Deep Reinforcement Learning based Group Recommender System

Course project for CSE 6240: Web Search and Text Mining, Spring 2021

Zefang Liu, Shuran Wen, Yinzhu Quan

Georgia Institute of Technology

April 26, 2021

Overview

Introduction

- **Problem description:** The group recommender system plays an important role in current web applications to recommend movies, news, songs, games, and other items according to group member preferences. Our project goal is to implement a novel group recommender system which can give the most suitable recommendations for groups.
- **Importance of this project:** Our model is meaningful to those people who want to get recommendations for their groups, such as entertainments with families and travels with friends. This model could help groups select their favorite items according to their common preferences and save their valuable time in the most effective way.

Approaches: Baselines

- **Attentive Group Recommendation (AGREE)** [1]: In the representation layer of this model, an attention mechanism is adopted to represent groups, where group members and items are embedded first, then sent to a neural attention network to get the group embeddings. Later, these group embeddings are stacked with item embeddings in the pooling layer, then sent to hidden layers and the following prediction layer. The neural collaborative filtering (NCF) is used here to learn the group/user-item interactions.
- **Group Information Maximization (GroupIM)** [2]: Data-driven regularization strategies are proposed to exploit both the preference covariance amongst users who are in the same group, as well as the contextual relevance of users' individual preferences to each group. The recommender architecture-agnostic framework GroupIM can integrate arbitrary neural preference encoders and aggregators for ephemeral group recommendation.

Approaches: DRGR

Effectiveness of DRGR: Markov Decision Process (MDP):

- State space $\mathcal{S}$: A state $s_t = [g, h_t] \in \mathcal{S}$ represents the state of a group at time t with its group id g and the browsing history h_t.
- Action space $\mathcal{A}$: An action $a_t \in \mathcal{A}$ is an item recommendation.
- Reward $\mathcal{R}$: A reward $r_t \in \{0, 1\}$ is the group response to one recommendation a_t at the state s_t.
- Transition probability $\mathcal{P}$: The $p(s_{t+1}|s_t, a_t)$ to measures how the environment evolve with the time t.
- Discount factor γ: The $\gamma \in [0, 1]$ valuates future reward.

An agent (recommender) will try to maximize its rewards through interactions with the environment (groups) by finding one policy (recommendation rule) $\pi : \mathcal{S} \rightarrow \mathcal{A}$. This allows DRGR solving this group recommendation problem. **Novelties of DRGR:**

- Generalization of the DRL framework to the group recommendation task with a group member self-attention mechanism.
- Consideration of the task dynamics property with a MDP formulation to learn the temporal structure of the data.

Approaches: DRGR

DRGR descriptions: Agent:

- **State embedding** is to embed one state $s_t = [g, h_t]$ to its embedding $\mathbf{s}_t$ by combining one group preference $\mathbf{g}$ with its browsing history $\mathbf{h}_t$.
- **Actor** is to input one embedded state $\mathbf{s}_t$ and output one action weight $\mathbf{w}_t$, where $\mathbf{w}_t$ can generate an action a_t.
- **Critic** is to assign the Q-value $Q(\mathbf{s}_t, \mathbf{a}_t)$ to one state-action pair.

The deep deterministic policy gradient algorithm is applied.

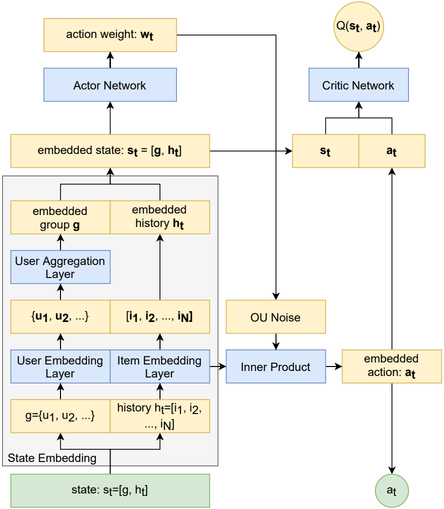

Figure 1: The framework of the agent.

