Exploring the Benefits of Multi-Task Learning for Session-Based Recommendation: A Tenrec Dataset Study

Masood Ahmed - 20379714 , Wilson Chau -20378374 , Liangjin Lin -20164732 , Rex Yang -20198864

Abstract—Session-based recommendation systems use users' previous or current behavior in a sequence of actions, known as a "session," to provide personalized suggestions. However, these systems may not always yield satisfactory results, particularly in finding cold-start users. To address this issue, we propose a combined method that uses both session-based recommendation and multi-task learning algorithms to obtain more accurate predictions. On the other hand, multi-task learning (MTL) involves learning multiple related tasks simultaneously to maximize performance. This method has better performance and efficiency when the tasks are related. On the other hand, the session-based recommendation is widely used in e-commerce or multimedia streaming services to predict users' next preferences based on their previous interactions in sessions. Combining the advantages of session-based recommendation and multi-task learning, we aim to compensate for each other's disadvantages and make more accurate predictions. We propose to test this combined approach on the Tenrec dataset produced by Tencent. Our proposed method may have better performance than using these algorithms separately.

Index Terms—Group 2, Multi-task learning, Session Based Recommendation

1 Introduction

The amount of information on the Internet is growing rapidly with an estimated 328.77 million terabytes of information created and shared daily [6]. With such a heavy information load, a precise recommendation is needed for users browsing online in various applications, including search engines, shopping webpages, news, and media.

Firstly, multi-task learning (MTL) is a method of learning multiple tasks simultaneously while maximizing the performance on one or all of them [Tenrec A large scale...]. It contrasts with traditional single-task learning that every model handles one task separately. MTL models usually have better performance and efficiency than single-task learning when the tasks are related to each other. MTL models aim to learn the generalized features which are meaningful to all the tasks. By understanding the generalized features, the model could be more powerful in capturing the essential features, as well as neglecting the noises. However, MTL models are not guaranteed to outperform all equivalent single-task models. When the tasks are not related, or the information learned has contradictions. It is crucial to analyze the underlying principles of different tasks before adopting MTL.

Session-based recommendation (SBR) is widely used in e-commerce or multimedia streaming services that aim to predict users' following preferences regarding their previous interactions in sessions while there is no historical information about them [1]. To boost the performance of SBRs, the idea of combining multi-task learning with session-based recommendation is the target to explore.

- *Masood, Wilson, and Liangjin are from Faculty of Computing at Queen's University and Rex is from Faculty of Commerce 22kgt,rex.yang,18ll51,22xl31@queensu.ca*

Owing to the nature of these algorithms, the intuition is to combine the advantage of session-based recommendation that it considers the session data and the nature of multi-task learning that all tasks share the same information across the different tasks to learn the generalized representation of the data. If these two algorithms can be used jointly, it may compensate for each other's disadvantages and make an overall more accurate prediction based on the same dataset Tenrec to investigate the overall performance.

2 Related Work

In this section, we will first review related work on session-based recommendation systems and multi-task learning algorithms used during our project's development.

2.1 Session-Based Recommendation with Graph Neural Networks (SR-GNN)

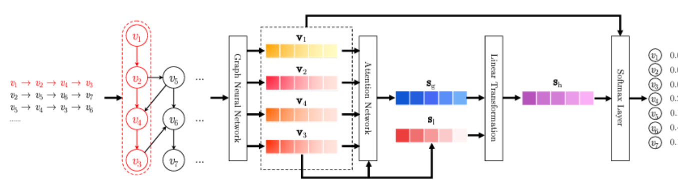

Figure 1: The workflow of the proposed SR-GNN method.

