Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)
Sister Conferences Best Papers Track

Methodological Issues in Recommender Systems Research (Extended Abstract)*

Maurizio Ferrari Dacrema[1] , **Paolo Cremonesi**[1] and
Dietmar Jannach[2]

[1]Politecnico di Milano, Italy
[2]University of Klagenfurt, Austria
maurizio.ferrari@polimi.it, paolo.cremonesi@polimi.it, dietmar.jannach@aau.at

Abstract

The development of continuously improved machine learning algorithms for personalized item ranking lies at the core of today's research in the area of recommender systems. Over the years, the research community has developed widely-agreed best practices for comparing algorithms and demonstrating progress with offline experiments. Unfortunately, we find this accepted research practice can easily lead to phantom progress due to the following reasons: limited reproducibility, comparison with complex but weak and non-optimized baseline algorithms, over-generalization from a small set of experimental configurations. To assess the extent of such problems, we analyzed 18 research papers published recently at top-ranked conferences. Only 7 were reproducible with reasonable effort, and 6 of them could often be outperformed by relatively simple heuristic methods, e.g., nearest neighbors. In this paper, we discuss these observations in detail, and reflect on the related fundamental problem of over-reliance on offline experiments in recommender systems research.

1 Introduction

Recommender systems are one of the most successful and visible application areas of machine learning technology in practice, and there is no doubt that personalized recommendations can lead to substantial benefits for businesses [Jannach and Jugovac, 2019]. Correspondingly, countless technical approaches were proposed during the last 25 years in the academic literature, from early nearest-neighbor and matrix factorization techniques [Ricci *et al.*, 2011] to the latest deep learning models [He *et al.*, 2017].

To evaluate and compare these approaches, the research community mostly relies on offline experimentation [Jannach *et al.*, 2012; Cremonesi *et al.*, 2010]. Even though there exists no truly standardized methodology for such data-based experiments, there are some widely accepted best practices that researchers typically adopt, e.g., to establish comparability with previous research. In a typical experimental setup, a newly proposed approach is compared to at least one, but usually more, existing baseline methods which are often claimed to represent the state-of-the-art. The actual comparison is then based on one or several of the most commonly-used performance (mostly accuracy) metrics for one or more datasets. Applying cross-validation is typically considered a good practice as well, but not always a necessity. In more recent years, statistical significance tests are reported more frequently, and sharing the code and the data used in the experiments is often encouraged.

Having such common practices without a doubt helps the community to make research comparable and reproducible, at least to a certain extent. However, researchers still have a lot of freedom to decide on the details of their experimental designs. For example, many technical proposals are not explicitly designed for a particular application domain but are—implicitly or explicitly—claimed to be advancing the state-of-the-art independently of the domain. Usually, despite this claim, the evaluation is limited to a small set of datasets and a selection of accuracy metrics and cut-off lengths (i.e., the number of recommended items). The criteria for those choices are often not well explained and the experimental design could therefore appear arbitrary. Furthermore, the combination of ample freedom and lack of justification for the experimental design contribute to a lack of clarity in what represents the state-of-the-art for a given scenario and what should thus be included as a baseline.

As a result of this freedom in the experimental design, it can become difficult to assess if a new technical proposal truly represents a generalizable advancement, in particular when there are additional reproducibility issues. Yet another potential problem of our common research approach is that the researcher that proposes a certain method is often also the only one that evaluates it before publication. This could have an impact on the experimental evaluation due to an unconscious search for a confirmation of the expected progress [Nickerson, 1998], and could manifest itself in an extensive optimization and tuning of the newly proposed approach, whereas less attention is paid to optimizing the baselines.

In the end, all these potential issues might lead to what we call **phantom progress**. With countless publications each

*This work is an extended abstract based on the publication "Are we really making much progress? A worrying analysis of recent neural recommendation approaches" which received the Best Long Paper Award at the ACM Conference on Recommender Systems (RecSys) 2019 [Ferrari Dacrema *et al.*, 2019b].

claiming to improve over the state-of-the-art we are seemingly making lots of progress but, given the potential limitations of our research methodology, this progress might be much smaller than expected or even non-existent.

