 1–34.

[8] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261* (2018).

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PmLR, 1597–1607.

[10] Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. 2021. You are allset: A multiset function framework for hypergraph neural networks. *arXiv preprint arXiv:2106.13264* (2021).

[11] Philip S Chodrow, Nate Veldt, and Austin R Benson. 2021. Generative hypergraph clustering: From blockmodels to modularity. *Science Advances* 7, 28 (2021), eabh1303.

[12] Akash Choudhuri, Hieu Vu, Kishlay Jha, and Bijaya Adhikari. 2025. Domain Knowledge Augmented Contrastive Learning on Dynamic Hypergraphs for Improved Health Risk Prediction. In *Proceedings of the 2025 SIAM International*

Conference on Data Mining (SDM). SIAM, 476–486.
[13] Akash Choudhuri, Yongjian Zhong, and Bijaya Adhikari. 2025. Implicit Hypergraph Neural Network. *arXiv preprint arXiv:2508.14101* (2025).
[14] Akash Choudhuri, Yongjian Zhong, Mehrdad Moharrami, Christine Klymko, Mark Heimann, Jayaraman J Thiagarajan, and Bijaya Adhikari. 2025. Conformal Edge-Weight Prediction in Latent Space. In *Proceedings of the 2025 SIAM International Conference on Data Mining (SDM)*. SIAM, 161–170.
[15] Jase Clarkson. 2023. Distribution free prediction sets for node classification. In *International conference on machine learning*. PMLR, 6268–6278.
[16] Alvaro Correia, Fabio Valerio Massoli, Christos Louizos, and Arash Behboodi. 2024. An information theoretic perspective on conformal prediction. *Advances in Neural Information Processing Systems* 37 (2024), 101000–101041.
[17] Ed Davis, Ian Gallagher, Daniel John Lawson, and Patrick Rubin-Delanchy. 2024. Valid conformal prediction for dynamic gnns. *arXiv preprint arXiv:2405.19230* (2024).
[18] Sofiane Ennadir, Amr Alkhatib, Henrik Bostrom, and Michalis Vazirgiannis. 2023. Conformalized adversarial attack detection for graph neural networks. In *Conformal and Probabilistic Prediction with Applications*. PMLR, 311–323.
[19] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. 2019. Hypergraph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 3558–3565.
[20] James H Fowler. 2006. Legislative cosponsorship networks in the US House and Senate. *Social networks* 28, 4 (2006), 454–465.
[21] Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*. PMLR, 1050–1059.
[22] Isaac Gibbs and Emmanuel Candes. 2021. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems* 34 (2021), 1660–1672.
[23] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*. PMLR, 1321–1330.
[24] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).
[25] Hans Hao-Hsun Hsu, Yuesong Shen, Christian Tomani, and Daniel Cremers. 2022. What Makes Graph Neural Networks Miscalibrated? *Advances in Neural Information Processing Systems* 35 (2022), 13775–13786.
[26] Kexin Huang, Ying Jin, Emmanuel Candes, and Jure Leskovec. 2024. Uncertainty quantification over graph with conformalized graph neural networks. *Advances in Neural Information Processing Systems* 36 (2024).
[27] Rafael Izbicki, Gilson T Shimizu, and Rafael B Stern. 2019. Flexible distribution-free conditional predictive bands using density estimators. *arXiv preprint arXiv:1910.05575* (2019).
[28] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144* (2016).
[29] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
[30] Bhawesh Kumar, Charlie Lu, Gauri Gupta, Anil Palepu, David Bellamy, Ramesh Raskar, and Andrew Beam. 2023. Conformal prediction with large language models for multi-choice question answering. *arXiv preprint arXiv:2305.18404* (2023).
[31] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems* 30 (2017).
[32] Dong Li, Zhiming Xu, Sheng Li, and Xin Sun. 2013. Link prediction in social networks based on hypergraph. In *Proceedings of the 22nd international conference on world wide web*. 41–42.
[33] Xiaoyu Li, Guangyu Tang, and Jiaojiao Jiang. 2025. Implicit Hypergraph Neural Networks: A Stable Framework for Higher-Order Relational Learning with Provable Guarantees. *arXiv preprint arXiv:2508.09427* (2025).
[34] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2022. Conformal prediction with temporal quantile adjustments. *Advances in Neural Information Processing Systems* 35 (2022), 31017–31030.
[35] Zong-Zhi Lin, Thomas D Pike, Mark M Bailey, and Nathaniel D Bastian. 2024. A hypergraph-based machine learning ensemble network intrusion detection system. *IEEE transactions on systems, man, and cybernetics: systems* (2024).
[36] Rui Luo and Nicolo Colombo. 2024. Conformal Load Prediction with Transductive Graph Autoencoders. *arXiv preprint arXiv:2406.08281* (2024).
[37] Rossana Mastrandrea, Julie Fournet, and Alain Barrat. 2015. Contact patterns in a high school: a comparison between data collected using wearable sensors, contact diaries and friendship surveys. *PloS one* 10, 9 (2015), e0136497.
