Uncertainty-Aware Graph Neural Network for Semi-Supervised Diversified Recommendation

Minjie (MINJIECAO@IEEE.ORG)

Research Article

Keywords: recommendation systems, uncertainty-aware pseudo-label selection, diversity recommendation

Posted Date: February 9th, 2024

DOI: https://doi.org/10.21203/rs.3.rs-3263186/v2

Additional Declarations: The authors declare no competing interests.

Uncertainty-Aware Graph Neural Network for Semi-Supervised Diversified Recommendation

Minjie Cao[1*] and Thomas Tran[1]

[1*]School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Ontario, Canada.

*Corresponding author(s). E-mail(s): mcao094@uottawa.ca;
Contributing authors: ttran@uottawa.ca;

Abstract

Graphs are a powerful tool for representing structured and relational data in various domains, including social networks, knowledge graphs, and molecular structures. Semi-supervised learning on graphs has emerged as a promising approach to address real-world challenges and applications. In this paper, we propose an uncertainty-aware pseudo-label selection framework for promoting diversity learning in recommendation systems. Our approach harnesses the power of semi-supervised Graph Neural Networks (GNNs), utilizing both labeled and unlabeled data, to address data sparsity issues often encountered in real-world recommendation scenarios. Pseudo-labeling, a prevalent semi-supervised method, combats label scarcity by enhancing the training set with high-confidence pseudo-labels for unlabeled nodes, enabling self-training cycles for supervised models. By incorporating pseudo-labels selected based on the model's uncertainty, our framework is designed to improve the model's generalization and foster diverse recommendations. The main contributions of this paper include introducing the uncertainty-aware pseudo-label selection framework, providing a comprehensive description of the framework, and presenting an experimental evaluation comparing its performance against baseline methods in terms of recommendation quality and diversity. Our proposed method demonstrates the effectiveness of uncertainty-aware pseudo-label selection in enhancing the diversity of recommendation systems and delivering a more engaging, personalized, and diverse set of suggestions for users.

Keywords: recommendation systems, uncertainty-aware pseudo-label selection, diversity recommendation.

1 Introduction

Graphs function as a universal language for representing structured and relational data, including social networks, knowledge graphs, and Molecules as graphs. Extracting information and learning from graphs can provide valuable insights for numerous real-world issues and applications [1, 2]. This study concentrates on the challenge of semi-supervised learning on graphs [3, 4], which seeks to categorize unlabeled nodes in a given graph using only a small fraction of labeled nodes. Graph neural networks (GNNs) have recently gained prominence as potent methods among the available solutions. GNNs primarily rely on a deterministic feature propagation mechanism to develop expressive node representations [5–7].

Recommendation systems have become an essential component of various online platforms, including e-commerce websites, social networks, and streaming services. These systems aim to provide personalized suggestions to users, enhancing their experience and engagement on the platforms. However, a common challenge faced by recommendation systems is the trade-off between accuracy [8] and diversity [9]. While many algorithms focus on maximizing accuracy [10], diverse recommendations are essential to cater to users' evolving preferences, avoid filter bubbles, and promote long-tail items [11, 12].

In this paper, we propose an uncertainty-aware pseudo-label selection framework that promotes diversity learning in recommendation systems. Our approach leverages semi-supervised learning, a technique that utilizes both labeled and unlabeled data, to address the data sparsity issue often encountered in real-world recommendation scenarios. By incorporating pseudo-labels selected based on the model's uncertainty, our framework is designed to improve the model's generalization and foster diverse recommendations.

The main contributions of this paper are as follows: (1) We introduce an uncertainty-aware pseudo-label selection framework for promoting diversity learning in recommendation systems, (2) We present an experimental evaluation of our proposed method, comparing its performance against baseline methods in terms of both recommendation quality and diversity.

By demonstrating the effectiveness of our uncertainty-aware pseudo-label selection framework, we hope to contribute to the ongoing efforts to improve the diversity of recommendation systems and provide users with a more engaging, personalized, and diverse set of suggestions.

2 Related Work

2.1 GNN for Diversified Recommendation

To promote diversity in recommender systems, the retrieved items should cover a wide range of topics, such as various product categories or music genres [13] . This requires the user embeddings generated by GNNs to be close to item embeddings with diverse topics[14]. However, the embedding aggregation operation in GNNs can result in user embeddings that are similar to the embeddings of items with which users have previously interacted [15]. This similarity can potentially hinder diversity

by predominantly recommending items that belong to the dominant topic in users' interaction history [14]. For instance, if a user primarily interacts with electronics, a GNN-generated embedding may be too close to electronic item embeddings, resulting in the system only recommending electronics and leading to low diversity [16].

To address the challenge of weak signals from less prevalent topics, researchers have explored ways to limit the impact of dominant topics by constructing diversified sub-graphs from the original user-item bipartite graph. Sun et al. [17] propose a model called Bayesian Graph Collaborative Filtering (BGCF), which creates augmented graphs using node copying [18] from high-order neighbors, allowing items of diverse topics with high similarity to be directly connected to user nodes. Zheng et al. [19] propose Diversified Graph Convolutional Networks (DGCN) and employ rebalanced neighbor sampling to reduce the weight of dominant topics and increase the importance of less prevalent topics in neighbor nodes. Graph Neural Network for Recommendation with Diversified Embedding Generation (DGRec) proposed by Yang et al. [20], is another current state-of-the-art diversified Recommendation System model. It design a submodular function to select a varied subset of neighbors during the aggregation process, aiming to boost diversity in its recommendations.