As part of our ongoing research work [Ferrari Dacrema *et al.*, 2019b; Ferrari Dacrema *et al.*, 2019a], we have therefore examined how severe these methodological problems are by analyzing the most recent approaches for top-n recommendation published at top-level conferences. The outcomes of this analysis are more than worrying. From the 18 considered algorithms, only 7 could be reproduced with reasonable effort. We then benchmarked these 7 algorithms by comparing them with conceptually much simpler and long-known techniques, e.g., based on nearest neighbors. To our surprise, 6 of the newest and complex algorithms were outperformed by at least one of the simpler techniques when using the same experimental setup that was used in the original papers.

We will review the outcomes that we originally reported in [Ferrari Dacrema *et al.*, 2019b] in Section 2. Afterwards, in Section 3, we summarize the identified methodological issues of our current research practice and give an outlook on possible remedies.

2 Experiment Setup and Results

2.1 Reproducibility

We analyzed the papers published between 2015 and 2018 in the following conference series: KDD, SIGIR, TheWebConf (WWW), and RecSys. According to the specific methodology reported in our paper, we identified 18 relevant articles but, based on the code provided by the authors, we could reproduce less than 40% of them. Table 1 reports the percentage of reproducible works per conference series. We can observe some variation in reproducibility at different conferences.

Conference	Reproducibility Ratio
KDD	3/4 (75%)
WWW	2/4 (50%)
SIGIR	1/3 (30%)
RecSys	1/7 (14%)
Total	7/18 (39%)

Table 1: Statistics of reproducibility of algorithms for *top-n* recommendation per conference series from 2015 to 2018.

2.2 Evaluation Methodology

The goal of our evaluation approach was to use the exact same experimental setup—including datasets, train-test splits, metrics, cut-off lengths, and hyper-parameters—that was used in the original papers, but to include additional baselines in the comparison. As baselines we included *(i)* a non-personalized method that recommends the most popular items to everyone, *(ii)* traditional user and item-based nearest neighbor techniques [Ricci *et al.*, 2011], *(iii)* two comparably recent, computationally simple graph-based methods ($P^3\alpha$ and $RP^3\beta$) [Cooper *et al.*, 2014; Paudel *et al.*, 2017], *(iv)* two content-based/collaborative hybrid methods based on nearest neighbors. We systematically optimized the hyper-parameters for these baselines per dataset. We share all code, data, and hyper-parameters used in our experiments online[1].

	CiteULike-a			
	HR@5	NDCG@5	HR@10	NDCG@10
TopPopular	0.1803	0.1220	0.2783	0.1535
UserKNN	**0.8213**	**0.7033**	**0.8935**	**0.7268**
ItemKNN	**0.8116**	**0.6939**	0.8878	**0.7187**
$P^3\alpha$	**0.8202**	**0.7061**	0.8901	**0.7289**
$RP^3\beta$	**0.8226**	**0.7114**	**0.8941**	**0.7347**
CMN	0.8069	0.6666	0.8910	0.6942
	Pinterest			
	HR@5	NDCG@5	HR@10	NDCG@10
TopPopular	0.1668	0.1066	0.2745	0.1411
UserKNN	**0.6886**	**0.4936**	0.8527	**0.5470**
ItemKNN	**0.6966**	**0.4994**	**0.8647**	**0.5542**
$P^3\alpha$	0.6871	**0.4935**	0.8449	**0.5450**
$RP^3\beta$	**0.7018**	**0.5041**	**0.8644**	**0.5571**
CMN	0.6872	0.4883	0.8549	0.5430
	Epinions			
	HR@5	NDCG@5	HR@10	NDCG@10
TopPopular	**0.5429**	**0.4153**	**0.6644**	**0.4547**
UserKNN	0.3506	0.2983	0.3922	0.3117
ItemKNN	0.3821	0.3165	0.4372	0.3343
$P^3\alpha$	0.3510	0.2989	0.3891	0.3112
$RP^3\beta$	0.3511	0.2980	0.3892	0.3103
CMN	0.4195	0.3346	0.4953	0.3592

Table 2: Results for the CMN method using metrics and cutoffs from the original paper. Numbers in bold indicate the best results in a column or when a baseline outperformed CMN.

