CS4681 Advanced Machine Learning

Enhancing Neural Collaborative Filtering with Self-Supervised Learning

Progress Evaluation Report

Name:	**Kariyawasam L.C.**
Index No:	210273B
Project ID:	REC001
Date of Sub:	2025.08.26

Abstract

Neural Collaborative Filtering (NCF) has established itself as a foundational deep learning framework for recommendation systems, moving beyond the linear constraints of traditional matrix factorization. However, its performance can be hampered by the inherent sparsity of user-item interaction data and its reliance on a simple pointwise loss function, which may not yield robust user and item representations. This paper proposes an enhancement to the NCF framework, termed NCF-SSL, which integrates a self-supervised contrastive learning task. The goal is to improve the quality of learned embeddings by enforcing consistency between augmented views of the same user or item. This auxiliary task acts as a powerful regularizer, encouraging the model to learn more generalizable features. We detail a methodology that is resource-efficient and can be implemented with minimal modifications to the original NCF architecture, making it a practical approach for improving recommendation accuracy.

Contents

1 Introduction

1.1 Background

Recommender systems are a cornerstone of modern online platforms, helping users navigate vast catalogs of items. Collaborative Filtering (CF) is a dominant paradigm in this domain, operating on the principle that users with similar past behaviors will have similar future preferences. For years, Matrix Factorization (MF) was the state-of-the-art method, modeling user-item interactions by factorizing the interaction matrix into low-dimensional latent feature vectors for users and items.

The advent of deep learning brought new possibilities to CF. He et al. [1] introduced Neural Collaborative Filtering (NCF), a general framework that replaces the simple inner product of MF with a non-linear neural network, specifically a Multi-Layer Perceptron (MLP). This allows the model to capture more complex and subtle user-item interaction patterns. The NCF framework elegantly combines a generalized version of MF (GMF) with an MLP, creating a powerful and flexible model for recommendation.

1.2 Problem Statement

Despite its success, the standard NCF model faces challenges. The primary issue stems from the supervision signal, which is often derived from sparse user-item interaction data. The model is typically trained with a pointwise loss function (e.g., binary cross-entropy), which treats each interaction as an independent classification task. This setup may not be sufficient for the model to learn high-quality, robust representations, especially for users or items with few interactions. The learned embeddings can overfit and may not generalize well to unseen data.

1.3 Project Objectives

The primary goal of this project is to develop a targeted enhancement to the baseline NCF model that improves its representation learning capabilities without requiring significant architectural overhaul or external data. The specific objectives are,

1. To conduct a thorough review of existing literature on enhancements to the NCF model, focusing on architectural modifications and loss function improvements.

2. To propose a novel methodology, NCF-SSL, that integrates a self-supervised contrastive learning task into the NCF training process.

3. To detail a practical implementation plan for NCF-SSL that is feasible to execute within a few weeks with limited computational resources.

2 Literature Review

Since the seminal work on NCF [7], a significant body of research has been dedicated to advancing deep learning-based collaborative filtering. These enhancements can be broadly grouped into several key areas.

2.1 Architectural Modifications

This line of research focuses on improving the core model architecture to better capture complex dependencies in the data.

2.1.1 Attention Mechanisms

Recognizing that not all latent features contribute equally to an interaction, attention mechanisms have been integrated to assign dynamic, context-aware weights to features. For instance, Attentional Collaborative Filtering (ACF) introduces an attention network to the MLP component of NCF to learn the importance of each latent dimension [1].

2.1.2 Graph Neural Networks (GNNs)

A major advancement has been the framing of recommendation as a graph learning problem. Models like Neural Graph Collaborative Filtering (NGCF) [10] explicitly model the bipartite user-item interaction graph. By propagating embeddings along the graph, NGCF captures high-order connectivity, encoding signals that a user is similar to another user because they have interacted with similar items. LightGCN [4] later simplified this approach by removing non-linearities and feature transformations, demonstrating that the core message-passing mechanism is the most critical component, leading to improved performance and efficiency.

2.1.3 Convolutional Neural Networks (CNNs)

To capture localized interaction patterns, CNNs have also been applied. ConvNCF [6] first performs an outer product between user and item embeddings to create a 2D interaction map, then applies a CNN to this map to learn higher-order correlations, similar to how CNNs learn features from images.

2.1.4 Generative Adversarial Networks (GANs)

GANs have been used to improve model robustness. IRGAN [9] frames the recommendation task as a game between a generator, which tries to produce realistic (user, item) pairs, and a discriminator, which aims to distinguish generated pairs from real interactions. This adversarial process serves as a strong regularizer.

2.2 Loss Function and Training Strategy Enhancements

This category focuses on improving the learning objective to better align with the goal of recommendation, which is often ranking.

2.2.1 Pairwise and Listwise Learning

The pointwise loss of NCF treats each interaction in isolation. In contrast, pairwise methods, such as Bayesian Personalized Ranking (BPR) [8], aim to optimize the relative ordering of items for a user, learning to rank observed items higher than unobserved ones. Listwise methods take this further by considering an entire list of items, directly optimizing ranking metrics like NDCG.

