Beyond Top-1: Addressing Inconsistencies in Evaluating Counterfactual Explanations for Recommender Systems

Amir Reza Mohammadi
University of Innsbruck
Innsbruck, Austria
amir.reza@uibk.ac.at

Andreas Peintner
University of Innsbruck
Innsbruck, Austria
andreas.peintner@uibk.ac.at

Michael Müller
University of Innsbruck
Innsbruck, Austria
Michael.M.Mueller@uibk.ac.at

Eva Zangerle
University of Innsbruck
Innsbruck, Austria
eva.zangerle@uibk.ac.at

Abstract

Explainability in recommender systems (RS) remains a pivotal yet challenging research frontier. Among state-of-the-art techniques, counterfactual explanations stand out for their effectiveness, as they show how small changes to input data can alter recommendations, providing actionable insights that build user trust and enhance transparency. Despite their growing prominence, the evaluation of counterfactual explanations in RS is far from standardized. Specifically, existing metrics show inconsistency since they are affected by variations in the performance of the underlying recommenders. Hence, we critically examine the evaluation of counterfactual explainers through consistency as the key principle of effective evaluation. Through extensive experiments, we assess how going beyond top-1 recommendation and incorporating top-k recommendations impacts the consistency of existing evaluation metrics. Our findings reveal factors that impact the consistency of existing evaluation metrics and offer a step toward effectively mitigating the inconsistency problem in counterfactual explanation evaluation.

CCS Concepts

• **Information systems** → **Recommender systems**.

Keywords

Recommender Systems, Counterfactual Explanation, Evaluation

ACM Reference Format:
Amir Reza Mohammadi, Andreas Peintner, Michael Müller, and Eva Zangerle. 2025. Beyond Top-1: Addressing Inconsistencies in Evaluating Counterfactual Explanations for Recommender Systems. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems (RecSys '25), September 22–26, 2025, Prague, Czech Republic.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3705328.3748028

1 Introduction

As the demand for transparency and interpretability in AI continues to grow [2, 10], eXplainable Artificial Intelligence (XAI) has emerged as a critical research domain. Within XAI, counterfactual explanations (CE) [7] have become powerful tools for advancing model interpretability. By illustrating how small, targeted changes to the input can lead to different model outputs, CEs provide intuitive, user-friendly insights into model behavior, fostering trust and understanding among end-users [17]. While much of the foundational work on CEs has focused on tabular or image data [15, 24, 36], there is an increasing shift toward adapting these methods to RS [9, 27, 30, 37, 38].

Despite growing interest, the evaluation of CEs in RS remains underexplored [6], lacking established standard evaluation setups and protocols, which has led to significant inconsistencies across current evaluation practices [22]. The typical approach involves training a recommender model (e.g., MF) on a dataset (e.g., ML-1M), followed by a CE method that generates counterfactual instances to alter the model's recommendations. However, as shown in a recent study [22], since counterfactual explainers aim to modify the input to change the model's output, the quality and behavior of the recommender itself, inevitably affect the resulting explanations. This dependency can be illustrated with a simple thought experiment. Suppose the recommender is entirely untrained and behaves randomly; in such a case, even arbitrary changes to the input would likely produce different outputs, misleadingly suggesting that the explainer is highly effective. While this dependency might be acceptable if the relative performance of different CE methods remained consistent—i.e., if method A consistently outperformed method B regardless of recommender quality. However, [22] demonstrated that the relative ranking of CE methods varies with the quality of the recommender, meaning that method A may not consistently outperform method B across different recommender qualities. Instead, the relative rankings of explanation methods vary depending on the recommender's performance level. This leads to inconsistencies in the evaluation of CE methods. *Consistency* in this context refers to an evaluation metric's ability to produce stable ranking across CE methods, despite changes in the performance of the recommender model. For example, if explainer method E_1 outperforms method E_2 in explaining a recommender R_1, then after minor changes in performance of RS, while the absolute performance of CEs may shift, E_1 should still outperform E_2 in relative terms. Although [22] has highlighted these inconsistencies in evaluation practices, the underlying causes have not been addressed.

