Graph Structure Aware Contrastive Knowledge Distillation for Incremental Learning in Recommender Systems

Yuening Wang*
yuening.wang@mail.mcgill.ca
McGill University
Montreal, Canada

Yingxue Zhang
yingxue.zhang@huawei.com
Huawei Noah's Ark Lab
Montreal, Canada

Mark Coates
mark.coates@mcgill.ca
McGill University
Montreal, Canada

ABSTRACT

Personalized recommender systems are playing an increasingly important role for online services. Graph Neural Network (GNN) based recommender models have demonstrated a superior capability to model users' interests thanks to rich relational information encoded in graphs. However, with the ever-growing volume of online information and the high computational complexity of training GNNs, it is difficult to perform frequent updates to provide the most up-to-date recommendations. There have been several attempts towards training GNN models in an incremental fashion to enable faster training times and permit more frequent model updates using the latest training data. The main technique is knowledge distillation, which aims to allow model updates while preserving key aspects of the model that were learned from the historical data. In this work, we develop a novel Graph Structure Aware Contrastive Knowledge Distillation for Incremental Learning in recommender systems, which is tailored to focus on the rich relational information in the recommendation context. We combine the contrastive distillation formulation with intermediate layer distillation to inject layer-level supervision. We demonstrate the effectiveness of our proposed distillation framework for GNN based recommendation systems on four commonly used datasets, showing consistent improvement over state-of-the-art alternatives.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

incremental learning, recommender system, graph neural networks, knowledge distillation

ACM Reference Format:
Yuening Wang, Yingxue Zhang, and Mark Coates. 2021. Graph Structure Aware Contrastive Knowledge Distillation for Incremental Learning in Recommender Systems. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3459637.3482117

*Work done as an intern at Huawei Noah's Ark Lab Montreal Research Center.

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

ACM ISBN 978-1-4503-8446-9/21/11…$15.00
https://doi.org/10.1145/3459637.3482117

1 INTRODUCTION

Personalized recommender systems have been playing an increasingly important role in online platforms and services, aiming to reduce information overload and to cater to the diverse interests of users. Deep learning models are becoming more prevalent in all aspects of recommender system design due to their superiority in end-to-end representation learning [3, 4, 9]. The recent Graph Neural Network (GNN) based recommendation systems [1, 11, 26, 36, 38, 45] have demonstrated superior recommendation performance. However, training an accurate deep learning model requires a vast number of training instances and a has a long convergence time. In particular, training a GNN based recommender system has high computational complexity, introduced by the neighbor sampling and message-passing steps. In practical scenarios, a sliding window mechanism has been used to select a smaller portion of the data (e.g., the most recent 10 days) to train the recommendation system. A large window size makes it very challenging to perform frequent updates to provide up-to-date recommendations that take into account evolving user interests [41]. However, a small window size can lead to catastrophic forgetting, with the recommendation system losing its knowledge of users' long-term preferences. The field of incremental learning has been proposed to tackle this dilemma and combat the catastrophic forgetting issue [2, 17, 18, 22–24, 40]. The two most commonly used techniques are *experience replay* (reservoir sampling) and *knowledge distillation* (regularization) based methods. In this work, we mainly investigate the knowledge distillation approach.

Because of the unique relational information in GNN models, there have been several attempts to design incremental learning techniques specially tailored for graph-structure data [34, 43]. LSP [43] introduces a local structure preserving module that explicitly accounts for the topological semantics of the teacher model. A recent work called GraphSAIL [41] proposes the first incremental learning framework for GNN-based recommender systems. It explicitly preserves each node's local and global structure by matching the respective structural distributions between teacher and student models. However, GraphSAIL inherits on of the shortcomings of the vanilla knowledge distillation techniques, in that it struggles to capture the correlations and higher-order dependencies between output dimensions when transferring knowledge from the teacher to the student model [31]. A recent contrastive representation distillation formulation has been proposed to tackle the above limitation and has been used in model compression and cross-domain transfer tasks in the field of computer vision [31]. Beyond this, GraphSAIL only distills the embedding layer for users and items. There is no attempt to preserve the intermediate layers' representations, which

usually have wider receptive fields (spanning more hops of neighborhood information). Usually, injecting intermediate layer-level supervision from the teacher to student model can better preserve the model properties [28, 39].

