Informed Dataset Selection with 'Algorithm Performance Spaces'

Joeran Beel
joeran.beel@uni-siegen.de
University of Siegen & Recommender-Systems.com
Siegen, Germany

Lukas Wegmeth
lukas.wegmeth@uni-siegen.de
University of Siegen
Siegen, Germany

Lien Michiels
lien.michiels@uantwerpen.be
University of Antwerp
Antwerp, Belgium
imec-SMIT, Vrije Universiteit Brussel
Brussels, Belgium

Steffen Schulz
steffen.schulz@student.uni-siegen.de
University of Siegen
Siegen, Germany

ABSTRACT

When designing recommender-systems experiments, a key question that has been largely overlooked is the choice of datasets. In a brief survey of ACM RecSys papers, we found that authors typically justified their dataset choices by labelling them as public, benchmark, or 'real-world' without further explanation. We propose the Algorithm Performance Space (APS) as a novel method for *informed* dataset selection. The APS is an n-dimensional space where each dimension represents the performance of a different algorithm. Each dataset is depicted as an n-dimensional vector, with greater distances indicating higher diversity. In our experiment, we ran 29 algorithms on 95 datasets to construct an actual APS. Our findings show that many datasets, including most Amazon datasets, are clustered closely in the APS, i.e. they are not diverse. However, other datasets, such as MovieLens and Docear, are more dispersed. The APS also enables the grouping of datasets based on the solvability of the underlying problem. Datasets in the top right corner of the APS are considered 'solved problems' because all algorithms perform well on them. Conversely, datasets in the bottom left corner lack well-performing algorithms, making them ideal candidates for new recommender-system research due to the challenges they present.

ACM Reference Format:
Joeran Beel, Lukas Wegmeth, Lien Michiels, and Steffen Schulz. 2024. Informed Dataset Selection with 'Algorithm Performance Spaces'. In *18th ACM Conference on Recommender Systems (RecSys '24), October 14–18, 2024, Bari, Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3640457.3691704

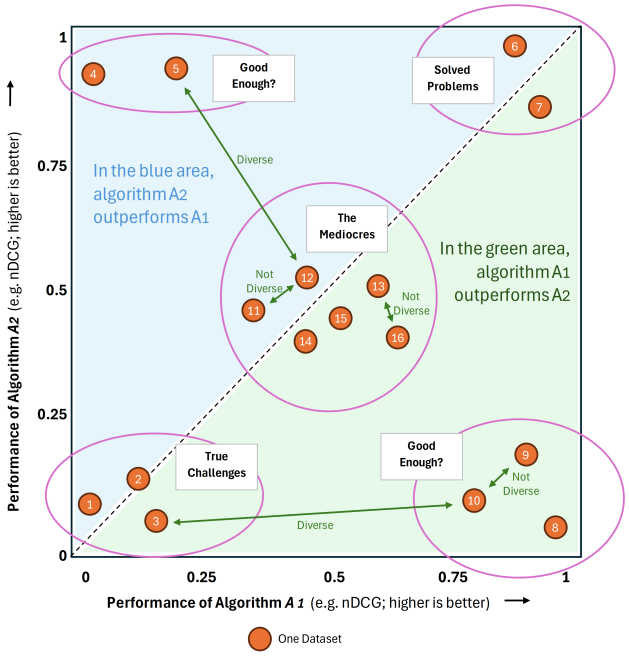

Figure 1: Illustration of the Algorithm Performance Space (APS). The x-axis and y-axis represent the performance (nDCG) of algorithms *A1* and *A2*. Orange circles indicate datasets. Datasets in the top right corner of the APS are considered 'solved problems,' as both algorithms perform well on them. Conversely, datasets in the bottom left corner lack well-performing algorithms.