Although session-based recommendation systems have all the advantages described above, they still have limitations. For example, the algorithm needs an adequate amount of user behavior within a session to make an accurate prediction. In addition, sessions are mostly anonymous and numerous, and the user behavior implicated in session clicks is often limited. What's more, from previous works, it is known that patterns of item transitions are an important part of the SBRs, but often the SBR methods always

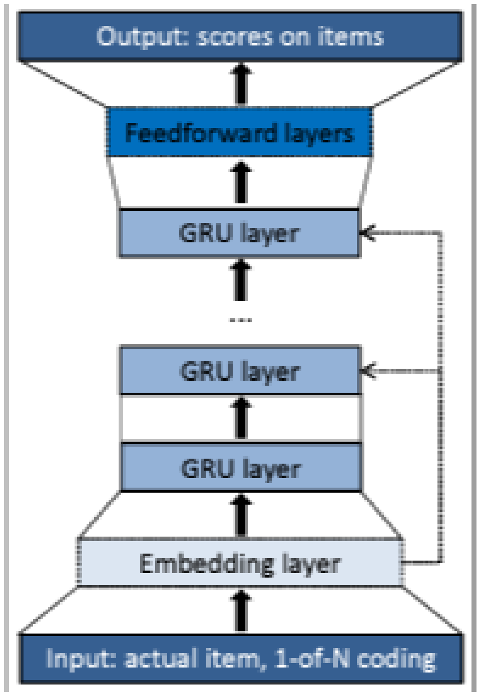

Fig. 1. Basic structure of the Gate Recurrent Unit for Recommendations (GRU4Rec) model

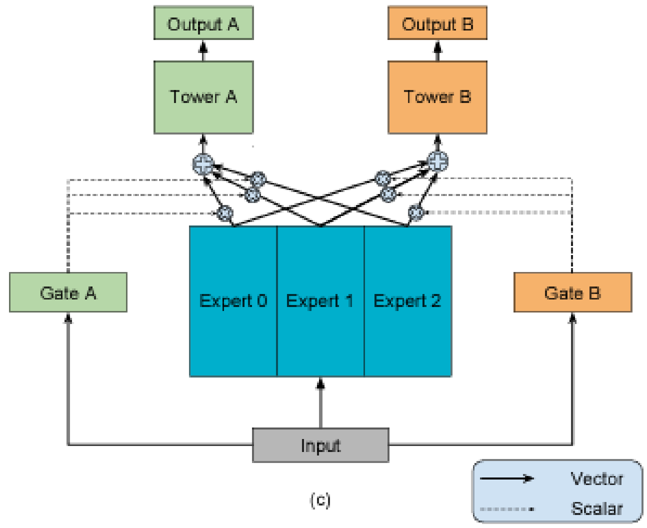

Fig. 2. Basic Structure of the Multi-gate Mixture-of-Experts (MMoE) model

model the single-way transitions among the contexts. Thus, session-based recommendation with graph neural networks is proposed to explore the transitions among items and therefore make further accurate predictions of user interaction [citation]. For the SR-GNN model, let V = v1, v2,..., vm where the set V denotes all unique items involved in all the sessions, and an anonymous session sequence s can be denoted as a list where s = [vs.,1, vs.,2, . . . , vs.,n] which is ordered by timestamps, with vs., i V represents a clicked item of one user within their session. The predicted next-click item will be form vs. n+1 for the session s. A session graph will then be constructed with each node representing an item vs. i V. Each session graph will be processed one by one, and the node vector will be obtained through a gated graph neural network. Then item embeddings will be learned on session graphs to generate the prediction through the Back-Propagation Through Time (BPTT) algorithm to train and obtain the final output model [1].

2.2 Gated Recurrent Unit for Recommendations (GRU4Rec)

Gated Recurrent Unit for Recommendations (GRU4Rec) is a session-based recommendation algorithm that uses Gated Recurrent Units (GRUs) as the core component of the Recurrent Neural Network (RNN) structure [3]. The algorithm consists of mainly four components: Input, RNN with GRU layers, output layer and loss function. The input of GRU4Rec is the actual state of the session while the output is the item of the next user click-interaction in their session. The most important part of the architecture is the GRU layer; the RNN is composed of multiple GRU layers that can capture dependencies and patterns in sessions. By the nature of RNN, the hidden GRU layers are updated based on the result and input from the previous state, which benefits the entire process by considering all data from the session. Lastly, GRU4Rec utilizes a ranking loss function to optimize the pairwise ranking of items within each individual session [2,3].

2.3 Multi-gate Mixture-of-Experts (MMoE)

Multi-gate Mixture-of-Experts (MMoE) is a novel multi-task learning approach that explicitly learns to model task relationships from data. It is driven from the Mixture-of-Experts (MoE) structure to multi-task learning by sharing expert submodels across all tasks [4]. The multi-gate mixture-of-experts (MMoE) builds on the MoE algorithm while having a gating network trained to optimize each task. The fundamental structure of MMoE is derived from the Shared-Bottom multi-task DNN structure, where input is passed into a shared layer across all the tasks and then fed into different task towers.