2.2 Uncertainty-Aware Pseudo-label Selection

The recent advancement of image classification task at semi-supervised learning are made mostly contributed by Consistency-regularization [21, 22] and Pseudo-label-based methods upgradation [23] . The previous state-of-the-art approaches for semi-supervised learning are usually a combination of these two methods [24] .

Consistency regularization methods dominate the current field of semi-supervised learning. The stunning performance of such techniques is however contributed by a large amount of prior work: for a classification task on a specific dataset, research often must spend a lot of effort in advance to search for the most domain-specific data augmentation strategy, but when it comes to some specified domains like video classification where these augmentation datasets are less effective, its capability will be limited [21].

One advantage of pseudo-label over consistency regularization is that it does not inherently require augmentations and can be generally applied to most domains. However, recent consistency regularization approaches tend to outperform pseudo-labeling on Semi-Supervised Learmomh benchmarks [21]. Rizve et al. [25] demonstrate that pseudo-label-based methods can perform on par with consistency regularization methods. Most of their experiments involve learning from noisy data and approach proposed greatly reduces noise by minimizing the effect of poor network calibration, allowing for competitive state-of-the-art results.

In the traditional pseudo-labeling method, a threshold is typically set to generate pseudo-label, and samples are given pseudo labels when the model's predicted probability for a certain class exceeds the threshold. The class with the highest probability is directly selected as the pseudo label. This is represented by the equation:

$$\tilde{y}_c^{(i)} = \mathbb{1}\left[p_c^{(i)} \geq \gamma\right] \tag{1}$$

where $\tilde{y}_c^{(i)}$ represent pseudo-labels generated for class c in $x^{(i)}$, $p_c^{(i)}$ represents the probability of class c being present in the sample.

However, relying solely on positive pseudo labels can lead to poor performance due to the large number of incorrectly pseudo-labeled samples used during training. Relying only on positive pseudo labels might introduce noise and also limit the diversity of the training data. Since only the samples that the model is confident about are pseudo-labeled and used for further training, the model might not get the chance to learn from more challenging or ambiguous cases in the unlabeled data. In Rizve et al.'s work [25], both positive and negative pseudo labels are generated to address this limitation.

Let $g^{(i)}$ indicate whether the pseudo label of $x^{(i)}$ will be counted for training, where $g^{(i)} = \left[g_1^{(i)}, \ldots, g_C^{(i)}\right] \subseteq \{0,1\}^C$,and the equation has been denoted with τ_p and τ_n are the confidence thresholds for positive and negative pseudo labels selection. Adding negative learning significantly reduces noise during training, leading to improved performance compared to the traditional pseudo-label method.

While confidence-based selection reduces pseudo-label error rates, it does not fully address the issue of poor network calibration. In poorly calibrated networks, incorrect predictions can have high confidence scores. Rizve's work [25] demonstrates that leveraging prediction uncertainties can mitigate the effects of poor calibration. An uncertainty-aware pseudo-label selection approach incorporates both the confidence and uncertainty of network predictions, resulting in a more accurate subset of pseudo-labels used in training. Equation 1 is modified as follows:

$$g_c^{(i)} = \mathbb{1}\left[u\left(p_c^{(i)}\right) \leq \kappa_p\right] \mathbb{1}\left[p_c^{(i)} \geq \tau_p\right] + \mathbb{1}\left[u\left(p_c^{(i)}\right) \leq \kappa_n\right] \mathbb{1}\left[p_c^{(i)} \leq \tau_n\right] \tag{2}$$

where $u(p)$ is the uncertainty of a prediction p, and κ_p and κ_n are the uncertainty thresholds.

In conclusion, the introduction of negative pseudo labels and an uncertainty-aware pseudo-label selection approach offer several advantages in training recommendation systems. By incorporating negative pseudo labels, the method effectively reduces the impact of incorrectly labeled samples during training, leading to improved performance. Additionally, leveraging prediction uncertainties through the uncertainty-aware pseudo-label selection helps mitigate the effects of poor network calibration, resulting in a more accurate subset of pseudo-labels used for training. This approach enhances the model's generalization capabilities and fosters diverse recommendations by promoting a balance between exploration and exploitation. Overall, the positive aspects of this method contribute to the improved accuracy, serendipity, and diversity of recommendations in recommendation systems.

3 Uncertainty-Aware Graph Neural Network

In this section, We will introduce the Uncertainty-Aware Graph Neural Network (UGNN) for semi-supervised learning on graphs. We will begin by introducing the definitions and notations utilized throughout the paper. Following that, we provide an overview of the proposed framework before delving into the specifics of each model

component. Finally, we discuss the process of learning the model parameters.