1 INTRODUCTION

A key question in recommender-systems offline evaluation is which datasets to use. Beel and Brunel [6] found that most researchers use MovieLens (40%), Amazon (35%), or Yelp (13%) datasets. This trend is confirmed by others [3, 12, 24, 25, 29]. The reasons for these choices are largely unclear. Cremonesi and Jannach [13] criticized the common lack of justification for dataset selection in the community. We surveyed 41 'Full Papers' from the ACM RecSys 2023 conference that evaluated algorithms offline. The authors of 23 papers (56%) indirectly justified their choices by a) referring to datasets as 'public' or 'real-world' (24%; 10 out of 41) or b) referring to the application domain, e.g. news or cross-domain, or the task, such as session-based recommendation (32%; 13 out of 41). Authors of 18 papers (44%) explained their choices by referring to datasets

ACM RecSys 2024, October 14–18, 2024, Bari, IT

ACM ISBN 979-8-4007-0505-2/24/10
https://doi.org/10.1145/3640457.3691704

as "benchmark" or "widely used" datasets. No authors justified their choice of datasets in detail.[1]

While it is common – and not necessarily problematic – to evaluate algorithms on benchmark datasets, especially in machine learning [17, 18, 21, 22, 26], the justifications provided by recommender-systems authors deserve closer scrutiny. Notably, there are no true benchmark datasets in the recommender-systems community. Therefore, we should critically examine any claims that refer to recommender-systems datasets as 'benchmarks'. Moreover, while choosing a dataset because it is widely used, public, or "real-world" may have its merits, this should not be the sole justification.

In summary, we agree with Cremonesi and Jannach that recommender systems researchers should make more informed decisions on which datasets to use for experiments. Evaluating a recommendation algorithm offline aims to estimate its performance on future unknown data. Choosing random, convenient, or popular datasets likely won't achieve this goal optimally. The community should discuss the factors influencing dataset selection and establish best practice guidelines. Researchers should also report their reasoning for choosing a dataset in their publications.

In this paper, we propose the Algorithm Performance Space (**APS**) as a novel method to make an informed decision on selecting datasets for recommender-system experiments to obtain good generalization power. We do not claim to have found the final answer but see the proposed method as one suggestion that will hopefully initiate a discussion in the community and eventually lead to accepted best practices on dataset selection.

2 RELATED WORK

Recommender-systems datasets have been subject to extensive research. Researchers introduced new datasets [7–9, 30, 31], introduced synthetic datasets [11, 19, 20, 23], augmented datasets [10, 14, 33], proposed methods for creating datasets [2] and they discussed how (not) to prune datasets [6]. Fan et al. [15] emphasized understanding data generation mechanisms behind datasets. They argue that the context in which interactions are generated can differ from real-world applications, limiting the datasets' ability to predict real-world model performance accurately. Specifically, they examined MovieLens and its data acquisition methods. Their findings revealed that nearly half of all users submitted all their ratings within a single day. Such insights are crucial for evaluating models in broader recommender-system scenarios.

To our knowledge, only Chin et al. [12] share our research goal of making informed decisions on dataset selection for offline evaluation experiments. Chin et al. [12] recommend choosing datasets with diverse characteristics, such as space, shape, density, and interaction distribution across users and items. They classified 51 datasets into five clusters using k-means clustering and selected three datasets from each cluster for their experiments. These experiments showed significant differences ($p < 0.05$) in algorithm performance (UserKNN, ItemKNN, RP3beta, WMF, and Mult-VAE) based on these characteristics. For example, RP3beta performed up to 45% better on relatively sparse datasets of moderate size but was least effective on denser, slightly larger datasets.

Analyzing the impact of dataset characteristics on algorithm performance is not new. Adomavicius and Zhang [1] found a correlation between dataset characteristics and algorithm performance in 2012. This finding is not surprising; if algorithms perform differently on different datasets, the reason must lie in the data. Therefore, it seems intuitive that recommender-systems researchers should choose datasets with varying characteristics, where algorithms might perform differently. However, other studies found that the similarity of datasets in terms of user characteristics, item characteristics, sparsity, etc., does not fully determine algorithm performance [5, 13, 16, 27]. Moreover, Chin et al. observed that several Amazon datasets appeared in different clusters, such as Amazon Movies & TV (cluster 1), Amazon Toys & Games (cluster 3), and Amazon Patio Lawn & Garden (cluster 4). Yet, algorithm performance on Amazon datasets was inconclusive both within and across clusters.

This gives us reason to assume that dataset characteristics may not be an ideal determinant of algorithm performance. Therefore, in this work, we consider dataset selection from the perspective of algorithm performance. That said, the work by Chin et al. and ours are not mutually exclusive. Combining dataset characteristics and algorithm performance into one selection method would be an exciting field of research for the future.