2.3 Results

Collaborative Memory Network for Recommendation Systems (CMN). Proposed by [Ebesu *et al.*, 2018] in three variants at SIGIR, this method combines memory networks and neural attention mechanisms with latent factor and neighborhood models. We report the results for the best variant (CMN-3) in Table 2. We omit the results for the other variants for space reasons. All details can however be found in [Ferrari Dacrema *et al.*, 2019b]. The results obtained for CMN are in various ways representative of what we observed for the other methods. For two datasets (CiteULike-a and Pinterest), the recent CMN method was outperformed by most of the simpler baselines. On the Epinions dataset, finally, CMN was much better than our personalized baselines, but this dataset has such a skewed distribution that recommending the most popular items to everyone was by far the most effective method in this evaluation. The relatively good performance in this settings of CMN is therefore attributed to the higher popularity bias of CMN.

Leveraging Meta-path based Context for *top-n* Recommendation with a Neural Co-attention Model (MCRec).

[1]https://github.com/MaurizioFD/RecSys2019_DeepLearning_Evaluation

Proposed by [Hu *et al.*, 2018] at KDD, MCRec is a meta-path based model, which applies a novel co-attention mechanism and a priority-based sampling technique to select higher-quality path instances. The authors provided an implementation which had the meta-paths hard-coded, and we therefore could reliably reproduce the results only for the small MovieLens dataset. For this dataset, however, it turned out that the traditional item-based nearest-neighbor technique was better than MCRec on all performance measures. In the context of this work, additional potential problems were identified based on the provided source code. For example, the accuracy metrics reported by the source code correspond to the maximum values that are obtained across different epochs when evaluating on the test set. Furthermore, in the original article, the hyperparameters of the examined baselines were said to be taken from the original papers and not optimized for the datasets used in the evaluation. Such issues were also found for other papers in our analysis. Finally, the NDCG metric was implemented in an uncommon way.

Collaborative Variational Autoencoder for Recommender Systems (CVAE). CVAE [Li and She, 2017] was also presented at KDD and is a hybrid technique that leverages collaborative information and content features. While CVAE, according to the original experiments, outperforms the previous neural methods including CDL [Wang *et al.*, 2015], our experiments indicate these neural methods were not necessarily strong baselines. In fact, our simple hybrid methods outperformed CVAE in all but one experimental configuration that had only *one* interaction per user in the training set.

Collaborative Deep Learning for Recommendation Systems (CDL). CDL [Wang *et al.*, 2015] was presented at SIGIR. It is a stacked denoising autoencoder which learns a hybrid representation of content and collaborative information. According to our experiments and in line with the results for CVAE, the simple hybrid baselines outperform CDL in three out of four dataset configurations. On a dense dataset, CDL is also outperformed by pure collaborative baselines with recommendation lists shorter than 100 items. Again, CDL is only better than our baselines on one small and very sparse dataset with only one interaction per user in the training set.

Neural Collaborative Filtering (NCF). NCF, proposed by [He *et al.*, 2017] at WWW, generalizes matrix factorization by replacing the inner product with a neural architecture. NCF has gained significant popularity in recent years and is considered as a baseline in all the papers we analyzed that were published afterwards. We reproduced the results for both datasets used in the original paper. For one dataset, we observed that traditional methods, like an item-based nearest-neighbor, outperform NCF. For the other dataset, NFC was better than the nearest neighbor techniques, but not better than the linear SLIM method [Levy and Jack, 2013]. Regarding methodological issues, we observed in the source code that the number of training epochs, which should be optimized on the validation set, was optimized on test data.