Notation

a	A scalar (integer or real)
$\mathcal{V}$	Set of nodes
$\mathbb{1}$	Indicator Function
$\boldsymbol{a}$	A vector
$\boldsymbol{A}$	Adjacency matrix
$\mathbf{A}$	A tensor
$\mathcal{G}$	The user-item interaction graph
$\boldsymbol{I}_n$	Identity matrix with n rows and n columns
$v_1, v_2, \ldots, v_m$	User nodes across the Graph
$\text{diag}(\boldsymbol{a})$	A square, diagonal matrix with diagonal entries given by $\boldsymbol{a}$
$\mathbf{a}$	A vector-valued random variable
$\boldsymbol{R} \in \mathbb{R}^{m \times n}$	Binary matrix with entries only 0 and 1 that represent user-item interactions in $\mathcal{G}$
$\boldsymbol{I}_i$	Element i of the item nodes, with indexing starting at 1

3.1 Overall Architecture

As previously introduced, we integrate uncertainty-aware selection into the matching process in conjunction with GNN. Our goal is to recommend items that align with users' interests while also maintaining diversity by ensuring dissimilarity between the recommended items. To achieve this, we employ unobserved samples as positive instances, allowing us to bring new items into the spotlight.

Figure 1 presents an overview of our proposed DGCN's comprehensive architecture. As shown in the Figure 1, our proposed model consist components: 1) Uncertainty-Aware Pseudo-Label Selection; 2) The GNN Encoder; 3) BCE loss and Contrastive Loss measured by Discriminator for optimization. Given a graph as input, a GNN encoder is employed to generate user embeddings and interaction predictions. Utilizing the generated embeddings and graph structure, we introduce an uncertainty-aware pseudo-label selection (UPS) measure to evaluate the confidence of the nodes, which aids in the selection of pseudo-labels. Based on the confidence measure and prediction probabilities, we assign pseudo-labels to selected reliable nodes and incorporate them into the existing label set for model retraining. Throughout the GNN retraining phase, the model concentrates on learning from more accurate and reliable pseudo-labels by taking into account both the confidence and uncertainty of predictions. This leads

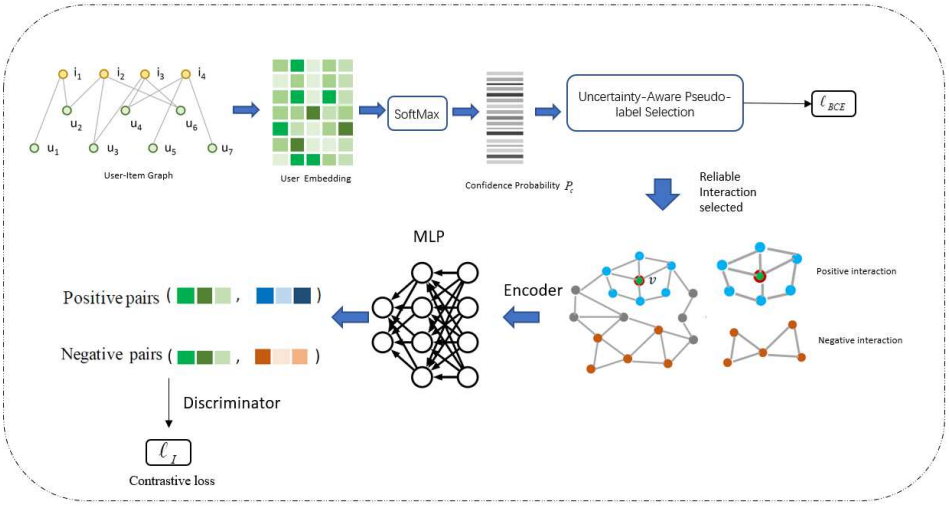

Fig. 1 Overview Architecture of the proposed Uncertainty-Aware Graph Neural Network

to enhanced generalization and improved model robustness against potential noise in pseudo-labels.

3.2 The GNN Encoder

The GNN encoder serves as the backbone for our framework [4], primarily responsible for generating user embeddings and providing label prediction probabilities to represent model confidence in its predictions. Any GNN focusing on node classification can be used for embedding learning and classification. A GNN encoder typically learns node embeddings by recursively aggregating and transforming node features from topological neighborhoods. For a node $v \in \mathcal{V}$, the embedding at the k-th layer can be computed as follows to generate the embedding H_v:

$$h_v^k = \sigma\left(\sum_{v' \in \mathcal{N}_v} \left(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}\right)_{v,v'} W^{k-1} h_{v'}^{k-1}\right), \tag{3}$$

Let $\sigma(\cdot)$ denote the activation function utilized in our model. The modified adjacency matrix of graph $\mathcal{G}$, incorporating self-connections, is given by $\tilde{A} = A + I$. Within this matrix, the entry $\tilde{D}ii$ is defined as the summation over the i-th row, mathematically represented as $\tilde{D}ii = \sum_j \tilde{A}_{ij}$. Furthermore, in each layer of our network structure, there exists a unique weight matrix, specified as W^k.

Finally, given the class prediction probabilities, the confidence score for each node can be calculated as v :

$$s_c(v) = \max_j f_\theta\left(\mathbf{x}_v\right)_j, \tag{4}$$

The confidence score $s_c(v)$ is utilized for node selection in combination with the representativeness score, which is detailed below.

3.3 Uncertainty-Aware Pseudo-Label Selection (UPS)

Incorporating uncertainty-aware selection into the recommendation task can contribute to the diversity of recommendations in many aspects:

Noise reduction: Uncertainty-aware selection helps in identifying and filtering out noisy pseudo-labels generated during the semi-supervised learning process. By considering both the confidence and uncertainty of predictions, the model focuses on learning from more accurate and reliable pseudo-labels, which leads to better generalization and more diverse recommendations.