3 THE CONCEPT OF ALGORITHM PERFORMANCE SPACES

To identify a diverse set of recommender-system datasets, we propose the utilization of the Algorithm Performance Space (**APS**). Initially introduced by Tyrrell et al. [28] to represent instances within a dataset for meta-learning with Siamese Neural Networks, we extend the APS concept to represent entire datasets for identifying diversity among them. The APS is an n-dimensional space, where each dimension corresponds to the performance of a distinct algorithm. Performance metrics can vary; however, for simplicity, we utilize normalized Discounted Cumulative Gain (nDCG) throughout this study and our examples. Each dataset within the APS is represented as an n-dimensional vector, with each dimension reflecting the performance of a specific algorithm on that dataset. Placing datasets in the APS allows the expression of a degree of diversity between the datasets: the larger the distance between two datasets in the APS, the greater the diversity. Unlike Chin et al., our definition of diversity is unrelated to dataset characteristics but instead refers to algorithm performance on the datasets.

We illustrate the Algorithm Performance Space with an example (Figure 1). Suppose there are only two algorithms, A_1 and A_2, creating a two-dimensional APS. The performance (nDCG) for A_1 is plotted on the x-axis, and the performance of A_2 on the y-axis. Datasets are represented as orange circles. For instance, dataset D_8 is represented as a two-dimensional vector (or point[2]) with coordinates $x \approx 1$ and $y \approx 0$. This indicates that algorithm A_1 performed well on dataset D_8, with an nDCG near 1, while A_2 performed poorly, with an nDCG near 0. Dataset D_8 is in close proximity to D_9, meaning the algorithms A_1 and A_2 performed on D_8 similar

[1] We acknowledge that although we critique the current practice in the community of not justifying dataset selection, we are not exempt from this critique. In our own work to date, we have also failed to provide thorough justifications.

[2] We use the terms point and vector interchangeably, even though this might be mathematically incorrect.

to the performance on D_9. Consequently, based on our definition, datasets D_8 and D_9 would be considered not diverse because the distance between them is small, meaning algorithms perform similarly on them. Conversely, on dataset D_{15} (positioned centrally), both algorithms A_1 and A_2 exhibited mediocre performance, each achieving an nDCG of around 0.5. The distance between D_{15} and D_8 is relatively large, meaning the algorithms performed differently on them. Therefore, datasets D_{15} and D_8 would be considered diverse.

The rationale behind the APS is as follows. If some datasets $D_1...D_m$ are close to each other in the APS, this indicates that *all* algorithms $A_1...A_n$ in the APS have performed similarly on them. To clarify, this does not imply that all *n* algorithms achieved the same performance on the *m* datasets. It could be, for instance, that algorithms A_1, A_2 and A_3 performed well on the *m* datasets; algorithms A_4 and A_5 performed poorly on the *m* datasets and algorithms $A_6...A_n$ exhibited mediocre performance on the *m* datasets. In other words, algorithm A_1 performed consistently across the *m* datasets, A_2 performed consistently across the *m* datasets (but not necessarily similarly to A_1) and all other *n* algorithms also performed consistently across the *m* datasets. Therefore, it seems likely to us that a novel algorithm A_{n+1} – that is not part of the APS – will also perform consistently across the *m* datasets (whether performance will be high, low or mediocre cannot be predicted). If, for instance, algorithm A_{n+1} performs poorly on one or two of the *n* datasets, we consider it extremely likely that algorithm A_{n+1} will perform similarly on the remaining *n* datasets because this behavior was true for all n algorithms. Consequently, evaluating the novel algorithm on one or two of the n datasets would be sufficient. Our assumption should hold, especially in a high-dimensional APS.

If all, or at least many, algorithms perform consistently across the datasets, there must be an underlying reason, which must lie in the data. Thus, the general idea by Chin et al. to group datasets by dataset characteristics is intuitive. However, we argue that it will be difficult, if not impossible, to identify all data characteristics that impact how an algorithm will perform on a particular dataset. Our approach focuses on the performance of algorithms, regardless of the reason for variances in performance.

