Temporal Evaluation of Recommender Algorithms: A Replication Study

Fiona Nlend
fiona.nlend@student.uni-siegen.de
University of Siegen
Siegen, Germany

Florian Paesler
florian.paesler@student.uni-siegen.de
University of Siegen
Siegen, Germany

Jonas Reising
jonas.reising@student.uni-siegen.de
University of Siegen
Siegen, Germany

ABSTRACT

Most recommender system evaluations rely on single-number metrics and assume that algorithm performance remains stable over time. However, Scheidt and Beel (2021)[7] demonstrated that this assumption can be misleading, as performance and algorithm rankings frequently fluctuate. In this study, we reproduce their findings using the original source code and extend the analysis to five additional datasets, including Amazon Software, Food.com and BeerAdvocate. Seven algorithms (FunkSVD, BiasedMF, Bias, UserKNN, ItemKNN, MostPopular and HPF) were evaluated across ten datasets using time-dependent evaluation, with performance reported over multiple time intervals. Our results confirm that, for 60% of the datasets, the best performing algorithm changed at least once over time. We also found that ranking instability was greatest in datasets covering long time spans and broad content categories (e.g, Amazon Electronics), whereas more homogeneous datasets (e.g, MovieLens) exhibited stable rankings. We found that nDCG exhibited stronger fluctuations than recall, underlining the importance of metric choice. All experiments were performed on consumer hardware and the code was made publicly available. These findings reinforce the need for time-sensitive evaluation protocols in recommender system research.

KEYWORDS

Recommender Systems, Time-Dependent Evaluation, Algorithm Ranking

1 INTRODUCTION

Recommender systems play a vital role in helping users to navigate large amounts of digital content by providing personalised recommendations. They are an integral part of platforms such as Netflix, Amazon and Yelp, and are used to recommend movies, products, music and restaurants. The effectiveness of recommender algorithms is usually evaluated offline using single-number metrics such as recall, nDCG or RMSE. While these metrics provide an easy means of comparing algorithmic performance, they implicitly assume that performance remains consistent over time.

This assumption is problematic in dynamic real-world environments where user preferences, item availability and interaction behaviour can change over time. If the evaluation is based on the full dataset without considering its temporal structure, misleading conclusions may be drawn. For instance, one algorithm may initially outperform others, only to be surpassed later, leading to inaccurate claims about which algorithm is 'best'. Nevertheless, most academic publications rely on static evaluations and report only aggregated metrics.

The 2021 study by Scheidt and Beel [7] addresses this gap by introducing a time-dependent evaluation approach. Their analysis of multiple datasets revealed significant temporal variability in algorithm performance, challenging the validity of single-number metrics. These findings suggest that evaluating recommender systems over time can provide more accurate and robust insights than static evaluations.

The aim of our study is to replicate these results. Using Scheidt and Beel's original codebase and publicly available datasets, we will verify whether their key findings about temporal performance variability can be replicated under our experimental conditions.

1.1 Background and Research Problem

Current research in recommender systems predominantly relies on static offline evaluations, which often report single-number metrics such as nDCG, recall or RMSE. However, these evaluations typically ignore the temporal dynamics inherent in longitudinal datasets. For example, the MovieLens 10M dataset spans 13 years. This simplification is problematic for two main reasons:

Algorithm robustness: Performance may fluctuate significantly over time. An algorithm that performed well in 2010 might perform poorly in 2020. Ignoring such fluctuations can lead to suboptimal real-world deployment.

Evaluation bias: Scheidt and Beel [7] found that 60% of datasets exhibited changes in algorithm rankings over time. However, such insights are lost when results are averaged into a single metric. Furthermore, their analysis of ACM RecSys 2020 revealed that 89% of evaluated papers used only static metrics, suggesting that time-dependent behaviour is widely neglected in current research.

1.2 Original Work

Scheidt and Beel (2021) provide a critical examination of the common practice of reporting single-number evaluation metrics in recommender system research. They hypothesise that this approach may obscure important temporal dynamics, such as changes in algorithm effectiveness and shifts in relative performance rankings over time.

To investigate this, they propose a time-dependent evaluation framework and apply it to ten variations of four real-world datasets (MovieLens, Netflix, Amazon and Yelp). The datasets were partitioned into four to eighteen temporal subsets based on monthly or yearly intervals, depending on data availability and timespan. In each evaluation step, all available data up to the given time point was used for model training, thereby simulating the progressive data accumulation observed in real-world systems.