RecSys '25, Prague, Czech Republic

ACM ISBN 979-8-4007-1364-4/25/09
https://doi.org/10.1145/3705328.3748028

In this paper, we hypothesize that extending evaluation to include lower-ranked recommendations—such as the top-3—can enhance the consistency of CE assessments. We critically examine standard evaluation protocols, highlighting a key limitation: the prevalent focus on changes in the top-1 recommendation. This practice, likely inherited from vision and graph neural network (GNN) research [14, 32], overlooks key characteristics unique to recommender systems—most notably—their inherently ranked outputs. Recent work has shown that relying solely on the top-1 item can amplify prediction volatility and randomness [19], especially in settings with lower model performance. To address this, we investigate how expanding evaluations to consider the top-k items ($1 \leq k \leq 5$) affects metric consistency.

We conduct extensive experiments across two recommender models, six state-of-the-art CE methods, and three widely used datasets. Our results demonstrate that top-k-based evaluation yields consistent and representative assessments, better aligning with the ranked nature of recommender outputs and real-world usage scenarios.

2 Related Work

The field of interpretable machine learning has witnessed substantial progress [9, 11, 27, 34]. Early research predominantly emphasized enhancing the interpretability of the models themselves [25, 26]. In parallel, the concept of "explainability" gained prominence, referring to approaches that treat the model as a black box and aim to elucidate their behavior retrospectively through *post-hoc* explanations [16, 35]. Counterfactual explanations have emerged as a popular approach to post-hoc explainability in RS, offering actionable insights by modifying user inputs to change model predictions [4, 14, 20, 21, 32]. Recent methods range from similarity-based heuristics [26] to model-agnostic frameworks like ACCENT [34] and LXR [5]. However, evaluation practices for CEs remain fragmented. Prior studies typically assess CE quality based on whether they can flip the top-1 recommendation [5, 29, 33, 38], often ignoring the ranked nature of recommender outputs. Furthermore, recent work [22] shows that CE evaluation can be inconsistent—changing the recommender's performance may alter the perceived ranking of explainers. In contrast, our work introduces a list-wise evaluation metric that improves consistency by considering multiple top-ranked items. This fills a critical gap in how CE methods are compared, moving toward more reliable and reproducible benchmarks for explainability in RS.

3 Methodology

To investigate the impact of extending the evaluation from top-1 items to top-k items, we propose the following approach.

3.1 Problem Setup and Preliminaries

Let $\mathcal{U}$ denote the set of users and $\mathcal{I}$ the set of items. Each user $u \in \mathcal{U}$ is represented by a binary interaction vector $\mathbf{x}_u \in \{0,1\}^{|\mathcal{I}|}$, where $\mathbf{x}_u[i] = 1$ indicates that user u has interacted with item i. A recommender model f_θ maps this input to a predicted score vector, with $f_\theta(\mathbf{x}_u)[i]$ representing the predicted relevance of item i. The RS generates a ranked list of recommendations for each user. We denote the top-k recommended items as $\mathcal{R}_u^k = \{i_1^*, i_2^*, \ldots, i_k^*\}$,

Algorithm 1: Consistent Evaluation of CE Methods

Input: Explainer e, recommender f_θ, test users $\mathcal{U}$, rank threshold $T \in \{5, 10, 20\}$, top-k recommendations tk
Output: Evaluation score POS-P@T
Load recommender parameters $f_\theta^{(c)}$;
foreach *rank threshold* $T \in \{5, 10, 20\}$ **do**
  Initialize accumulators: POS_total $\leftarrow 0$;
  foreach *user* $u \in \mathcal{U}$ **do**
    $\mathbf{x}_u \leftarrow$ user's interaction vector ;
    $\mathcal{R}_u^k \leftarrow$ top-k recommendations from $f_\theta^{(c)}(\mathbf{x}_u)$;
    foreach $tk \in \mathcal{R}_u^k$ **do**
      $i_{tk}^* \leftarrow tk$-th item in $\mathcal{R}_u^k$;
      $CE \leftarrow e(\mathbf{x}_u, i_{tk}^*)$
      Generate positive sequence $\{\mathbf{x}_u^{(1)}, \ldots\}$ by removing most relevant items from CE ;
      $\text{rank}_{\text{pos}} \leftarrow \text{rank}(i^*; \mathbf{x}_u^{(t)})$;
      POS_total += $\mathbb{1}[\text{rank}_{\text{pos}} > k]$;
    end
  end
  POS-P@T$_{(k)}$ $\leftarrow$ POS_total$/(|\mathcal{U}| \cdot k)$
end
return All POS-P@T$_{(k)}$

where i_j^* is the j-th item in the ranked recommendation list. These are the items for which we seek to evaluate CEs.