Based on the above rationale, we argue that researchers typically should choose datasets with high diversity, i.e., datasets that are highly distant from each other in the APS. This approach allows researchers to determine whether their algorithm is an "all-rounder" that performs well across various scenarios or excels only in specific areas of the APS. However, there may be situations where selecting several non-diverse datasets is appropriate, too.

The APS serves another purpose: it enables the grouping of datasets based on the solvability of the underlying problem. Datasets in the top right corner of the APS can be considered 'solved problems', as each algorithm in the APS performs well on them. Such datasets might not be ideal candidates for new recommender-system experiments since numerous algorithms have already achieved high performance. Further development for these datasets would likely yield minimal value, as existing algorithms are near-optimal performance. Consequently, the likelihood of developing an algorithm that significantly outperforms the current state of the art on these 'solved problems' is low or even impossible.

In contrast, datasets placed in the bottom left corner are those for which no or few well-performing algorithms currently exist. These datasets could be prime candidates for new recommender-system research as they represent true challenges. Significant progress would be achieved if a researcher developed a novel algorithm that performs well on such a dataset. Similarly, datasets where algorithms perform mediocrely are positioned in the middle of the chart. Developing an algorithm that performs well on these datasets would also signify real progress. For datasets in the top-left and bottom-right corners, some algorithms perform well while others do not. Whether further efforts should be directed toward finding more algorithms that perform well on these datasets or whether the current state is sufficient remains a topic for debate.

We want to emphasize that we are not providing definitive recommendations on choosing datasets based on the APS. While we have offered examples of potential arguments, these are illustrative rather than prescriptive. A key benefit of the APS is that it allows individual researchers to apply their own reasoning to dataset selection. If researchers explain their reasoning in their manuscripts, reviewers and readers can evaluate the validity of the authors' choices. There may be, for instance, valid reasons to include solved-problems datasets for experiments or to select multiple datasets from the same area within the APS. The APS provides a framework for researchers to consider differences among datasets and systematically justify their decisions.

4 EXPERIMENT

In the previous section, we introduced the concept of an Algorithm Performance Space (APS) for informed dataset selection. In this section, we present the results of an experiment involving 29 algorithms, including a random recommender, and 95 datasets. The goal of this experiment was to explore the practical appearance of an APS, rather than to obtain definitive evidence of its effectiveness.

4.1 Methodology

We ran 29 recommendation algorithms from RecBole [32] with default hyperparameters on 95 recommender-systems datasets to construct and examine an actual APS. Details on the algorithms and datasets are provided in the supplemental material[3]. The datasets include 74 with explicit feedback and 21 with implicit feedback. To generate top-n recommendations and calculate nDCG, we converted the explicit feedback into implicit feedback by considering each rating as a positive interaction. We applied 5-core pruning to all datasets. The training and evaluation were limited to 7,000 GPU hours on our university's GPU cluster (NVIDIA Tesla V100). We employed 5-fold cross-validation, allocating 30 minutes for training each fold of the 29 algorithms across the 95 datasets[4]. While default hyperparameters and a 30-minute training period may not lead to optimal algorithm performance, we deem this methodology appropriate for obtaining an initial understanding of the APS.

[3] https://code.isg.beel.org/Informed-Dataset-Selection-via-APS/

[4] 102 out of the possible 29x95=2755 pairings (3.7%) of datasets and algorithms failed in training, e.g., due to the time limit or failing to create an embedding, and are therefore missing in our evaluation e.g. the APS and reduced APS

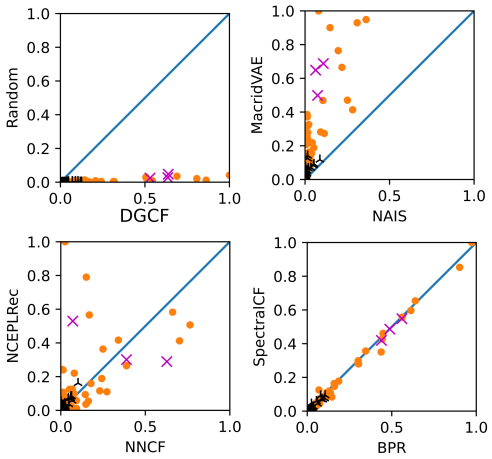

Figure 2: Four of the 812 "mini" 2-dimensional Algorithm Performance Spaces. MovieLens datasets are illustrated by violet crosses, Amazon datasets black, other datasets by blue circles. Axes show the relative nDCG performance (1 = best-performing nDCG; 0 = worst-performing nDCG)

