Are We Really Making Recommendations Robust? Revisiting Model Evaluation for Denoising Recommendation

Guohang Zeng
University of Technology Sydney
Sydney, Australia
Guohang.Zeng@student.uts.edu.au

Jie Lu*
University of Technology Sydney
Sydney, Australia
jie.lu@uts.edu.au

Guangquan Zhang
University of Technology Sydney
Sydney, Australia
guangquan.zhang@uts.edu.au

Abstract

Implicit feedback data has emerged as a fundamental component of modern recommender systems due to its scalability and availability. However, the presence of noisy interactions—such as accidental clicks and position bias—can potentially degrade recommendation performance. Recently, denoising recommendation have emerged as a popular research topic, aiming to identify and mitigate the impact of noisy samples to train robust recommendation models in the presence of noisy interactions. Although denoising recommendation methods have become a promising solution, our systematic evaluation reveals critical reproducibility issues in this growing research area. We observe inconsistent performance across different experimental settings and a concerning misalignment between validation metrics and test performance caused by distribution shifts. Through extensive experiments testing 6 representative denoising methods across 4 recommender models and 3 datasets, we find that no single denoising approach consistently outperforms others, and simple improvements to evaluation strategies can sometimes match or exceed state-of-the-art denoising methods. Our analysis further reveals concerns about denoising recommendation in high-noise scenarios. We identify key factors contributing to reproducibility defects and propose pathways toward more reliable denoising recommendation research. This work serves as both a cautionary examination of current practices and a constructive guide for the development of more reliable evaluation methodologies in denoising recommendation.

CCS Concepts

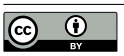
• **Information systems** → **Collaborative filtering**; • **Computing methodologies** → *Learning from implicit feedback.*

Keywords

Recommender Systems, Evaluation, Methodology

ACM Reference Format:
Guohang Zeng, Jie Lu, and Guangquan Zhang. 2025. Are We Really Making Recommendations Robust? Revisiting Model Evaluation for Denoising Recommendation. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems (RecSys '25), September 22–26, 2025, Prague, Czech Republic.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3705328.3748153

*Corresponding Author



RecSys '25, Prague, Czech Republic

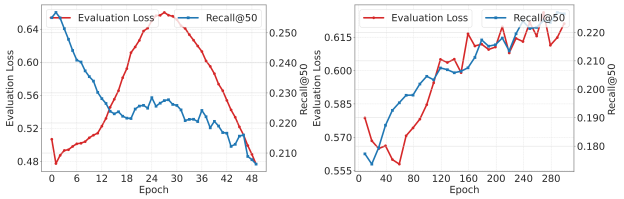
ACM ISBN 979-8-4007-1364-4/25/09
https://doi.org/10.1145/3705328.3748153

1 Introduction

Recommender systems aim to uncover user preferences from user-item interaction data, guiding users toward content that matches their interests and potential needs. Compared to explicit feedback (e.g., ratings), implicit feedback (clicks, market browsing behaviors, and other user interactions) serves as the preferred training resource due to its easier acquisition process and consequently larger data volume. However, research has shown that these implicit interactions face significant data quality challenges [21, 24] , including position bias [18], caption bias [17], and accidental clicks [30]. These inaccurate false-positive interactions have been demonstrated to negatively impact user experience [38], as they misrepresent actual user preferences and lead to suboptimal recommendations.

To address this challenge, Wang et al. [35] pioneered a novel denoising recommendation paradigm that addresses noisy interactions during the training process. Denoising recommendation methods aim to learn robust models from training and validation sets containing noisy interactions, which can minimize the expected risks on clean test sets where noisy interactions are absent. The core concept of this paradigm posits that even when training datasets inevitably contain noisy interactions, models should be able to adaptively identify and mitigate their negative impact, thereby building more robust recommender systems. Building upon this foundation, the research community has introduced various innovative approaches to enhance the noise-resistance capabilities of recommender systems, exploring different dimensions such as optimization perspectives [37], memory effects [8], and leveraging the unique data patterns of noisy interactions [16]. With continuous improvements in the state of the art for denoising recommendations [5, 8, 44], this research direction has garnered increasing attention in the academic community.

Despite denoising recommendation evolving into a flourishing new research area, we observe two phenomena worth contemplating: (1) In experimental results from recent papers, denoising recommendation methods do not significantly outperform traditional methods without denoising methods in some scenarios [5, 16]; (2) The performance of various denoising methods reported across different papers often lacks consistency, with different denoising methods failing to demonstrate consistent partial order relationships in their performance. These observations have prompted our reflection. In recent years, the reproducibility issue [6, 7, 27] in recommender systems has gradually attracted widespread attention in academia, making the research community realize that representation learning in recommender systems and its reproducibility remain challenging topics. In this study, we are committed to deeply exploring how this emerging field of denoising recommendation



(a) R-CE [35] on GMF (b) DCF [16] on LightGCN

Figure 1: Evaluation loss fails to provide accurate model selection criteria. (a) shows the GMF model trained on adressa with R-CE [35] denoising. (b) shows the LightGCN model trained on MovieLens using the DCF [16] denoising.

is shrouded in the fog of reproducibility, and attempt to reveal its internal mechanisms and key issues.

Our work began with the observation of an anomalous experimental phenomenon. Figure 1 illustrates the misalignment between validation loss and test set performance during model training. In Figure 1(a), despite the model's true performance on the test set continuously declining, the evaluation loss begins to fit noise after 30 epochs of training, making model selection based on validation performance prone to choosing a poorly performing model. In Figure 1(b), although model performance continues to improve, the validation loss reaches its minimum value in an underfitted state, and after reaching its minimum, the evaluation loss increases synchronously with test performance (normally, evaluation loss and test set performance should be contradictory indicators). We attribute this anomalous phenomenon to the data distribution differences between the validation set and test set. In standard supervised learning settings, samples from training, validation, and test sets should be independently and identically distributed. However, in the specific scenario of denoising recommendation, the data distribution $P_{\mathcal{D}_{val}}$ of the validation set containing noisy interactions is clearly not equivalent to the data distribution $P_{\mathcal{D}_{test}}$ of the clean test set, leading to distribution drift between validation and test sets. Our results indicate that models with lower empirical error on the validation set are not necessarily optimal on the test set, which has led to frequent misassessment of model performance in previous research studies.