In this work, we propose a novel graph structure aware contrastive knowledge distillation objective for incremental learning in recommender systems. The contrastive learning objective is well-tailored for graph-based recommendation models with a specific design of the contrastive loss positive and negative sample pairs using graph structure information. In order to better distill the information from multiple receptive fields, capturing different neighborhood scales on the graph, we combine the contrastive distillation formulation with intermediate layer distillation to inject layer-level supervision. To further incorporate the multi-relation information coming from the additional user-user and item-item context graphs, we extend our designed contrastive distillation objective to support multiple graph structures. We demonstrate the effectiveness of our proposed novel distillation framework for two commonly used GNN based recommendation systems (PinSAGE [45] and Multi-GCCF [27]) on four commonly used datasets. We observe on average 5%∼10% improvement over alternatives. The additional ablation studies establish the effectiveness of all proposed model components.

2 RELATED WORK

Graph-based Recommender Systems Graphs are a natural tool for modeling rich relational information in the context of recommender systems. Classic works [7, 13, 42] use random-walk on the user-item interaction graph to compute a similarity score for each pair of user and item. With the recently emerging field of representation learning on graphs [6, 8, 10, 16, 21, 30, 46], Graph Neural Network (GNN) based recommendation systems have shown their superior ability to model user-item interactions and learn from additional contextual graph information [5, 12, 25, 26, 32, 38, 45, 48]. PinSAGE [45] applies a GNN on the item-item graph with mean aggregation to model the similarity between items (Pins) and boards. Knowledge graphs are incorporated to enrich the relation information between items leading to more effective representation learning [33, 37]. Multi-GCCF [27] leverages GNNs to exploit proximity information not only from user-item interaction graphs but also from user-user and item-item graphs separately. It also accounts for the intrinsic difference between user nodes and item nodes by employing different aggregation functions.

Knowledge Distillation for Incremental Learning Knowledge distillation was originally proposed to transfer knowledge between a large and complex (teacher) model into a small (student) model by matching the output logits to reduce the computational cost [14, 35]. Subsequent works focus on further improving the transferability and exploring its application in various domains [15, 20, 28, 29, 31, 39, 43]. One of the interesting applications is to use knowledge distillation to tackle the catastrophic forgetting problem in incremental training [2, 17, 18, 22–24, 40]. There are several recent works aim to tackle the incremental learning challenge specifically for training GNN models to better preserve the structural properties [22, 34, 41, 43, 47]. LSP [43] utilizes a local structure preserving module that explicitly accounts for the topological semantics of the teacher model. [34] proposes a data replaying and model regularization for existing neighborhood pattern consolidation. GraphSAIL [41] proposes the first incremental learning framework for GNN-based recommender systems. It aims to explicitly preserve each node's local and global structure by extracting the structure information as a distribution and minimizing the distance between the teacher and student models to enable structure-aware knowledge transfer.

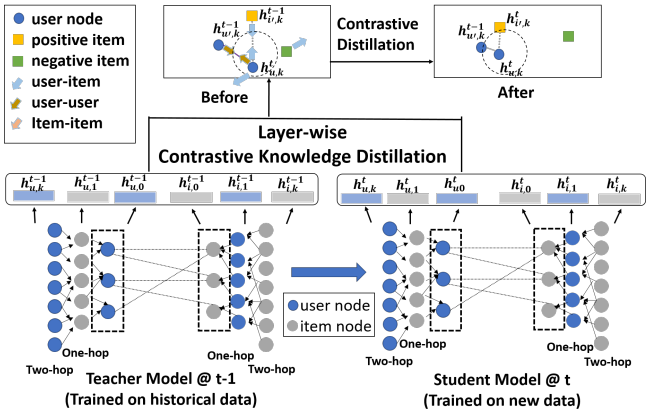

Figure 1: The overall framework of our proposed layer-wise structure-aware contrastive distillation. Contrastive distillation is applied on each layer's representation between student and teacher model. User-item, iter-item and user-user graphs contribute to forming the positive samples.