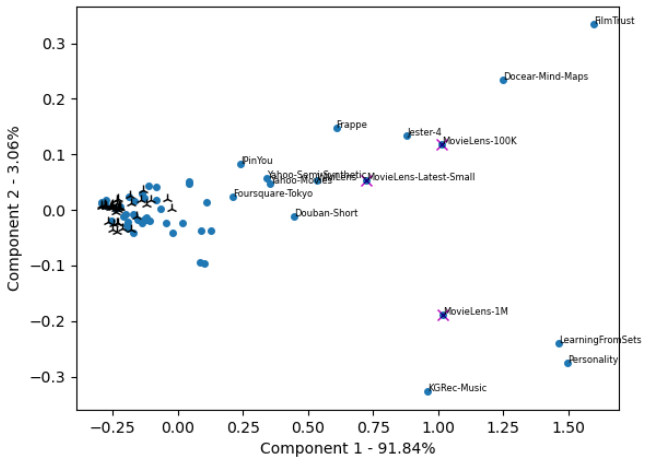

Figure 3: The reduced APS for which the 29 dimensions were reduced to two with Principal Component Analysis (PCA)

4.2 812 Mini-APS

Given 29 algorithms, we would typically build a 29-dimensional APS. However, a 29-dimensional space cannot be visualized. Therefore, we first build 29x28 = 812 two-dimensional Algorithm Performance Spaces, which we term "Mini-APS". We normalized each axis so that the best-performing algorithm (measured by nDCG) would get a value of 1 and the worst-performing algorithm a value of 0. A selection of 4 of these 812 Mini-APS is shown in Figure 2. All 812 APS are available in the supplemental material[5]. Given that MovieLens and Amazon datasets are among the most popular datasets in recommender-systems research [3, 6, 12], we highlighted these datasets in the charts. MovieLens datasets are highlighted as a violet cross, and Amazon datasets are highlighted in black.

The top-left of Figure 2 shows the mini-APS with the *DGCF* and *Random* algorithm. The Random algorithm served as a control group. The chart shows that the random algorithm performed poorly on all datasets, whereas results for DGCF are more spread. DGCF performed very well on some datasets, and on others, poorly.

The top-right of Figure 2 shows the mini-APS with the NAIS and MacridVAE algorithms. The APS shows that MacridVAE outperforms NAIS on all datasets. In the bottom-right of Figure 2, the algorithms BPR and SpectralCF perform almost alike. Almost all datasets are plotted on the diagonal. The bottom-left of Figure 2 shows NNCF vs. NCEPLRec. While there is a concentration of datasets near the origin, the spread of datasets is relatively large. Overall, it is notable that the Amazon datasets (black; Figure 2) are relatively close together in nearly all 812 Mini-APS. In contrast, the MovieLens datasets (violet) have more distance from each other. Our observation regarding the Amazon datasets is particularly interesting compared to the results by Chin et al. [12]. In their analysis, focusing on dataset characteristics, the Amazon datasets were spread out across four out of five clusters. This means, based on dataset characteristics, the Amazon datasets are diverse. Based on algorithm performance, the Amazon datasets are not (very) diverse.

The Mini-APS (Figure 2) differ somewhat from our expectations (Figure 1). Contrary to our expectation, the datasets in the Mini-APS usually are not widely spread and achieve relatively low nDCGs on most datasets. The datasets tend to cluster around the origin and/or the diagonals. The clustering around the diagonals indicates that algorithms tend to perform similarly on dataset, meaning if one algorithm performs well (poorly), the other algorithms also tend to perform well (or poorly). There are exceptions, though. In our view, this finding emphasizes the importance of an informed dataset selection: Apparently, most datasets are very similar in terms of how algorithms perform on them, thus making it even more critical and difficult to identify diverse datasets.