Given a counterfactual explainer e, we obtain a relevance ranking over user history items with respect to each item i_j^* in $\mathcal{R}_u^k$. Based on this relevance, we iteratively perturb the user input vector $\mathbf{x}_u$ by removing one item at a time, yielding a sequence $\mathbf{x}_u^{(t)}$ of perturbed inputs. Following the literature [5, 22], We compute the following metrics:

Positive Perturbation (POS-P@T). This metric quantifies how quickly the top-k recommended items fall below the rank threshold T during perturbation. For user u, the POS-P@T score is:

$$\text{POS-P@T}_u = \sum_{j=1}^{k} \sum_{t=1}^{|\mathbf{x}_u|} \mathbb{1}\left[\text{rank}(i_j^*; \mathbf{x}_u^{(t)}) > T\right] \tag{1}$$

where $\text{rank}(i_j^*; \mathbf{x}_u^{(t)})$ is the rank of i_j^* in the output of f_θ given the perturbed input $\mathbf{x}_u^{(t)}$ and the indicator function $\mathbb{1}[\cdot]$ checks whether the item still appears within the rankd threshold T. The key difference between this formulation and the literature [5, 22] lies in computing the inner value over the top-k items ($\sum_{j=1}^{k}$) rather than only the top-1 item.

The final POS-P@T score is the average over all users:

$$\text{POS-P@T} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \text{POS-P@T}_u \tag{2}$$

We have detailed the computation process of this metric in Algorithm 1.

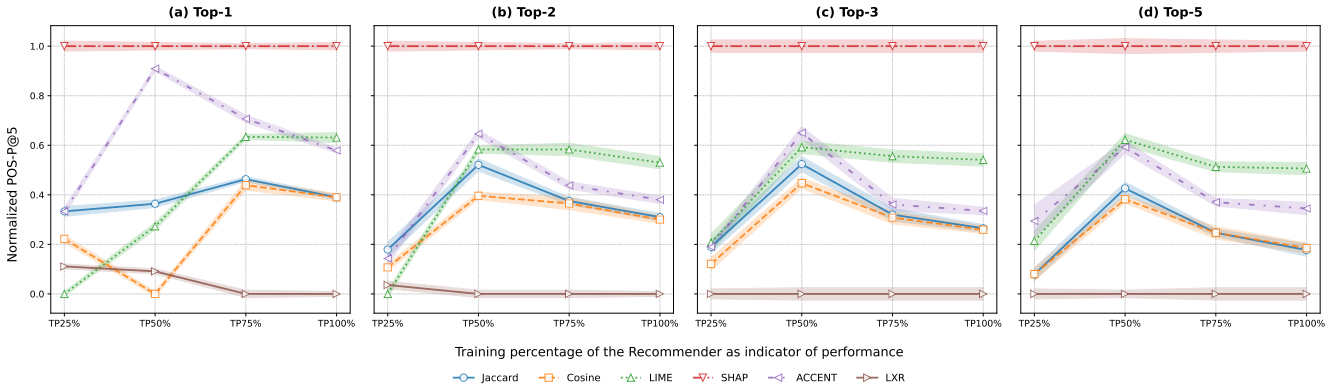

Figure 1: Comparison of CE methods based on POS@5 ↓ (lower value is the better) across four performance levels of the VAE recommender on the ML-1M dataset. The figure shows the impact of going beyond top-1 (a) and considering top-k (b-d) recommendations on improving consistency when evaluating CE models. To facilitate clearer comparisons, the values are normalized using Min-max normalization, and shading is used to represent the variance in the results.