Coverage of less prevalent topics: By accounting for the uncertainty associated with predictions, the model can effectively capture and consider less prevalent topics or items that might have otherwise been overlooked. This results in a more diverse set of recommendations, as the model is able to recognize and recommend items from a broader range of topics or categories.

Balancing exploration and exploitation: Uncertainty-aware selection helps balance the trade-off between exploration (discovering new items) and exploitation (recommending known, relevant items). By factoring in the uncertainty of predictions, the model is encouraged to explore less certain or less frequently recommended items, thereby promoting diversity in the recommendations.

Robustness to poor calibration: [25] Neural networks can suffer from poor calibration, where incorrect predictions may have high confidence scores. Uncertainty-aware selection addresses this issue by considering the uncertainty of predictions alongside confidence scores, ensuring that the model is not misled by poorly calibrated predictions and can still generate diverse recommendations.

The UPS algorithm is illustrated in Figure 2. By considering both the probability score and confidence thresholds, we derive the selection criterion to construct the pseudo-label set U_p:

$$U_p = \mathbb{1}_{\{v \in \mathcal{U} | s_c(v) > \tau_p\}} + \mathbb{1}_{\{v \in \mathcal{U} | s_c(v) < \tau_n\}} \tag{5}$$

We generate the pseudo labels for U_p by employing the GNN encoder:

$$\hat{\mathbf{y}}^v = \underset{j}{\operatorname{argmax}} f_\theta \left(\mathbf{x}_v\right)_j ; v \in U_p \tag{6}$$

We use the BCE loss to optimize GCN for item label classification:

$$L_{\mathrm{BCE}}\left(\tilde{\boldsymbol{y}}^{(v)}, \hat{\boldsymbol{y}}^{(v)}, \boldsymbol{g}^{(v)}\right) = -\frac{1}{s^{(v)}} \sum_{c=1}^{C} g_c^{(v)} \left[\tilde{y}_c^{(v)} \log\left(\hat{y}_c^{(v)}\right) + \left(1 - \tilde{y}_c^{(v)}\right) \log\left(1 - \hat{y}_c^{(v)}\right)\right] \tag{7}$$

where $g_C^{(v)} \subseteq \{0,1\}^C$ is a binary vector representing the selected pseudo-labels in node v, where $g_c^{(v)} = 1$ when $\tilde{y}_c^{(v)}$ is selected and $g_c^{(v)} = 0$ when $\tilde{y}_c^{(v)}$ is not selected.

In both cases, the selection of high confidence pseudo-labels removes noise during training, allowing for improved performance when compared to traditional pseudo-label methods.

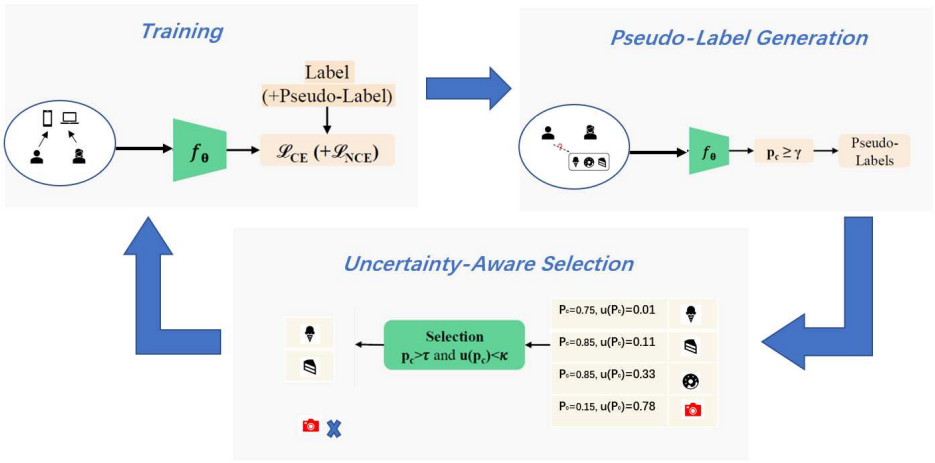

Fig. 2 Descriptions for UPS

3.4 Model Training

When the model is required to distinguish between positive and negative context subgraphs, it is encouraged to learn diverse and complementary features. This can lead to a more comprehensive understanding of the underlying graph structure and improve the diversity of recommendations, avoiding filter bubbles and over-personalization issues [26]. As a result, the contrastive loss is applied and further achieved by employing a discriminator adopted from Li's work [4]. With regard to the discriminator $\mathcal{D}(\cdot)$, we implement it using a bilinear layer:

$$\mathcal{D}\left(\varphi\left(h_v\right), \phi\left(H_{\mathcal{N}_v}\right)\right) = \sigma\left(\varphi\left(h_v\right) B \phi\left(H_{\mathcal{N}_v}\right)^T\right) \tag{8}$$

$$\ell_I = -\frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \left[\log \mathcal{D}\left(\varphi\left(h_v\right), \phi\left(H\right)\right) + \log\left(1 - \mathcal{D}\left(\varphi\left(h_v\right), \phi\left(H\right)\right)\right)\right] \tag{9}$$

Where B represent the learnable parameter, $\mathcal{N}_v$ denote as the generated subgraph, $\varphi(\cdot)$ serve as an MLP encoder for node embedding transformation. $\phi(\cdot)$ is a subgraph encoder, responsible for aggregating embeddings from all nodes within the subgraph to generate a subgraph embedding.