To conduct rigorous benchmark testing, we investigated 12 denoising recommendation method papers [2, 5, 9, 16, 22, 29, 33, 35–37, 44] published in top conferences in recent years. From these, we selected 6 representative denoising methods and trained four typical recommender models on three widely used datasets, completing tests across 73 experimental configurations. Through these large-scale experiments, our contributions are:

- We reveal the misalignment of the performance between validation set and test set that might be caused by distribution shift, which lead to inaccurate benchmarking of existing denoising methods.
- We propose a simple sampling-based evaluation method, finding that just by improving evaluation strategies, empirical error minimization without any denoising techniques can be comparable or even exceed the performance of state-of-the-art denoising recommendation methods in some experimental scenarios.
- By observing the evolution of denoising model accuracy on test sets through numerous experiments, we found that no single denoising method consistently outperforms others across different scenarios.
- We find that denoising recommendation is essentially ineffective on datasets like Adressa with noise rates exceeding 60%, revealing previously unnoticed limitations of existing denoising methods, and indicating that their applicability in extreme noise scenarios needs to be reassessed.
- We reveal and discuss two factors which lead to reproducibility defects in denoising recommendation, and provide clues for improving reproducibility and reliability in future research.

2 Related works

2.1 Denoising Recommendation

Implicit feedback is inherently susceptible to various types of noise, such as user misclicks and popularity bias [1, 3] . Recent studies have clearly demonstrated that large amounts of noisy implicit feedback can mislead the learning process of recommendation models, causing them to fit incorrect user preference patterns [20, 28, 32, 35]. To address this critical issue, denoising recommendation have emerged and attracted widespread attention in recommender systems research community [32, 43].

Existing denoising recommendation methods primarily focus on reducing the impact of noisy interactions on model training by adjusting their weights. Among these, R-CE and T-CE methods [35] use loss values as indicators of noise, dynamically reducing the weights of high-loss samples or directly dropping them. The BOD method [37] formulates weight learning as a bi-level optimization problem to automatically learn optimal denoising weights. Model consistency-based methods such as DeCA [36] assume that different models make consistent predictions on clean data while showing significant disagreement on noisy data, identifying noise through dual-model training. The SGDL method [8] observes the memorization effect of deep models and eliminates noise influence during the pre-training phase. The DCF method [16] not only considers sample dropping strategies but also addresses the handling of hard positive samples. The PLD method [44] identifies noisy interactions by analyzing each user's personal loss distribution to reduce the probability of noisy interactions being optimized during training. LLaRD [33] utilizes large language models to generate denoising knowledge through semantic insights and Chain-of-Thought reasoning on user-item graphs, then applies Information Bottleneck to filter noise and improve recommendation accuracy. UDT [5] separates user behavior into willingness and action phases, identifies high-uncertainty willingness and user-specific inconsistency patterns, then adjusts interaction importance weights based on these patterns to reduce noise during training.

Besides dedicated denoising methods, there are also related robustness-enhancing approaches, including adversarial training methods such as AMF [14], contrastive learning-based methods

such as SGL [39] which improves the robustness of user-item representations through graph augmentation and contrastive learning, and scenario-specific denoising techniques such as specialized denoising modules for micro-video recommendation [23], next-basket recommendation [26], and social recommendation [34]. However, these auxiliary methods are typically effective in specific scenarios and are difficult to serve as universal denoising solutions. Therefore, this article focuses primarily on specifically designed denoising recommendation methods.

2.2 Related Studies in Computer Vision

In the domain of computer vision, a small amount of research has noted the distribution shift problem between noisy validation sets and clean test sets. Several theoretical frameworks have emerged to validate the reliability of model selection using noisy validation data under specific noise assumptions [4, 31]. Alternative approaches circumvent the use of potentially unreliable noisy validation sets altogether by determining optimal early stopping points through monitoring prediction dynamics on training samples [42]. Furthermore, some work has focused on correcting performance metrics on noisy validation sets by estimating noise transition matrices, thereby more accurately approximating performance on clean data distributions [25, 41]. Despite these advances, the fundamental differences between recommender systems and computer vision present significant challenges when attempting to transfer these methodologies directly to denoising recommendation tasks. To the best of our knowledge, our work is the first to notice and attempt to address the distribution shift problem caused by noisy validation sets in the context of recommendation systems.

3 Preliminaries

In this section, we introduce the formal definition of denoising recommendation. Consider a recommendation problem based on implicit feedback, where our objective is to learn user preferences through a recommendation model. Let $\mathcal{U} = \{u_1, u_2, \dots\}$ denote the set of users and $\mathcal{I} = \{i_1, i_2, \dots\}$ represent the set of items. The supervision information is derived from an interaction matrix $\mathbf{Y} \in \{0,1\}^{|\mathcal{U}|\times|\mathcal{I}|}$, where $y_{ui} = 1$ indicates that user u has interacted with item i, and $y_{ui} = 0$ indicates otherwise. A recommendation model can be formulated as a function $\hat{y}_{ui} = f(u, i \mid \Theta)$ parameterized by Θ, where $\hat{y}_{ui} \in [0,1]$ is the learned preference of the user for the item. We denote an interaction as a tuple (u, i, y^*_{ui}), where $y^*_{ui} \in \{0,1\}$ indicates whether there exists an interaction record between user u and item i. The traditional setting of recommendation is to learn a model f, such that minimize the expected risk:

$$R(f) = \mathop{\mathbb{E}}_{(u,i,y_{ui})\sim P_{\mathcal{D}}} \left[\ell(f(u, i \mid \Theta), y_{ui})\right] \tag{1}$$

where $P_{\mathcal{D}}(\cdot)$ denotes the unknown distribution over the interaction data. In standard recommendation scenarios, we assume that interactions data $\mathcal{D}^* = \left\{\left(u, i, y^*_{ui}\right) \mid u \in \mathcal{U}, v \in \mathcal{V}, y^*_{ui} \in \{0,1\}\right\}$ is sampled from $P_{\mathcal{D}}(\cdot)$, then we can train the recommendation models by *empirical risk minimization* (ERM) as

$$\arg\min_{\Theta} \mathcal{L}(\mathcal{D}^*) = \frac{1}{|\mathcal{D}^*|} \sum_{(u,i,y^*_{ui})\in\mathcal{D}^*} \ell\left(f\left(u, i \mid \Theta\right), y^*_{ui}\right) \tag{2}$$

where ℓ is an arbitrary loss function. Minimizing Eq. (1) means training a model f to learn user preferences for items, thereby generalizing to unseen items for recommendations.

