DEEP LEARNING BASED RECOMMENDER SYSTEMS

Tselepidis Nikolaos
ntselepidis@student.ethz.ch

Goetschmann Philippe
pgoetsch@student.ethz.ch

Maksimov Anton
antonma@student.ethz.ch

Pollak Georg Richard
pollakg@student.ethz.ch

Abstract

TODO: Write abstract

Introduction

Recommender systems are information filtering techniques that aim to predict the level of preference of a user over a specific item. In the era of big data, such techniques have attracted the interest of the scientific community, as they provide a natural approach to improving the user experience on various services, through personalization. Classical recommender systems usually make use of either content-based or collaborative filtering approaches. Content-based filtering techniques utilize specific characteristics of an item in order to recommend additional items with similar properties, while collaborative filtering approaches utilize users' past behaviour i.e. preferences and interactions with items, as well as decisions of other users with similar interests. In most cases, collaborative filtering (CF) techniques yield improved predictions compared to the content-based approaches. There are two main categories of methods when it comes to CF; (i) the Nearest-Neighbor techniques, and (ii) the Matrix Factorization (aka Latent Factor) methods. As the Netflix Prize competition has demonstrated, Matrix Factorization methods are superior to classic Nearest-Neighbor techniques, as they allow the incorporation of additional information to the models, and can thus achieve improved model capacity [8].

Recently, both academia and industry have been in a race to design deep learning based recommender systems in an attempt to overcome the obstacles of conventional models and to achieve higher recommendation quality. In fact, deep learning can effectively capture non-linear and non-trivial user-item relationships, and also enable codification of more complex abstractions as data representations in the higher levels [13]. Various deep neural network architectures have been proposed and shown to be effective for predicting user preferences. Neural Collaborative Filtering (NCF) generalizes the Matrix Factorization (MF) approach by replacing the inner product utilized in MF models by a multi-layer perceptron that can learn non-linear user-item interaction functions, and thus increases the expressiveness of the MF model [6]. Collaborative Memory Networks (CMN) unify the two classes of collaborative filtering models into a hybrid approach, combining the strengths of the global structure of the latent factor model, and the local neighborhood-based structure in a nonlinear fashion, by fusing a memory component and a neural attention mechanism as the neighborhood component [4]. Neural Graph Collaborative Filtering (NGCF) injects the collaborative signal into the embedding process by exploiting the user-item graph structure, so that it can effectively model high-order connectivity in the user-item interaction graph, and thus achieves improved recommendation quality [12]. Other deep learning based recommendation methods include Autoencoders, [11], Variational Autoencoders (VAE), [9], and Restricted Boltzmann Machines (RBMs) [10]. However, as authors have stated in [3] there has been a reproducibility issue with regards to neural recommendation approaches.

In this work, we conduct an objective study of four recently proposed neural recommendation approaches, namely NCF [6], CMN [4], NGCF [12], and VAE [9], that can be used in the context of collaborative filtering for implicit feedback, in an attempt to contribute to the resolution of the reproducibility crisis [3]. It should be stated, that implicit feedback reflects users' preference through behaviours like watching videos, purchasing products, and clicking items [7]. As opposed to explicit feedback, i.e. ratings and reviews, implicit feedback can be tracked automatically and in vast amounts, but is more challenging to utilize, since only user-item interactions are collected instead of user preferences.

In Section 2, we briefly present the aforementioned approaches along with the main underlying concepts. In Section 3, we give extensive comparative results of the selected approaches on three datasets from different application domains, i.e MovieLens (movie recommendations) [5], Epinions (product recommendations) [2], and Jester (joke recommendations) [1]. In Section 4, we discuss the strengths and weaknesses of the selected methods based on the results, and finally, in Section 5, we summarize our work.