4.3 Reduced Algorithm Performance Space

To get a more comprehensive overview, we reduced the APS from 29 to 2 dimensions via Principle Component Analysis (**PCA**) (Figure 3). In this reduced APS, the spread of datasets is notably larger.

Once again, we observe that the Amazon datasets (black) are clustered relatively close together, indicating that algorithms tend to perform similarly on all Amazon datasets. While the MovieLens datasets (100K, 1M, Latest-Small) are further apart, they are similar in terms of the first component, which explains 91.84% of the variance. Interestingly, on the right side of the chart, at the top and bottom are relatively unpopular datasets such as FilmTrust, Docear [8], Personality, KGRec-Music, and LearningFromSets, with the exception of MovieLens 1M. It would be interesting to explore why these datasets are far from the others.

It must be noted that the axes do not represent performance any more. Hence, datasets, e.g. in the top-right corner, do not necessarily represent datasets on which algorithms perform very well.

[5]https://code.isg.beel.org/Informed-Dataset-Selection-via-APS/

5 SUMMARY AND DISCUSSION

We introduced Algorithm Performance Spaces (APS) for informed dataset selection. Although our current research is preliminary, we see great potential. We envision an extensive APS containing the performance values of many algorithms on numerous datasets. Unlike inflexible benchmark datasets in machine learning, APS would avoid focusing on a fixed set of datasets. Researchers could extend APS with new datasets (public or private), use APS to identify diverse or similar datasets based on their use case, and explain their reasoning in their manuscripts, allowing reviewers to judge the validity of the reasoning.

Many open questions remain. For instance, can and should APS be combined with dataset characteristics? How should distance be calculated in a high-dimensional APS? How exactly should datasets be chosen based on APS? Should there be a standardized APS for all recommender-systems researchers, or should each researcher build their own? Should there be 'sub-spaces' for different evaluation metrics, algorithms, and data preprocessing methods, such as one APS for movies and another for eCommerce? Most importantly, how can the effectiveness of APS be shown empirically? It is also crucial to debate the impact of hyperparameter optimization (HPO), training duration, and data preprocessing (e.g., 5-core pruning or converting explicit to implicit feedback) on APS. These factors likely significantly influence dataset placement in APS. Also, there is a trade-off to consider when developing APS. On the one hand, APS should include many algorithms to highlight their strengths and weaknesses across datasets effectively. On the other hand, researchers need to easily place new datasets in the APS, requiring all algorithms to be run on the dataset. The more algorithms an APS has, the more challenging it becomes to add a new dataset.

In summary, Algorithm Performance Spaces (APS) uniquely visualize and analyze how different algorithms perform across various datasets, focusing on performance outcomes rather than dataset characteristics. APS help researchers identify diverse or similar datasets based on algorithmic performance, likely enhancing generalization to new, unseen data. APS also distinguish between well-solved and challenging datasets, guiding researchers to prioritize those that can drive meaningful advancements in algorithm development. By offering a clear and extendable performance-based method for dataset selection, APS can enhance informed decision-making in recommender-systems research, improving the transparency, rigor, and generalizability of offline evaluations.

6 ACKNOWLEDGEMENTS

This work was in part supported by funding from the Ministry of Culture and Science of the German State of North Rhine Westphalia, grant no. 311-8.03.03.02-149514. We used ChatGPT to improve the writing of this manuscript. We mostly used it as an "advanced" grammar and phrasing tool. For further details, please refer to [4].

REFERENCES

[1] Gediminas Adomavicius and Jingjing Zhang. 2012. Impact of data characteristics on recommender systems performance. *ACM Trans. Manage. Inf. Syst.* 3, 1, Article 3 (apr 2012), 17 pages. https://doi.org/10.1145/2151163.2151166

[2] Marcia Barros, Francisco M. Couto, Matilde Pato, and Pedro Ruas. 2021. Creating Recommender Systems Datasets in Scientific Fields. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (Virtual Event, Singapore) *(KDD '21)*. Association for Computing Machinery, New York, NY, USA, 4029–4030. https://doi.org/10.1145/3447548.3470805

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

[4] Joeran Beel. 2024. Our use of AI-tools for writing research papers. In *Intelligent Systems Group, Blog*. https://isg.beel.org/blog/2024/08/19/my-use-of-ai-tools-for-writing-research-papers/

[5] Joeran Beel, Corinna Breitinger, Stefan Langer, Andreas Lommatzsch, and Bela Gipp. 2016. Towards reproducibility in recommender-systems research. *User Modeling and User-Adapted Interaction* 26, 1 (mar 2016), 69–101. https://doi.org/10.1007/s11257-016-9174-x

[6] Joeran Beel and Victor Brunel. 2019. Data Pruning in Recommender Systems Research: Best-Practice or Malpractice?. In *13th ACM Conference on Recommender Systems (RecSys)*, Vol. 2431. CEUR-WS, 26–30.