Negative Perturbation (NEG-P@T). Analogous to the positive perturbation setting, we define the NEG-P@K metric to evaluate the robustness of the top-m recommended items under negative perturbations. Here, instead of removing the most important items, we iteratively remove the least important items from the user's interaction vector $\mathbf{x}_u$, again based on the importance ranking returned by the counterfactual explainer e. This metric evaluates how well each top-k items ($1 \leq k \leq 5$) maintain their position within the T rank threshold under negative perturbation, where less relevant items are removed:

$$\text{NEG-P@T}_u = \sum_{j=1}^{k} \sum_{t=1}^{|\mathbf{x}_u|} \mathbb{1}\left[\text{rank}(i_j^*; \mathbf{x}_u^{(t)}) \leq T\right] \tag{3}$$

where $\mathbf{x}_u^{(t)}$ now represents the user interaction vector after removing the t-th *least important* item, and $\text{rank}(i_j^*; \mathbf{x}_u^{(t)})$ is the resulting rank of the explained item and the overall NEG-P@T score is:

$$\text{NEG-P@T} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \text{NEG-P@T}_u \tag{4}$$

3.2 Evaluation Setup

We use the MovieLens 1M (ML-1M) [12], Yahoo! Music [8] and Pinterest [13] datasets following the literature [5, 22], focusing on collaborative filtering models based on implicit user-item interactions. To simulate implicit ratings in ML-1M, following [5], we retained only ratings of 3.5 or higher and included users and items with at least two ratings. This resulted in 575,128 ratings from 6,037 users across 3,381 items. For the Yahoo! Music dataset, we retained ratings of 70 or higher, and following [5] we took a random sample of 486,744 ratings from 19,155 users for 9,362 items from the Pinterest dataset.

For each user, we generate a ranked list of recommendations and evaluate the explanation of the top-k recommendations. For each k value, we use four levels of recommender performance by training the recommender based on Hit Rate and saving model checkpoints after 25%, 50%, 75%, and full training, to simulate recommendations of various quality levels. We evaluate 24 configurations (3 dataset x 2 Recommenders x 4 performance levels), repeat our experiments three times, and report the average performance. We share our dataset processing scripts, the source code, and the hyperparameters[1].

3.3 Evaluated Methods

Our evaluation includes a range of CE methods, from established baselines to recent advancements. Specifically, we tested two similarity-based approaches, two traditional explainers, and two state-of-the-art methods:

Jaccard [3]/*Cosine* [31] Similarity generate explanations by comparing an item to those in the user's history using Jaccard [3] or Cosine [31] similarity metrics based on co-interacted users.

LIME-RS [23] is a modified version of the LIME framework designed specifically for recommender systems. It produces explanations by approximating the complex recommendation model with a locally interpretable linear model, capturing relationships within a neighborhood around the user's data. We applied LIME-RS to the user's historical interactions, carefully adjusting the number of samples based on performance on the validation set.

SHAP (SHapley Additive exPlanations) [38] quantifies the influence of individual features on a model's predictions. By leveraging game theory, it assigns a value to each feature according to its contribution to the final output. SHAP, though effective, is computationally intensive due to the exponential number of possible perturbations in a user's history.

ACCENT (Action-based Counterfactual Explanations for Neural Recommenders for Tangibility) [34] is a CE framework that extends influence functions to provide actionable, model-agnostic insights for neural recommenders. It extends techniques originally designed for latent factor models, making them applicable to a broader class of neural recommender systems.

[1] https://github.com/dbis-uibk/CE4RS-Eval

LXR [5], the state-of-the-art CE approach, employs self-supervised learning to generate explanation masks that highlight critical user data without perturbations using a gradient learning approach.

Our study includes two collaborative filtering recommenders: (1) *Matrix Factorization (MF)*, which decomposes the user-item interaction matrix into lower-dimensional latent factors [1]. Recent reproducibility studies indicate that MF continues to achieve competitive performance [28]. We used the implementation from [5] for easier comparison. (2) *Variational Autoencoder (VAE)*, a generative model that encodes and decodes data while learning a probabilistic latent representation [18]. VAE-based models have demonstrated strong performance in collaborative filtering tasks. In our experiments, we adopt a VAE-based recommender with an architecture similar to that of [18].