Six algorithms from the LensKit library were evaluated: FunkSVD, BiasedMF, Bias, UserKNN, ItemKNN and MostPopular. The metrics nDCG, Recall and RMSE were used for evaluation. At each time step, the models were trained using 5-fold cross-validation and hyperparameter tuning via grid search, and performance was computed using held-out test data. Users with fewer than three ratings and subsets with fewer than 500 ratings were excluded to ensure statistical validity.

The results showed that, in 90% of the datasets, the performance of the algorithms changed over time. In 60% of cases, the relative ranking of the algorithms shifted at least once. For instance, the top-performing algorithm in the Amazon Toys and Games dataset changed multiple times over a 14-year period. Amazon-based datasets exhibited greater volatility than MovieLens datasets, likely due to variations in data density and user activity thresholds.

The magnitude of performance shifts varied between evaluation metrics. For example, nDCG values fluctuated by over 90% in some datasets (e.g. Amazon Music), whereas RMSE varied by up to 35%. This highlights that the choice of evaluation metric can significantly impact the perception of algorithm stability.

Additionally, the authors conducted a meta-analysis of 67 papers from ACM RecSys 2020. They found that 89% relied solely on static, aggregate metrics, highlighting that temporal aspects are largely overlooked in current recommender systems research.

Scheidt and Beel conclude that reporting performance metrics as time series reveals essential trends, instabilities and robustness that static evaluations cannot capture. To facilitate further research and encourage the wider adoption of time-dependent evaluation practices, they have publicly released their codebase.

1.3 Research Goal

Our goal is to replicate and reproduce the findings of Scheidt and Beel (2021) regarding the temporal variability of recommender system performance. More specifically, we will evaluate whether common recommendation algorithms such as UserKNN, ItemKNN and SVD fluctuate in performance over time when applied to various real-world datasets.

2 METHODOLOGY

2.1 Replication to Reproduction

We began our replication by using the unmodified source code provided by Scheidt and Beel[7] on a selection of datasets already used in the original study, namely MovieLens 100k, MovieLens 1M [3], Amazon Instant Video and Amazon Toys and Games[1]. However, the results we obtained varied significantly, both when compared to the original paper and across repeated runs using the same dataset under identical conditions. Upon closer inspection, we discovered that the original implementation lacked fixed random seeds, which resulted in nondeterministic behavior in both data sampling and algorithm execution (particularly for BiasedMF and FunkSVD).

To address this, we modified the codebase to include fixed random seeds in all relevant components, leading to stable and reproducible results. Although our replicated results still did not perfectly match the original findings, we were able to achieve a consistent evaluation framework. We additionally updated the codebase to support LensKit v1.14.0.

2.2 Algorithms and Datasets

In line with the original evaluation, we tested six algorithms: FunkSVD, BiasedMF, Bias, UserKNN, ItemKNN, and MostPopular. Furthermore, we included one additional algorithm provided by LensKit: HPF. For evaluation, we focused on the nDCG and Recall metrics, excluding RMSE to reduce computation time.

Beyond the original datasets, we incorporated five additional datasets. Due to hardware limitations and the requirement for both explicit feedback and timestamped ratings, the selection was constrained. We prioritized datasets under five million interactions to ensure feasible runtime on consumer-grade machines. The added datasets include two Amazon subsets[2] (Software, Video Games), as well as Food.com[3], BeerAdvocate[4] and MovieTweetings[5].

Table 1 summarizes the properties of these datasets and the corresponding temporal splits. Unlike Scheidt and Beel [7], we used coarser splitting intervals (e.g., every 5 years for 25-year spans) to mitigate the exponential runtime growth associated with increased number of temporal evaluation points. However, our splitting strategy follows the same core principle: each set contains all data up to a given cutoff point, thus preserving historical accumulation and aligning with real-world system deployment.

2.3 Evaluation

Our evaluation methodology closely mirrors that of Scheidt and Beel [7]. At each time step, we set aside the most recent 20% of each user's ratings as a test set. The remaining 80% was used to perform grid search with five-fold cross-validation (5 iterations) for each algorithm. Hyperparameter tuning was conducted independently for each temporal split.

Due to the considerable computational overhead caused by repeated hyperparameter optimization, we limited the evaluation to subsets of the available datasets. As shown in Table 1, we either reduced the dataset size through sampling or adjusted the temporal split intervals depending on the dataset's total size and timespan. Specifically, we first removed users with fewer than three interactions to ensure meaningful user histories. We then applied uniform random sampling (using a fixed seed for reproducibility) to reduce the dataset size while preserving the overall distribution of user-item interactions. This allowed us to maintain a sufficiently large number of temporal evaluation steps while keeping the overall runtime manageable.

