; revised 14 March, 2025; date of current version 14 March, 2025.

Exploring Methods of Optimizing User Item Recommendation Systems

MIE 424 Survey-Oriented Project

Jiacheng (Jason) Chen [1], Max Zhang[1], Ken Kato[1], Dav Vrat Chadha[1], Punyaphat Sukcharoenchaikul[1]

[1]Department of Engineering Science, University of Toronto, Toronto, ON Canada

I. INTRODUCTION

USER-item recommendation systems are crucial in modern digital platforms, driving personalized experiences in domains such as e-commerce, streaming services, and social media. These systems leverage machine learning to predict user preferences based on historical interactions, ensuring that users receive relevant content, products, or services. However, optimizing recommendation systems remains a significant challenge due to the need for high accuracy, computational efficiency, and adaptability to evolving user behavior.

In this project, we aim to study different optimization techniques for user-item recommendation systems, focusing on methods that enhance recommendation accuracy and efficiency. Specifically, we investigate the effectiveness of Collaborative Filtering, Content-enriched recommendation, temporal models, and Deep Reinforcement Learning (DRL) in improving recommendation performance. Mathematically, all these methods are trying to optimize the following problem:

$$\hat{R}_{ui} = f(u, i \mid \Theta) \tag{1}$$

where $\hat{R}_{ui}$ is the predicted rating for user u on item i, f is the optimization function, and Θ represents the model parameters. The goal is to minimize the prediction error between the predicted ratings and the actual ratings, typically using a loss function such as Mean Squared Error (MSE) or Mean Absolute Error (MAE).

TODO: fix the explaining sentences by reflecting the new section separations more Different learning-based optimization techniques approach this problem in distinct ways. Autoencoders treat recommendation as a latent representation learning problem, compressing user-item interactions into a lower-dimensional space to extract meaningful embeddings that reveal hidden relationships. Convolutional Neural Networks (CNNs) take a spatial feature extraction approach, applying convolutional filters to structured representations of user-item interactions to capture local patterns and shared item attributes. Neural Attention Mechanisms cast recommendation as a context-aware learning problem, dynamically weighting past interactions to focus on the most relevant items and personalize recommendations. Recurrent Neural Networks (RNNs) frame recommendation as a sequential prediction problem, leveraging temporal dependencies in user behavior to anticipate future preferences. Meanwhile, Deep Reinforcement Learning (DRL) treats recommendation as a decision-making problem under uncertainty, optimizing long-term user engagement by learning policies that maximize cumulative satisfaction.

By systematically comparing these approaches, we aim to provide insights into their respective strengths, weaknesses, and practical trade-offs.

II. Related Works

A. Collaborative Filtering

Collaborative Filtering (CF) is a widely used technique for building recommendation systems, leveraging past user-item interactions (such as ratings) to generate personalized recommendations. Traditional CF models like matrix factorization and neighborhood-based methods both have limitations in capturing complex patterns in user preferences [1].

The limitation lies in high computational cost and parameter inefficiency. Biased Matrix Factorization Collaborative Filtering (RBM-CF) requires contrastive divergence for training, which can be slow and it models discrete rating levels separately, leading to increased memory requirements. Furthermore, generalization is limited because matrix factorization techniques assume a linear latent representation, which may not effectively capture nonlinear user-item interactions. Therefore one of the gaps lies in developing a more efficient and expressive CF model using deep learning, one that maintains computational efficiency while improving predictive performance [1].

The authors of [1] introduced AutoRec, an autoencoder-based framework for collaborative filtering. The core idea is to treat user-item interactions as input to an autoencoder, which learns latent representations and reconstructs the missing ratings. The paper proposes two variants of AutoRec: Item-based AutoRec (I-AutoRec), which models item embeddings and reconstructs user ratings, and User-based AutoRec (U-AutoRec), which models user embeddings and reconstructs item ratings. Compared to previous models, AutoRec introduces several key improvements. First, it offers more efficient training by using gradient-based backpropagation instead of the computationally expensive contrastive divergence required by RBM-CF. Second, it reduces parameter count as it does not require separate parameter estimation for different rating levels, unlike RBM-CF. Finally, AutoRec enables nonlinear representation learning through activation functions such as sigmoid, allowing it to capture more complex user-item interactions compared to traditional matrix factorization, which assumes a linear latent representation [1].

AutoRec has several limitations. It is a relatively simple model, and while a deeper version has been tested, the performance improvements are minimal, suggesting that deeper models may require further adjustments to be effective. Additionally, AutoRec does not address the cold-start problem, making it challenging to provide recommendations for new users or items that lack prior ratings [1].

B. Content-enriched Recommendation

1) Convolutional Neural Networks

Traditional recommender systems mainly model a user's general preferences based on past interactions and fail to account for sequential dependencies, where more recent interactions are stronger indicators of future behavior. Top-N sequential recommendation addresses this by predicting the next items a user will likely engage with based on their recent activity but current Markov Chain-based models have limitations. They treat past items independently rather than collectively which it would fail to capture skip behaviors, where earlier actions may still have an impact even if intermediate steps are unrelated. To address these gaps, the authors [2] propose Caser (Convolutional Sequence Embedding Recommendation Model), which takes advantage of Convolutional Neural Networks (CNNs) to model sequential patterns in recommendation. Caser represents a sequence of recent items as an embedding matrix and applies horizontal and vertical convolutional filters to detect both union-level and point-level sequential patterns. Additionally, it incorporates user embeddings to personalize recommendations and allows for skip behaviors, making it more flexible in comparison to traditional Markov models [2].

