How to perform reproducible experiments in the ELLIOT recommendation framework: data processing, model selection, and performance evaluation

Discussion Paper

Vito Walter Anelli[1], Alejandro Bellogín[2], Antonio Ferrara[1], Daniele Malitesta[1], Felice Antonio Merra[1], Claudio Pomo[1], Francesco Maria Donini[3], Eugenio Di Sciascio[1] and Tommaso Di Noia[1]

[1]*Politecnico di Bari, via Orabona, 4, 70125 Bari, Italy*

[2]*Universidad Autónoma de Madrid, Ciudad Universitaria de Cantoblanco, 28049 Madrid, Spain*

[3]*Università degli Studi della Tuscia, via Santa Maria in Gradi, 4, 01100 Viterbo, Italy*

Abstract

Recommender Systems have shown to be an effective way to alleviate the over-choice problem and provide accurate and tailored recommendations. However, the impressive number of proposed recommendation algorithms, splitting strategies, evaluation protocols, metrics, and tasks, has made rigorous experimental evaluation particularly challenging. ELLIOT is a comprehensive recommendation framework that aims to run and reproduce an entire experimental pipeline by processing a simple configuration file. The framework loads, filters, and splits the data considering a vast set of strategies. Then, it optimizes hyperparameters for several recommendation algorithms, selects the best models, compares them with the baselines, computes metrics spanning from accuracy to beyond-accuracy, bias, and fairness, and conducts statistical analysis. The aim is to provide researchers a tool to ease all the experimental evaluation phases (and make them reproducible), from data reading to results collection. ELLIOT is freely available on GitHub at https://github.com/sisinflab/elliot.

Keywords

Recommender Systems, Reproducibility, Adversarial Learning, Visual Recommenders, Knowledge Graphs

1. Introduction and Background

Recommendation Systems (RSs) have risen to prominence in the recent decade as the go-to option for personalized decision-support systems. Recommendation is a retrieval task in which a catalog of products is scored, and the highest-scoring items are shown to the user. Both academia and industry have focused their attention on RSs, as they were proven to supply customized goods to users. This collaborative effort yielded a diverse set of recommendation algorithms, spanning from memory-based to latent factor-based and deep learning-based approaches. However, the RSs community has become increasingly aware that adequately evaluating a model is not limited

IIR 2021 – 11th Italian Information Retrieval Workshop, September 13–15, 2021, Bari, Italy

vitowalter.anelli@poliba.it (V. W. Anelli); claudio.pomo@poliba.it (C. Pomo)

to measuring accuracy metrics alone. Another aspect that has attracted much attention concerns the evaluation of these models. While it is widely recognized the importance of beyond-accuracy metrics, an additional effort needed to compare models rigorously and fairly with each other in order to justify why one model performs differently from another. The problem of reproducing the experiments recurs when the need to recompute the whole set of experiments emerges. Be it a new experiment or not, it opens the doors to another class of problems: the number of possible design choices often imposes the researcher to define and implement only the chosen (usually limited) experimental setting. As highlighted in Konstan and Adomavicius [1], RS assessment is an essential and developing research issue connected to reproducibility, which is a cornerstone of the scientific process. Recently, academics have taken a closer look at this topic, also because the relevance and effect of such discoveries would rise depending on how well we assess the performance of a system. Some academics suggest that at least the following four steps should be defined within the assessment procedure to improve replicability and allow fair comparisons across various works (either frameworks, research papers, or published artifacts) [2]: data splitting, item suggestions, candidate item creation, and performance monitoring, which may be all done with data. These phases were completed with *dataset collecting* and *statistical testing* in a recent study [3]. Depending on the performance dimension to examine, several of these phases can be further classified, like performance measurement. Gunawardana and Shani [4] reviews different performance characteristics of RSs, comparing some measures, e.g., accuracy, coverage, confidence, trust, novelty, variety, and serendipity. However, to the best of our knowledge, no public implementation that provides more than one or two of these aspects exists. Furthermore, other dimensions such as bias (in particular, popularity bias [5]) and fairness [6] have lately been explored by the community [7, 8].