3 METHODOLOGY

In this section, we introduce our proposed method, *Structure-Aware Contrastive Distillation* with *Layer-wise Distillation* .

3.1 Structure-Aware Contrastive Distillation

In the vanilla KD [10] and its variants, the final objective ψ assumes the output dimensions are independent: $\psi\left(\mathbf{y}^S, \mathbf{y}^T\right) = \sum_i \phi_i\left(\mathbf{y}_i^S, \mathbf{y}_i^T\right)$. Directly minimizing the distance (KL divergence between the probabilistic outputs [10] or the mean squared error between the embeddings [2, 17, 41]) of a teacher and student networks ignores important structural knowledge of the teacher network [31]. A recent contrastive representation distillation formulation has been proposed to tackle the model compression and cross-domain transfer tasks in the CV field [31]. This motivates us to use a novel contrastive distillation objective to better preserve the model parameters as well as the relational information between the *teacher model*, trained on the data acquired from history data and the *student model*, trained on the knowledge acquired from new data.

The contrastive distillation objective aims to maximize a lower bound on the mutual information between the teacher and student representations [19, 31]. In other words, the contrastive objective encourages the teacher and student to map similar inputs (positive pairs) to close representations in the latent space and different inputs (negative pairs) to distant representations. To tailor the original contrastive distillation objective to suit our graph-based recommendation system setting, we construct the positive pairs between the embedding from the *student model* (trained using the

data at time t) and the embedding from the one-hop neighbors of the user-item bipartite graph from the *teacher model* (trained using the data at time $t-1$). The negative samples are randomly sampled from the non-adjacent nodes of the teacher model. For each user, we use the same number of positive samples and negatives samples. Our structure-aware contrastive distillation loss on the user-item bipartite graph is defined as:

$$\mathcal{L}_{const} =$$
$$\frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{-1}{|\mathcal{N}_{UI}^{t-1}(u)|}\sum_{i\in\mathcal{N}_{UI}^{t-1}(u)}\log\frac{\exp\left(\boldsymbol{h}_{u,0}^{t-1}\cdot\boldsymbol{h}_{i,0}^{t}/\tau\right)}{\sum_{\hat{i}\in\mathcal{D}_{UI}^{t-1}(u)}\exp\left(\boldsymbol{h}_{u,0}^{t-1}\cdot\boldsymbol{h}_{\hat{i},0}^{t}/\tau\right)}$$
$$+\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\frac{-1}{|\mathcal{N}_{UI}^{t-1}(i)|}\sum_{u\in\mathcal{N}_{UI}^{t-1}(i)}\log\frac{\exp\left(\boldsymbol{h}_{i,0}^{t-1}\cdot\boldsymbol{h}_{u,0}^{t}/\tau\right)}{\sum_{\hat{u}\in\mathcal{D}_{UI}^{t-1}(i)}\exp\left(\boldsymbol{h}_{i,0}^{t-1}\cdot\boldsymbol{h}_{\hat{u},0}^{t}/\tau\right)},$$

where $\boldsymbol{h}_{u,0}^{t}$ is the embedding for node u at time t, $\mathcal{U}$ is the user node set, $\mathcal{I}$ is the item node set, and $\mathcal{N}_{UI}^{t-1}(u)$ is the neighborhood set of user node u from the user-item interaction graph at time $t-1$, which provides the positive samples. $\mathcal{D}_{UI}^{t}(u)$ is the collection (union) of positive and negatives samples of the user u generated from the user-item bipartite graph from time t. τ is a temperature that adjusts the concentration level.