Spectral Collaborative Filtering (SpectralCF). This method, proposed by [Zheng *et al.*, 2018] at RecSys, uses a novel convolution operation to make collaborative recommendations directly in the spectral domain. In our initial experiments, we found that the algorithm was competitive with our baselines only for one of three datasets. Specifically, it was the dataset for which the authors shared the train-test split. An investigation revealed the distribution of the data in the provided test set was very different from what we would likely obtain by applying a random sampling procedure. After creating the train-test splits by our own, we found that SpectralCF does not work as expected and consistently exhibits lower performance when compared with personalized and even non-personalized baselines.

Variational Autoencoders for Collaborative Filtering (Mult-VAE). Proposed by [Liang *et al.*, 2018] at WWW, Mult-VAE implements a variational autoencoder for implicit feedback datasets. We could reproduce the results reported in the original paper for two datasets and found that the proposed method outperforms all our baselines on all metrics by a large margin. Thus, we identified at least one neural method in our analysis that was consistently better than our simple baselines. Like for NCF, we therefore also trained the SLIM method to investigate how Mult-VAE compares with more modern methods. It turned out that Mult-VAE was also up to 5% better than SLIM for many metrics. For some metrics and cut-off lengths, these improvements however tend to vanish, in particular when the evaluation measure and the optimization target are the same. In some cases and depending on the cut-off length, SLIM was also slightly better than Mult-VAE.

3 Discussion and Ways Forward

3.1 Summary of Issues

In summary, our experimental evaluations revealed the following issues of today's research practice, which can easily lead to a lack of progress in our field.

1. Lack of Reproducibility
2. Comparison with Complex yet Weak Baselines
3. Lack of Proper Optimization of Baselines
4. Arbitrariness of Experimental Configurations
5. Technical Issues in the Evaluation

Reproducibility. Only 40% of the papers in question could be reliably reproduced. Sharing code and data has become more common in recent years, but even at top-level publication outlets this is not a common practice yet.

Weak Baselines. Regarding the choice of the baselines, in recent years complex neural methods are typically considered the state-of-the-art. Our analysis however shows that some of these methods are actually not strong baselines. The achieved progress is therefore sometimes non-existent as these complex models do not outperform previous methods.

Optimization of Baselines. A related problem is that researchers sometimes do not carefully optimize all the baselines, but take hyper-parameter configurations from the original papers, even though these parameters were sometimes obtained using different datasets and experimental designs. In fact, any experimental analysis in which not all algorithms are optimized for all datasets is mostly meaningless. In order to substantiate a claim regarding the strength of the chosen

Method	Datasets	Split	Metrics	Cutoffs
CMN	Epinions, CiteULike-a, Pinterest	leave-one-out, 100 negatives	HR, NDCG	5, 10
MCRec	ML100K, LastFM, Yelp	holdout 80/20, 50 negative per positive	Precision, Recall, NDCG	10
CVAE	CiteULike-a, CiteULike-t	1 or 10 interactions in training	Recall	50-300, step 50
CDL	CiteULike-a, CiteULike-t, Netflix	1 or 10 interactions in training	Recall	50-300, step 50
NCF	ML1M, Pinterest	leave-one-out, 100 negatives	HR, NDCG	1, 5, 10
SpectralCF	ML1M, HetRec, Amazon Video	holdout 80/20	Recall, MAP	20-100 step 10
Mult-VAE	ML20M, Netflix, MSD	cold user, holdout 80/20 profile	Recall, NDCG	20, 50, 100

Table 3: Evaluation protocol used in the reproducible articles: datasets, train-test split, evaluation metrics and recommendation list lengths.

baselines, selecting a state-of-the-art algorithm is therefore required, but not sufficient. Usually, any algorithm, even the latest one, which has not been optimized for the given experimental scenario will often lead to non-competitive accuracy and therefore does not constitute a strong baseline.

Arbitrariness of Experiments. We found that all sorts of metrics and cut-off lengths were used in the analyzed papers. The choice of the evaluation datasets seems almost arbitrary and not driven by an application problem or by theory. Nonetheless, claims regarding general improvements over the state-of-the-art are common, even though the algorithms are evaluated only in a very specific experimental configuration. Table 3 gives us an impression of the various experiment configurations that were used in the seven reproduced papers.