Reproducibility is the keystone of modern RSs research. Dacrema et al. [9] and Rendle et al. [10] have recently raised the need of comprehensive and fair recommender model evaluation. However, the outstanding success and the community interests in Deep Learning (DL) recommendation models raised the need for novel instruments. LibRec [11], Spotlight [12], and OpenRec [13] were the first open-source projects that made DL-based recommenders available – with less than a dozen of available models without filtering, splitting, and hyper-optimization tuning strategies. However, they do not provide a general tool for extensive experiments on the pre-elaboration and the evaluation of a dataset. Indeed, after the reproducibility hype [9, 10], DaisyRec [14] and RecBole [15] raised the bar of framework capabilities, making available both large set of models, data filtering/splitting and, above all, hyper-parameter tuning features.

From the researcher's point of view, our framework solves the issues mentioned above. ELLIOT [16] natively provides widespread research evaluation features, like the analysis of multiple cut-offs and several RSs. ELLIOT supplies, to date, 36 metrics, 13 splitting strategies, and 8 prefiltering policies to evaluate the diverse tasks and domains. Moreover, the framework offers, to date, 27 similarities, and 51 hyperparameter tuning combined approaches.

2. Elliot

ELLIOT (Figure 1) is an extendable framework with eight functional modules, each of which is in charge of a different aspect of the experimental suggestion process. The user is only meant

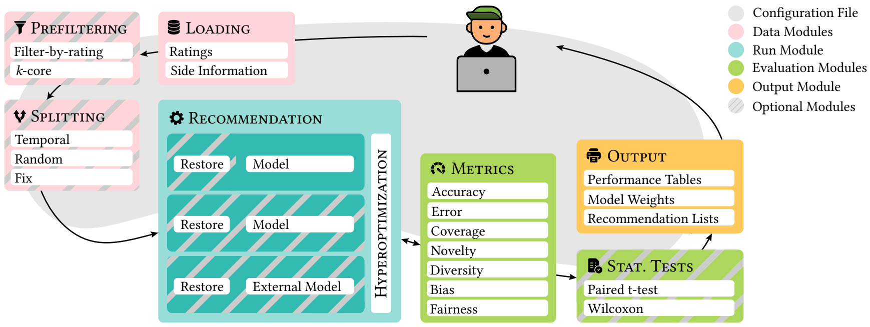

Figure 1: Overview of the modules involved in ELLIOT's operation.

to input human-level experimental flow information via a configurable configuration file, so what happens behind the scenes is transparent. Hence, ELLIOT allows the execution of the whole pipeline. In the following, we detail each module and how to create a configuration file.

2.1. Data Preparation

Loading. Different data sources, such as user-item feedback, and extra side information, may be required for RSs experiments. Hence, ELLIOT has a variety of *Loading* module implementations. The researcher/experiment designer may create prefiltering and splitting custom methods that can be saved and loaded to save time in the future. Additional data, such as visual [17, 18] and semantic features [19, 20, 21, 22], can be handled through a specific data loader.

Prefiltering. ELLIOT offers data filtering options based on two different techniques. The first is *Filter-by-rating*, whose purpose is to eliminate user-item interactions if the preference score is below a certain level. It can be (i) a *Numerical* value, e.g., 3.5, (ii) a *Distributional* detail, e.g., global rating average value, or (iii) a user-based distributional (*User Dist.*) value, e.g., user's average rating value. The second, *k-core*, filters out users, items, or both, with less than k recorded interactions. It can proceed iteratively (*Iterative k-core*) on both users and items until the filtering condition is met, i.e., all the users and items have at least k recorded interaction. Finally, the *Cold-Users* filtering allows retaining cold-users only.