Modeling the additional relationships between users and items has been demonstrated to be effective in recommender systems [27, 33, 37], leading to more accurate embeddings by enforcing a stronger collaborative signal constraint. This inspires us to design an additional contrastive distillation objective to transfer the proximity information of the user-user and item-item similarity graphs from the teacher to the student model. We adopt the same positive and negative construction strategy as the user-item graph. The mathematical formulation is similar to the above equation. We provide the overall formulation in the following section.

3.2 Layer-wise Structure-Aware Contrastive Distillation

Intermediate Layer Distillation Injecting intermediate layer-level supervision from teacher to student models can improve the transfer of the model's properties [20, 28, 39]. This ensures that not only the final embeddings of the student model are transferred from the corresponding teacher embeddings, but also the representations from the intermediate layers are well preserved. Prior works have proposed to use mean squared error to distill the representation from the intermediate layer or a combination of the intermediate layers if the number of layers between the teacher and student model does not match [20, 28, 39].

Layer-Wise Structure-Aware Contrastive Distillation Our final model design takes advantage of both layer-wise distillation (allowing the representation from different receptive fields, i.e., neighborhood-scales on the graph, to be preserved) and the contrastive objective, aiming to maximize the knowledge transferred from the teacher to the student model. The overall framework of our model design is shown in Figure 1. Our final layer-wise structure-aware contrastive distillation objective is defined as:

$$\mathcal{L}_{lw-const} =$$
$$\frac{1}{K}\sum_{k=0}^{K}\Big(\frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{-1}{|\mathcal{N}_{UI}^{t-1}(u)|}\sum_{i\in\mathcal{N}_{UI}^{t-1}(u)}\log\frac{\exp\left(\boldsymbol{h}_{u,k}^{t-1}\cdot\boldsymbol{h}_{i,k}^{t}/\tau\right)}{\sum_{\hat{i}\in\mathcal{D}_{UI}^{t-1}(u)}\exp\left(\boldsymbol{h}_{u,k}^{t-1}\cdot\boldsymbol{h}_{\hat{i},k}^{t}/\tau\right)}$$
$$+\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\frac{-1}{|\mathcal{N}_{UI}^{t-1}(i)|}\sum_{u\in\mathcal{N}_{UI}^{t-1}(i)}\log\frac{\exp\left(\boldsymbol{h}_{i,k}^{t-1}\cdot\boldsymbol{h}_{u,k}^{t}/\tau\right)}{\sum_{\hat{u}\in\mathcal{D}_{UI}^{t-1}(i)}\exp\left(\boldsymbol{h}_{i,k}^{t-1}\cdot\boldsymbol{h}_{\hat{u},k}^{t}/\tau\right)}$$
$$+\frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{-1}{|\mathcal{N}_{UU}^{t-1}(u)|}\sum_{u'\in\mathcal{N}_{UU}^{t-1}(u)}\log\frac{\exp\left(\boldsymbol{h}_{u,k}^{t-1}\cdot\boldsymbol{h}_{u',k}^{t}/\tau\right)}{\sum_{\hat{u}\in\mathcal{D}_{UU}^{t-1}}\exp\left(\boldsymbol{h}_{u,k}^{t-1}\cdot\boldsymbol{h}_{\hat{u},k}^{t}/\tau\right)}$$
$$+\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\frac{-1}{|\mathcal{N}_{II}^{t-1}(i)|}\sum_{i'\in\mathcal{N}_{II}^{t-1}(i)}\log\frac{\exp\left(\boldsymbol{h}_{i,k}^{t-1}\cdot\boldsymbol{h}_{i',k}^{t}/\tau\right)}{\sum_{\hat{i}\in\mathcal{D}_{II}^{t-1}}\exp\left(\boldsymbol{h}_{i,k}^{t-1}\cdot\boldsymbol{h}_{\hat{i},k}^{t}/\tau\right)}\Big),$$

where $\boldsymbol{h}_{u,k}^{t}$ is the embedding for node u at time t, $\mathcal{N}_{II}^{t-1}(i)$ is the neighborhood set of user i from the item-item similarity graph at time $t-1$. $\mathcal{N}_{UU}^{t-1}(u)$ is the neighborhood set of user u from the user-user similarity graph at time $t-1$. Both $\mathcal{N}_{UU}^{t-1}(u)$ and $\mathcal{N}_{II}^{t-1}(i)$ provide positive samples for the contrastive objective.