To ensure high replicability and meaningful data every new dataset was evaluated three times using different random seeds (1, 42, 123). We evaluated the original datasets only once using the original code for comparison.

Our updated code can be found here[6].

[1] Downloaded from Amazon review data 2014 (ucsd.edu)[4]

[2] Downloaded from Amazon Reviews'23[5]

[3] Downloaded from Food.com Recipes and Interactions[6]

[4] Downloaded from Beer Reviews from BeerAdvocate (1.5 Million)

[5] Downloaded from MovieTweetings (Kaggle)[2]

[6] GitHub: https://github.com/florian-p0/time-recsys/

Table 1: Datasets used for Evaluation

Dataset	Number of Ratings	Subset of Ratings Used (% of Total)	Timespan	Split (# of sets)
Amazon Software	4.8 million	50%	1999-2024	every 5 years (5)
Amazon Video Games	4.6 million	50%	1998-2024	every 5 years (5)
BeerAdvocate	1.5 million	10%	1998-2011	yearly (11)
Food.com	1.1 million	100%	2000-2019	every 3 years (6)
MovieTweetings	900 thousand	100%	2013-2022	every 3 years (4)

3 RESULTS & DISCUSSION

Our time-dependent evaluation of recommender algorithms reaffirmed the key finding from Scheidt and Beel (2021): the performance and ranking of algorithms can change substantially over time. This section expands on the dynamics observed across different datasets, evaluates the implications of these changes, and compares our results to the original study in more detail.

3.1 Algorithm Ranking Instability

We observed substantial ranking fluctuations in several datasets. The Amazon Electronics dataset stood out with six ranking changes in 15 epochs, signaling a high degree of algorithmic volatility. Similar patterns emerged in Amazon Software and BeerAdvocate, both demonstrated two swaps over 5 and 11 epochs, respectively. These results mirror those reported by Scheidt and Beel, who noted that 60% of datasets exhibited at least one ranking change. Our results suggest that this phenomenon holds true even with new datasets not included in the original work.

Interestingly, datasets with longer timespans and broader item categories, such as Amazon Electronics and BeerAdvocate, showed greater temporal instability. This implies that the diversity and evolution of user behavior over time contribute to shifting algorithm performance. In contrast, more homogeneous datasets such as MovieLens 1M, Amazon Instant Video, and Amazon Video Games displayed stable rankings throughout the evaluation period. These datasets likely exhibit less variation in item availability or user demographics over time, leading to fewer abrupt shifts in performance.

3.2 Early Phase Volatility

We also identified a common pattern of early stage ranking volatility followed by later convergence. For example, on Food.com and ML-100k, a ranking change occurred in the first epoch, after which the best performing algorithm remained consistent. This suggests that early data dynamics can disproportionately influence model behavior, especially when limited data leads to overfitting or underrepresentation of general trends. As the training set grows and stabilizes, the models seem to reach a more robust performance plateau.

This observation matches the insight of the original study that initial training phases are often the most dramatic changes in performance. It further highlights the need for evaluation frameworks that are sensitive to these critical early periods, especially in systems deployed in rapidly evolving domains (e.g., new product launches or seasonal recommendation contexts).

3.3 Metric-Specific Trends

Although we excluded RMSE due to runtime constraints, we did compare nDCG and Recall across time steps. In general, both metrics showed parallel trends, but with different magnitudes of variation. nDCG tended to exhibit more pronounced temporal fluctuations, especially in Amazon datasets. This supports Scheidt and Beel's observation that nDCG is more sensitive to ranking changes, while metrics such as RMSE (or possibly Recall) can smooth out meaningful shifts due to their aggregative nature.

In data sets with dense feedback, such as MovieTweetings, we observed relatively stable performance across all algorithms, though nDCG remained more volatile than Recall. This suggests that the choice of metric can significantly influence the perceived stability or effectiveness of an algorithm and reinforces the original recommendation of presenting multiple metrics over time.

3.4 Effect of Dataset Size and Sampling

To manage computational constraints, we applied sampling and coarser temporal splits (e.g., 5-year intervals for long-running datasets). Despite these differences in methodology, the core findings were preserved: algorithms vary in performance over time, and single-number metrics do not capture this complexity. The consistency of these conclusions across datasets, metrics, and methodological variants supports the robustness of the original study's claims.