Splitting. ELLIOT implements three splitting strategies: (i) *Temporal*, (ii) *Random*, and (iii) *Fix*. The *Temporal* method divides user-item interactions depending on the transaction timestamp, either by setting the timestamp, selecting the best one [23, 24], or using a hold-out (*HO*) mechanism. Hold-out (*HO*), *K*-repeated hold-out (*K-HO*), and cross-validation (*CV*) are all part of the *Random* methods. Finally, the *Fix* approach leverages an already split dataset.

Recommendation Models. The *Recommendation* module provides the functionalities to train (and restore) the ELLIOT recommendation models and the new ones integrated by users. To date, ELLIOT integrates around 50 recommendation models partitioned into two sets: (i) 38 *popular* models implemented in at least two of the other reviewed frameworks, (ii) other well-known state-of-the-art recommendation models implemented in less than two frameworks, like, MultiDAE [25], graph-learning, e.g., NGCF [26], visual [27], e.g., VBPR [28], adversarial-robust, e.g., AMR [29], and MSAPMF [30], content-aware, e.g., KaHFM [19], and KGFlex [31].

Hyper-parameter Tuning. According to Rendle et al. [10], Anelli et al. [32], hyperparameter optimization has a significant impact on performance. *Grid Search*, *Simulated Annealing*, *Bayesian Optimization*, and *Random Search* are all offered by ELLIOT. Additionally, it supports four different traversal techniques in the search space. *Grid Search* is automatically inferred when the user specifies the available hyperparameters.

2.2. Performance Evaluation

Metrics. ELLIOT provides a set of 36 evaluation metrics, partitioned into seven families: *Accuracy* [33, 34], *Error*, *Coverage*, *Novelty* [35], *Diversity* [36], *Bias* [37, 38, 39, 40, 41], and *Fairness* [42, 43]. It is worth mentioning that ELLIOT is the framework that exposes both the largest number of metrics and the only one considering bias and fairness measures. Moreover, the practitioner can choose any metric to drive the model selection and the tuning.
Statistical Tests. All other cited frameworks do not support statistical hypothesis tests, probably due to the need for computing fine-grained (e.g., per-user or per-partition) results and retaining them for each recommendation model. Conversely, ELLIOT helps computing two statistical hypothesis tests, i.e., *Wilcoxon* and *Paired t-test*, with a flag in the configuration file.

2.3. Framework Outcomes

When the training of recommenders is over, ELLIOT uses the *Output* module to gather the results. Three types of output files can be generated: (i) *Performance Tables*, (ii) *Model Weights*, and (iii) *Recommendation Lists*. Performance Tables come in the form of spreadsheets, including all the metric values generated on the test set for each recommendation model given in the configuration file. Cut-off-specific and model-specific tables are included in a final report (i.e., considering each combination of the explored parameters). Statistical hypothesis tests are also presented in the tables, as well as a JSON file that summarizes the optimal model parameters. Optionally, ELLIOT stores the model weights for the sake of future re-training.

2.4. Preparation of the Experiment

ELLIOT is triggered by a single configuration file written in YAML (e.g., refer to the toy example `sample_hello_world.yml`). The first section details the data loading, filtering, and splitting information defined in Section 2.1. The `models` section represents the recommendation models' configuration, e.g., Item-*k*NN. Here, the model-specific hyperparameter optimization strategies are specified, e.g., the grid-search. The `evaluation` section details the evaluation strategy with the desired metrics, e.g., nDCG in the toy example. Finally, `save_recs` and `top_k` keys detail, for example, the *Output* module abilities described in Section 2.3.

3. Conclusion and Future Work

ELLIOT is a framework that perform the entire recommendation process from an RS researcher's perspective. It requires the practitioner/researcher to write a configuration file to conduct a rigorous and reproducible experimental evaluation. The framework provides several functionalities: loading, prefiltering, splitting, hyperparameter optimization strategies, recommendation models, and statistical hypothesis tests. To the best of our knowledge, ELLIOT is the first recommendation framework providing an entire multi-recommender experimental pipeline based on a simple configuration file. We plan to extend ELLIOT in various directions to include: sequential recommendation scenarios, adversarial attacks, reinforcement learning-based recommendation systems, differential privacy facilities, sampled evaluation, and federated/ distributed recommendation.