Accordingly, network parameters are updated by:

$$\ell = \ell_I + \beta \ell_{BCE} \tag{10}$$

4 EXPERIMENTAL RESULTS

In our experiments, we carry out extensive evaluations with in-depth analyses on a real-world recommendation system to validate the effectiveness of UGNN in enhancing diversity. In this section, we aim to address the following three research questions:

(RQ1): How does the proposed UGNN model compare to various competitive models in terms of recommendation accuracy in the matching process (refer to Sec. 4.3)?

(RQ2): How does UGNN perform against competitive baselines in terms of recommendation diversity at the element level, list level, and global level (refer to Sec. 4.3)?

(RQ4): How do uncertainty-aware pseudo-label selection influence UGNN recommendation accuracy and diversity (refer to Sec. 4.4)?

4.1 Datasets and Evaluation Metrics

DataSets: We utilise three public datasets in our experiments: Last.fm, Douban-Book and Amazon-CDs [27] to evaluate UGNN. The statistics of the three datasets are listed in Table 1. The benchmark datasets utilized in our experiments are publicly available, encompassing real-world data from various domains, sizes, and sparsity levels. For each user, we randomly select 20% of the rated items as ground truth for testing. The remaining 70% and 10% of the data constitute the training and validation sets, respectively. The split ratio for labeled data is 10%, meaning that out of all interactions in the training set, only 10% are treated as labeled data with the rest being considered as unlabeled data, simulating the real-world conditions where explicit feedback is scarce compared to the vast number of unobserved user-item interactions. The Douban-book dataset is derived from the book domain, while the Last-FM1 dataset is sourced from the music domain. Amazon-CDs is a subset from Amazon-review which is a popular dataset for product recommendations [27] .

Metrics: In all experiments, we evaluate the recommendation accuracy of our model and the baselines using Recall@10 as the metric. However, since accuracy alone does not guarantee satisfactory recommendations, we also assess Serendipity@10 [28], which takes into account the surprise and relevance of a recommendation. It can be computed by [28]:

$$\text{SRDP@k} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \left(\frac{1}{|I_k(u)|} \sum_{i \in I_k(u)} \max\left(P_i(u) - P_i(\mathcal{U}), 0\right) * \text{rel}_i(u) \right) \tag{11}$$

Here, $P_i(u) = \frac{|I_k(u)| - rank_i}{|R_{u,k}| - 1}$ represents the probability of recommending item i to a specific user u, and $P_i(\mathcal{U}) = \frac{D(i)}{\sum_{u \in \mathcal{U}} D(u)}$ represents the approximate probability of recommending that item to any user. We use $D(i)$ and $D(u)$ to denote the degrees of item i and user u in the observed graph, respectively.

4.2 Baseline

To validate the efficacy of our proposed UGNN, we compare its performance against several diversification methods as follows:

NGCF [15]: Neural Graph Collaborative Filtering (NGCF) is a framework exploits the useritem graph structure by propagating embeddings on it. This method achieves the target by leveraging high-order connectivities in the user-item integration graph.

MMR [29]: Maximal Marginal Relevance (MMR) is a pioneer work designed to balance and diversify personalized recommendation lists in order to reflect the users complete spectrum of interests. This method introduced the intra-list similarity metric to evaluate the topical diversity of recommendation lists and implement the topic diversification approach to reduce intra-list similarity.

DGCN [19]: Diversified Graph Convolutional Networks (DGCN) conduct rebalanced neighbor sampling, which down-weights dominant topics and boosts the importance of disadvantaged topics in neighbor nodes. Zhangs work investigate the potential of negative sampling in diversification. The authors propose to choose those similar but negative items, which means items of the same category with the positive sample. By sampling negative items from the positive category, the recommendation model is optimized to distinguish users preference within a category. And those negative items in the same category are less likely to be retrieved, which increases the possibility of recommending items from other more diverse categories.

DGRec [20]: Graph Neural Network for Recommendation with Diversified Embedding Generation (DGRec) aims to enhance diversity in the retrieval stage of the GNN-based recommender system model. It utilize the three modules: submodular neighbor selection to find a subset of diverse neighbors to aggregate for each GNN node, layer attention to assign attention weights for each layer, and loss reweighting to focus on the learning of items belonging to long-tail categories.

Table 1 Features of the considered datasets

Dataset	# User	# Item	# Interaction	Density
Last.fm	1,892	17,047	92,834	0.28%
Douban-Book	13,024	22,347	16,506	0.27%
Amazon-CDs	43,169	35,648	777,426	0.051%

4.3 Comparative Analysis

We compare our method with the baselines in terms of overall performance. To evaluate the performance for Top-K recommendation task, we conduct statistical significance testings which are averaged over 100 runs with random weight initializations when K=10 and K=20. We can reach the same conclusion for other top-N retrievals. As can be observed from Table 2 & 3 and Figure 3:

4.3.1 Recall Performance Analysis

The recall performance, denoted as R@10 and R@20, serves as a primary metric for evaluating the recommendation quality of the models. Our findings are as follows:

Table 2 Recall Performance Comparison (R@10 and R@20)