[38] Mark EJ Newman. 2003. The structure and function of complex networks. *SIAM review* 45, 2 (2003), 167–256.
[39] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2015. A review of relational machine learning for knowledge graphs. *Proc. IEEE* 104, 1 (2015), 11–33.
[40] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).
[41] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. 2019. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems* 32 (2019).
[42] Yaniv Romano, Evan Patterson, and Emmanuel Candes. 2019. Conformalized quantile regression. *Advances in neural information processing systems* 32 (2019).
[43] Yaniv Romano, Matteo Sesia, and Emmanuel Candes. 2020. Classification with valid and adaptive coverage. *Advances in Neural Information Processing Systems* 33 (2020), 3581–3591.
[44] Nabeel Seedat, Jonathan Crabbé, and Mihaela van der Schaar. 2022. Data-SUITE: Data-centric identification of in-distribution incongruous examples. In *International Conference on Machine Learning*. PMLR, 19467–19496.
[45] Kamile Stankeviciute, Ahmed M Alaa, and Mihaela Van der Schaar. 2021. Conformal time-series forecasting. *Advances in neural information processing systems* 34 (2021), 6216–6228.
[46] David Stutz, Krishnamurthy Dj Dvijotham, Ali Taylan Cemgil, and Arnaud Doucet. 2022. Learning Optimal Conformal Classifiers. In *International Conference on Learning Representations*.
[47] Dirar Sweidan and Ulf Johansson. 2021. Probabilistic Prediction in scikit-learn. In *The 18th International Conference on Modeling Decisions for Artificial Intelligence, On-line (from Umeå, Sweden), September 27-30, 2021*.
[48] Jiaye Teng, Chuan Wen, Dinghuai Zhang, Yoshua Bengio, Yang Gao, and Yang Yuan. 2022. Predictive Inference with Feature Conformal Prediction. In *The Eleventh International Conference on Learning Representations*.
[49] Ze Tian, TaeHyun Hwang, and Rui Kuang. 2009. A hypergraph-based learning algorithm for classifying gene expression and arrayCGH data with prior knowledge. *Bioinformatics* 25, 21 (2009), 2831–2838.
[50] Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. 2019. Conformal prediction under covariate shift. *Advances in neural information processing systems* 32 (2019).
[51] Puja Trivedi, Mark Heimann, Rushil Anirudh, Danai Koutra, and Jayaraman J Thiagarajan. 2023. A Stochastic Centering Framework for Improving Calibration in Graph Neural Networks. In *The Twelfth International Conference on Learning Representations*.
[52] Puja Trivedi, Danai Koutra, and Jayaraman J Thiagarajan. 2024. On Estimating Link Prediction Uncertainty Using Stochastic Centering. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 6810–6814.
[53] Aad W Van der Vaart. 2000. *Asymptotic statistics*. Vol. 3. Cambridge university press.
[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
[55] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. 2017. Graph attention networks. *stat* 1050, 20 (2017), 10–48550.
[56] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. 2005. *Algorithmic learning in a random world*. Vol. 29. Springer.
[57] Peihao Wang, Shenghao Yang, Yunyu Liu, Zhangyang Wang, and Pan Li. 2023. Equivariant Hypergraph Diffusion Neural Operators. In *The Eleventh International Conference on Learning Representations*. https://openreview.net/forum?id=RiTjKoscnNd
[58] Tuo Wang, Jian Kang, Yujun Yan, Adithya Kulkarni, and Dawei Zhou. 2025. Non-exchangeable Conformal Prediction for Temporal Graph Neural Networks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 3031–3042.
[59] Xiao Wang, Hongrui Liu, Chuan Shi, and Cheng Yang. 2021. Be confident! towards trustworthy graph neural networks via confidence calibration. *Advances in Neural Information Processing Systems* 34 (2021), 23768–23779.
[60] Yuxin Wang, Quan Gan, Xipeng Qiu, Xuanjing Huang, and David Wipf. 2023. From hypergraph energy functions to hypergraph neural networks. In *International Conference on Machine Learning*. PMLR, 35605–35623.
[61] Tianxin Wei, Yuning You, Tianlong Chen, Yang Shen, Jingrui He, and Zhangyang Wang. 2022. Augmentations in hypergraph contrastive learning: Fabricated and generative. *Advances in neural information processing systems* 35 (2022), 1909–1922.
[62] Hugo Werner, Lars Carlsson, Ernst Ahlberg, and Henrik Boström. 2021. Evaluation of updating strategies for conformal predictive systems in the presence of extreme events. In *Conformal and Probabilistic Prediction and Applications*. PMLR, 229–242.
[63] Ran Xu, Mohammed K Ali, Joyce C Ho, and Carl Yang. 2023. Hypergraph transformers for ehr-based clinical predictions. *AMIA Summits on Translational Science Proceedings* 2023 (2023), 582.
[64] Ran Xu, Yue Yu, Chao Zhang, Mohammed K Ali, Joyce C Ho, and Carl Yang. 2022. Counterfactual and factual reasoning over hypergraphs for interpretable clinical predictions on ehr. In *Machine Learning for Health*. PMLR, 259–278.