However, due to the existence of noisy interactions, denoising recommendation considers that the observed interaction data may not follow the true distribution. We denote the observed interactions set as $\bar{\mathcal{D}} = \{(u, i, \bar{y}_{ui}) \mid u \in \mathcal{U}, i \in \mathcal{I}, \bar{y}_{ui} \in \{0,1\}\}$. Note that due to the existence of noisy interactions, there is an inconsistency between D^* and $\bar{D}$. Specifically, we consider that $\bar{D}$ contains noisy interactions, which can be represented as $\{(u, i) \mid y^* = 0 \wedge \bar{y} = 1\}$. These noisy interactions are typically introduced by users' accidental clicks or position bias. Wang et al. [35] provided a formal definition of *denoising recommendation training task* as:

$$\Theta^* = \min \mathcal{L}_{CE}(denoise(\bar{\mathcal{D}})), \tag{3}$$

where $\mathcal{L}_{CE}$ denotes Binary Cross Entropy loss, aiming to learn a reliable recommender with parameters Θ^* by denoising implicit feedback, such as pruning the impact of noisy interactions.

3.1 Revisiting Denoising Recommendation

Since the true distribution in the test set is invisible, we need to select models based on the validation set. Let the samples in the validation set be denoted as $\bar{D}_{val}$ following distribution $P_{\bar{\mathcal{D}}}$. Note that our optimization target is the expected risk $R(f)$ over distribution $P_{\mathcal{D}}$, however $P_{\mathcal{D}} \neq P_{\bar{\mathcal{D}}}$, which leads to validation set-test set distribution shift. Simply optimizing equation (3) will cause the model to incorrectly select models, resulting in many denoising methods being incorrectly evaluated.

In this work, we demonstrate a simple sampling approach for the validation set:

$$\hat{D}_{\text{val}} = \{(u, i, y_{ui}) \in D_{\text{val}} \mid \ell(f(u, i|\Theta), y_{ui}) < \tau_a\} \tag{4}$$

where $\hat{D}_{val}$ is sampled from $\bar{D}_{val}$, where τ_a represents the a-th percentile of loss values for all interactions in D_{val}, and we set $a = 0.8$. This means we retain the interactions with loss values below this threshold. Specifically, we discard the 20% of interactions with the highest loss values and retain the remaining interactions as validation data. This method is based on the small-loss criterion [11, 45], which assumes samples with small losses are more likely to originate from the true noise-free distribution.

Next, we only need to apply the Empirical Risk Minimization (ERM) method without using any denoising training techniques, and evaluate on $\hat{D}_{val}$:

$$\Theta^* = \min \mathcal{L}_{CE}(\hat{\mathcal{D}}_{val}), \tag{5}$$

In the following section, we will demonstrate how existing methods have been incorrectly evaluated, and how by only changing the validation set sampling method, it is possible to outperform existing denoising methods in some scenarios.

4 Settings

4.1 Benchmark Selection

In this section, we explain how we conduct a fair benchmark in denoising recommendation, which includes the selection of denoising methods, recommendation models, and datasets.

4.1.1 Denoising Methods. To rigorously evaluate denoising evaluation methods, we examined relevant papers published in top-tier conferences over the past 4 years (2021-2025), including RecSys, KDD, SIGIR, WWW, and WSDM, with a total of 12 articles on recommendation denoising. The following papers, although related to denoising recommendation, were not considered as evaluation candidates: [2, 22, 29] focused on denoising for special recommendation scenarios (e.g., sequential recommendation, graph collaborative filtering, CTR prediction) rather than general collaborative filtering denoising; PLD [44] did not follow the original denoising setting, but created new experimental scenarios; LLaRD [33] utilized text information, which is a new denoising setting. SGDL [8] was not considered due to its high computational cost [1]. Additionally, some methods employed non-rigorous evaluation approaches: BOD [37] performed evaluation directly on the test set; AutoDenoise [9] evaluated directly on the clean validation set. Therefore, we examined 6 denoising methods from the remaining 4 papers, including two latest denoising methods, which are:

- **ERM**: without denoising, directly performs empirical risk minimization on noisy interactions (referred to as the *normal* method or *base* method in previous literature)
- **R-CE** [35]: adjusts the importance weights of user interaction data based on the value of binary cross-entropy loss
- **T-CE** [35]: dropout user-item interactions with loss values exceeding a predefined threshold during training.
- **DeCA** [36]: leverages the disagreement between two models on interactions to identify noisy interactions, where higher disagreement is more likely to indicate a noisy interaction.
- **DeCAp** [36]: DeCA method with model pre-training.
- **DCF** [16]: beyond simply discarding noisy interactions, it re-labels highly deterministic noisy samples to mitigate the increased data sparsity caused by data dropout.
- **UDT** [5]: separates user behavior into willingness and action phases, then identifies uncertainty patterns to adjust interaction weights for noise reduction during training.
- **ERM-RE (Revisiting Evaluation)**: ERM-RE employs standard Empirical Risk Minimization without any denoising techniques. As described by Eq. (4) and Eq. (5), it modifies the validation process.

4.1.2 Datasets. For fair and rigorous dataset selection for evaluation, we first reviewed all datasets adopted in denoising recommendation, which are: MovieLen-100k, MovieLen-1M, Adressa, Yelp, Amazon Book, Amazon Elec, iFashion, and Beauty; among these, only MovieLen-100k, Adressa, and Yelp have been used three or more times. Therefore, we chose to conduct evaluations on the following 3 datasets where their statistics can be found at Table 1:

- **MovieLens** [12]: This is movie rating datasets from the MovieLens web site. We used the MovieLens-100k version. Following the setting in [35], ratings below 3 are considered false-positive feedback.
- **Adressa**[2] [10]: A real-world news reading dataset that includes user clicks on news and the dwell time for each click. Following the setting in [19], we consider clicks with a dwell time of less than 10 seconds as false-positive clicks.
- **Yelp**[3]: This is a user review dataset for the catering industry. Following the setting in [35], ratings below 3 are considered false-positive feedback.

Dataset	#User	#Item	#Iter	#FP Iter	FP ratio
ml-100k	944	1,611	79,619	12,649	15.89%
Adressa	212,231	6,596	419,491	254,487	60.67%
yelp	45,548	57,396	1,672,520	260,581	15.58%

Table 1: Dataset statistics. #FP Iter represents false-positive interaction, FP ratio represents the proportion of FP iter within the total interactions

Consistent with other denoising recommendation approaches, we retain false-positive interactions in both the training set and validation set, while keeping only clean interactions in the test set.