Table 1: *CE methods performance for Explaining the MF Recommender on the Yahoo! Music Dataset based on top-3 ranks.* The best results in each metric are highlighted in bold, and the second-best results are *underlined*. Arrows next to the metrics indicate performance direction: a downward arrow (↓) signifies that lower values are better, while an upward arrow (↑) signifies that higher values are better. As shown in the table, the performance of the explainers are consistent across different recommender performance levels, shown as training percentage.

	Method	T = 5		T = 10	
		POS ↓	NEG ↑	POS ↓	NEG ↑
TP50%	Jaccard	0.767	0.837	0.874	0.921
	Cosine	0.766	0.836	0.876	0.920
	LIME-RS	0.779	<u>0.840</u>	0.882	<u>0.924</u>
	SHAP	0.813	0.820	0.915	0.902
	ACCENT	<u>0.751</u>	0.830	<u>0.856</u>	0.910
	LXR	**0.743**	**0.861**	**0.829**	**0.929**
TP75%	Jaccard	0.507	0.751	0.623	0.815
	Cosine	0.502	<u>0.753</u>	0.618	<u>0.817</u>
	LIME-RS	0.488	0.715	0.544	0.798
	SHAP	0.631	0.665	0.712	0.767
	ACCENT	<u>0.459</u>	0.7310	<u>0.538</u>	0.800
	LXR	**0.447**	**0.764**	**0.529**	**0.827**
TP100%	Jaccard	0.463	0.759	0.512	0.795
	Cosine	0.457	<u>0.776</u>	0.507	<u>0.801</u>
	LIME-RS	0.476	0.765	0.527	0.793
	SHAP	0.618	0.616	0.666	0.663
	ACCENT	<u>0.453</u>	0.692	<u>0.491</u>	0.748
	LXR	**0.431**	**0.792**	**0.471**	**0.811**

4 Experiments and Results

Based on our hypothesis that extending evaluations to include lower-ranked recommendations—such as the top-3 items—enhances consistency, we incorporate lower-ranked items in the evaluation. Particularly, we compute performance separately for top-1, top-2, top-3, and higher-ranked recommendations, then aggregate these performance values to obtain a comprehensive assessment. For instance, computing the value for top-3 involves selectively masking data from the user's history to ensure that the third-highest recommended item drops to a lower rank (rank ≥ 4). This method ensures that the evaluation captures the influence of multiple recommendation ranks rather than relying solely on the highest-ranked item, leading to a more stable comparison of CE methods.

To test this hypothesis, we evaluate six representative CE methods using the POS and NEG metrics, analyzing their performance across four thresholds of recommendation length (top-k) and four levels of recommender performance. As shown in Figure 1, evaluations based solely on the top-1 recommendation (Fig. 1a) exhibit significant fluctuations on every two recommenders, particularly when transitioning from the lowest-performing recommender (TP25%) to TP50%. However, when extending the evaluation to the top-2 recommendations (Fig. 1b), we can already observe more consistency, especially for high-performing recommenders (TP50%–TP100%). Following this, setting the threshold to include the top-5 recommendations (Fig. 1d) stabilizes the evaluation outcomes for the highest performing recommenders (TP50%–TP100%). We observed a similar pattern in other configurations, albeit with different thresholds. For the MF recommender, the ranking of methods remained stable even at smaller k values across all datasets, in contrast to the VAE. An example of this observation is presented in Table 1, which shows the performance of CE methods across three levels of recommender performance (determined by training percentage). The results, based on top-3 values, demonstrate the consistency of the comparative rankings among the methods. The observed consistency when considering only the top-3 items stems from the simplicity of the MF model, which results in a more straightforward recommendation task and lower performance variability among recommenders, ultimately leading to less variability in explainers. Additionally, for the NEG metric, we found that a lower k, specifically $k = 2$, consistently yielded stable performance across all recommender configurations. More detailed results are available in the accompanying Git repository presented in section 3.2. Overall, this evidence highlights the importance of considering list-wise recommendations in CE evaluations. Adopting such an approach not only enhances evaluation consistency but also ensures better alignment with the inherent characteristics of recommendation systems.