[65] Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha Talukdar. 2019. Hypergcn: A new method for training graph convolutional networks on hypergraphs. *Advances in neural information processing systems* 32 (2019).
[66] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 974–983.
[67] Margaux Zaffran, Olivier Féron, Yannig Goude, Julie Josse, and Aymeric Dieuleveut. 2022. Adaptive conformal predictions for time series. In *International Conference on Machine Learning*. PMLR, 25834–25866.
[68] Soroush H Zargarbashi, Simone Antonelli, and Aleksandar Bojchevski. 2023. Conformal prediction sets for graph neural networks. In *International Conference on Machine Learning*. PMLR, 12292–12318.
[69] Soroush H Zargarbashi and Aleksandar Bojchevski. 2023. Conformal Inductive Graph Neural Networks. In *The Twelfth International Conference on Learning Representations*.
[70] Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. 2018. Efficient neural network robustness certification with general activation functions. *Advances in neural information processing systems* 31 (2018).
[71] Jize Zhang, Bhavya Kailkhura, and T Yong-Jin Han. 2020. Mix-n-match: Ensemble and compositional methods for uncertainty calibration in deep learning. In *International conference on machine learning*. PMLR, 11117–11128.
[72] Liyan Zhang, Jingfeng Guo, Jiazheng Wang, Jing Wang, Shanshan Li, and Chunying Zhang. 2022. Hypergraph and uncertain hypergraph representation learning theory and methods. *Mathematics* 10, 11 (2022), 1921.
[73] Zizhao Zhang, Haojie Lin, Yue Gao, and KLISS BNRist. 2018. Dynamic hypergraph structure learning.. In *IJCAI*. 3162–3169.
[74] Tianyi Zhao, Jian Kang, and Lu Cheng. 2024. Conformalized link prediction on graph neural networks. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4490–4499.