References

[1] J. A. Konstan, G. Adomavicius, Toward identification and adoption of best practices in algorithmic recommender systems research, in: A. Bellogín, P. Castells, A. Said, D. Tikk (Eds.), Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation, RepSys 2013, Hong Kong, China, October 12, 2013, ACM, 2013, pp. 23–28. URL: https://doi.org/10.1145/2532508.2532513. doi:10.1145/2532508.2532513.

[2] A. Said, A. Bellogín, Comparative recommender system evaluation: benchmarking recommendation frameworks, in: A. Kobsa, M. X. Zhou, M. Ester, Y. Koren (Eds.), Eighth ACM Conference on Recommender Systems, RecSys '14, Foster City, Silicon Valley, CA, USA - October 06 - 10, 2014, ACM, 2014, pp. 129–136. URL: https://doi.org/10.1145/2645710.2645746. doi:10.1145/2645710.2645746.

[3] A. Bellogín, A. Said, Improving accountability in recommender systems research through reproducibility, CoRR abs/2102.00482 (2021). URL: https://arxiv.org/abs/2102.00482. arXiv:2102.00482.

[4] A. Gunawardana, G. Shani, Evaluating recommender systems, in: F. Ricci, L. Rokach, B. Shapira (Eds.), Recommender Systems Handbook, Springer, 2015, pp. 265–308. URL: https://doi.org/10.1007/978-1-4899-7637-6_8. doi:10.1007/978-1-4899-7637-6_8.

[5] H. Abdollahpouri, Popularity bias in ranking and recommendation, in: V. Conitzer, G. K. Hadfield, S. Vallor (Eds.), Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019, ACM, 2019, pp. 529–530. URL: https://doi.org/10.1145/3306618.3314309. doi:10.1145/3306618.3314309.

[6] M. D. Ekstrand, R. Burke, F. Diaz, Fairness and discrimination in retrieval and recommendation, in: B. Piwowarski, M. Chevalier, É. Gaussier, Y. Maarek, J. Nie, F. Scholer (Eds.), Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019, ACM, 2019, pp. 1403–1404. URL: https://doi.org/10.1145/3331184.3331380. doi:10.1145/3331184.3331380.

[7] C. Ardito, T. D. Noia, E. D. Sciascio, D. Lofú, G. Mallardi, C. Pomo, F. Vitulano, Towards a trustworthy patient home-care thanks to an edge-node infrastructure, in: HCSE, volume 12481 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 181–189.

[8] F. M. Donini, F. Narducci, C. Pomo, A. Ragone, Explanation in multi-stakeholder recommendation for enterprise decision support systems, in: CAiSE Workshops, volume 423 of *Lecture Notes in Business Information Processing*, Springer, 2021, pp. 39–47.

[9] M. F. Dacrema, P. Cremonesi, D. Jannach, Are we really making much progress? A worrying analysis of recent neural recommendation approaches, in: T. Bogers, A. Said,

P. Brusilovsky, D. Tikk (Eds.), Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019, ACM, 2019, pp. 101–109. URL: https://doi.org/10.1145/3298689.3347058. doi:10.1145/3298689.3347058.

[10] S. Rendle, W. Krichene, L. Zhang, J. R. Anderson, Neural collaborative filtering vs. matrix factorization revisited, in: R. L. T. Santos, L. B. Marinho, E. M. Daly, L. Chen, K. Falk, N. Koenigstein, E. S. de Moura (Eds.), RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020, ACM, 2020, pp. 240–248. URL: https://doi.org/10.1145/3383313.3412488. doi:10.1145/3383313.3412488.

[11] G. Guo, J. Zhang, Z. Sun, N. Yorke-Smith, Librec: A java library for recommender systems, in: A. I. Cristea, J. Masthoff, A. Said, N. Tintarev (Eds.), Posters, Demos, Late-breaking Results and Workshop Proceedings of the 23rd Conference on User Modeling, Adaptation, and Personalization (UMAP 2015), Dublin, Ireland, June 29 - July 3, 2015, volume 1388 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2015. URL: http://ceur-ws.org/Vol-1388/demo_paper1.pdf.