	Last.fm		Douban-Books		Amazon-CDs	
Method	**R@10**	**R@20**	**R@10**	**R@20**	**R@10**	**R@20**
NGCF	0.1586	0.2104	0.1120	0.1500	0.0778	0.1050
MMR	0.1920	0.2450	0.0638	0.0850	0.0862	0.1150
DGCN	0.1489	0.2000	0.0820	0.1100	0.0859	0.1120
DGRec	0.1650	0.2200	0.0950	0.1250	0.0880	0.1180
UGNN(Ours)	0.1721	0.2300	0.1023	0.1350	0.0932	0.1220

Table 3 Socially Responsible Data Processing Performance Comparison (SRDP@10 and SRDP@20)

	Last.fm		Douban-Books		Amazon-CDs	
Method	**SRDP@10**	**SRDP@20**	**SRDP@10**	**SRDP@20**	**SRDP@10**	**SRDP@20**
NGCF	0.0122	0.0165	0.0774	0.0822	0.0101	0.0135
MMR	0.0175	0.0230	0.0110	0.0150	0.0081	0.0108
DGCN	0.0188	0.0250	0.0084	0.0120	0.0097	0.0129
DGRec	0.0200	0.0270	0.0115	0.0155	0.0090	0.0120
UGNN(Ours)	0.0221	0.0295	0.0132	0.0175	0.0105	0.0140

- On the **Last.fm** dataset, UGNN achieves superior recall rates of 17.21% (R@10) and 23% (R@20), outperforming other methods.
- In the context of the **Douban-Books** dataset, UGNN continues to lead with recall rates of 10.23% and 13.5% for R@10 and R@20, respectively.
- For the **Amazon-CDs** dataset, UGNN's recall rates of 9.32% (R@10) and 12.2% (R@20) further confirm its robust performance across diverse domains.

4.3.2 Diversity Analysis

The result indicated by SRDP@10 and SRDP@20, reflects the model's ability to recommend items while considering diversity implications. Our analysis shows:

- UGNN sets the benchmark with the highest SRDP scores across all datasets. Specifically, it scores 0.0221 (SRDP@10) and 0.0295 (SRDP@20) on Last.fm, 0.0132 and 0.0175 on Douban-Books, and 0.0105 and 0.0140 on Amazon-CDs.
- It is more difficult to balance the two aspects to serve as a greedy algorithms. In the Last.fm dataset, MMR has a higher recommendation accuracy (0.1920) compared to NGCF (0.1586) and DGCN (0.1489), but its serendipity (0.0175) is lower than the proposed UGNN model (0.02219). we can observe that the MMR (Maximal Marginal Relevance), does not consistently achieve a balance between recommendation accuracy (R) and serendipity (SRDP) across the three datasets.

This result demonstrates the effectiveness of our model in recommendation scenarios.

4.4 Ablation Studies

In this section, we present an ablation study to evaluate the effectiveness of the Uncertainty-Aware Pseudo-Label Selection (UPS) component in the Uncertainty-Aware Graph Neural Network (UGNN) model. We compare the performance of UGNN with and without UPS in terms of accuracy (R@10) and diversity (SRDP@10) with the subset from **Douban-Book**.

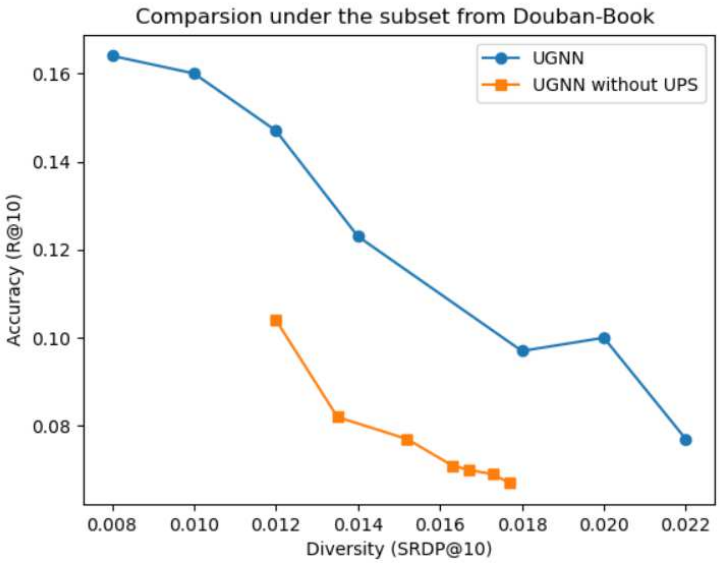

Fig. 3 Comparison of UGNN and UGNN without UPS

Figure 4 illustrates the comparison between UGNN and UGNN without UPS. The x-axis represents the diversity (SRDP@10), and the y-axis represents the accuracy (R@10). The plot shows that the UGNN model consistently outperforms the UGNN without UPS model in both diversity and accuracy across various SRDP@10 values. This indicates that the UPS component in UGNN plays a crucial role in enhancing the model's performance.

The UGNN model exhibits a trade-off between diversity and accuracy, as the diversity increases, the accuracy decreases. This trade-off is consistent with the expectation that as the recommendations become more diverse, the overall accuracy may be slightly compromised. However, the UGNN model maintains a higher accuracy compared to the UGNN without UPS model, which demonstrates the effectiveness of UPS in improving the model's performance.

The UGNN without UPS model shows lower diversity and accuracy than the UGNN model across different SRDP@10 values. This indicates that the absence of UPS in the model results in a less robust and less diverse recommendation system.