4.1.3 Recommendation Models. In the denoising recommendation scenario, 5 models were adopted as experimental subjects, namely: GMF, NeuMF, CDAE, NGCF, and LightGCN, where both NGCF and LightGCN are graph collaborative filtering models. We selected the more representative LightGCN model as a candidate. Therefore, we chose the following 4 models.

- **GMF** [15]: a matrix factorization extension using element-wise product and a linear layer for interaction modeling.
- **NeuMF** [40]: combines GMF and MLP to model user-item relationships, capturing both linear and non-linear patterns.
- **CDAE**[13]: it adds noise to interactions and uses a neural network (often MLP or linear layers) to reconstruct the original, improving robustness.
- **LightGCN** [15]: is a graph-based model that learns user/item embeddings by linear propagation on the interaction graph.

4.2 Experiment settings

4.2.1 Implementation Details. Notably, since we need to characterize the changes in the denoising method on the test set during the training process, we need to frequently conduct global testing which is very time consuming. For most experiment scenarios, we perform a global evaluation once per epoch. For models like LightGCN and CDAE that require many epochs to converge, we conduct global evaluations intermittently, with intervals ranging from once every 5 epochs to once every 25 epochs, depending on the model's convergence speed and computational cost. Due to the different convergence speeds of different recommendation models, we selected 50 epochs for GMF and NeuMF, 300 epochs for LightGCN, and 500 epochs for CDAE. As for other model parameters, we maintained consistency with the descriptions provided by the authors. All code and hyperparameter choices, as well as training logs and results can be found at https://github.com/bachml/EvalDenoisingRec.

4.2.2 Evaluation protocols. Following [16, 35], we divided the dataset into training set, validation set, and clean test set with an 8:1:1 ratio.

[1] In the experiments of [5], SGDL was also not adopted as a baseline due to its high computational cost.
[2] https://www.adressa.no/
[3] https://www.yelp.com/dataset

Table 2: Recommendation Performance on early stop epoch. R stands for Recall and N stands for NDCG. Underlined values represent the best performing method, while Imprv indicates the percentage improvement of ERM-RE compared to standard ERM (not relative to sub-optimal methods).

Dataset		MovieLens				Adressa				Yelp			
Base Model	Method	R@5	R@20	N@5	N@20	R@5	R@20	N@5	N@20	R@5	R@20	N@5	N@20
GMF	ERM	0.0433	0.1178	0.0542	0.0738	0.1154	0.2178	0.0879	0.1233	0.0148	0.0436	0.0151	0.0245
	R-CE	0.0501	0.1196	0.0628	0.0794	0.0961	0.1557	0.0724	0.0915	0.0159	0.0456	0.0160	0.0257
	T-CE	0.0486	0.1147	0.0600	0.0766	0.1153	0.2177	0.0880	0.1234	0.0139	0.0451	0.0140	0.0243
	DeCA	0.0390	0.0977	0.0520	0.0675	0.1071	0.2147	0.0824	0.1190	-	-	-	-
	DeCAp	0.0430	0.1019	0.0522	0.0679	0.0987	0.2006	0.0751	0.1192	-	-	-	-
	DCF	0.0426	0.0972	0.0530	0.0668	0.1139	0.2145	0.0868	0.1216	0.0191	0.0572	0.0193	0.0316
	UDT	0.0420	0.1018	0.0545	0.0696	0.1285	0.2138	0.0916	0.1212	0.0160	0.0466	0.0158	0.0259
	ERM-RE	0.0485	0.1232	0.0632	0.0826	0.1173	0.2032	0.0850	0.1148	0.0141	0.0439	0.0142	0.0240
	Imprv	12.01%	4.58%	16.61%	11.92%	1.65%	-6.70%	-3.30%	-6.89%	-4.73%	0.69%	-5.96%	-2.08%
NeuMF	ERM	0.0439	0.1179	0.0542	0.0728	0.1619	0.3119	0.1257	0.1746	0.0116	0.0375	0.0115	0.0201
	R-CE	0.0543	0.1215	0.0647	0.0816	0.0805	0.1132	0.0691	0.0795	0.0125	0.0382	0.0124	0.0208
	T-CE	0.0453	0.1153	0.0537	0.0719	0.1631	0.3094	0.1243	0.1725	0.0114	0.0384	0.0114	0.0203
	DeCA	0.0070	0.0198	0.0085	0.0122	0.1702	0.3123	0.1314	0.1809	-	-	-	-
	DeCAp	0.0321	0.0801	0.0361	0.0516	0.1693	0.3140	0.1296	0.1788	-	-	-	-
	DCF	0.0328	0.0991	0.0430	0.0619	0.1611	0.3117	0.1254	0.1744	0.0157	0.0499	0.0157	0.0271
	UDT	0.0290	0.0937	0.0401	0.0599	0.1654	0.3106	0.1289	0.1766	0.0151	0.0451	0.0150	0.0248
	ERM-RE	0.0539	0.1180	0.0667	0.0823	0.1645	0.3088	0.1258	0.1734	0.0109	0.0369	0.0106	0.0192
	Imprv	22.78%	0.08%	23.06%	13.05%	1.61%	-0.99%	0.08%	-0.69%	-6.03%	-1.60%	-7.82%	-4.47%
CDAE	ERM	0.0418	0.1148	0.0525	0.0727	0.1688	0.3100	0.1303	0.1776	0.0197	0.0577	0.0201	0.0323
	R-CE	0.0444	0.1267	0.0561	0.0791	0.1487	0.2322	0.1171	0.1451	0.0200	0.0577	0.0201	0.0322
	T-CE	0.0416	0.1144	0.0523	0.0726	0.1696	0.3085	0.1305	0.1770	0.0224	0.0653	0.0227	0.0365
	DeCA	0.0354	0.1086	0.0445	0.0661	0.1619	0.3187	0.1216	0.1771	-	-	-	-
	DeCAp	0.0367	0.1096	0.0449	0.0668	0.1446	0.2752	0.1063	0.1499	-	-	-	-
	UDT	0.0364	0.1050	0.0408	0.0636	0.1747	0.3135	0.1325	0.1818	0.0252	0.0708	0.0255	0.0400
	ERM-RE	0.0479	0.1223	0.0606	0.0801	0.1688	0.3100	0.1303	0.1776	0.0203	0.0596	0.0191	0.0322
	Imprv	14.59%	6.53%	15.43%	10.18%	0.00%	0.00%	0.00%	0.00%	3.05%	3.29%	-4.98%	-0.31%
LightGCN	ERM	0.0559	0.1275	0.0680	0.0858	0.1238	0.2226	0.0937	0.1281	0.0189	0.0532	0.0192	0.0302
	R-CE	0.0525	0.1241	0.0660	0.0838	0.1214	0.2234	0.0920	0.1281	0.0127	0.0376	0.0132	0.0211
	T-CE	0.0567	0.1382	0.0693	0.0907	0.1214	0.2077	0.0930	0.1224	0.0146	0.0418	0.0159	0.0246
	DeCA	0.0390	0.0995	0.0420	0.0612	0.0728	0.1384	0.0542	0.0770	-	-	-	-
	DeCAp	0.0491	0.1231	0.0596	0.0806	0.1134	0.2251	0.0822	0.1209	-	-	-	-
	DCF	0.0432	0.1178	0.0581	0.0775	0.1247	0.2188	0.0946	0.1278	0.0194	0.0575	0.0200	0.0321
	UDT	0.0353	0.0932	0.0480	0.0626	0.1247	0.2247	0.0948	0.1299	0.0180	0.0514	0.0178	0.0286
	ERM-RE	0.0581	0.1364	0.0702	0.0915	0.1189	0.1992	0.0906	0.1171	0.0193	0.0547	0.0196	0.0311
	Imprv	3.94%	6.98%	3.24%	6.64%	-3.96%	-10.51%	-3.31%	-8.59%	2.12%	2.82%	2.08%	2.98%