2.2.2 Adversarial Training

To enhance robustness, adversarial training techniques add small, worst-case perturbations to the embedding vectors during training [5]. This forces the model to learn smoother decision boundaries and become less sensitive to minor variations in the input data.

2.2.3 Contrastive Learning

Most recently, self-supervised contrastive learning has gained significant traction. This paradigm aims to learn representations by maximizing the agreement between different augmented "views" of the same data point. For recommendation, Self-supervised Graph Learning (SGL) [11] applies augmentations like node or edge dropout to the user-item graph and uses a contrastive loss to pull representations of the same node from different views closer. This provides a powerful auxiliary signal that alleviates data sparsity issues. Our proposed work is directly inspired by this trend.

2.3 Integration of Side Information

To address data sparsity and the cold-start problem, many models incorporate side information (e.g., user demographics, item attributes). The Wide & Deep model [2], while not a direct NCF successor, exemplifies the principle of combining a simple linear model (wide part) for memorization with a deep neural network (deep part) for generalization, often incorporating rich features. DeepFM [3] integrates factorization machines with an MLP to effectively model both low and high order feature interactions.

3 Methodology

The proposed method is an NCF with Self-Supervised Learning (NCF-SSL), a model that augments the standard NCF training objective with a contrastive learning task. The methodology is designed to be simple and efficient.

3.1 Data Augmentation

To create the different "views" required for contrastive learning, a simple stochastic data augmentation technique will be applied, embedding dropout. For each user u and item i in a training batch, two distinct augmented views of their respective embeddings ($\mathbf{p}_u$, $\mathbf{q}_i$) are generated by applying dropout independently.

$$\mathbf{p}_u^{(k)} = \text{Dropout}(\mathbf{p}_u), \quad \mathbf{q}_i^{(k)} = \text{Dropout}(\mathbf{q}_i) \quad \text{for } k \in \{1, 2\}$$

This process creates two correlated but different representations for each user and item within the same training step.

3.2 Multi-Task Learning Objective

The model is trained using a multi-task loss function that combines the primary recommendation task with the auxiliary self-supervised task.

3.2.1 Recommendation Loss

The primary objective is the standard NCF loss. The binary cross-entropy loss for the recommendation task is computed on both augmented views and average the results

$$\mathcal{L}_{NCF} = -\frac{1}{N} \sum_{(u,i)\in\mathcal{Y}^+\cup\mathcal{Y}^-} \frac{1}{2} \sum_{k=1}^{2} \left(y_{ui} \log \hat{y}_{ui}^{(k)} + (1 - y_{ui}) \log(1 - \hat{y}_{ui}^{(k)}) \right)$$

where $\hat{y}_{ui}^{(k)}$ is the predicted score from the NCF model for the k-th augmented view, y_{ui} is the ground truth label, and N is the number of training samples.

3.2.2 Self-Supervised Contrastive Loss

The auxiliary objective is a contrastive loss that encourages the embeddings of the two augmented views of the same entity (user or item) to be more similar to each other than to the embeddings of other entities in the same batch. The InfoNCE loss is used for that. For users, the loss is: $\mathcal{L}_{SSL}^{user} = \sum_{u\in\mathcal{B}} -\log \frac{\exp(\text{sim}(\mathbf{p}_u^{(1)}, \mathbf{p}_u^{(2)})/\tau)}{\sum_{v\in\mathcal{B}} \exp(\text{sim}(\mathbf{p}_u^{(1)}, \mathbf{p}_v^{(2)})/\tau)}$ where $\text{sim}(\cdot, \cdot)$ is the cosine similarity, τ is a temperature hyperparameter, and $\mathcal{B}$ is the set of users in the current batch. A similar loss, $\mathcal{L}_{SSL}^{item}$, is computed for items. The total self-supervised loss is: $\mathcal{L}_{SSL} = \mathcal{L}_{SSL}^{user} + \mathcal{L}_{SSL}^{item}$

3.2.3 Final Loss

The final training objective is a weighted sum of the two losses, $\mathcal{L} = \mathcal{L}_{NCF} + \lambda\mathcal{L}_{SSL}$ where λ is a hyperparameter that balances the contribution of the recommendation task and the self-supervised task. By training the model with this combined objective, the NCF framework is encouraged to learn more robust and generalizable embeddings.