Models and Methods

1. Neural Collaborative Filtering

TODO: Summary by Nik

2. Collaborative Memory Network

TODO: Summary by Georg

3. Neural Graph Collaborative Filtering

TODO: Summary by Anton

4. Variational Autoencoder

TODO: Summary by Phil

Results

1. Experimental Setup

TODO: *Here we should present the datasets, the data filtering and data splits, as well as the evaluation (i.e. sampling of k negatives and one positive) and the metrics we are using (i.e. Hit Ratio and NDCG). Also make a table with dataset characteristics. Also talk about the reformulation of the problems into the implicit feedback setting.*

Dataset	#Users	#Items	#Interactions	Density
Movielens	6,040	3,706	1,000,209	0.0447
Jester	24,938	100	616,912	0.2474
Epinions	27,453	37,274	99,321	0.0001

Table 1: Dataset statistics.

2. Performance Comparison

TODO: *Here we will present the tables and plots of the comparative results. Also state the methods' codenames. For example, Generalized Matrix Factorization with embedding size 8 has a codename GMF(8), Multilayer Perceptron with two layers will be called MLP(2), Neural Matrix Factorization with embedding size 8 and 2 layers will be called NeuMF(8,2) ... Something like that.*

Movielens	HR@10	NDCG@10
GMF(32)	0.7023	0.4202
MLP(4)	0.6616	0.3897
NeuMF(32,2)	0.7012	0.4233
CMN	0	0
NGCF	0	0
VAE	0	0

Table 2: Best performance achieved by each method (state configurations on codename) on Movielens dataset.

Jester	HR@10	NDCG@10
GMF(16)	0.8411	0.7589
MLP(3)	0.8427	0.7569
NeuMF(8,2)	0.8404	0.7563
CMN	0	0
NGCF	0	0
VAE	0	0

Table 3: Best performance achieved by each method (state configurations on codename) on Jester dataset.

Epinions	HR@10	NDCG@10
GMF(64)	0.2348	0.1351
MLP(2)	0.3641	0.2135
NeuMF(32,2)	0.3750	0.2167
CMN	0	0
NGCF	0	0
VAE	0	0

Table 4: Best performance achieved by each method (state configurations on codename) on Epinions dataset.

Discussion

TODO: *Here we will make comments on the results that were presented in the previous section, as well as on the advantages and limitations of the methods.*

Summary

TODO: *Here we will summarize our work. We conducted an objective experimental study of the methods in order to contribute to the resolution of the reproducibility crisis. What were our main conclusions? Furthermore, in order to be able to objectively compare the methods some modification/standarization of the authors' codes were needed. We need to briefly discuss these things.*

Appendix

References

[1] Anonymous Ratings Data from the Jester Online Joke Recommender System. https://goldberg.berkeley.edu/jester-data/. Accessed: 2019-12-12.

[2] Recommender Systems Datasets. http://cseweb.ucsd.edu/ jmcauley/datasets.html#social_data. Accessed: 2020-01-12.

[3] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 101–109. ACM, 2019.

[4] Travis Ebesu, Bin Shen, and Yi Fang. Collaborative Memory Network for Recommendation Systems. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 515–524. ACM, 2018.

[5] F Maxwell Harper and Joseph A Konstan. The Movielens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems (TIIS)*, 5(4):19, 2016.

[6] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural Collaborative Filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182. International World Wide Web Conferences Steering Committee, 2017.

[7] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, pages 263–272. IEEE, 2008.

[8] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix Factorization Techniques for Recommender Systems. *Computer*, (8):30–37, 2009.

[9] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. Variational Autoencoders for Collaborative Filtering. In *Proceedings of the 2018 World Wide Web Conference*, pages 689–698. International World Wide Web Conferences Steering Committee, 2018.

[10] Ruslan Salakhutdinov, Andriy Mnih, and Geoffrey Hinton. Restricted Boltzmann Machines for Collaborative Filtering. In *Proceedings of the 24th international conference on Machine learning*, pages 791–798. ACM, 2007.

[11] Suvash Sedhain, Aditya Krishna Menon, Scott Sanner, and Lexing Xie. AutoRec: Autoencoders Meet Collaborative Filtering. In *Proceedings of the 24th International Conference on World Wide Web*, pages 111–112. ACM, 2015.

[12] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural Graph Collaborative Filtering. *arXiv preprint arXiv:1905.08108*, 2019.

[13] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep Learning based Recommender System: A Survey and New Perspectives. *ACM Computing Surveys (CSUR)*, 52(1):5, 2019.