We report results using two widely utilized metrics in the denoising recommendation domain: NDCG@K and Recall@K, with K values set at 5 and 20 for all datasets. To characterize the differences across different random seeds, we ran experiments 3 times with different seeds on MovieLens and Adressa, and 2 times on Yelp. It should be noted that our experiments require frequent global evaluations on the test set, making our work very time-consuming. Repeating the Yelp experiments twice already reached the upper limit of what our limited experimental equipment could handle.

4.2.3 Hyperparameters selections. Regarding the hyperparameters of the denoising methods, we strictly followed the settings claimed by the authors whenever they were mentioned in the respective papers. For the DeCA method, the authors provided sufficient hyperparameters, so we adopted their exact settings. Considering that the hyperparameters of denoising methods might be sensitive

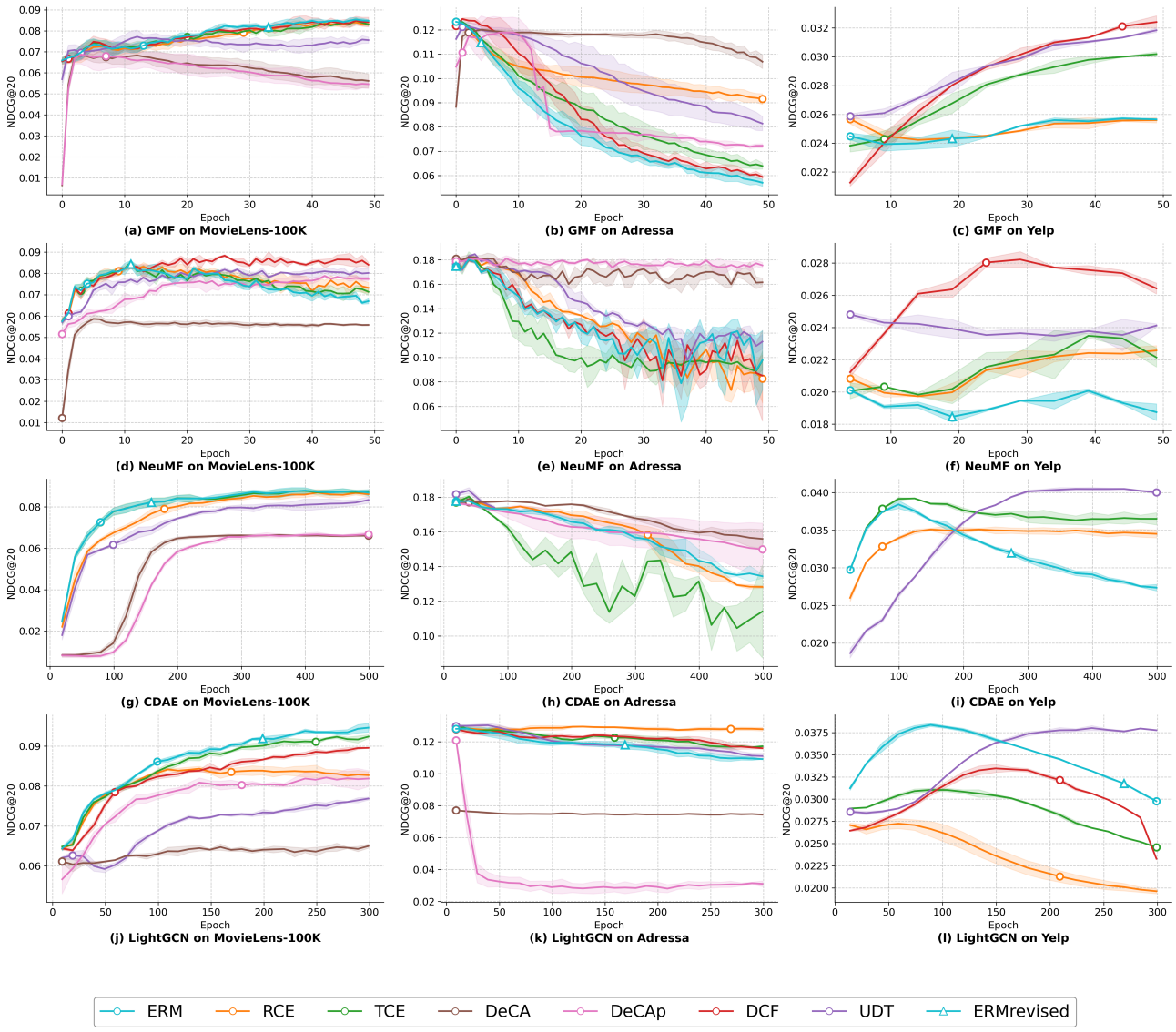
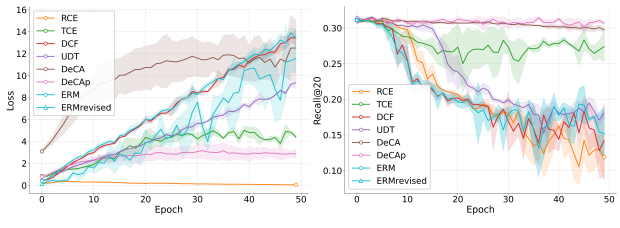