4 EXPERIMENTS

4.1 Datasets

To evaluate the effectiveness of our framework, we apply it to the following datasets: **(1) Gowalla** is a real-world dataset collected from users' checking-in history; **(2) Yelp** is a real-world dataset provided by the Yelp mobile app. We use the most recent 5-years of data for training. **(3) Taobao2014** is a real-world dataset collected from users-commodities behavioral data on Alibaba's mobile commerce platforms in 2014. These datasets vary in size, time-span, domain, and sparsity. To mimic a real-world incremental learning scenario, we split all datasets over absolute timestamps. We filtered duplicated interactions and nodes with less than ten records, following the same procedure as suggested in [41]. The statistics of the processed datasets are listed in Table 2.

4.2 Training and Evaluation

We split the data, in chronological order, into a 60% base block and four 10% incremental blocks. The base block is randomly split into training, validation and testing sets. To mimic the real scenario, block t is used as the training set, and half of block t+1 is used for validation, while the other half is used for testing. Recall@20 is calculated for items in test set and is used to evaluate our methods and baselines. Recall@20 is computed as the fraction of positive items among the top 20 items which have the highest preference scores. The average Recall@20 over all consecutive blocks is used as the final evaluation metric.

Base Models We evaluate the performance of our proposed incremental learning framework on two well-known GNN-based recommender models *PinSage [45]* and *MGCCF [26]*. The same base models are adopted in the GraphSAIL experiments [41].

Table 1: The overall performance comparison among our model and baselines. The results are averaged over 3 repeated experiments. Average boost is the relative improvement over the fine-tune result.

Dataset	Algorithm	MGCCF [26]				PinSage [45]			
		Inc.b1	Inc.b2	Inc. b3	Avg. Recall@20	Inc.b1	Inc.b2	Inc. b3	Avg. Recall@20
Gowalla	Fine Tune	0.1159	0.1253	0.1470	0.1294(+0.00%)	0.1138	0.1074	0.1238	0.1150(+0.00%)
	LSP_s	0.1302	0.1349	0.1532	0.1394(+7.73%)	0.1198	0.0832	0.1332	0.1121(-2.52%)
	Uniform	0.1319	0.1397	0.1571	0.1429(+10.43%)	0.1233	0.1289	**0.1397**	0.1306(+13.56%)
	GraphSAIL	0.1339	0.1399	0.1549	0.1429(+10.43%)	0.0970	0.1248	0.1300	0.1173(+2.00%)
	LWC-KD	**0.1372**	**0.1498**	**0.1652**	**0.1507(+16.46%)**	**0.1296**	**0.1425**	0.1375	**0.1365(+18.70%)**
Yelp	Fine Tune	0.0705	0.0638	0.0640	0.0661(+0.00%)	0.0692	0.0614	**0.0561**	0.0622(+0.00%)
	LSP_s	0.0722	0.0661	0.0644	0.0676(+2.27%)	0.0437	**0.0626**	0.0548	0.0537(-13.67%)
	Uniform	0.0718	0.0635	0.0610	0.0654(-1.06%)	0.0676	0.0608	0.0535	0.0606(-2.57%)
	GraphSAIL	0.0714	0.0622	0.0626	0.0654(-1.06%)	0.0670	0.0611	0.0358	0.0546(-12.22%)
	LWC-KD	**0.0732**	**0.0675**	**0.0652**	**0.0686(+3.78%)**	**0.0721**	0.0618	0.0548	**0.0629(+1.13%)**
Taobao2014	Fine Tune	0.0208	0.0112	0.0138	0.0153(+0.00%)	0.0141	0.0092	0.0107	0.0113(+0.00%)
	LSP_s	0.0213	0.0106	0.0138	0.0152(-0.65%)	0.0172	0.0106	0.0107	**0.0129(+14.16%)**
	Uniform	0.0195	0.0127	0.0148	0.0157(+2.61%)	0.0158	0.0110	**0.0117**	0.0128(+13.27%)
	GraphSAIL	0.0222	0.0105	0.0139	0.0155(+1.31%)	**0.0178**	0.0094	0.0091	0.0121(+7.08%)
	LWC-KD	**0.0231**	**0.0152**	**0.0174**	**0.0186(+21.57%)**	0.0162	**0.0114**	0.0105	0.0127(+12.39%)