[7] Joeran Beel, Zeljko Carevic, Johann Schaible, and Gabor Neusch. 2017. RARD: The Related-Article Recommendation Dataset. *D-Lib Magazine* 23, 7/8 (July 2017), 1–14.

[8] Joeran Beel, Stefan Langer, Bela Gipp, and Andreas Nuernberger. 2014. The Architecture and Datasets of Docear's Research Paper Recommender System. *D-Lib Magazine* 20, 11/12 (2014). https://doi.org/10.1045/november14-beel

[9] Joeran Beel, Barry Smyth, and Andrew Collins. 2019. RARD II: The 94 Million Related-Article Recommendation Dataset. In *Proceedings of the 1st Interdisciplinary Workshop on Algorithm Selection and Meta-Learning in Information Retrieval (AMIR)*. CEUR-WS, 39–55.

[10] Shuqing Bian, Wayne Xin Zhao, Jinpeng Wang, and Ji-Rong Wen. 2022. A Relevant and Diverse Retrieval-enhanced Data Augmentation Framework for Sequential Recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) *(CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 2923–2932. https://doi.org/10.1145/3511808.3557071

[11] Jesús Bobadilla, Abraham Gutiérrez, Raciel Yera, and Luis Martínez. 2023. Creating synthetic datasets for collaborative filtering recommender systems using generative adversarial networks. *Knowledge-Based Systems* 280 (2023), 111016. https://doi.org/10.1016/j.knosys.2023.111016

[12] Jin Yao Chin, Yile Chen, and Gao Cong. 2022. The datasets dilemma: How much do we really know about recommendation datasets?. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 141–149.

[13] Paolo Cremonesi and Dietmar Jannach. 2021. Progress in Recommender Systems Research: Crisis? What Crisis? *AI Magazine* 42, 3 (Nov. 2021), 43–54. https://doi.org/10.1609/aimag.v42i3.18145

[14] Gordian Edenhofer, Andrew Collins, Akiko Aizawa, and Joeran Beel. 2019. Augmenting the DonorsChoose.org Corpus for Meta-Learning. In *Proceedings of The 1st Interdisciplinary Workshop on Algorithm Selection and Meta-Learning in Information Retrieval (AMIR)*. CEUR-WS, 32–38.

[15] Yu-chen Fan, Yitong Ji, Jie Zhang, and Aixin Sun. 2023. Our Model Achieves Excellent Performance on MovieLens: What Does it Mean? *ACM Transactions on Information Systems* (2023).

[16] 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* (Copenhagen, Denmark) *(RecSys '19)*. Association for Computing Machinery, New York, NY, USA, 101–109. https://doi.org/10.1145/3298689.3347058

[17] Sebastian Felix Fischer, Matthias Feurer, and Bernd Bischl. 2023. OpenML-CTR23–a curated tabular regression benchmarking suite. In *AutoML Conference 2023 (Workshop)*.

[18] Pieter Gijsbers, Marcos LP Bueno, Stefan Coors, Erin LeDell, Sébastien Poirier, Janek Thomas, Bernd Bischl, and Joaquin Vanschoren. 2024. Amlb: an automl benchmark. *Journal of Machine Learning Research* 25, 101 (2024), 1–65.

[19] Mark Grennan, Martin Schibel, Andrew Collins, and Joeran Beel. 2019. GIANT: The 1-Billion Annotated Synthetic Bibliographic-Reference-String Dataset for Deep Citation Parsing. In *27th AIAI Irish Conference on Artificial Intelligence and Cognitive Science*. 101–112.

