Mid term Report - RecPress

Deepak
MA20BTECH11019
ma20btech11019@iith.ac.in

Tapishi Kaur
MA20BTECH11017
ma20btech11017@iith.ac.in

Safder Shakil
EM23MTECH11007
em23mtech11007@iith.ac.in

Abdul Waris
CE22RESCH11001
ce22resch11001@iith.ac.in

1. Problem Statement

The problem at hand is the inefficiency of contemporary recommender systems, which often demand substantial computational resources for high-accuracy recommendations. This inefficiency poses a significant challenge, particularly in resource-constrained environments and edge devices, where such systems struggle to operate effectively. The main objective of RecPress: Knowledge Distillation for Efficient Recommender Systems project is to employ knowledge distillation techniques to address this problem, with the aim of developing efficient recommender systems that balance accuracy and computational efficiency, thereby making them suitable for broader deployment.

2. Word Done after PPR

Prior to the submission of the Preliminary Project Report (PPR), our initial focus revolved around formulating a concise problem statement and gaining a comprehensive grasp of the fundamental concepts underlying our research. In the PPR document, we have provided an overview of recommender systems, delved into the principles of knowledge distillation, conducted a thorough literature survey concerning various recommender system architectures, and commenced an exploration of knowledge distillation techniques applicable to recommender systems.

Within the scope of the present Mid-term Report (MTR), we intend to encapsulate the cumulative knowledge we have acquired regarding knowledge distillation techniques for recommender systems. Additionally, we will elucidate the outcomes derived from our implementation and experimentation with one of these techniques. The MTR will underscore the forthcoming phases and milestones in our project, outlining our planned course of action as we move forward.

3. Literature Survey

The papers we studied further are the follow up papers by the authors of DE-RRD framework paper, which we studied and summarised in our PPR.

3.1. Topology Distillation for Recommender System

The paper introduces a general topology distillation approach for Recommender Systems (RS), which guides the student's learning by using the topological structure based on relational knowledge in the teacher's representation space. Two topology distillation methods are proposed: 1) FTD (Full Topology Distillation), which transfers the complete topology. FTD is suitable when the student has the capacity to learn all the teacher's knowledge. 2) HTD (Hierarchical Topology Distillation), which transfers the decomposed topology hierarchically. HTD is employed in scenarios where the student has limited capacity compared to the teacher. Extensive experiments on real-world datasets consistently demonstrate that the proposed approach outperforms state-of-the-art competitors in RS. The paper suggests potential directions for advancing the topology distillation approach, including investigating layer selection and simultaneous distillation from multiple layers, extending topology distillation across different base models, and utilizing prior knowledge of user/item groups to improve the method's effectiveness by providing better supervision on relational knowledge.

3.2. Distillation from Heterogeneous Models for Top-K Recommendation

Knowledge distillation methods for recommender systems focus on distillation from a homogeneous teacher that has the same model type to the student model. Distillation from heterogeneous teachers, which have distinct architectures and learning objectives to the student model, is an even more complex and less explored problem.

Kang et al (2023) proposed a new KD framework for recommender systems, called HetComp, that is specifically designed to distill from heterogeneous teachers. HetComp addresses the challenges of distilling from heterogeneous teachers by using a curriculum learning approach, a dynamic knowledge construction approach, and an adaptive knowledge transfer approach.

Curriculum learning HetComp uses a curriculum learning approach to guide the student model to learn the easy-to-hard knowledge from the heterogeneous teachers. In the curriculum learning stage, HetComp generates a sequence of training data for the student model, starting with easy examples and gradually increasing the difficulty of the examples as the student model learns. The easy examples are generated by selecting items that are highly ranked by the heterogeneous teachers. The difficulty of the examples is gradually increased by selecting items that are less highly ranked by the heterogeneous teachers.

Dynamic knowledge construction HetComp uses a dynamic knowledge construction approach to provide the student model with progressively difficult ranking knowledge. In the dynamic knowledge construction stage, HetComp generates new training data for the student model based on its current learning state. The new training data is generated by selecting items that are difficult for the student model to rank correctly. This ensures that the student model is always learning the most challenging knowledge.

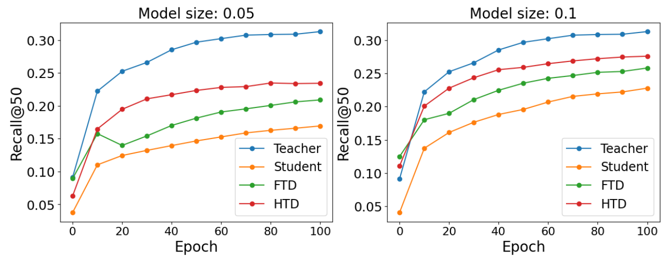
Adaptive knowledge transfer In adaptive knowledge transfer stage, HetComp adjusts the distillation objective according to the student's learning state. This ensures that the student model learns the ranking knowledge from the heterogeneous teachers in a gradual and effective manner.