Table 2: Statistics of evaluation datasets.

Dataset	#Users	#Items	#Interactions	Density (%)	Timespan (Months)
Gowalla	29,858	40,998	1,027,464	0.0839	19
Yelp	40,863	25,338	942,395	0.0910	60
Taobao2014	8,844	39,103	749,438	0.2167	1

Baselines: To demonstrate the effectiveness of our method, we compare it to the similar baselines used in GraphSAIL [41] including *1) Fine-Tune 2) LSP_s* [44], *3) GraphSAIL* [41]. We add a simple reservoir sampling baseline to sample a subset of old data (5%) from a reservoir to tackle catastrophic forgetting problem as *4) Uniform*.

4.3 Comparison with Baselines

From Table 1, we can make the following observations:

1) Our framework outperforms the fine-tune baseline for both base models and all datasets on the average Recall@20 metric. To be precise, it outperforms on Gowalla by 16.46% with MGCCF and 18.70% with PinSage; on the Yelp dataset, it outperforms by 3.78% with MGCCF and 1.13% with PinSage; on the Taobao2014 dataset, it outperforms by 21.57% with MGCCF and 12.39% with PinSage.

2) Our model outperforms all other methods, which indicates the framework can generalize to different graph-based recommendation systems. The only exception is the experiment on Taobao2014 using PinSage as the base model. In this case, performance of our model is very close to the best results given by LSP_s. All knowledge distillation based methods have similar training efficiency.

4.4 Ablation Study

We study the effectiveness of all proposed model components. We compare our method with: **1) Layer-wise** uses layer-wise structure-aware distillation with an L_2 loss; **2) SGCT** uses contrastive distillation on each node's embedding where a single user-item bipartite graph is used to construct positive samples; **3) MGCT** uses contrastive distillation on each node's embedding where multiple (user-item, user-user and item-item) graphs are used to construct positive samples; **4) GraphSAIL-local** is the baseline local structure distillation proposed in [41], used to validate the effectiveness of the contrastive objective. As shown in Table 3, we observe that the final model that combines all components yields the best average recall@20 for all three datasets; second, each method outperforms GraphSAIL-local, which indicates that each component is effective.

Table 3: Ablation analysis of applying different components with MGCCF base model. Average Boost is the relative improvement over the GraphSAIL-local distillation.

Dataset	Algorithm	Inc. b1	Inc. b2	Inc. b3	Avg. Recall@20
Gowalla	GraphSAIL-local	0.1207	0.1285	0.1478	0.1323(+0.00%)
	Layer-wise	0.1356	0.1425	0.1571	0.1440(+8.84%)
	SGCT	0.1335	0.1392	0.1546	0.1424(+7.63%)
	MGCT	0.1337	0.1412	0.1549	0.1433(+8.31%)
	LWC-KD	**0.1372**	**0.1498**	**0.1652**	**0.1507(+13.91%)**
Yelp	GraphSAIL-local	0.0716	0.0634	0.0635	0.0662(+0.00%)
	Layer-wise	0.0724	0.0672	0.0645	0.0680(+2.71%)
	SGCT	0.0708	0.0648	0.0644	0.0667(+0.76%)
	MGCT	0.0718	0.0664	0.0645	0.0676(+2.11%)
	LWC-KD	**0.0732**	**0.0675**	**0.0652**	**0.0686(+3.63%)**
Taobao2014	GraphSAIL-local	0.0202	0.0103	0.0135	0.0147(+0.00%)
	Layer-wise	0.0213	0.0135	0.0142	0.0163(+10.88%)
	SGCT	0.0208	0.0115	0.0140	0.0154(+4.76%)
	MGCT	0.0228	0.0147	0.0169	0.0181(+23.13%)
	LWC-KD	**0.0231**	**0.0152**	**0.0174**	**0.0186(+26.53%)**