Figure 2: Evolution of NDCG@20 on test sets during model training for different recommendation models and datasets. Lines represent mean performance across multiple seeds, with shaded areas showing variance. Markers indicate early stopping points selected based on minimum validation loss.

to different datasets, we performed a degree of hyperparameter tuning for experimental settings where the authors did not provide parameters, to ensure that these methods were not unfairly underestimated. For the R-CE and T-CE methods, the authors provided hyperparameters for the Adressa and Yelp datasets. We conducted a simple grid search within the authors' suggested ranges for the Movielens dataset. For the DCF method, the authors provided hyperparameters for the GMF and NeuMF models but not for the LightGCN model. We performed a simple grid search on LightGCN and achieved good performance. For the UDT method, with its four hyperparameters, conducting a grid search across all possible combinations proved impractical. We applied the default parameters provided by the authors on GitHub across all scenarios, and UDT still achieved satisfactory performance, indicating that the UDT method is not particularly sensitive to parameter selection.



(a) Validation Loss (b) Recall@20

Figure 3: Performance of various denoising methods on the NeuMF model for the Adressa dataset.

5 Results and Analysis

Figure 2 shows the evolution of true performance[4][5] on the test set for various denoising methods during training The nodes on each line indicate the early stopping points selected based on minimum validation loss. Performance metrics at these early stopping points can be found in Table 2. Note that since the DCF method did not provide experiments on the CDAE model and is difficult to implement on the CDAE model, we don't report its performance.

Based on our experiments, we observe the following findings:

5.1 The Dilemma of Model Selection

By observing Figure 2, we have identified a systemic issue: existing model selection mechanisms based on validation set loss largely fail in denoising recommendation tasks. Specifically, almost all methods exhibit a phenomenon where the early stopping point (marked by nodes in the figures) does not align with the best performance point on the test set. With the exception of the DCF method performing well on the Yelp dataset, other scenarios do not show reliable model selection patterns. (The performance on the Adressa dataset is even more unusual, which we will discuss in the next section.)

This inconsistency further suggests that there may be complex distribution differences between validation and test sets, making model selection signals unreliable. Although robust machine learning in the computer vision field indicates that validating the reliability of model selection using noisy validation data under specific noise assumptions is possible [4], these prerequisites do not seem to be met in the recommender system domain. This phenomenon suggests two issues: (1) the benchmarking reported in existing papers on denoising recommendations may be unreliable, and (2) we need to rethink model selection strategies in environments with noise, potentially developing validation mechanisms more robust to distribution differences, or exploring adaptive training methods that do not require validation sets.

[4]The performance of the DeCA [36] is significantly worse compared to other methods, which is consistent with the experimental results reported in [5].

[5]Similar to [5], we found that DeCA [37] cannot execute with reasonable GPU storage on large datasets such as Yelp, therefore we are unable to provide results for DeCA on the Yelp dataset.

Method	R@5	R@20	N@5	N@20
R-CE	0.0501	0.1196	0.0628	0.0794
R-CE-RE	0.0471 ↓	0.1174 ↓	0.0581 ↓	0.0759 ↓
T-CE	0.0486	0.1147	0.0600	0.0766
T-CE-RE	0.0471 ↑	0.1174 ↑	0.0602 ↑	0.0759 ↓
DeCA	0.0390	0.0977	0.0520	0.0675
DeCA-RE	0.0413 ↑	0.0973 ↓	0.0526 ↑	0.0669 ↓
DeCAp	0.0430	0.1019	0.0522	0.0679
DeCAp-RE	0.0434 ↑	0.1027 ↑	0.0511 ↓	0.0675 ↓
DCF	0.0426	0.0972	0.0530	0.0668
DCF-RE	0.0474 ↑	0.1187 ↑	0.054 ↑	0.0736 ↑
UDT	0.0420	0.1018	0.0545	0.0696
UDT-RE	0.0445 ↑	0.1023 ↑	0.0563 ↑	0.0705 ↑

Table 3: Performance comparison of denoising methods with and without Revisiting Evaluation (RE) on the MovieLens dataset using GMF model. ↑ indicates improvement over the baseline method, ↓ indicate performance decrease.

5.2 Denoising Effectiveness in Highly Noisy Datasets

In Figure 2, we notice that all denoising methods on the Adressa dataset show a trend of performance degradation as training progresses. In this section, we will delve deeper and further discuss whether denoising methods can truly achieve robust models on datasets with low signal-to-noise ratios like Adressa. Figure 3 shows that on the Adressa dataset with a noise ratio as high as 60.67% (see Table 1), the performance of various denoising methods presents a concerning trend. First, Figure 3(a) demonstrates that almost all denoising methods show continuously increasing validation loss from the beginning of training, while Figure 3(b) shows a continuous decline in test set performance across all denoising methods, indicating that the model may not have truly learned robust representations against noisy interactions. More importantly, all methods achieve similar recall rates in the early training stages, but their performance continuously degrades as training epochs increase. Current denoising methods may not have truly learned how to handle noisy interactions. The final performance of the model seems to depend more on "luckily" stopping at a higher performance point in the early training stages, rather than truly learning the ability to distinguish between genuine preferences and noisy interactions. In fact, many denoising methods set the early stopping interval to be very small, which gives an ineffective denoising method the opportunity to randomly achieve high performance on the Adressa dataset. Noting that Adressa has the lowest signal-to-noise ratio among the three commonly used datasets (in addition to a noise rate as high as 60.67%, we observe that Adressa also has very limited interaction data per user), this suggests that existing denoising methods may have unnoticed limitations, and their applicability in extreme noise scenarios needs to be reassessed.

5.3 Effects of Model Selection in Denoising Recommendation

Analyzing the experimental results, we found that on the MovieLens dataset, our proposed ERM-RE method can achieve performance comparable to or even better than specially designed denoising methods. These results suggest that in denoising recommendation tasks, simply by improving validation set sampling methods, it may be possible to achieve performance comparable to complex denoising algorithms without introducing additional model complexity.

A natural question arises: can our modified evaluation method also be used to enhance existing denoising recommendation methods? Table 3 shows the performance comparison between original denoising methods and their RE versions (incorporating our proposed evaluation strategy) when using the GMF model on the MovieLens dataset. The results show that among the six denoising methods, T-CE, DCF, and UDT achieve performance improvements after incorporating the RE strategy. However, not all denoising methods benefited from this approach. We observed that the R-CE method experienced a slight decrease in performance after applying the RE strategy, possibly because R-CE itself already handles high-loss samples by adjusting sample weights, creating functional overlap with the RE strategy, which may lead to excessive denoising when combined.