However, the reduced granularity in our temporal splits may have led to underreporting of smaller performance shifts, particularly in fast-changing datasets. For example, Amazon Software might exhibit even more instability if evaluated yearly instead of every five years. Future work could investigate this by varying the granularity of temporal splits and examining its effect on stability measures

4 LIMITATIONS

As discussed in the methodology section, we had to reduce both the number and size of datasets due to computational constraints. In particular, we focused on datasets with fewer than five million interactions, whereas the original study also included datasets exceeding ten million ratings. This limits the comparability of our results, especially regarding the behavior of algorithms on very large datasets.

Furthermore, we restricted our evaluation to two metrics, nDCG and Recall, in order to reduce execution time. While this allowed us to focus on ranking performance, it excluded RMSE as a measure of rating prediction quality.

Finally, although it would have been desirable to include a wider range of algorithms and additional datasets, doing so would have

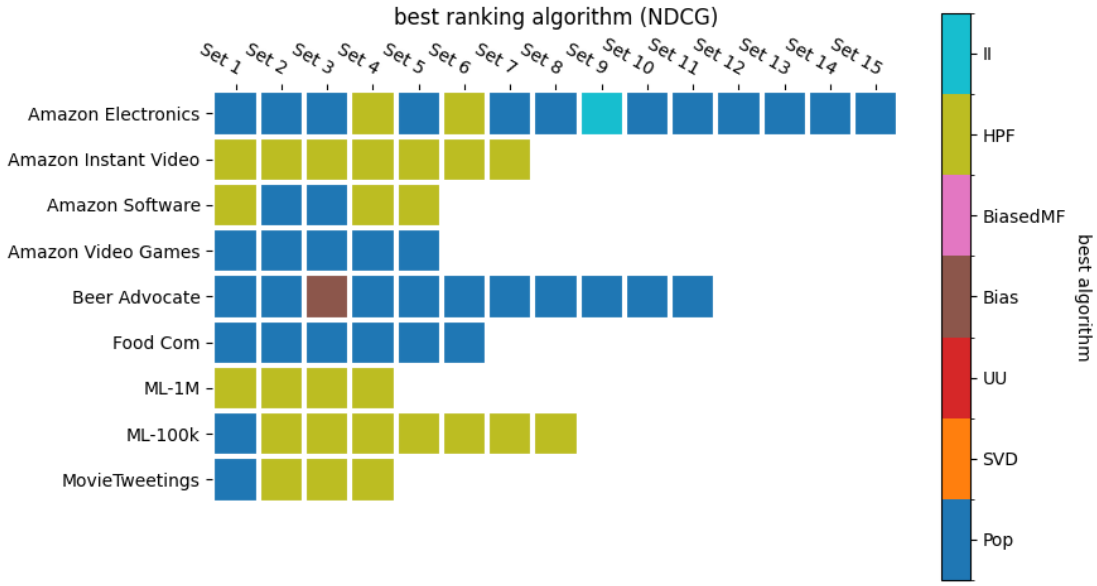

Figure 1: best performing algorithm in each dataset and timestep

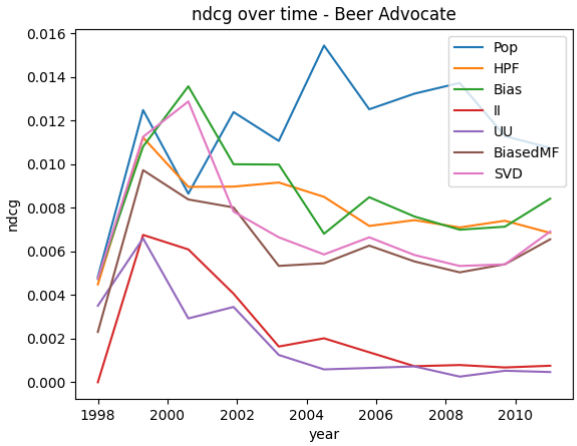

Figure 2: BeerAdvocate

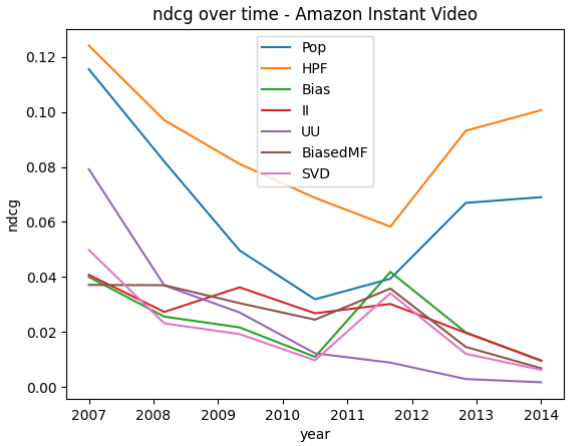

Figure 3: Amazon Instant Video

significantly increased runtime and was therefore not feasible within our computational budget.