5 CONCLUSION

In this work, we proposed a method which uses layer-wise contrastive distillation for incremental learning in graph-based recommender systems. The contrastive distillation is designed to support multi-graph structure. We conducted experiments over two GNN-based models and three datasets and demonstrated empirically that our proposed method outperforms various baseline methods.

REFERENCES

[1] Rianne van den Berg, Thomas N Kipf, and Max Welling. 2017. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263* (2017).

[2] Francisco M. Castro, Manuel J. Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. 2018. *End-to-End Incremental Learning*.

[3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, et al. 2016. Wide & Deep Learning for Recommender Systems. In *The ACM Recommender Systems conference. DLRS workshop*.

[4] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proc.of the ACM conference on recommender systems*.

[5] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Yihong Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph Neural Networks for Social Recommendation. In *WWW*. ACM, 417–426.

[6] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. 2017. Neural message passing for quantum chemistry. In *International Conference on Machine Learning*. PMLR, 1263–1272.

[7] Marco Gori and Augusto Pucci. 2007. ItemRank: A Random-Walk Based Scoring Algorithm for Recommender Engines. In *Proc. Int. Joint Conf. Artificial Intelligence*. Hyderabad, India.

[8] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[9] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. Deepfm: a factorization-machine based neural network for ctr prediction. In *Proc. Int. Joint. Conf. Artificial Intelligence*.

[10] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Proc. Adv. Neural Inf. Proc. Systems*.

[11] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. *Proc. ACM Int. Conf. Research and Development in Information Retrieval* (2020).

[12] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*. ACM.

[13] Xiangnan He, Ming Gao, Min-Yen Kan, and Dingxian Wang. 2017. BiRank: Towards Ranking on Bipartite Graphs. *CoRR* abs/1708.04396 (2017). arXiv:1708.04396 http://arxiv.org/abs/1708.04396

[14] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv:1503.02531* (2015).

[15] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Neural Information Processing Systems*.

[16] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proc. Int. Conf. Learning Representations*.

[17] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* (2017).

[18] Arun Mallya and Svetlana Lazebnik. 2018. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Conf. on Computer Vision and Pattern Recognition (CVPR)*.

[19] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[20] Peyman Passban, Yimeng Wu, Mehdi Rezagholizadeh, and Qun Liu. 2020. ALP-KD: Attention-Based Layer Projection for Knowledge Distillation. In *Proc. AAAI Int. Conf. Artificial Intelligence*.

[21] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 701–710.

[22] Ruihong Qiu, Hongzhi Yin, Zi Huang, and Chen Tong. 2020. GAG: Global Attributed Graph Neural Network for Streaming Session-based Recommendation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[23] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*.

[24] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. 2017. Incremental Learning of Object Detectors Without Catastrophic Forgetting. In *Int. Conf. on Computer Vision (ICCV)*.

[25] Weiping Song, Zhiping Xiao, Yifan Wang, Laurent Charlin, Ming Zhang, and Jian Tang. 2019. Session-based Social Recommendation via Dynamic Graph Attention Networks. In *Proc. ACM Int. Conf. Web Search and Data Mining*.

[26] Jianing Sun, Yingxue Zhang, Chen Ma, Mark Coates, Huifeng Guo, Ruiming Tang, and Xiuqiang He. 2019. Multi-Graph Convolution Collaborative Filtering. In *IEEE Int. Conf. on Data Mining (ICDM)*.

[27] Jianing Sun, Yingxue Zhang, Chen Ma, Mark Coates, Huifeng Guo, Ruiming Tang, and Xiuqiang He. 2019. Multi-graph convolution collaborative filtering. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1306–1311.