In summary, these results indicate that improving model selection methods to further enhance denoising recommendation systems is feasible. However, different denoising methods have distinct core mechanisms and working principles, resulting in varying compatibility with the RE strategy. For methods that already have built-in sample reweighting mechanisms (such as R-CE), directly applying RE may not bring additional benefits and might even produce negative effects. Therefore, we believe future research could explore customized validation set sampling strategies tailored to different denoising methods, which may further improve the performance and stability of denoising recommendation systems.

5.4 Performance Inconsistencies Across Scenarios

Based on Figure 2, we observed a phenomenon worthy of attention: in current research, no single denoising method has been found to perform excellently across all experimental scenarios. This lack of a universal solution may reflect the complexity faced in the field of denoising recommendation.

When comparing specially designed denoising methods with the basic ERM method, experimental data indicates that the former has not demonstrated the expected widespread advantages. Notably, in specific dataset-model combinations, the basic ERM method combined with the RE strategy performs comparably to some complex denoising techniques, and in some cases even outperforms them.

This inconsistency in performance may be influenced by multiple factors, including but not limited to: differences in noise distribution across datasets, the compatibility between model architectures and denoising methods, and limitations in our current understanding of the nature of noise. Based on these findings, we believe future research directions may need to focus more on solutions targeted at specific scenarios rather than pursuing universally applicable methods. Additionally, in-depth research into noise generation mechanisms and their impact patterns may provide important insights for developing more effective denoising strategies.

6 Discussion

Based on our analysis in the previous section, we propose two pathways to enhance the reproducibility of denoising recommendation.

6.1 Underfitting Issue Due to Epoch Selection

Many previous works set the maximum number of training epochs to 10. This might be because validation loss tends to continually increase as the model fits noise, leading to model selection favoring earlier epochs, while increasing the number of epochs seemingly doesn't affect training results. However, this doesn't mean the model stops improving on the test set (as shown in Figure 2). In fact, except for the Adressa dataset, models trained for fewer than 10 epochs are underfitted on the test set.

From our experimental results, we observe that on the MovieLens dataset, almost all models show continuously improving performance trends within 50 epochs. The LightGCN model in particular continues to improve even after 300 epochs. This suggests that stopping training too early may prevent models from fully learning useful patterns in the data, thereby reducing recommendation performance. Therefore, we recommend researchers consider longer training cycles in their experimental design to give models sufficient time to learn true user preferences, especially on datasets with lower noise ratios.

6.2 Limitations of Fine-grained Early Stopping

Many previous works set very fine-grained early stopping criteria, evaluating models every few hundred iterations based on validation loss and treating iteration steps as hyperparameters. Given the high variance characteristics of models in early training stages, treating iteration steps as hyperparameters essentially performs grid search on test performance using unreliable validation set metrics. Our experimental results show that model performance often fluctuates during training, especially in early stages. As shown in Figure 2, different random seeds lead to performance curves with varying fluctuation patterns under the same model architecture, and overly fine-grained early stopping strategies may result in model selections based on accidentally captured performance peaks rather than true denoising capabilities.

To improve reproducibility of denoising recommendation methods, we suggest researchers adopt coarser-grained early stopping strategies, such as evaluating once per epoch rather than every few hundred iterations. Researchers should also clearly distinguish between true denoising effects and simple hyperparameter optimization effects, avoiding tuning the number of training steps as an implicit hyperparameter to more accurately evaluate the effectiveness of different denoising methods.

7 Summary and Future work

In this study, we examined the effectiveness and stability of denoising methods in recommendation systems, finding that model selection mechanisms based on validation set loss are generally

unreliable in denoising recommendation tasks. Experiments show significant mismatches between minimum validation loss points and actual best performance points on the test set, especially in high-noise datasets. This distribution shift between validation and test sets may be one of the fundamental reasons for the unstable performance of existing denoising methods, and is a key issue currently overlooked in research. In future work, we will be interested in discovering the patterns in generalization capabilities of denoising recommendation across more datasets, despite the high computational costs associated with frequent global unbiased evaluations on large-scale datasets.

We hope our work can draw the research community's attention to the model selection issue in denoising recommendation, encourage a re-examination of the basic assumptions behind existing denoising methods, and ensure that future denoising recommendation research can be built on the foundation of reliable evaluation.

Acknowledgments

This work is supported by ARC Discovery Project DP220102635 and Laureate Fellow Project FL190100149.

References

[1] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2019. Managing popularity bias in recommender systems with personalized re-ranking. *arXiv preprint arXiv:1901.07555* (2019).

[2] Huiyuan Chen, Yusan Lin, Menghai Pan, Lan Wang, Chin-Chia Michael Yeh, Xiaoting Li, Yan Zheng, Fei Wang, and Hao Yang. 2022. Denoising self-attentive sequential recommendation. In *Proceedings of the 16th ACM conference on recommender systems*. 92–101.

[3] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and debias in recommender system: a survey and future directions (2020). *arXiv preprint arXiv:2010.03240* (2020).

[4] Pengfei Chen, Junjie Ye, Guangyong Chen, Jingwei Zhao, and Pheng-Ann Heng. 2021. Robustness of accuracy metric and its inspirations in learning with noisy labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11451–11461.

[5] Haoyan Chua, Yingpeng Du, Zhu Sun, Ziyan Wang, Jie Zhang, and Yew-Soon Ong. 2024. Unified Denoising Training for Recommendation. In *Proceedings of the 18th ACM Conference on Recommender Systems*. 612–621.

[6] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. 2021. A troubling analysis of reproducibility and progress in recommender systems research. *ACM Transactions on Information Systems (TOIS)* 39, 2 (2021), 1–49.

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

[8] Yunjun Gao, Yuntao Du, Yujia Hu, Lu Chen, Xinjun Zhu, Ziquan Fang, and Baihua Zheng. 2022. Self-guided learning to denoise for robust recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 1412–1422.

[9] Yingqiang Ge, Mostafa Rahmani, Athirai Irissappane, Jose Sepulveda, James Caverlee, and Fei Wang. 2023. Automated data denoising for recommendation. *arXiv preprint arXiv:2305.07070* (2023).

[10] Jon Atle Gulla, Lemei Zhang, Peng Liu, Özlem Özgöbek, and Xiaomeng Su. 2017. The adressa dataset for news recommendation. In *Proceedings of the international conference on web intelligence*. 1042–1048.

