ECE 380L Project Proposal

Tianqi Chen, Yilin He, Qiujiang Jin

October 26th 2021

1 Problem description

Since 1994[1], online advertisement has become an important part of marketing strategy in industries such as e-commerce, entertainment and social networking. In order to pick personalized contents for every user automatically as well as maximize the return on advertising spend, engineers and researchers have developed a variety of efficient recommender systems (i.e. advertising algorithms). Despite the success of those algorithms, there are still many challenges in this area:

- Large scale: the algorithms have to be scalable as the number of users and items can easily reach tens of millions[2].
- Real-time demand response: in real settings, advertising contents need to be ready within milliseconds.
- Non-staticity: Data are being generate every moment and general trends and individual user preferences vary over time, so the advertising model should reflect the temporary change of user interest.
- Cold start: for new users, the algorithms need to come up with proper recommendations as personal as possible that are neither too vague (just following the general trend and recommending popular items) nor too specific (misled by some unique behaviors/information and recommending some extremely rare items).

The problem of online advertisement and the recommender system in particular can be summarized as either a classification problem or a ranking problem that directly predicts the user reaction (e.g. click or not, like or dislike, and rating) on each item (usually within a pre-selected subset) or ranks items based on some index reflecting user interest.

[1] The first online advertisement was posted on October 27, 1994.

[2] For example, 2nd quarter active users on Alibaba reached 828M; over 75.1M products are being sold on Amazon

2 Data available

We have found are several sources of data for recommender system, including Kaggle, GroupLens[3] and *the movie database* (TMDb). We also find a curated list very helpful in search of datasets [Jul]. For simplicity, we will start our project with the movies dataset on Kaggle[4], which contains the metadata over 45,000 movies listed in the full MovieLens Dataset[5] and 26 million rating from over 270,000 users. The metadata for movies include cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, TMDB vote counts and vote averages. The user ratings are on a scale of 1-5.

3 Possible approaches

After careful literature review and out of our practical concern about computing resources and time complexity, we preliminarily consider the following algorithms potential solutions to our rating prediction (classification) task on the movie dataset:

- SVD++ [KBV09]: generalization of *singular value decomposition* (SVD) that can be applied on sparse interaction data between user and item; it also accounts for user-specific and item-specific bias.
- Factorization Machine (FM) [Ren10]: a latent factor model that embeds user-item features (user id, item id, timestamp, last rating, etc.) into the latent space and computes a prediction score based on interaction.
- GBDT+LR [He+14]: a combined ML model that uses gradient boosting decision trees to transform contextual and historical features and then inputs the transformed features into a linear classifier (logistic regression).
- Wide & Deep [Che+16]: a general framework that first encodes the input data by a feed-forward neural network (deep component, generalization) and cross-product transformation (wide component, memorization) and then predicts by a logistic regression classifier.
- DeepFM [Guo+17]: a deep learning version of FM under the Wide & Deep framework.
- Deep Interest Network (DIN) [Zho+18]: a deep neural network (DNN) based model which incorporates the attention mechanism to obtain ad-aware embeddings and employs two techniques (mini-batch aware regu-

[3]GroupLens is a research lab in the Department of Computer Science and Engineering at the University of Minnesota, Twin Cities specializing in recommender systems, online communities, mobile and ubiquitous technologies, digital libraries, and local geographic information systems

[4]https://www.kaggle.com/rounakbanik/the-movies-dataset

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

larization and data adaptive activation) that can be useful in practical industrial setting.

We plan to implement the aforementioned algorithms on our own and experiment on the movie dataset. The focus of this project for the moment is to provide a case study that compares and analyzes the performances of those algorithms, identifying the respective strengths and weaknesses as well as the reasons behind, figuring out the space for potential improvement and proposing methods/techniques to refine them.

References

[KBV09] Yehuda Koren, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems". In: *Computer* 42.8 (2009), pp. 30–37.

[Ren10] Steffen Rendle. "Factorization machines". In: *2010 IEEE International conference on data mining*. IEEE. 2010, pp. 995–1000.

[He+14] Xinran He et al. "Practical lessons from predicting clicks on ads at facebook". In: *Proceedings of the Eighth International Workshop on Data Mining for Online Advertising*. 2014, pp. 1–9.

[Che+16] Heng-Tze Cheng et al. "Wide & deep learning for recommender systems". In: *Proceedings of the 1st workshop on deep learning for recommender systems*. 2016, pp. 7–10.

[Guo+17] Huifeng Guo et al. "DeepFM: a factorization-machine based neural network for CTR prediction". In: *arXiv preprint arXiv:1703.04247* (2017).

[Zho+18] Guorui Zhou et al. "Deep interest network for click-through rate prediction". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 1059–1068.

[Jul] UCSD Julian McAuley. *Recommender Systems and Personalization Datasets*. URL: `https://cseweb.ucsd.edu/~jmcauley/datasets.html`.