Technical Issues. Finally, we also found a number of other methodological issues in the evaluation. For example, in more than one paper certain parameters were optimized on the test set. Cross-validation and significance tests are common, but do not seem to be strictly required in our community. For some papers, we actually found major mistakes like a non-random selection of data points for the test set.

3.2 Is This a Problem of Deep Learning?

The described issues of today's research practice in applied machine learning are actually not entirely new, and they are not tied to recommender systems research or deep learning techniques. More than ten years ago, researchers found that the claimed improvements for a certain information retrieval task over a decade "don't add up" [Armstrong *et al.*, 2009]. In 2019, it was then found by Lin that some of the problems from ten years ago, e.g., regarding the choice of the baselines, are still there in the Information Retrieval field [Lin, 2019]. In 2018, [Makridakis *et al.*, 2018] compared various statistical and machine learning techniques for the problem of time series prediction. They found that for certain problems some more recent and complex techniques are less accurate than relatively simple and long-known approaches. In the context of recommender systems, [Rendle *et al.*, 2019] analyzed recent works for the problem of rating prediction and found that progress is quite limited. Finally, for the problem of session-based recommendations, recent works indicate that often very simple techniques are able to outperform the latest neural approaches [Ludewig and Jannach, 2018; Ludewig *et al.*, 2019].

Overall, while we focused on recent neural techniques in our analysis, we find that some of the partially long-standing underlying methodological issues can be found also for research works that are not based on deep learning. A particular problem with deep learning might however lie in the computational complexity of some methods. According to our experiments, systematic hyper-parameter tuning for one single neural baseline can take several days or even weeks, depending on the dataset size, even when using modern GPUs.

3.3 Ways Forward

Some of the observed issues, in particular reproducibility, should be relatively easy to address, e.g., through stricter publication requirements. Technically, establishing reproducibility in our research area in general seems easier than in other domains, and with the use of virtualization technology even sharing the execution environments has become possible. At least a part of the other problems, like the choice and optimization of the baselines or the justification of a certain experimental design, can probably be alleviated by increased awareness and improved review processes. Given the current boom in machine learning, there is however already now a certain thinness of the reviewer pool [Lipton and Steinhardt, 2018], which makes it difficult to maintain the high levels of review quality that we aim for.

However, even if all problems regarding methodology and reproducibility were fixed, we are still facing one further fundamental issue, which is the *over-reliance on offline experimentation*. In the analyzed papers—and in most published research on algorithms—the generated recommendations are never shown to any user. While accurate relevance predictions are clearly important for any recommender systems, higher prediction accuracy on historical datasets does not necessarily lead directly to better recommendations. Various industry reports as well as user studies in fact indicate that better offline performance does not necessarily translate into better value for users or providers [Cremonesi *et al.*, 2012; Gomez-Uribe and Hunt, 2015; Maksai *et al.*, 2015]. It therefore stands to question if small accuracy improvements for particularly chosen datasets and experimental configurations that are often not well justified would matter in the real world.

Ultimately, providing recommendations is not only an algorithmic problem, it is much more multi-faceted and to a large part a problem of human-computer interaction [Jannach *et al.*, 2016]. Therefore, future works should more often consider humans in the loop when evaluating and go beyond the somewhat narrow and probably over-simplifying problem abstraction that we rely on today.

References

[Armstrong *et al.*, 2009] Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. Improvements that don't add up: Ad-hoc retrieval results since 1998. In *CIKM '09*, pages 601–610, 2009.

[Cooper *et al.*, 2014] Colin Cooper, Sang Hyuk Lee, Tomasz Radzik, and Yiannis Siantos. Random walks in recommender systems: exact computation and simulations. In *WWW '14*, pages 811–816, 2014.

[Cremonesi *et al.*, 2010] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys '10)*, pages 39–46, 2010.

[Cremonesi *et al.*, 2012] Paolo Cremonesi, Franca Garzotto, and Roberto Turrin. Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 2(2):1–41, 2012.

[Ebesu *et al.*, 2018] Travis Ebesu, Bin Shen, and Yi Fang. Collaborative memory network for recommendation systems. *SIGIR '18*, 2018.