The mixture of Experts (MoE) is a machine learning algorithm that combines predictions of multiple specialized models called “experts” to make the output predictions. It involves multiple “experts” to keep the focus on predicting the correct answer for the cases where it is already doing better than the other experts (specialization).

In this scenario, each “expert” is a feed-forward network, as shown in Figure 3. Then gating networks are introduced for each task in the model to take the input features and output SoftMax gates, assembling the experts with different weights, thus allowing different tasks to utilize experts differently. After this, the results from assembled experts are passed into the task-specific tower structure, where gating networks for various tasks will learn different mixture patterns of experts assembling, thus capturing the task relationships [4].

The key idea of MMoE is to guide the shared bottom part to allow partial sharing across tasks. To do this, we split the shared bottom part into smaller into smaller expert networks. The output of the networks will be weighted and summed up together and passed into another neural network (tower), and the output will be the predicted element of the task (predicted like, predicted share, etc.).

3 METHODOLOGY

The problem we would like to explor is to combine session-based recommendation with multi-task learning

3.1 SR-GNN model

The group first tried the SR-GNN model, however, the existing code for the model is non-executable. Therefore, the SR-GNN model is abandoned.

3.2 MMoE and GRU4Rec

Judging from the nature of the session-based recommendation and multi-task learning, or GRU4REC and MMoE to be specific, there are several potential benefits of combining them:

Enhanced information capturing and learning: MMoE is a technique that focuses on modeling and capturing diverse and complicated relationships in the input dataset through the featuring "experts". In addition, GRU4Rec symbolizes itself with the ability to capture long-range dependencies in each user session in the dataset using GRU layers and RNN. If these two algorithms can be combined, it is possible to produce an even more powerful model that takes different angles on the dataset with efficiency from multi-task learning and the consideration of session data from session-based learning.

To combine these two algorithms, the following procedure is taken:

1) Process the dataset. The dataset preprocessing follows the same step as the paper "Tenrec: A Large-scale Multipurpose Benchmark Dataset for Recommender Systems", where we keep the most recent 30 click-interactions from user history (keep the most 30 recent one if session exceeds the length of 30, or add padding of 0s for those sessions are shorter than 30, we discard any sessions with a length short than 10). Following the same technique, we keep the last item in the session for testing, the second to last for validating and the rest for training [5].
2) Implement GRU4Rec algorithm to capture the dependencies and patterns within each session in the dataset. The changes occur before feeding the session embedding data into the GRU output layer to softmax layer for a result, we use the current processed data and feed into the MMoE model.
3) The later gating networks from MMoE model will then take the GRU output of the session data and produce a set of weights for later computation with the experts.
4) The process should then go to a softmax layer to compute the probability of each clicking on each possible item, and then be ranked based on the likelihood of them happening in the next click-interaction. Model training and validation.

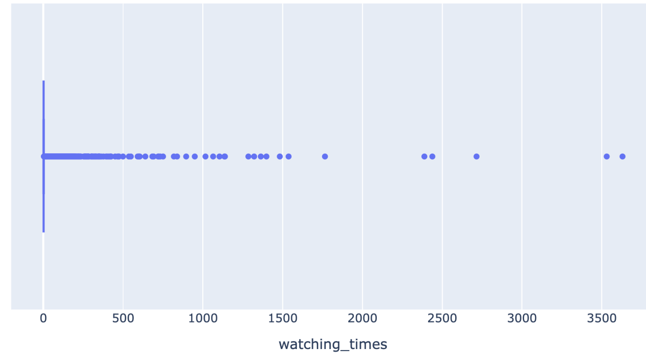

Fig. 3. Figure 4: Boxplot for number of times QB-video being watched by selected users

5) The result will then be compared with the baseline benchmark from the paper "Tenrec: A Large-scale Multipurpose Benchmark Dataset for Recommender Systems" to see if improvements are found.