Importantly, our findings show that evaluation consistency is not uniformly affected across all metrics. POS, which measures how quickly the explained item drops out of the top-K list under positive perturbations, is particularly sensitive to the quality of the recommender. In contrast, NEG, which checks whether the explained item remains within the top-K under less relevant perturbations, proves to be more stable even when the recommender is weaker. These results suggest that the choice of metric—and the value of k—should be treated as a tunable hyperparameter, influenced by both the dataset and the recommender architecture.

Moreover, the use of multiple performance checkpoints for the recommender further reinforces the robustness of our evaluation framework. By capturing performance across training stages (25-100%), we ensure that the evaluation of explainers is not biased by any single fixed model state. This approach not only provides a more comprehensive view of the explainer's effectiveness but

also reveals how susceptible existing metrics are to fluctuations in model quality.

5 Conclusion and Future Work

In this paper, we address a critical methodological gap in the evaluation of counterfactual explanations for recommender systems, focusing on evaluation consistency. Our findings highlight the importance of considering top-k recommendations when assessing CEs. By extending evaluations beyond the top-1 recommendation, we demonstrate consistency on current mainstream metrics for evaluation. These findings not only address key methodological gaps in counterfactual evaluation but also lay the groundwork for future research into domain-adaptive CE metrics.

An important direction for future work is the development of unified evaluation metrics that jointly consider both recommender performance and explanation quality. As our study highlights, the effectiveness of counterfactual explanation is closely tied to the behavior and accuracy of the underlying recommender system. Evaluating explanations in isolation may lead to misleading conclusions, especially when the recommender is poorly calibrated or underperforms. By integrating recommendation performance metrics—such as Hit Rate—with explanation metrics, we can better assess the fidelity and trustworthiness of explanations in context. Such composite metrics would not only provide a more holistic view of system behavior but also help identify explanation methods that maintain reliability across varying levels of recommendation quality.

References

[1] Mohamed Hussein Abdi, George Onyango Okeyo, and Ronald Waweru Mwangi. 2018. Matrix Factorization Techniques for Context-Aware Collaborative Filtering Recommender Systems: A Survey. *Comput. Inf. Sci.* 11, 2 (2018), 1–10.

[2] Saugat Aryal and Mark T. Keane. 2024. Even-Ifs from If-Onlys: Are the Best Semi-factual Explanations Found Using Counterfactuals as Guides?. In *Case-Based Reasoning Research and Development - 32nd International Conference, ICCBR 2024, Merida, Mexico, July 1-4, 2024, Proceedings (Lecture Notes in Computer Science, Vol. 14775)*, Juan A. Recio-García, Mauricio Gabriel Orozco-del-Castillo, and Derek Bridge (Eds.). Springer, 33–49. doi:10.1007/978-3-031-63646-2_3

[3] Sujoy Bag, Sri Krishna Kumar, and Manoj Kumar Tiwari. 2019. An efficient recommendation generation using relevant Jaccard similarity. *Inf. Sci.* 483 (2019), 53–64.

[4] Mohit Bajaj, Lingyang Chu, Zi Yu Xue, Jian Pei, Lanjun Wang, Peter Cho-Ho Lam, and Yong Zhang. 2021. Robust Counterfactual Explanations on Graph Neural Networks. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual.* 5644–5655. https://proceedings.neurips.cc/paper/2021/hash/2c8c3a57383c63caef6724343eb62257-Abstract.html

[5] Oren Barkan, Veronika Bogina, Liya Gurevitch, Yuval Asher, and Noam Koenigstein. 2024. A Counterfactual Framework for Learning and Evaluating Explanations for Recommender Systems. In *WWW '24*. ACM, 3723–3733.

[6] Christine Bauer, Eva Zangerle, and Alan Said. 2024. Exploring the Landscape of Recommender Systems Evaluation: Practices and Perspectives. *Trans. Recomm. Syst.* 2, 1 (2024), 11:1–11:31. doi:10.1145/3629170

[7] Finale Doshi-Velez and Been Kim. 2017. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv preprint arXiv:1702.08608* (2017).