[Ferrari Dacrema *et al.*, 2019a] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. A troubling analysis of reproducibility and progress in recommender systems research. *arXiv:1911.07698*, 2019.

[Ferrari Dacrema *et al.*, 2019b] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches. *RecSys '19*, 2019.

[Gomez-Uribe and Hunt, 2015] Carlos A. Gomez-Uribe and Neil Hunt. The Netflix recommender system: Algorithms, business value, and innovation. *Transactions on Management Information Systems*, 6(4):13:1–13:19, 2015.

[He *et al.*, 2017] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *WWW '17*, pages 173–182, 2017.

[Hu *et al.*, 2018] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Philip S Yu. Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In *KDD '18*, pages 1531–1540, 2018.

[Jannach and Jugovac, 2019] Dietmar Jannach and Michael Jugovac. Measuring the business value of recommender systems. *ACM Transactions on Management Information Systems*, 10(4), 2019.

[Jannach *et al.*, 2012] Dietmar Jannach, Markus Zanker, Mouzhi Ge, and Marian Gröning. Recommender systems in computer science and information systems - a landscape of research. In *EC-Web 2012*, pages 76–87, 2012.

[Jannach *et al.*, 2016] Dietmar Jannach, Paul Resnick, Alexander Tuzhilin, and Markus Zanker. Recommender systems - beyond matrix completion. *Communications of the ACM*, 59(11):94–102, 2016.

[Levy and Jack, 2013] Mark Levy and Kris Jack. Efficient top-n recommendation by linear regression. In *RecSys Large Scale Recommender Systems Workshop*, 2013.

[Li and She, 2017] Xiaopeng Li and James She. Collaborative variational autoencoder for recommender systems. In *KDD '17*, pages 305–314, 2017.

[Liang *et al.*, 2018] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. Variational autoencoders for collaborative filtering. In *WWW '18*, pages 689–698, 2018.

[Lin, 2019] Jimmy Lin. The neural hype and comparisons against weak baselines. *SIGIR Forum*, 52(2):40–51, January 2019.

[Lipton and Steinhardt, 2018] Zachary C. Lipton and Jacob Steinhardt. Troubling trends in machine learning scholarship. *arXiv:1911.07698*, 2018.

[Ludewig and Jannach, 2018] Malte Ludewig and Dietmar Jannach. Evaluation of session-based recommendation algorithms. *User-Modeling and User-Adapted Interaction*, 28(4–5):331–390, 2018.

[Ludewig *et al.*, 2019] Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. Performance comparison of neural and non-neural approaches to session-based recommendation. In *RecSys '19*, 2019.

[Makridakis *et al.*, 2018] Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. Statistical and machine learning forecasting methods: Concerns and ways forward. *PloS one*, 13(3), 2018.

[Maksai *et al.*, 2015] Andrii Maksai, Florent Garcin, and Boi Faltings. Predicting online performance of news recommender systems through richer evaluation metrics. In *RecSys '15*, pages 179–186, 2015.

[Nickerson, 1998] Raymond S. Nickerson. Confirmation bias: A ubiquitous phenomenon in many guises. *Review of General Psychology*, 2(2):175–220, 1998.

[Paudel *et al.*, 2017] Bibek Paudel, Fabian Christoffel, Chris Newell, and Abraham Bernstein. Updatable, accurate, diverse, and scalable recommendations for interactive applications. *ACM Transactions on Interactive Intelligent Systems*, 7(1):1, 2017.

[Rendle *et al.*, 2019] Steffen Rendle, Li Zhang, and Yehuda Koren. On the difficulty of evaluating baselines: A study on recommender systems. *arXiv:1905.01395*, 2019.

[Ricci *et al.*, 2011] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer, 2011.

[Wang *et al.*, 2015] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *KDD '15*, pages 1235–1244, 2015.

[Zheng *et al.*, 2018] Lei Zheng, Chun-Ta Lu, Fei Jiang, Jiawei Zhang, and Philip S. Yu. Spectral collaborative filtering. In *RecSys '18*, pages 311–319, 2018.