A Hypergraph Neural Networks

The early structure of HGNNs mimicked the convolution step of GNNs. In particular, Feng et al. [19] proposed the first spectral hypergraph convolution, formulated as

$$\mathbf{X}' = \sigma\left(\mathbf{D}_v^{-\frac{1}{2}}\mathbf{H}\mathbf{W}_e\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-\frac{1}{2}}\mathbf{X}\mathbf{W}\right), \tag{7}$$

where $\mathbf{H}$ is the incidence matrix, $\mathbf{D}_v$ and $\mathbf{D}_e$ are vertex and hyperedge degree matrices, $\mathbf{W}_e$ is a diagonal hyperedge weight matrix, and $\mathbf{W}$ is a trainable weight matrix.

Later, [5] introduced a simplified hypergraph convolution operation, expressed as

$$\mathbf{X}' = \sigma\big(\mathbf{D}_v^{-1}\mathbf{H}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{X}\mathbf{W}\big), \tag{8}$$

which removes the symmetric normalization and leads to a message-passing view of hypergraph learning. In all our experiments, we have used the formulation by [5].

B Examples of Violation of Assumption 1 in Theorem 2

There are some extreme examples where the contrastive augmentations will violate this assumption:

(1) In multi-class node classification with extreme label imbalance, where even small augmentation may disproportionately isolate nodes from minority classes. In such cases, the topology-aware correction mechanism may no longer propagate reliable information through the local neighborhood, causing calibration to break down and resulting in lower empirical marginal coverage on average.
(2) In hypergraphs with extremely poor connectivity, such as containing a single bridging hyperedge that connects two or more large, otherwise disconnected hypergraph components, if a contrastive augmentation removes or perturbs this bridging hyperedge, the connectivity between the components is disrupted. As a result, the local neighborhood information used in the topology-aware correction may no longer reflect the true label dependencies across the hypergraph, potentially violating the Bounded Coverage assumption and leading to miscalibrated prediction sets.

C Proof of Proposition 3

Proposition 6. *Let $I(Y;\mathbf{Z}_1)$ and $I(Y;\mathbf{Z}_0)$ denote the mutual information between the labels and latent embeddings for CCF-HGNN and CF-HGNN, respectively, and $\Delta \in \mathbf{R}^+$ then,*

$$I(Y;\mathbf{Z}_1) \geq I(Y;\mathbf{Z}_0) + \Delta. \tag{9}$$

Proof. From Proposition E.2 in [40], we know

$$I(Y;\mathbf{Z}) \geq \log(N) - \mathcal{L}_{\mathbf{Z}}^{\text{InfoNCE}},$$

where N is the number of samples and $\mathcal{L}_{\mathbf{Z}}^{\text{InfoNCE}}$ the InfoNCE loss.

For our method, $\mathcal{L}_{\mathbf{Z}_1}^{\text{InfoNCE}}$ is explicitly minimized, compared to CF-HGNN, which consequently means that $\log(N) - \mathcal{L}_{\mathbf{Z}_1}^{\text{InfoNCE}} \geq \log(N) - \mathcal{L}_{\mathbf{Z}_0}^{\text{InfoNCE}}$.

This implies the proof statement:

$$I(Y;\mathbf{Z}_1) \geq I(Y;\mathbf{Z}_0) + \Delta$$

□

D Proof of Theorem 8

Lemma 7. *[16] For any conformal prediction scheme with the coverage guarantee of $1-\alpha$, and any distribution $q(\cdot)$, we have:*

$$\begin{aligned}\mathbb{E}([\log|C(x)|]^+) \geq (1-\alpha)\frac{H(Y|X) - h_b(a) - a\log M}{1-\alpha+\frac{1}{n+1}} \\ (1-\alpha)\frac{\alpha\mathbb{E}_{P_{Y,X,\mathcal{D}_{cal}|E=0}}\left[-\log\hat{Q}^0_{Y|X} + \log\mathbb{E}_{u(y_{C(x)})}[q(y|x)]\right]}{1-\alpha+\frac{1}{n+1}} \\ -(1-\alpha)\mathbb{E}_{P_{Y,X,\mathcal{D}_{cal}|E=1}}\left[-\log\hat{Q}^1_{Y|X} + \log\mathbb{E}_{u(y_{C(x)})}[q(y|x)]\right],\end{aligned} \tag{10}$$

where $\hat{Q}^0_{Y|X} = q(y|x)\mathbb{I}[y \notin C(x)]$ and $\hat{Q}^1_{Y|X} = q(y|x)\mathbb{I}[y \in C(x)]$. Here, $|C(x)|$ denotes the size of the prediction set for input x, $H(Y|X)$ the conditional entropy of Y given X, $h_b(\cdot)$ the binary entropy function, a the error probability, and $M = |\mathcal{Y}|$ the number of classes.