[12] M. Kula, Spotlight, https://github.com/maciejkula/spotlight, 2017.

[13] L. Yang, E. Bagdasaryan, J. Gruenstein, C. Hsieh, D. Estrin, Openrec: A modular framework for extensible and adaptable recommendation algorithms, in: Y. Chang, C. Zhai, Y. Liu, Y. Maarek (Eds.), Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018, ACM, 2018, pp. 664–672. URL: https://doi.org/10.1145/3159652.3159681. doi:10.1145/3159652.3159681.

[14] Z. Sun, D. Yu, H. Fang, J. Yang, X. Qu, J. Zhang, C. Geng, Are we evaluating rigorously? benchmarking recommendation for reproducible evaluation and fair comparison, in: R. L. T. Santos, L. B. Marinho, E. M. Daly, L. Chen, K. Falk, N. Koenigstein, E. S. de Moura (Eds.), RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020, ACM, 2020, pp. 23–32. URL: https://doi.org/10.1145/3383313.3412489. doi:10.1145/3383313.3412489.

[15] W. X. Zhao, S. Mu, Y. Hou, Z. Lin, K. Li, Y. Chen, Y. Lu, H. Wang, C. Tian, X. Pan, Y. Min, Z. Feng, X. Fan, X. Chen, P. Wang, W. Ji, Y. Li, X. Wang, J. Wen, Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms, CoRR abs/2011.01731 (2020). URL: https://arxiv.org/abs/2011.01731. arXiv:2011.01731.

[16] V. W. Anelli, A. Bellogín, A. Ferrara, D. Malitesta, F. A. Merra, C. Pomo, F. M. Donini, T. D. Noia, Elliot: A comprehensive and rigorous framework for reproducible recommender systems evaluation, in: SIGIR, ACM, 2021, pp. 2405–2414.

[17] W. Kang, C. Fang, Z. Wang, J. J. McAuley, Visually-aware fashion recommendation and design with generative image models, in: V. Raghavan, S. Aluru, G. Karypis, L. Miele, X. Wu (Eds.), 2017 IEEE International Conference on Data Mining, ICDM 2017, New Orleans, LA, USA, November 18-21, 2017, IEEE Computer Society, 2017, pp. 207–216. URL: https://doi.org/10.1109/ICDM.2017.30. doi:10.1109/ICDM.2017.30.

[18] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, T. Chua, Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention, in: N. Kando, T. Sakai, H. Joho, H. Li, A. P. de Vries, R. W. White (Eds.), Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017, ACM, 2017, pp. 335–344. URL: https://doi.org/10.1145/3077136.3080797. doi:10.1145/3077136.3080797.

[19] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ragone, J. Trotta, How to make latent factors interpretable by feeding factorization machines with knowledge graphs, in: C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. F. Cruz, A. Hogan, J. Song, M. Lefrançois, F. Gandon (Eds.), The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I, volume 11778 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 38–56. URL: https://doi.org/10.1007/978-3-030-30793-6_3. doi:10.1007/978-3-030-30793-6_3.

[20] V. W. Anelli, A. Calì, T. D. Noia, M. Palmonari, A. Ragone, Exposing open street map in the linked data cloud, in: IEA/AIE, volume 9799 of *Lecture Notes in Computer Science*, Springer, 2016, pp. 344–355.

[21] V. W. Anelli, T. D. Noia, P. Lops, E. D. Sciascio, Feature factorization for top-n recommendation: From item rating to features relevance, in: RecSysKTL, volume 1887 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 16–21.

[22] V. W. Anelli, P. Basile, D. G. Bridge, T. D. Noia, P. Lops, C. Musto, F. Narducci, M. Zanker, Knowledge-aware and conversational recommender systems, in: RecSys, ACM, 2018, pp. 521–522.