The UPS component's ability to account for the uncertainty in predictions and to balance exploration and exploitation makes it an essential part of the UGNN model for achieving better performance.

4.5 Hyperparameter Sensitivity Analysis

UGNN has two hyper-parameters τ_p and κ_p, which control the confidence thresholds for pseudo-labelling and uncertainty threshold, respectively. Figure 4 shows the effect of τ_p and κ_p. In the figure, we can observe that UGNN is highly stable when the value of $\kappa_p \leq 0.09$; In the range from 0.5 to 0.9, UGNN consistently achieves high R@10, and further increases of the threshold lead to predictable performance drops, with increasing levels of noise as the threshold value rises. After selecting uncertainty threshold, confidence thresholds $\tau_p > 0.5$ also contribute to similar performance. It is noteworthy that despite variations in κ_p, the SRDP@10 remains stable, underscoring UGNN's capability to sustain diversity standards irrespective of the hyperparameter settings.

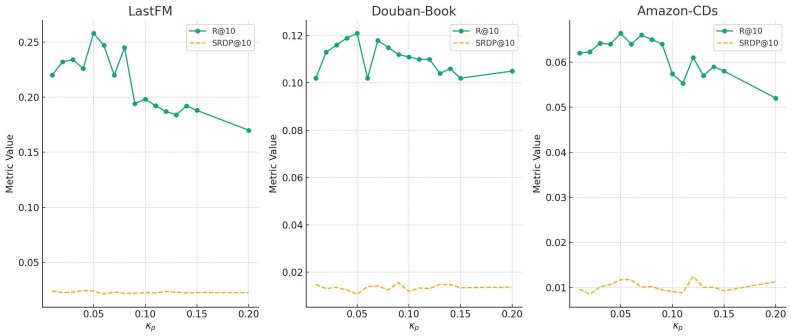

Fig. 4 Sensitivity analysis of τ_p and κ_p

5 Discussion

In this section, we provide an in-depth discussion to distinguish our proposed Uncertainty-Aware Graph Neural Network (UGNN) model from existing recommendation models and to highlight the specific value and research contributions of our work.

6 Conclusion and Future Work

In this paper, we introduced a novel Uncertainty-Aware Graph Neural Network (UGNN) model designed to promote diversity learning in recommendation systems. Leveraging semi-supervised learning, our approach effectively utilized both labeled and unlabeled data to overcome the data sparsity issue common in real-world recommendation scenarios. Comprehensive evaluation on three distinct datasets (Last-FM,

Douban-Book, and Amazon-CD) demonstrated the superiority of UGNN over several baseline models (including DGCN, MMR, and NGCF) in terms of both recall and serendipity, solidifying the value of incorporating uncertainty-aware modeling in recommendation systems.

Looking ahead, our future research will be focusing on handling Large-Scale Graphs. The growth of online platforms and services has led to an increase in the size and complexity of graph data used in recommendation systems. This increase, however, presents a significant challenge for conventional models due to memory constraints and computational limitations. As demonstrated in Figure 5, when dealing with large graphs that cannot fit in GPU memory, training on a CPU can still be a practical solution.

The results of this study provide an important stepping stone in the field of recommendation systems. We anticipate our future work will bring us closer to fully realizing the potential of uncertainty-aware modeling for diverse and personalized recommendations, which will significantly benefit users in their exploration and discovery process in various online platforms.

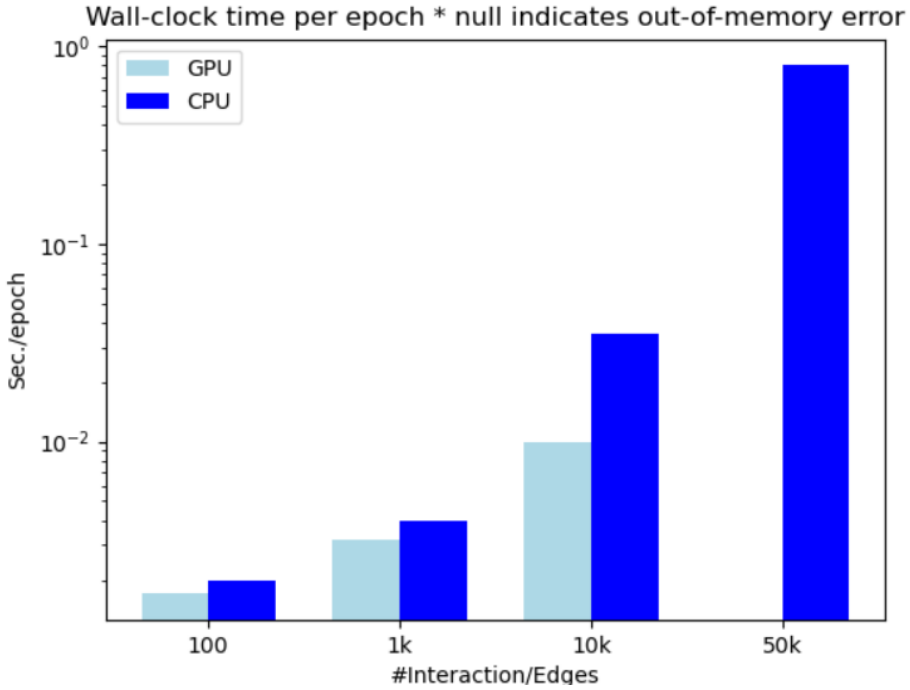

Fig. 5 Wall-clock time per epoc

References

[1] Fout, A., Byrd, J., Shariat, B., Ben-Hur, A.: Protein interface prediction using

graph convolutional networks. Advances in neural information processing systems **30** (2017)