[8] Gideon Dror, Noam Koenigstein, Yehuda Koren, and Markus Weimer. 2012. The Yahoo! Music Dataset and KDD-Cup '11. In *KDD Cup 2011 (JMLR Proceedings, Vol. 18)*. JMLR.org, 8–18.

[9] Azin Ghazimatin, Oana Balalau, Rishiraj Saha Roy, and Gerhard Weikum. 2020. PRINCE: Provider-side Interpretability with Counterfactual Explanations in Recommender Systems. In *WSDM '20*. ACM, 196–204.

[10] Amirata Ghorbani, James Wexler, James Y. Zou, and Been Kim. 2019. Towards Automatic Concept-based Explanations. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). 9273–9282. https://proceedings.neurips.cc/paper/2019/hash/77d2afcb31f6493e350fca61764efb9a-Abstract.html

[11] Amirata Ghorbani and James Y. Zou. 2020. Neuron Shapley: Discovering the Responsible Neurons. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.* https://proceedings.neurips.cc/paper/2020/hash/41c542dfe6e4fc3deb251d64cf6ed2e4-Abstract.html

[12] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (dec 2015), 19 pages.

[13] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *WWW '17*. ACM, 173–182.

[14] Zexi Huang, Mert Kosan, Sourav Medya, Sayan Ranu, and Ambuj K. Singh. 2023. Global Counterfactual Explainer for Graph Neural Networks. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023*. ACM, 141–149. doi:10.1145/3539597.3570376

[15] Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. 2024. Text-to-Image Models for Counterfactual Explanations: A Black-Box Approach. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 4757–4767.

[16] Vassilis Kaffes, Dimitris Sacharidis, and Giorgos Giannopoulos. 2021. Model-Agnostic Counterfactual Explanations of Recommendations. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, UMAP 2021, Utrecht, The Netherlands, June, 21-25, 2021*. ACM, 280–285. doi:10.1145/3450613.3456846

[17] Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. 2021. Algorithmic Recourse: from Counterfactual Explanations to Interventions. In *FAccT '21*. ACM, 353–362.

[18] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. 2018. Variational Autoencoders for Collaborative Filtering. In *WWW '18*. ACM, 689–698.

[19] Yang Liu, Alan Medlar, and Dorota Glowacka. 2023. What We Evaluate When We Evaluate Recommender Systems: Understanding Recommender Systems' Performance using Item Response Theory. In *RecSys '23*. ACM, 658–670.

[20] Ana Lucic, Maartje A. ter Hoeve, Gabriele Tolomei, Maarten de Rijke, and Fabrizio Silvestri. 2022. CF-GNNExplainer: Counterfactual Explanations for Graph Neural Networks. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2022, 28-30 March 2022, Virtual Event (Proceedings of Machine Learning Research, Vol. 151)*. PMLR, 4499–4511. https://proceedings.mlr.press/v151/lucic22a.html

[21] Jing Ma, Ruocheng Guo, Saumitra Mishra, Aidong Zhang, and Jundong Li. 2022. CLEAR: Generative Counterfactual Explanations on Graphs. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022.* http://papers.nips.cc/paper_files/paper/2022/hash/a69d7f3a1340d55c720e572742439eaf-Abstract-Conference.html

[22] Amir Reza Mohammadi, Andreas Peintner, Michael Müller, and Eva Zangerle. 2024. Are We Explaining the Same Recommenders? Incorporating Recommender Performance for Evaluating Explainers. In *RecSys '24*. ACM, 1113–1118.

[23] Caio Nóbrega and Leandro Balby Marinho. 2019. Towards explaining recommendations through local surrogate models. In *ACM/SIGAPP '19*. ACM, 1671–1678.