Lemma 7 shows that the expected prediction set size is lower bounded by the conditional entropy $H(Y|X)$, penalized by calibration-dependent terms.

Theorem 8. *Under the assumptions:*

(1) ***Bounded coverage:*** *Contrastive augmentations do not reduce conformal coverage (marginal coverage $\geq 1-\alpha$ is preserved on average).*
(2) ***Large Mutual Information gap:*** *$I(Y;\mathbf{Z}_1) - I(Y;\mathbf{Z}_0)$ is sufficiently large (Lemma 6).*

Then, the expected conformal prediction set size under CCF-HGNN is smaller than under CF-HGNN:

$$\mathbb{E}\big[|C_1(\mathbf{x})|\big] \leq \mathbb{E}\big[|C_0(\mathbf{x})|\big]. \tag{11}$$

Proof. By Lemma 6, $I(Y;Z_1) \geq I(Y;Z_0) + \Delta$. Equivalently, $H(Y|Z_1) \leq H(Y|Z_0) - \Delta$.

Lemma 7 lower bounds the expected log set size in terms of $H(Y|X)$. Since Z_1 captures more information about Y than Z_0, the effective conditional entropy $H(Y|Z_1)$ is smaller. Thus, the bound for $C_1(X)$ is tighter than for $C_0(X)$.

Formally,

$$\mathbb{E}[\log |C_0(X)|]^+ \geq f(H(Y|Z_0)),$$
$$\mathbb{E}[\log |C_1(X)|]^+ \geq f(H(Y|Z_1)),$$

where $f(\cdot)$ is the lower-bound functional in Lemma 7. Since $H(Y|Z_1) < H(Y|Z_0)$, the bound for $C_1(X)$ is strictly smaller, which implies:

$$\mathbb{E}[|C_1(X)|] \leq \mathbb{E}[|C_0(X)|].$$

□

E Convergence of CCF-HGNN

Theorem 9. *If the calibration model $\tilde{\mu}(\cdot)$ produces stable predictions $\hat{p}(y_i|\mathcal{X}_i)$ as the number of calibration samples $n_{cal} \to \infty$, the expected prediction set size $\mathbb{E}[|C(\mathbf{x})|]$ for a test point converges in probability to a fixed value:*

$$\mathbb{E}[|C(\mathbf{x})|] \to \sum_{y \in \mathcal{Y}} \mathbb{P}(\hat{p}(y|\mathbf{x}) \geq 1 - q^*), \tag{12}$$

where $q^* = F^{-1}(1-\alpha)$ is the $(1-\alpha)^{th}$-quantile of the true non-conformity score distribution.

Proof. Let $F_n(v) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}(V_i \leq v)$ be the empirical CDF of the non-conformity scores $V_i = 1 - \hat{p}(y_i|\mathcal{X}_i)$. Using Glivenko-Cantelli Theorem [53] with Assumption 1, $\sup_v |F_n(v) - F(v)| \to 0$ as $n \to \infty$. Assuming the calibration model $\tilde{\mu}(\cdot)$ produces stable predictions $\hat{p}(y_i|\mathcal{X}_i)$ as the number of calibration samples $n_{cal} \to \infty$, and as F is continuous and strictly increasing, F^{-1} is continuous at $1-\alpha$. For any $\epsilon > 0$, choose $\delta > 0$ such that

$$F(q^* - \epsilon) < 1 - \alpha - \delta, \quad F(q^* + \epsilon) > 1 - \alpha + \delta.$$