4 DATASET

TABLE 1
Data Statistics

Name	QK-video	QK-article	QB-video	QB-article
#users	5,022,750	1,325,838	34,240	24,516
#items	3,753,436	220,122	130,637	7,355
#click	142,321,193	46,111,728	1,701,171	348,736
#like	10,141,195	821,888	29,687	/
#share	1,128,312	591,834	2,541	/
#follow	857,678	62,239	2,487	/
#read	/	44,228,593	/	/
#favorite	/	316,627	/	/
#exposure	493,458,970	/	2,422,299	/
avg #clicks	28.34	34.78	49.69	14.22

The data in our dataset consists of feedback behaviors on various kinds of content and user profiles. These were collected from two different feed recommendation platforms of Tencent. These two platforms are QQ Bow and QQ Kan. The data are collected from 81 days (about 2 and a half months)' period from September 17th, 2021, to December 7th, 2021.

Tencent was founded in 1998 in Shenzhen, China. It is an internet and technology company, focusing on technology and entertainment and. Some of their services and product includes but are not limited to social media, FinTech, cloud computing and video game.

Dataset includes user feedback from four scenarios, QK-videos, QK-Article, QB-Video, and QB-Article. In addition, user profile and usage behavior are also included. There are 142 million clicks, 10 million likes, 1 million shares, and 860,000 follows were recorded. In addition, it also contained 3.75 million of videos as well as its type of feature. More importantly, basic demographic and usage data were also extracted, such as: Age, Gender, Watching Time/Behavior. In addition, the dataset also include the negative feedback, it refers to the time that when the user was presented with article or video and video, they did not choose to view it. Looking at the dataset about QB-video's distribution of times, a skewness to the right is observed. Majority of the video in QB-video platform are being watched under 500 times by selected users. (Figure 3).

The dataset has the following features:

Multiple categories for positive user feedback. One of the features of the Tenrec dataset is it contains multiple positive user feedback, which makes itself stand out from other datasets. In QK-video and QB-video files, the positive user feedback includes likes, shares and follows besides the explicit interaction like click-action. By having extra positive feedback, we can derive more from the dataset just by their semantic meaning (like, share, follow) to denote the users' preferences directly. Which serves the purpose well with multi-task learning models.

Timestamp information removed. The other thing we should take note of is that the timestamp information has been removed from the dataset. However, the time information is kept in place with a different form of representation: the user interaction behaviors are presented according to the time order.

5 Experiments and Results

Result obtained from SR-GNN model (Failed)

Since the original model was found non-funcitonal, we had to abandon it to start all over again.

Result obtained from combing GRU4Rec with MMoE (incomplete)

The group is unable to complete the combined model since the problem with the previous model was found at a very late stage of the project.

6 Group Member Contributios

All of our group member has made considerable contribution to the final product of this project. Everyone contributed to the project from different perspective. Masood mainly contributed toward the code testing, Wilson coded for SR-GNN model and little on the report. Liangjin diligently worked in both coding area(architecture designing and coding for SR-GNN, GRU4Rec MMoE) and presentation. Liangjin also involves in writing the technical part of the project. Rex contributed toward the project by using visualization to make powerpoint and speak for the majority of the slide. He also created do some data visualization and convert final report to LaTex.

7 Replication Package

GitHub available at: https://github.com/AdnumaCLin/CISC-372-Group_2.git

8 Conclusion and Future Work

In conclusion, we have proposed a combined approach that uses both session-based recommendation (SBR) and multi-task learning (MTL) to provide more accurate predictions in recommendation systems. As the amount of information on the internet is rapidly increasing, a precise recommendation system is necessary to aid users in browsing various applications.

While SBR is effective in predicting users' next preferences based on their previous interactions in sessions, it has limitations, such as the need for enough user behavior within a session to make accurate predictions. On the other hand, MTL is a more efficient and effective method of learning multiple tasks simultaneously, but its prediction quality may suffer in certain scenarios.

To overcome the limitations of SBR and MTL, we proposed a combined approach that uses both algorithms. The intuition is to combine the advantage of SBR, which considers the session data, and the nature of MTL, which learns generalized representations of the data across different tasks. Our proposed approach compensates for each algorithm's disadvantages, leading to a more accurate prediction based on the same dataset.