[24] Amir Hossein Ordibazar, Omar Hussain, and Morteza Saberi. 2021. A Recommender System and Risk Mitigation Strategy for Supply Chain Management Using the Counterfactual Explanation Algorithm. In *Service-Oriented Computing - ICSOC 2021 Workshops - AIOps, STRAPS, AI-PA and Satellite Events, Dubai, United Arab Emirates, November 22-25, 2021, Proceedings (Lecture Notes in Computer Science, Vol. 13236)*, Hakim Hacid, Monther Aldwairi, Mohamed Reda Bouadjenek, Marinella Petrocchi, Noura Faci, Fatma Outay, Amin Beheshti, Lauritz Thamsen, and Hai Dong (Eds.). Springer, 103–116. doi:10.1007/978-3-031-14135-5_8

[25] Andreas Peintner, Amir Reza Mohammadi, and Eva Zangerle. 2023. SPARE: Shortest Path Global Item Relations for Efficient Session-based Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, Jie Zhang, Li Chen, Shlomo Berkovsky, Min Zhang, Tommaso Di Noia, Justin Basilico, Luiz Pizzato, and Yang Song (Eds.). ACM, 58–69. doi:10.1145/3604915.3608768

[26] Andreas Peintner, Amir Reza Mohammadi, and Eva Zangerle. 2025. Efficient Session-based Recommendation with Contrastive Graph-based Shortest Path Search. *ACM Trans. Recomm. Syst.* 3, 4, Article 46 (April 2025), 24 pages. doi:10.1145/3701764

[27] Niloofar Ranjbar, Saeedeh Momtazi, and MohammadMehdi Homayoonpour. 2024. Explaining recommendation system using counterfactual textual explanations. *Mach. Learn.* 113, 4 (2024).

[28] Steffen Rendle, Walid Krichene, Li Zhang, and John R. Anderson. 2020. Neural Collaborative Filtering vs. Matrix Factorization Revisited. In *RecSys '20*. ACM, 240–248.

[29] Javier Del Ser, Alejandro Barredo Arrieta, Natalia Díaz Rodríguez, Francisco Herrera, Anna Saranti, and Andreas Holzinger. 2024. On generating trustworthy counterfactual explanations. *Inf. Sci.* 655 (2024), 119898. doi:10.1016/J.INS.2023.

119898

[30] Shahrzad Shashaani. 2024. Explainability in Music Recommender System. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys 2024, Bari, Italy, October 14-18, 2024*, Tommaso Di Noia, Pasquale Lops, Thorsten Joachims, Katrien Verbert, Pablo Castells, Zhenhua Dong, and Ben London (Eds.). ACM, 1395–1401. doi:10.1145/3640457.3688028

[31] Ramni Harbir Singh, Sargam Maurya, Tanisha Tripathi, Tushar Narula, and Gaurav Srivastav. 2020. Movie recommendation system using cosine similarity and KNN. *International Journal of Engineering and Advanced Technology* 9, 5 (2020), 556–559.

[32] Juntao Tan, Shijie Geng, Zuohui Fu, Yingqiang Ge, Shuyuan Xu, Yunqi Li, and Yongfeng Zhang. 2022. Learning and Evaluating Graph Neural Network Explanations based on Counterfactual and Factual Reasoning. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*. ACM, 1018–1027. doi:10.1145/3485447.3511948

[33] Juntao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual Explainable Recommendation. In *CIKM '21*. ACM, 1784–1793.

[34] Khanh Hiep Tran, Azin Ghazimatin, and Rishiraj Saha Roy. 2021. Counterfactual Explanations for Neural Recommenders. In *SIGIR '21*. ACM, 1627–1631.

[35] Sandra Wachter, Brent D. Mittelstadt, and Chris Russell. 2017. Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR. *CoRR* abs/1711.00399 (2017). arXiv:1711.00399 http://arxiv.org/abs/1711.00399

[36] Arran Zeyu Wang, David Borland, and David Gotz. 2025. Beyond Correlation: Incorporating Counterfactual Guidance to Better Support Exploratory Visual Analysis. *IEEE Transactions on Visualization and Computer Graphics* 31, 1 (2025), 776–786. doi:10.1109/TVCG.2024.3456369

[37] Yi Yu, Kazunari Sugiyama, and Adam Jatowt. 2025. Domain Counterfactual Data Augmentation for Explainable Recommendation. *ACM Trans. Inf. Syst.* 43, 3, Article 58 (Feb. 2025), 30 pages. doi:10.1145/3711856

[38] Jinfeng Zhong and Elsa Negre. 2022. Shap-enhanced counterfactual explanations for recommendations. In *SAC '22*. ACM, 1365–1372.