As n_{cal} grows, $\sup_v |F_{n_{\text{cal}}}(v) - F(v)| < \delta$, which means

$$F_{n_{\text{cal}}}(q^* - \epsilon) \geq F(q^* - \epsilon) - \delta < 1 - \alpha,$$

and

$$F_{n_{\text{cal}}}(q^* + \epsilon) \leq F(q^* + \epsilon) + \delta > 1 - \alpha.$$

So $q^* - \epsilon < \hat{q} < q^* + \epsilon$, which means

$$\mathbb{P}(|\hat{q} - q^*| > \epsilon) \to 0 \quad \text{as } n_{cal} \to \infty.$$

The prediction set is thus

$$C(\mathbf{x}) = \{y \in \mathcal{Y} : \hat{p}(y|\mathbf{x}) \geq 1 - \hat{q}\}.$$

So the expected set size is

$$\mathbb{E}[|C(\mathbf{x})|] = \mathbb{E}\left[\sum_{y \in \mathcal{Y}} \mathbf{1}(\hat{p}(y|\mathbf{x}) \geq 1 - q)\right] = \sum_{y \in \mathcal{Y}} \mathbb{P}(\hat{p}(y|\mathbf{x}) \geq 1 - q).$$

As $\hat{q} \to q^*$, and since $g(\cdot)$ is stable,

$$\mathbf{1}(\hat{p}(y|\mathbf{x}) \geq 1 - \hat{q}) \to \mathbf{1}(\hat{p}(y|\mathbf{x}) \geq 1 - q^*).$$

So,

$$\mathbb{E}[|C(\mathbf{x})|] \to \sum_{y \in \mathcal{Y}} \mathbb{P}(\hat{p}(y|\mathbf{x}) \geq 1 - q^*).$$

This limit is a fixed value determined by the distribution of $\hat{p}(y|\mathbf{x})$ and q^*. Conformal prediction ensures that as long as $\hat{q}$ is calibrated,

$$\mathbb{P}(y \in C(\mathbf{x})) \geq 1 - \alpha.$$

□

F Descriptions of the Baselines

The baseline models used in this work can be characterized into the following categories:

- **Traditional UQ Methods:** These methods do not provide any statistical guarantee about marginal coverage. The 3 baseline methods used under this category are as follows:
  (1) **Temperature Scaling (TS) [23]:** It is a post-processing calibration method for UQ. It takes the model's logits (pre-softmax outputs) and divides them by a learned scalar parameter called the temperature. Higher temperature values produce softer probability distributions with lower confidence.
  (2) **Vector Scaling (VS) [23]:** Vector scaling is a more flexible version of temperature scaling. Instead of using a single global adjustment for all classes, it assigns each class its own adjustment with a small bias. This allows the model to adjust situations where some classes are consistently overconfident or underconfident, thereby improving the calibration of predicted probabilities across all classes.
  (3) **Ensemble Temperature Scaling (ETS) [71]:** Ensemble Temperature Scaling applies temperature scaling to the aggregated outputs of a model ensemble. A single temperature parameter is learned on the ensemble's averaged logits to adjust overall confidence. This method preserves the accuracy advantages of ensembling while improving calibration, resulting in more reliable uncertainty estimates.
- **Conformal Prediction Methods:** These methods have a theoretical guarantee for marginal coverage. We adapted two prior works as baselines:
  (1) **Conformal Predictor (CP) [56]:** For this model, the mean estimator (HGNN) was trained on the classification task on the training data. After that, the non-conformity scores were obtained for the calibration data (node set), a quantile was selected (based on the type of the non-conformity score function), and predictive bands were constructed for test nodes.
  (2) **Conformalized Hypergraph Neural Network [26] (CF-HGNN):** This model integrates conformal prediction with hypergraph neural networks to provide uncertainty estimates with guaranteed marginal coverage. The key idea is to adapt non-conformity scores to hypergraph learning tasks, where nodes, edges, and higher-order relationships need to be considered simultaneously. CF-HGNN first trains a base HGNN to produce class probability estimates, then applies a conformal calibration step using a held-out calibration set. Unlike CP, CF-HGNN explicitly accounts for hypergraph structures, leading to tighter predictive sets and better utilization of higher-order relational information.