Data

Data source:

- MovieLens-1M [3] dataset: 5-star rating from MovieLens, a movie recommendation service, downloaded from MovieLens website[1].
- Data preprocessing (`MovieLens-Rand`): randomly generating groups with 2-5 users; assigning group ratings to movies based on group members' ratings; 100 rating-missed items are randomly sampled for each rating.

Table 1: **Properties** of the `MovieLens-Rand` (U-I and G-I for user-item and group-item).

Property	Number
# Users	1626
# Items	1998
# Groups	1000
# U-I ratings	438 129
# G-I ratings	53 248
Avg.# ratings/user	269.45
Avg.# ratings/group	53.25
Avg. group size	2.19

[1] https://grouplens.org/datasets/movielens/

Experiments and Evaluations

- **Experimental setup:** Both user and group rating data from `MovieLens-Rand` are split into training, validation, and testing datasets with the ratio of 70%, 10%, and 20% respectively by the temporal order. All models are trained on the training set, the hyper-parameters are tuned on the validation set, and the evaluation of models is done on the testing set. The codes are run on the Google Colab [2], where one Tesla P100-PCIE-16GB GPU is used. The Python version is 3.7.10, and the PyTorch [4] version is 1.8.1.
- **Evaluation metrics:** The evaluation metrics are recall (RECALL@K) and normalized discounted cumulative gain (NDCG@K), where K is the number of recommendations and $K = \{5, 10, 20\}$.

[2] `https://colab.research.google.com/`

Results

Table 2: **Results** on the `MovieLens-Rand` dataset.

Metric	R@5	R@10	R@20	N@5	N@10	N@20
AGREE	**0.4006**	**0.5591**	**0.7335**	**0.2737**	**0.3250**	**0.3691**
GroupIM	0.1576	0.1688	0.2497	0.1597	0.1602	0.1891
DRGR	0.2885	0.4328	0.5572	0.1874	0.2336	0.2653

Comparisons of DRGR with baselines:

- DRGR performs better than GroupIM but worse than AGREE.
- GroupIM targets ephemeral groups lack of historical interactions. But `MovieLens-Rand` has a relative large average number of group ratings (53.25), where DRGR can learn long interaction history.
- AGREE has a neural attention mechanism to learn influences of group members for different items. However, DRGR only uses one self-attention mechanism to aggregate group member preferences but ignore their interactions with different recommendation items. This makes DRGR inflexible when items are diverse.

Conclusion and Future Work

- **Conclusion**: We propose a novel Deep Reinforcement learning based Group Recommender system (DRGR). Actor-critic networks are implemented with the deep deterministic policy gradient algorithm. The DRGR model is applied on the MovieLens-Rand dataset with two baselines, AGREE and GroupIM. Comparing their results, we find that DRGR performs better than GroupIM due to long interaction histories but worse than AGREE because of the self-attention mechanism.
- **Future work**:
  - An updated attention network in the state embedding layer with one neural attention mechanism to combine the group member embeddings and history item embeddings then generate one new state embedding. This neural attention can allow different group members have different influences when the group has various item histories.
  - The Deep Sets [5] algorithm to aggregate group member state embeddings, where each group member embedding will consist its user preference embedding and its browsing history item embeddings.

References I

[1] Da Cao, Xiangnan He, Lianhai Miao, Yahui An, Chao Yang, and Richang Hong. Attentive group recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 645–654, New York, NY, USA, 2018. Association for Computing Machinery.

[2] Aravind Sankar, Yanhong Wu, Yuhang Wu, Wei Zhang, Hao Yang, and Hari Sundaram. Groupim: A mutual information maximization framework for neural group recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1279–1288, 2020.

[3] F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4), December 2015.

References II

[4] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[5] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.