[2] Feng, Y., You, H., Zhang, Z., Ji, R., Gao, Y.: Hypergraph neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 3558–3565 (2019)

[3] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)

[4] Li, Y., Yin, J., Chen, L.: Informative pseudo-labeling for graph neural networks with few labels. Data Mining and Knowledge Discovery **37**(1), 228–254 (2023)

[5] Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. Advances in neural information processing systems **30** (2017)

[6] Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., *et al.*: Graph attention networks. stat **1050**(20), 10–48550 (2017)

[7] Yadav, N., Pal, S., Singh, A.K.: Diversified recommendation using implicit content node embedding in heterogeneous information network. Multimedia Tools and Applications, 1–31 (2023)

[8] Gunawardana, A., Shani, G., Yogev, S.: Evaluating recommender systems. In: Recommender Systems Handbook, pp. 547–601. Springer, ??? (2012)

[9] Liu, J., Zhou, T., Kuscsik, Z., Medo, M., Wakeling, J.R., Zhang, Y.-C.: Solving the apparent diversity-accuracy dilemma of recommender systems (2010)

[10] McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: CHI'06 Extended Abstracts on Human Factors in Computing Systems, pp. 1097–1101 (2006)

[11] De Gemmis, M., Lops, P., Semeraro, G., Basile, P.: Integrating tags in a semantic content-based recommender. In: Proceedings of the 2008 ACM Conference on Recommender Systems, pp. 163–170 (2008)

[12] Ge, M., Delgado-Battenfeld, C., Jannach, D.: Beyond accuracy: evaluating recommender systems by coverage and serendipity, recomm. (2010). Syst

[13] Zhang, Y., Saghdha, D.., Quercia, D., Jambor, T.: Auralist: Introducing serendipity into music recommendation. In: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, pp. 13–22. ACM, ??? (2012)

[14] Gao, C., Zheng, Y., Li, N., Li, Y., Qin, Y., Piao, J., Quan, Y., Chang, J., Jin, D., He, X., *et al.*: A survey of graph neural networks for recommender systems: Challenges, methods, and directions. ACM Transactions on Recommender Systems **1**(1), 1–51 (2023)

[15] Wang, X., He, X., Wang, M., Feng, F., Chua, T.-S.: Neural graph collaborative filtering. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 165–174 (2019)

[16] Abdollahpouri, H., Adomavicius, G., Burke, R.: Controlling popularity bias in learning-to-rank recommendation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, pp. 42–46. ACM, ??? (2017)

[17] Sun, J., Guo, W., Zhang, D., Zhang, Y., Regol, F., Hu, Y., Guo, H., Tang, R., Yuan, H., He, X., *et al.*: A framework for recommending accurate and diverse items using bayesian graph convolutional neural networks. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2030–2039 (2020)

[18] Pal, S., Regol, F., Coates, M.: Bayesian graph convolutional neural networks using node copying. arXiv preprint arXiv:1911.04965 (2019)

[19] Zheng, Y., Gao, C., Chen, L., Jin, D., Li, Y.: Dgcn: Diversified recommendation with graph convolutional networks. In: Proceedings of the Web Conference 2021, pp. 401–412 (2021)

[20] Yang, L., Wang, S., Tao, Y., Sun, J., Liu, X., Yu, P.S., Wang, T.: Dgrec: Graph neural network for recommendation with diversified embedding generation. In: Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, pp. 661–669 (2023)

[21] Sajjadi, M., Javanmardi, M., Tasdizen, T.: Regularization with stochastic transformations and perturbations for deep semi-supervised learning. Advances in neural information processing systems **29** (2016)

[22] Miyato, T., Maeda, S.-i., Koyama, M., Ishii, S.: Virtual adversarial training: a regularization method for supervised and semi-supervised learning. IEEE transactions on pattern analysis and machine intelligence **41**(8), 1979–1993 (2018)

[23] Yi, K., Wu, J.: Probabilistic end-to-end noise correction for learning with noisy labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7017–7025 (2019)

[24] Zhou, T., Wang, S., Bilmes, J.: Time-consistent self-supervision for semi-supervised learning. In: International Conference on Machine Learning, pp. 11523–11533 (2020). PMLR

[25] Rizve, M.N., Duarte, K., Rawat, Y.S., Shah, M.: In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. arXiv preprint arXiv:2101.06329 (2021)

[26] Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)

[27] He, R., McAuley, J.: Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In: Proceedings of the 25th International Conference on World Wide Web, pp. 507–517 (2016)

[28] Silveira, T., Zhang, M., Lin, X., Liu, Y., Ma, S.: How good your recommender system is? a survey on evaluations in recommendation. International Journal of Machine Learning and Cybernetics **10**, 813–831 (2019)

[29] Ziegler, C.-N., McNee, S.M., Konstan, J.A., Lausen, G.: Improving recommendation lists through topic diversification. In: Proceedings of the 14th International Conference on World Wide Web, pp. 22–32 (2005)