[20] Marden Pasinato, Carlos Eduardo Mello, Marie-Aude Aufaure, and Geraldo Zimbrão. 2013. Generating Synthetic Data for Context-Aware Recommender Systems. In *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*. 563–567. https://doi.org/10.1109/BRICS-CCI-CBIC.2013.99

[21] Petar Ristoski, Gerben Klaas Dirk De Vries, and Heiko Paulheim. 2016. A collection of benchmark datasets for systematic evaluations of machine learning on the semantic web. In *The Semantic Web–ISWC 2016: 15th International Semantic Web Conference, Kobe, Japan, October 17–21, 2016, Proceedings, Part II 15*. Springer, 186–194.

[22] Joseph D Romano, Trang T Le, William La Cava, John T Gregg, Daniel J Goldberg, Praneel Chakraborty, Natasha L Ray, Daniel Himmelstein, Weixuan Fu, and Jason H Moore. 2021. PMLB v1.0: an open-source dataset collection for benchmarking machine learning methods. *Bioinformatics* 38, 3 (10 2021), 878–880. https://doi.org/10.1093/bioinformatics/btab727 arXiv:https://academic.oup.com/bioinformatics/article-pdf/38/3/878/49007845/btab727.pdf

[23] Manel Slokom. 2018. Comparing recommender systems using synthetic data. In *Proceedings of the 12th ACM Conference on Recommender Systems* (Vancouver, British Columbia, Canada) *(RecSys '18)*. Association for Computing Machinery, New York, NY, USA, 548–552. https://doi.org/10.1145/3240323.3240325

[24] Zhu Sun, Hui Fang, Jie Yang, Xinghua Qu, Hongyang Liu, Di Yu, Yew-Soon Ong, and Jie Zhang. 2023. DaisyRec 2.0: Benchmarking Recommendation for Rigorous Evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 7 (2023), 8206–8226. https://doi.org/10.1109/TPAMI.2022.3231891

[25] Zhu Sun, Di Yu, Hui Fang, Jie Yang, Xinghua Qu, Jie Zhang, and Cong Geng. 2020. Are we evaluating rigorously? benchmarking recommendation for reproducible evaluation and fair comparison. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 23–32.

[26] Jeyan Thiyagalingam, Mallikarjun Shankar, Geoffrey Fox, and Tony Hey. 2022. Scientific machine learning benchmarks. *Nature Reviews Physics* 4, 6 (2022), 413–420.

[27] Dana Thomas, Amy Greenberg, and Pascal Calarco. 2011. Scholarly Usage Based Recommendations: Evaluating bX for a Consortium.

[28] Bryan Tyrrell, Edward Bergman, Gareth Jones, and Joeran Beel. 2020. 'Algorithm-Performance Personas' for Siamese Meta-Learning and Automated Algorithm Selection. In *7th ICML Workshop on Automated Machine Learning*. 1–16. https://www.automl.org/wp-content/uploads/2020/07/AutoML_2020_paper_48.pdf

[29] Tobias Vente, Lukas Wegmeth, Alan Said, and Joeran Beel. 2024. From Clicks to Carbon: The Environmental Toll of Recommender Systems. In *Proceedings of the 18th ACM Conference on Recommender Systems* (2024-09-02).

[30] Kai Wang, Zhene Zou, Minghao Zhao, Qilin Deng, Yue Shang, Yile Liang, Runze Wu, Xudong Shen, Tangjie Lyu, and Changjie Fan. 2023. RL4RS: A Real-World Dataset for Reinforcement Learning based Recommender System. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2935–2944. https://doi.org/10.1145/3539618.3591899

[31] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, and Ming Zhou. 2020. MIND: A Large-scale Dataset for News Recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 3597–3606. https://doi.org/10.18653/v1/2020.acl-main.331

[32] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *CIKM*. ACM, 4653–4664.

[33] Yuhan Zhao, Rui Chen, Riwei Lai, Qilong Han, Hongtao Song, and Li Chen. 2023. Augmented Negative Sampling for Collaborative Filtering. In *Proceedings of the 17th ACM Conference on Recommender Systems* (Singapore, Singapore) *(RecSys '23)*. Association for Computing Machinery, New York, NY, USA, 256–266. https://doi.org/10.1145/3604915.3608811