Caser is evaluated on MovieLens, Gowalla, Foursquare, and Tmall datasets, consistently outperforming state-of-the-art models such as Factorized personalized Markov chains (FPMC), Fossil, and GRU4Rec in Precision@N, Recall@N, and MAP (Mean Average Precision). Results show that horizontal filters improve sequential modeling by capturing dependencies where multiple past actions jointly influence recommendations, while vertical filters effectively model point-level patterns. Caser also has some limitations: its performance varies across datasets, it requires more computation than simpler models like FPMC, and it is sensitive to hyperparameters, making training more complex [2].

2) Attention Networks

C. Temporal models

1) Recurrent Neural Networks

Recommender systems rely on long-term user histories to generate personalized recommendations, but in many real-world scenarios—such as e-commerce sites and media platforms—user tracking is unreliable or infeasible due to privacy concerns and short user engagement. In these cases, session-based recommendation is important, where recommendations are made based only on a user's recent interactions within a session. Existing methods, including item-to-item similarity models and Markov Chain-based approaches, are limited because they either consider only the last interaction or struggle to model long-term dependencies in user behavior. To address these shortcomings, the authors of [3] propose a Recurrent Neural Network (RNN)-based model with Gated Recurrent Units (GRUs), which captures sequential dependencies by processing an entire session as a sequence. Unlike traditional models, GRUs dynamically update their hidden state with each user interaction, learning richer session representations. Additionally, the model incorporates a ranking-based loss function for better recommendation accuracy and optimizations like session-parallel mini-batches and negative sampling for efficient training [3].

The GRU-based model is evaluated on RecSys Challenge 2015 (RSC15) and a YouTube-like VIDEO dataset, showing up to a 25% improvement in Recall@20 compared to state-of-the-art baselines like item-KNN and Bayesian Personalized Ranking Matrix Factorization (BPR-MF). Results indicate that the model effectively captures sequential dependencies, leading to more relevant recommendations. However, it has some limitations: higher training complexity, requiring GPU acceleration for scalability; lack of content-based features, meaning it relies purely on session interactions; and difficulty in hyperparameter tuning, as ranking loss functions require extensive experimentation. Additionally, like most collaborative filtering models, it still suffers from the cold-start problem, struggling to recommend items for new users with no prior interactions [3].

D. Deep Reinforcement Learning Frameworks

Traditional news recommendation methods is formulated as an optimization problem that maximize the immediate engagement, assuming a fixed user preference. However, this approach fails to optimize long-term user engagement, as news consumption is time-sensitive and user interests evolve over time. Moreover, current recommendations influence future engagement, making only optimizing current reward suboptimal [4]. To achieve sustained engagement over an infinite horizon, the recommendation system is better modeled as a Markov Decision Process (MDP), where the objective is to maximize cumulative future rewards. In this formulation, the state represents a user's recent reading history and current news, the action is recommending some news articles, and the reward is engagement-based metrics [4]. The policy improvement theorem and contraction mapping of state-action value functions (Q-function) guarantee the existence and convergence of the optimal recommendation policy. However, solving this MDP exactly is intractable due to the high-dimensional state-action space, as there are infinitely many news states and infinitely many recommendation combinations. To address this, the authors propose using Deep Q-Networks (DQN) for functional approximation to compress state and action spaces, enabling generalization across unseen states [4]. Further enhancements include Double Q-Learning (DDQN) to mitigate overestimation bias, experience replay for stable learning, activeness scores to dynamically weight user engagement, and Dueling Bandit Gradient Descent (DBGD) to refine exploration-exploitation trade-off via pairwise comparisons.

The proposed method and its enhanced variant, Double DQN (DDQN) + all tricks, outperform all baseline methods, including Logistic Regression (LR), Factorization Machines (FM), Wide & Deep (W&D), and bandit-based approaches (LinUCB, HLinUCB). The DDQN model achieves the highest CTR (0.1662) and nDCG (0.4877), demonstrating the effectiveness of reinforcement learning in optimizing long-term engagement. The dueling network architecture improves user-news interaction modeling, while future reward consideration in DDQN leads to additional gains in recommendation accuracy. However, in the offline setting, incorporating user activeness and exploration does not show significant improvements due to dataset limitations, as interactions are static, preventing effective exploration. The authors note that naïve exploration strategies like ε-greedy can harm recommendation accuracy, suggesting that exploration should be designed more carefully. The authors propose evaluating the model in an online setting, where user engagement evolves dynamically, allowing better assessment of exploration strategies and activeness-aware recommendations. Additionally, integrating more advanced exploration methods and personalized reward functions could further enhance long-term engagement optimization.

REFERENCES

[1] S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "Autorec: Autoencoders meet collaborative filtering," in *Proceedings of the 24th international conference on World Wide Web*, 2015, pp. 111–112.

[2] J. Tang and K. Wang, "Personalized top-n sequential recommendation via convolutional sequence embedding," in *Proceedings of the eleventh ACM international conference on web search and data mining*, 2018, pp. 565–573.

[3] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," *arXiv preprint arXiv:1511.06939*, 2015.

[4] G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, N. J. Yuan, X. Xie, and Z. Li, "DRN: A Deep Reinforcement Learning Framework for News Recommendation," in *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18*. Lyon, France: ACM Press, 2018, pp. 167–176. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3178876.3185994