We reviewed related works on session-based recommendation systems, including SR-GNN, which models item transitions among contexts, and MMoE, which explicitly learns to model task relationships from data. MMoE builds on the MoE algorithm and uses gating networks to take input features and output SoftMax gates to assemble experts with different weights, thus allowing different tasks to utilize experts differently.

Overall, our proposed combined approach can improve the performance of recommendation systems and provide more accurate recommendations to users. However, further research is needed to evaluate its effectiveness in real-world scenarios.

Based on the research questions that we had, there are many possibilities that we can dive into regarding future work. One such scope can be to investigate the impact of different parameters on the performance of the SBR algorithms. For example, the impact of the session length, size of the item embedding, regularization parameter, and the number of neighbors used in collaborative filtering can be evaluated. The sensitivity analysis can be conducted on the Tenrec dataset suite, and the performance can be evaluated using standard evaluation metrics.

While another possibility can be to compare the performance of a single SBR algorithm with a hybrid approach that combines multiple SBR algorithms, the proposed approach can be assessed using the Tenrec dataset suite. The performance can be evaluated using standard evaluation metrics. The complexity and computational cost of using a hybrid approach can also be evaluated.

Overall, the proposed future work can enhance the performance of Session-Based Recommendation (SBR) algorithms by investigating Multi-Task Learning (MTL), sensitivity analysis of parameters, and comparing the efficiency of single SBR algorithm and hybrid approaches. The proposed experiments can be conducted using the Tenrec dataset suite, which is a suitable dataset for studying the task of SBR algorithms.

9 Response to Presentation Questions

Q1: what is the negative feedback in this dataset? So you have considered them?

Negative feedback in this dataset means "with exposure but no action" (only read the article without reaction – no like, share, or follow). We are no longer considering the negative feedback, but it was something we thought of using.

Q2: What are the statistics of the overlapping users?

There are 269,207 users between QK-video and QK-article, 3,261 users between QK-video and QB-video, and 58 between QK-video and QB-article. Regarding overlapped items, 78,482 videos are overlapped between QK-video and QB-video.

Q3: what's this? Is this the baseline performance? How was MTL formed on this dataset/task? Why should this be considered as a baseline? i think the baseline should be the single-task session-based recommendation. (experimental result slide)

This is the result of MTL algorithms running their sample code with the first 50,000 rows of data. We planned on running the baseline benchmarks of session-based recommendation and multi-task learning algorithms with their first 50,000 rows of data. After a successful implementation of the combined algorithm, we compare the result to them. The baseline can be a single-task session-based recommendation for sure, back then we were still trying to figure out the SR-GNN model, so we didn't have that yet.

10 Reference

[1] Yeganegi, Reza, and Saman Haratizadeh. "Papers with Code - Star: A Session-Based Time-Aware Recommender System." STAR: A Session-Based Time-Aware Recommender System — Papers With Code, 2022. https://paperswithcode.com/paper/star-a-session-based-time-aware-recommender.

[2] Tan, Yong Kiam, Xinxing Xu, and Yong Liu. "Improved Recurrent Neural Networks for Session-Based Recommendations." arXiv.org, September 16, 2016. https://arxiv.org/abs/1606.08117.

[3] Hidasi, Balázs, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. "Session-Based Recommendations with Recurrent Neural Networks." arXiv.org, March 29, 2016. https://arxiv.org/abs/1511.06939.

[4] Ma, Jiaqi, Zhe Zhao, Xinyang Yi, Jilin Chen, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed Chi. "Modeling Task Relationships in Multi-Task Learning with Multi-Gate Mixture-of-Experts: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining." ACM Other conferences, July 1, 2018. https://dl.acm.org/doi/10.1145/3219819.3220007.

[5] Yuan, Guanghu, Fajie Yuan, Yudong Li, Beibei Kong, Shujie Li, Lei Chen, Min Yang, et al. "Tenrec: A Large-Scale Multipurpose Benchmark Dataset for Recommender Systems." arXiv.org, October 20, 2022. https://arxiv.org/abs/2210.10629.

[6] Duarte, Fabio. "Amount of Data Created Daily (2023)." Exploding Topics. Exploding Topics, April 3, 2023. https://explodingtopics.com/blog/data-generated-per-day.