HetComp framework can compress the ensemble knowledge of heterogeneous recommendation models into a lightweight model, which inturn reduce the huge inference costs while retaining high accuracy. The paper provided extensive experiments showing that HetComp significantly improves the distillation efficacy and the generalization of the student model. Based on its great compatibility with existing models, HetComp framework can be a solution for the accuracy-efficiency trade-off of the recommender systems.

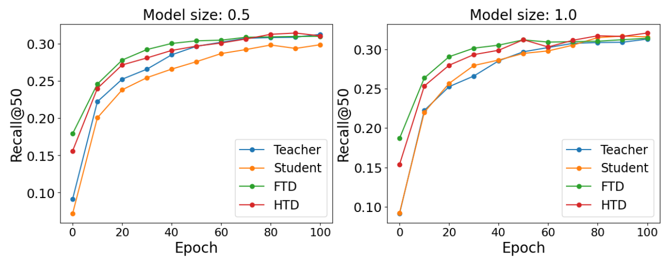
4. Code explanation and Replication of Results from Topology Distillation

Setup

This section sets the random number generators, loads the dataset, and creates the training and test data loaders.



(a) Recall value for models with 10 and 20 parameters respectively



(b) Recall value for models with 100 and 200 parameters respectively

Hyperparameters

This section sets the hyperparameters for training the models, including the learning rate, batch size, regularization coefficient, and maximum epoch.

Training

The selection of hyperparameters must take into account factors such as the dataset, the base model (BPR in our case), the capacity gap, and the specific layer chosen for the distillation process. For our experimentation, we have opted for the CiteULike dataset.

Evaluation

This section evaluates the three models on the test set and reports the results.

We verify the claims made in paper:

1. When the capacity of the student model is highly limited, the student model learns best with HTD.

2. As the capacity gap between the teacher model and student model decreases, the student model takes more benefits from FTD.

Model Descriptions

The baseline model is a simple recommendation model that uses only the user-item rating matrix to predict ratings.

The FTD model is a topology distillation model that learns to mimic the topology of a teacher model.FTD transfers the full topology, and it is used in the scenario where the student has enough capacity to learn all the teacher's knowledge. The HTD model is a hierarchical topology distillation model that learns to mimic the topology of a teacher model at multiple levels. HTD transfers the decomposed topology hierarchically, and it is adopted in the classical KD scenario where the student has a very limited capacity compared to the teacher.

The code snippet also includes a dictionary called `history_dict`. This dictionary stores the training and validation losses of the three models for different student dimensions.

To replicate the results reported in the paper, we use the Python implementation of topology distillation provided in the following Colab notebook:https://colab.research.google.com/drive/1zKAiw6879ek9zLmp4YGgP_Nr5TfR7vRR?usp=sharing

5. Future plans

Looking ahead, the future scope of our work encompasses couple of critical aspects. Firstly, we will work on expanding the dataset, incorporating more diverse real-world examples to further validate the effectiveness of the topology-based knowledge distillation approach. Additionally, we will delve deeper into the existing literature, seeking opportunities to optimize and refine the codebase, ensuring that we stay aligned with the latest advancements in the field. Our goal is to refine our methodology, explore variations, and evaluate its performance across different domains and use cases.

6. References

1. Topology Distillation for Recommender System SeongKu Kang, Junyoung Hwang, Wonbin Kweon, Hwanjo Yu

2. Distillation from Heterogeneous Models for Top-K Recommendation SeongKu Kang, Wonbin Kweon, Dongha Lee, Jianxun Lian, Xing Xie, Hwanjo Yu

3. DE-RRD: A Knowledge Distillation Framework for Recommender System SeongKu Kang et al (2020).

4. Neural Collaborative Filtering (2017) Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, Tat-Seng Chua

5. Wide & Deep Learning for Recommender Systems (2016) Heng-Tze Cheng et al.

6. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction (2017) Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, Xiuqiang He

7. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer (2019) Fei Sun et al.

8. Distilling the Knowledge in a Neural Network (2015) Geoffrey Hinton et al

9. Knowledge Distillation: A Survey (2021) Jianping Gou et al.

10. A novel Enhanced Collaborative Autoencoder with knowledge distillation for top-N recommender systems. Pan, Yiteng et al. Neurocomputing 2019

11. Scene-adaptive Knowledge Distillation for Sequential Recommendation via Differentiable Architecture Search (2021) Lei chen et al.