[23] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ragone, J. Trotta, Local popularity and time in top-n recommendation, in: L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, D. Hiemstra (Eds.), Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I, volume 11437 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 861–868. URL: https://doi.org/10.1007/978-3-030-15712-8_63. doi:10.1007/978-3-030-15712-8_63.

[24] A. Bellogín, P. Sánchez, Revisiting neighbourhood-based recommenders for temporal scenarios, in: M. Bieliková, V. Bogina, T. Kuflik, R. Sasson (Eds.), Proceedings of the 1st Workshop on Temporal Reasoning in Recommender Systems co-located with 11th International Conference on Recommender Systems (RecSys 2017), Como, Italy, August 27-31, 2017, volume 1922 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017, pp. 40–44. URL: http://ceur-ws.org/Vol-1922/paper8.pdf.

[25] D. Liang, R. G. Krishnan, M. D. Hoffman, T. Jebara, Variational autoencoders for collaborative filtering, in: P. Champin, F. L. Gandon, M. Lalmas, P. G. Ipeirotis (Eds.), Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018, ACM, 2018, pp. 689–698. URL: https://doi.org/10.1145/3178876.3186150. doi:10.1145/3178876.3186150.

[26] X. Wang, X. He, M. Wang, F. Feng, T. Chua, Neural graph collaborative filtering, in: B. Piwowarski, M. Chevalier, É. Gaussier, Y. Maarek, J. Nie, F. Scholer (Eds.), Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019, ACM, 2019, pp. 165–174. URL: https://doi.org/10.1145/3331184.3331267. doi:10.1145/3331184.3331267.

[27] V. W. Anelli, A. Bellogín, A. Ferrara, D. Malitesta, F. A. Merra, C. Pomo, F. M. Donini, T. D. Noia, V-elliot: Design, evaluate and tune visual recommender systems, in: RecSys 2021: Fifteenth ACM Conference on Recommender Systems (RecSys '21), September 27-October 1, 2021, Amsterdam, Netherlands, ACM, 2021. URL: https://doi.org/10.1145/3460231.3478881. doi:10.1145/3460231.3478881.

[28] R. He, J. J. McAuley, VBPR: visual bayesian personalized ranking from implicit feedback,

in: D. Schuurmans, M. P. Wellman (Eds.), Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, AAAI Press, 2016, pp. 144–150. URL: http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11914.

[29] J. Tang, X. Du, X. He, F. Yuan, Q. Tian, T. Chua, Adversarial training towards robust multimedia recommender system, IEEE Trans. Knowl. Data Eng. 32 (2020) 855–867. URL: https://doi.org/10.1109/TKDE.2019.2893638. doi:10.1109/TKDE.2019.2893638.

[30] V. W. Anelli, A. Bellogín, Y. Deldjoo, T. Di Noia, F. A. Merra, Msap: Multi-step adversarial perturbations on recommender systems embeddings, The International FLAIRS Conference Proceedings 34 (2021). doi:10.32473/flairs.v34i1.128443.

[31] V. W. Anelli, T. D. Noia, E. D. Sciascio, A. Ferrara, A. C. M. Mancino, Sparse feature factorization for recommender systems with knowledge graphs, in: RecSys 2021: Fifteenth ACM Conference on Recommender Systems (RecSys '21), September 27-October 1, 2021, Amsterdam, Netherlands, ACM, 2021. URL: https://doi.org/10.1145/3460231.3474243. doi:10.1145/3460231.3474243.

[32] V. W. Anelli, T. D. Noia, E. D. Sciascio, C. Pomo, A. Ragone, On the discriminative power of hyper-parameters in cross-validation and how to choose them, in: T. Bogers, A. Said, P. Brusilovsky, D. Tikk (Eds.), Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019, ACM, 2019, pp. 447–451. URL: https://doi.org/10.1145/3298689.3347010. doi:10.1145/3298689.3347010.