5 ACKNOWLEDGMENTS

This work was conducted as part of the Machine Learning Praktikum at the University of Siegen [1].

We used ChatGPT (OpenAI) as a writing assistant to improve the clarity and phrasing of certain sections. All scientific content was authored, verified, and interpreted by the authors.

REFERENCES

[1] Joeran Beel and Lukas Wegmeth. 2025. Machine Learning Praktikum. *Universisät Siegen* (2025).

[2] Simon Dooms, Toon De Pessemier, and Luc Martens. 2013. MovieTweetings: A Movie Rating Dataset Collected From Twitter. In *Proceedings of the Workshop on Crowdsourcing and Human Computation for Recommender Systems (CrowdRec 2013) co-located with ACM RecSys 2013.*

[3] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 5, 4, Article 19 (Dec. 2015), 19:1–19:19 pages. https://doi.org/10.1145/2827872

[4] Ruining He and Julian McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proceedings of the 25th International Conference on World Wide Web (WWW)*. International World Wide Web Conferences Steering Committee, 507–517. https://doi.org/10.1145/2872427.2883037

[5] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging Language and Items for Retrieval and Recommendation. *arXiv preprint arXiv:2403.03952* (2024).

[6] Shuyang Li. 2019. Food.com Recipes and Interactions. https://doi.org/10.34740/KAGGLE/DSV/783630

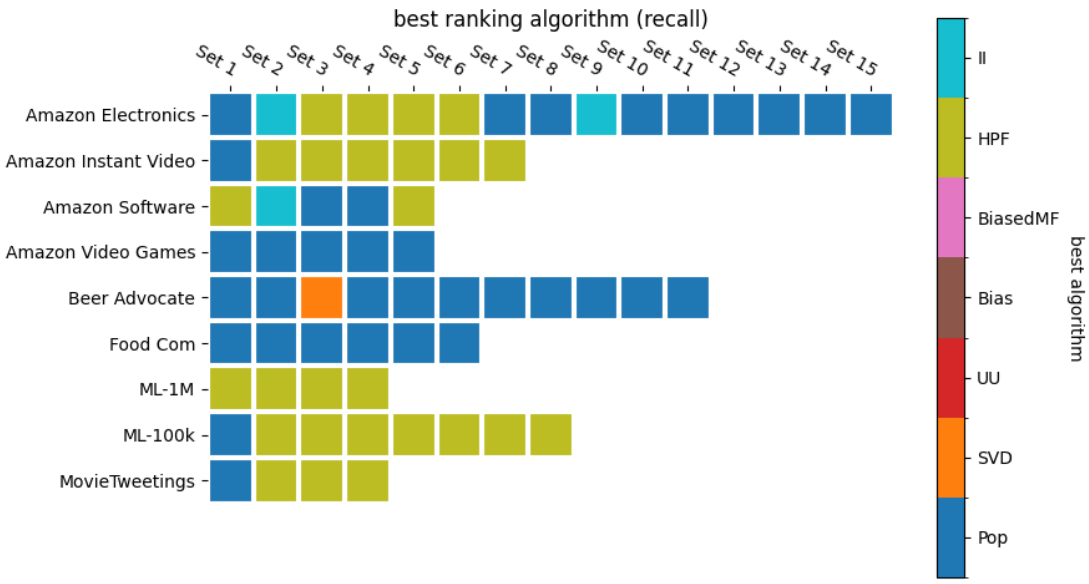

Figure 4: best performing algorithm in each dataset and timestep

[7] Teresa Scheidt and Joeran Beel. 2021. Time-dependent evaluation of recommender systems. In *Perspectives 2021 (CEUR Workshop Proceedings, Vol. 2955).* CEUR Workshop Proceedings. https://ceurspt.wikidata.dbis.rwth-aachen.de/Vol-2955/paper10.pdf Publisher Copyright: © 2021 Copyright for this paper by its authors.; 2021 Perspectives on the Evaluation of Recommender Systems Workshop, Perspectives 2021 ; Conference date: 25-09-2021.