[11] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems* 31 (2018).

[12] F Maxwell Harper and Joseph A Konstan. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)* 5, 4 (2015), 1–19.

[13] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. 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*. 639–648.

[14] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. 2018. Adversarial personalized ranking for recommendation. In *The 41st International ACM SIGIR conference on research & development in information retrieval*. 355–364.

[15] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[16] Zhuangzhuang He, Yifan Wang, Yonghui Yang, Peijie Sun, Le Wu, Haoyue Bai, Jinqi Gong, Richang Hong, and Min Zhang. 2024. Double correction framework for denoising recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1062–1072.

[17] Katja Hofmann, Fritz Behr, and Filip Radlinski. 2012. On caption bias in interleaving experiments. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. 115–124.

[18] Rolf Jagerman, Harrie Oosterhuis, and Maarten de Rijke. 2019. To model or to intervene: A comparison of counterfactual and online learning to rank from user interactions. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*. 15–24.

[19] Youngho Kim, Ahmed Hassan, Ryen W White, and Imed Zitouni. 2014. Modeling dwell time to predict click-level satisfaction. In *Proceedings of the 7th ACM international conference on Web search and data mining*. 193–202.

[20] Dongha Lee, SeongKu Kang, Hyunjun Ju, Chanyoung Park, and Hwanjo Yu. 2021. Bootstrapping user and item representations for one-class collaborative filtering. In *Proceedings of the 44th international ACM SIGIR conference on Research and Development in information retrieval*. 317–326.

[21] Lukas Lerche. 2016. Using implicit feedback for recommender systems: characteristics, applications, and challenges. (2016).

[22] Weilin Lin, Xiangyu Zhao, Yejing Wang, Yuanshao Zhu, and Wanyu Wang. 2023. Autodenoise: Automatic data instance denoising for recommendations. In *Proceedings of the ACM Web Conference 2023*. 1003–1011.

[23] Yiyu Liu, Qian Liu, Yu Tian, Changping Wang, Yanan Niu, Yang Song, and Chenliang Li. 2021. Concept-aware denoising graph neural network for micro-video recommendation. In *Proceedings of the 30th ACM international conference on information & knowledge management*. 1099–1108.

[24] Hongyu Lu, Min Zhang, and Shaoping Ma. 2018. Between clicks and satisfaction: Study on multi-phase user preferences and satisfaction for online news reading. In *The 41st international acm sigir conference on research & development in information retrieval*. 435–444.

[25] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. 2017. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1944–1952.

[26] Yuqi Qin, Pengfei Wang, and Chenliang Li. 2021. The world is binary: Contrastive learning for denoising next basket recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 859–868.

[27] Faisal Shehzad and Dietmar Jannach. 2023. Everyone'sa winner! on hyperparameter tuning of recommendation models. In *Proceedings of the 17th ACM Conference on Recommender Systems*. 652–657.

[28] Yanchao Tan, Carl Yang Member, Xiangyu Wei, Ziyue Wu, and Xiaolin Zheng. 2022. Partial relaxed optimal transport for denoised recommendation. *arXiv preprint arXiv:2204.08619* (2022).

[29] Changxin Tian, Yuexiang Xie, Yaliang Li, Nan Yang, and Wayne Xin Zhao. 2022. Learning to denoise unreliable interactions for graph collaborative filtering. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. 122–132.

[30] Gabriele Tolomei, Mounia Lalmas, Ayman Farahat, and Andrew Haines. 2019. You must have clicked on this ad by mistake! Data-driven identification of accidental clicks on mobile ads with applications to advertiser cost discounting and click-through rate prediction. *International Journal of Data Science and Analytics* 7 (2019), 53–66.

[31] William Toner and Amos Storkey. 2024. Noisy Early Stopping for Noisy Labels. *arXiv preprint arXiv:2409.06830* (2024).

[32] Pengfei Wang, Chenliang Li, Lixin Zou, Zhichao Feng, Kaiyuan Li, Xiaochen Li, Xialong Liu, and Shangguang Wang. 2023. Tutorial: Data Denoising Metrics in Recommender Systems. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 5224–5227.

[33] Shuyao Wang, Zhi Zheng, Yongduo Sui, and Hui Xiong. 2025. Unleashing the Power of Large Language Model for Denoising Recommendation. In *Proceedings of the ACM on Web Conference 2025*. 252–263.

[34] Tianle Wang, Lianghao Xia, and Chao Huang. 2023. Denoised self-augmented learning for social recommendation. *arXiv preprint arXiv:2305.12685* (2023).

[35] Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. 2021. Denoising implicit feedback for recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*. 373–381.

[36] Yu Wang, Xin Xin, Zaiqiao Meng, Joemon M Jose, Fuli Feng, and Xiangnan He. 2022. Learning robust recommenders through cross-model agreement. In *Proceedings of the ACM web conference 2022*. 2015–2025.

[37] Zongwei Wang, Min Gao, Wentao Li, Junliang Yu, Linxin Guo, and Hongzhi Yin. 2023. Efficient bi-level optimization for recommendation denoising. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*. 2502–2511.

[38] Hongyi Wen, Longqi Yang, and Deborah Estrin. 2019. Leveraging post-click feedback for content recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 278–286.

[39] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*. 726–735.

[40] Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. 2016. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the ninth ACM international conference on web search and data mining*. 153–162.

[41] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. 2019. Are anchor points really indispensable in label-noise learning? *Advances in neural information processing systems* 32 (2019).

[42] Suqin Yuan, Lei Feng, and Tongliang Liu. 2025. Early stopping against label noise without validation data. *arXiv preprint arXiv:2502.07551* (2025).

[43] Kaike Zhang, Qi Cao, Fei Sun, Yunfan Wu, Shuchang Tao, Huawei Shen, and Xueqi Cheng. 2023. Robust recommender system: a survey and future directions. *arXiv preprint arXiv:2309.02057* (2023).

[44] Kaike Zhang, Qi Cao, Yunfan Wu, Fei Sun, Huawei Shen, and Xueqi Cheng. 2025. Personalized Denoising Implicit Feedback for Robust Recommender System. In *Proceedings of the ACM on Web Conference 2025*. 4470–4481.

[45] Yuxiang Zheng, Zhongyi Han, Yilong Yin, Xin Gao, and Tongliang Liu. 2024. Can We Treat Noisy Labels as Accurate? *arXiv preprint arXiv:2405.12969* (2024).