[33] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, K. Gai, Deep interest network for click-through rate prediction, in: Y. Guo, F. Farooq (Eds.), Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018, ACM, 2018, pp. 1059–1068. URL: https://doi.org/10.1145/3219819.3219823. doi:10.1145/3219819.3219823.

[34] G. Schröder, M. Thiele, W. Lehner, Setting goals and choosing metrics for recommender system evaluations, volume 811, 2011, pp. 78–85. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84891939277&partnerID=40&md5=c5b68f245b2e03725e6e5acc1e3c6289.

[35] S. Vargas, P. Castells, Rank and relevance in novelty and diversity metrics for recommender systems, in: B. Mobasher, R. D. Burke, D. Jannach, G. Adomavicius (Eds.), Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, USA, October 23-27, 2011, ACM, 2011, pp. 109–116. URL: https://dl.acm.org/citation.cfm?id=2043955.

[36] C. Zhai, W. W. Cohen, J. D. Lafferty, Beyond independent relevance: methods and evaluation metrics for subtopic retrieval, in: C. L. A. Clarke, G. V. Cormack, J. Callan, D. Hawking, A. F. Smeaton (Eds.), SIGIR 2003: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 28 - August 1, 2003, Toronto, Canada, ACM, 2003, pp. 10–17. URL: https://doi.org/10.1145/860435.860440. doi:10.1145/860435.860440.

[37] H. Abdollahpouri, R. Burke, B. Mobasher, Managing popularity bias in recommender systems with personalized re-ranking, in: R. Barták, K. W. Brawner (Eds.), Proceedings of the Thirty-Second International Florida Artificial Intelligence Research Society Conference, Sarasota, Florida, USA, May 19-22 2019, AAAI Press, 2019, pp. 413–418. URL: https://aaai.org/ocs/index.php/FLAIRS/FLAIRS19/paper/view/18199.

[38] H. Abdollahpouri, R. Burke, B. Mobasher, Controlling popularity bias in learning-to-rank

recommendation, in: P. Cremonesi, F. Ricci, S. Berkovsky, A. Tuzhilin (Eds.), Proceedings of the Eleventh ACM Conference on Recommender Systems, RecSys 2017, Como, Italy, August 27-31, 2017, ACM, 2017, pp. 42–46. URL: https://doi.org/10.1145/3109859.3109912. doi:10.1145/3109859.3109912.

[39] H. Yin, B. Cui, J. Li, J. Yao, C. Chen, Challenging the long tail recommendation, Proc. VLDB Endow. 5 (2012) 896–907. URL: http://vldb.org/pvldb/vol5/p896_hongzhiyin_vldb2012.pdf. doi:10.14778/2311906.2311916.

[40] Z. Zhu, J. Wang, J. Caverlee, Measuring and mitigating item under-recommendation bias in personalized ranking systems, in: J. Huang, Y. Chang, X. Cheng, J. Kamps, V. Murdock, J. Wen, Y. Liu (Eds.), Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, ACM, 2020, pp. 449–458. URL: https://doi.org/10.1145/3397271.3401177. doi:10.1145/3397271.3401177.

[41] V. Tsintzou, E. Pitoura, P. Tsaparas, Bias disparity in recommendation systems, in: R. Burke, H. Abdollahpouri, E. C. Malthouse, K. P. Thai, Y. Zhang (Eds.), Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems (RecSys 2019), Copenhagen, Denmark, September 20, 2019, volume 2440 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019. URL: http://ceur-ws.org/Vol-2440/short4.pdf.

[42] Y. Deldjoo, V. W. Anelli, H. Zamani, A. Bellogin, T. Di Noia, A flexible framework for evaluating user and item fairness in recommender systems, User Modeling and User-Adapted Interaction (2020) 1–47.

[43] Z. Zhu, X. Hu, J. Caverlee, Fairness-aware tensor-based recommendation, in: A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, H. Wang (Eds.), Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018, ACM, 2018, pp. 1153–1162. URL: https://doi.org/10.1145/3269206.3271795. doi:10.1145/3269206.3271795.