[28] Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression. *arXiv preprint arXiv:1908.09355*.

[29] Siqi Sun, Zhe Gan, Yu Cheng, Yuwei Fang, Shuohang Wang, and Jingjing Liu. 2020. Contrastive Distillation on Intermediate Representations for Language Model Compression. arXiv:2009.14167 [cs.CL]

[30] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*. 1067–1077.

[31] Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2020. Contrastive representation distillation. In *ICLR*.

[32] Rianne van den Berg, Thomas N Kipf, and Max Welling. 2018. Graph Convolutional Matrix Completion. In *Proc. ACM Conf. Knowledge Discovery and Data Mining (DLD)*.

[33] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge graph convolutional networks for recommender systems. In *WWW*. 3307–3313.

[34] Junshan Wang, Guojie Song, Yi Wu, and Liang Wang. 2020. Streaming Graph Neural Networks via Continual Learning. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1515–1524.

[35] Lung-Chuang Wang and Kuk-Jin Yoon. 2020. Knowledge Distillation and Student-Teacher Learning for Visual Intelligence: A Review and New Outlooks. *arXiv: 2004.05937* (2020).

[36] Shoujin Wang, Liang Hu, Yan Wang, Xiangnan He, Quan Z Sheng, Mehmet A Orgun, Longbing Cao, Francesco Ricci, and Philip S Yu. 2021. Graph learning based recommender systems: a review. In *Proc. Int. Joint. Conf. Artificial Intelligence*.

[37] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *KDD*. ACM.

[38] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *Proc. ACM Int. Conf. Research and Development in Information Retrieval*.

[39] Yimeng Wu, Peyman Passban, Mehdi Rezagholizade, and Qun Liu. 2020. Why Skip If You Can Combine: A Simple Knowledge Distillation Technique for Intermediate Layers. In *Proc. Empirical Methods in Natural Language Processing (EMNLP)*.

[40] Ju Xu and Zhanxing Zhu. 2018. Reinforced continual learning. In *Advances in Neural Information Processing Systems*.

[41] Yishi Xu, Yingxue Zhang, Wei Guo, Huifeng Guo, Ruiming Tang, and Mark Coates. 2020. GraphSAIL: Graph Structure Aware Incremental Learning for Recommender Systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2861–2868.

[42] Jheng-Hong Yang, Chih-Ming Chen, Chuan-Ju Wang, and Ming-Feng Tsai. 2018. HOP-rec: high-order proximity for implicit recommendation. In *Proc ACM Conf. Recommender Systems*.

[43] Yiding Yang, Jiayan Qiu, Mingli Song, Dacheng Tao, and Xinchao Wang. 2020. Distilling Knowledge from Graph Convolutional Networks. *Conf. on Computer Vision and Pattern Recognition (CVPR)* (2020).

[44] Yiding Yang, Jiayan Qiu, Mingli Song, Dacheng Tao, and Xinchao Wang. 2021. Distilling Knowledge from Graph Convolutional Networks. arXiv:2003.10477 [cs.CV]

[45] 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 *Proc. ACM Conf. Knowledge Discovery and Data Mining*.

[46] Yingxue Zhang, Soumyasundar Pal, Mark Coates, and Deniz Üstebay. 2019. Bayesian graph convolutional neural networks for semi-supervised classification. In *Proc. AAAI Int. Conf. Artificial Intelligence*.

[47] Fan Zhou, Chengtai Cao, Ting Zhong, Kunpeng Zhang, Goce Trajcevski, and Ji Geng. 2020. Continual Graph Learning. *arXiv:2003.09908* (2020).

[48] Hongmin Zhu, Fuli Feng, Xiangnan He, Xiang Wang, Yan Li, Kai Zheng, and Yongdong Zhang. 2020. Bilinear graph neural network with neighbor interactions. In *Proc. Int. Joint. Conf. Artificial Intelligence*, Vol. 5.