3.3 Hyperparameter Sensitivity

The introduction of a multi-task learning framework brings several new hyperparameters that can significantly impact performance. The model's effectiveness will likely be highly sensitive to,

- **The contrastive loss weight (λ):** Balancing the primary recommendation task and the auxiliary self-supervised task is critical. If λ is too small, the benefit of contrastive learning may be negligible. If it is too large, it could overwhelm the main objective, leading to poor recommendation performance.
- **The temperature parameter (τ):** In the InfoNCE loss, the temperature controls the sharpness of the distribution over negative samples. A proper setting is crucial for effective learning, as it helps to scale the similarity scores and prevent the model from focusing only on the easiest negatives.
- **The augmentation strategy:** The dropout rate used for creating augmented views must be carefully selected. Too little dropout will result in views that are too similar, providing a weak learning signal. Too much dropout may destroy valuable information within the embeddings, hindering the learning process.

Tuning these hyperparameters will require extensive experimentation and can be computationally expensive.

3.4 Negative Sampling and Batch Size

The effectiveness of the contrastive learning task is directly tied to the quality and quantity of negative examples, which in this framework are the other samples within a training batch.

- **Batch Size:** A small batch size provides a limited number of negative examples, which can weaken the contrastive signal and slow down convergence. Conversely, a very large batch size increases the computational cost per step due to the pairwise similarity calculations ($O(B^2)$ for a batch of size B).
- **False Negatives:** In-batch negative sampling carries the risk of "false negatives"—items that are semantically similar to the anchor item but are treated as negatives. This can send a conflicting signal to the model, potentially hurting representation quality.

3.5 Evaluation and Interpretability

Isolating the source of performance gains can be challenging. An improvement in recommendation metrics could be due to the regularization effect of dropout, the improved embedding structure from the contrastive loss, or a synergistic combination of both. Futher, qualitatively assessing the "goodness" of the learned embeddings beyond offline metrics will require visualization techniques (e.g., t-SNE) and careful analysis.

4 Project Planning and Timeline

This section outlines the overall planning and timeline for the project. It details the key milestones, deliverables, and estimated completion dates to ensure systematic progress throughout the project duration. By establishing a clear schedule, resources can be managed efficiently and project objectives can be achieved within the allocated timeframe.

A large portion of the upcoming weeks will focus on the implementation of the NCF-SSL model, including coding the data augmentation strategies and multi-task learning framework. Afterwards the results will be analyzed against the baseline benchmarks to evaluate the effectiveness of the proposed enhancements. The final weeks will be dedicated to compiling the findings into a comprehensive report, ensuring that all aspects of the project are thoroughly documented and presented.

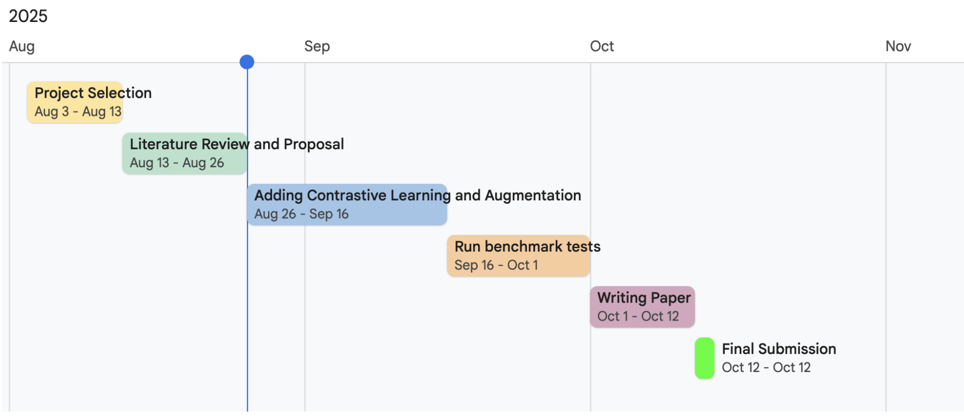

Figure 1: Project Timeline and Key Milestones

References

[1] Chao Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. Attentional collaborative filtering: Recommending images with item- and component-level attention. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 953–956, 2017.

[2] Heng-Tze Cheng et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 7–10, 2016.

[3] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiangnan He. Deepfm: A factorization-machine based neural network for ctr prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1725–1731, 2017.

[4] Xiangnan He, Kaige Deng, Xiang Wang, Yong Li, Yan Wang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 639–648, 2020.

[5] Xiangnan He, Zhiqiang Du, Xiang Wang, Fuli Feng, Jinyang He, and Tat-Seng Chua. Adversarial personalized ranking for recommendation. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 415–424, 2018.

[6] Xiangnan He, Chao He, Meng Du, and Raymond C. K. Chan. Outer product-based neural collaborative filtering. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 237–245, 2018.

[7] 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 (WWW)*, pages 173–182, 2017.

[8] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 452–461, 2009.

[9] Jun Wang, Lantao Yu, Weinan Zhang, Yikang Gong, Ying Xu, Ben Wang, Peng Zhang, and Defu Zhang. Irgan: A minimax game for unifying generative and discriminative information retrieval models. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 515–524, 2017.

[10] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 165–174, 2019.

[11] Jiaxi Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jian Lian, and Xing Xie. Self-supervised graph learning for recommendation. In *Proceedings of the 44th*

International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), pages 726–735, 2021.