Progress in Recommender Systems Research: Crisis? What Crisis?*

Paolo Cremonesi, Dietmar Jannach

Scholars in algorithmic recommender systems research have developed a largely standardized scientific method, where progress is claimed by showing that a new algorithm outperforms existing ones on or more accuracy measures. In theory, reproducing and thereby verifying such improvements is easy, as it merely involves the execution of the experiment code on the same data. However, as recent work shows, the reported progress is often only virtual, because of a number of issues related to (i) a lack of reproducibility, (ii) technical and theoretical flaws, and (iii) scholarship practices that are strongly prone to researcher biases. As a result, several recent works could show that the latest published algorithms actually do not outperform existing methods when evaluated independently. Despite these issues, we currently see no signs of a crisis, where researchers re-think their scientific method, but rather a situation of stagnation, where researchers continue to focus on the same topics. In this paper, we discuss these issues, analyze their potential underlying reasons, and outline a set of guidelines to ensure progress in recommender systems research.

The central components in any recommender system are the algorithms that determine which items will be shown to individual users based on specific contexts. Correspondingly, a core topic of recommender systems research lies in the continuous improvement of these algorithms. Early research from more than a quarter-century ago relied on comparably simple algorithms like nearest-neighbor heuristics [Resnick et al 1994]. Since then, increasingly more complex machine learning approaches were proposed, from linear models, to matrix factorization techniques, to the latest deep learning models that we see today [Koren 2008, Ning and Karypis 2011, He et al. 2017, Wu et al. 2019]. Although the type of problems covered by the research has gradually expanded over the years, the classical rating prediction and top-N recommendation problems still attract most of the attention and the research community has developed a seemingly standardized way of operationalizing these problems. In almost all published research, algorithms are compared through offline experiments on historical data. In such experiments, progress is claimed if a new algorithm is better in predicting

*Appeared in AI Magazine, Vol. 42(3), 2021

held-out data than previous ones in terms of measures such as RMSE, Precision, Recall, MAP, or NDCG.

Given the huge amount of published research works over twenty-five years and the agreed-upon standards for conducting experimental research, one would certainly expect to observe continuing and strong progress in this area. This should particularly be the case because the majority of algorithms papers claim to improve the state-of-the art. However, a number of scholars repeatedly voiced concerns regarding this progress over the years. Ten years ago, for example, [Ekstrand et al. 2011], found it ''[...] *difficult to reproduce and extend recommender systems research results.*'' Later, in 2013, a workshop on reproducibility and replication was held in conjunction with the ACM Conference on Recommender Systems. In that context, [Konstan and Adomavicius 2013] observed that the community faces a situation of limited progress, in particular because of a lack of rigor and problematic evaluation practices. Related phenomena were investigated in more depth in [Beel et al. 2016]. Here, the authors not only found that the recommender systems community ''[...] *widely seems to accept that research results are difficult to reproduce.*''. Their experiments also showed that minor variations in the evaluation setup can have significant effects on the observed outcomes.

In our own past works, we benchmarked more than twenty of the most recent deep learning (neural) methods against a set of non-neural algorithms, most of them published many years earlier. Very surprisingly, these studies showed that---except for a very small set of experimental configurations---the latest neural methods, despite their computational complexity, were almost consistently outperformed by long-known existing methods, see [Ferrari Dacrema et al. 2019, Ferrari Dacrema et al. 2021, Ferrari Dacrema et al. 2020a] and [Ludewig et al. 2020]. In several cases, even nearest-neighbor techniques developed 25 years ago were better than the latest machine learning models. These observations were made both for the traditional top-N recommendation task and for more recent session-based recommendation scenarios. These findings, which were reproduced and confirmed also by other researchers [Rendle et al. 2020, Kouki et al. 2020, Sun et al. 2020], indicate that we are facing major issues, which may prevent us from truly moving forward as a field.

Furthermore, our studies revealed that the reproducibility of existing works is actually not high. In the context of the works presented in [Ferrari Dacrema et al. 2019, Ferrari Dacrema et al. 2021], for example, we found that about 55% of the papers published at top-level conferences could not be reproduced based on shared code and data, nor by contacting authors for help and clarification. Put differently, for 55% of the papers the verification or falsification of the made claims---which is a central element of any scientific research---is almost impossible.

All in all, it seems that algorithms research is showing signs of stagnation, where constantly new models are proposed, but where the claimed improvements ''don't add up'', as observed previously in the domain of information retrieval [Armstrong et al. 2009, Yang et al. 2019]. Various reasons contribute to this phenomenon. Researchers for example sometimes compare their new methods to existing methods (baselines), which are too weak in general or which are not

properly tuned. Furthermore, the fact that researchers are generally flexible regarding their experimental designs in terms of baselines, datasets, evaluation protocol, and performance measures may easily lead to researcher bias, where algorithm designers may only look for evidence that supports the superiority of their own method. Unfortunately, despite the existing evidence of the problem, we do not observe clear signs of a crisis yet, where researchers would re-think their methodological approaches. In contrast, with the current boom of AI and machine learning research, even more algorithms are published every year that rely on research practices that are only partially suited to demonstrate progress.

This paper aims to stimulate a discussion on the lack of progress in some areas of research on recommender systems. While supporting the points described here, we do not question the overall quality of research in recommender systems: in many aspects, the community has advanced far beyond what we had achieved a decade ago. Also, the bad practices identified here are not specific to any individual or institution. We made such mistakes by ourselves, but by making researchers more aware of them, we aim to avoid them in the future.

In the remainder of the paper, we first discuss the issues of today's research practice in machine learning applied to recommender systems. We then look at the potential underlying reasons for the observed phenomena. Finally, we sketch a number of ways forward to increase the reproducibility of our research and ensure progress in the future.

The Problems

According to our observations, it is not one single problem that leads to the apparent stagnation, but multiple issues that can all contribute to a lack of progress:

- Lack of reproducibility: If a work has never been reproduced---either because it is not reproducible or because the research community never attempted to reproduce it---its claimed gains can neither be verified nor denied, and as such, any progress described in the work cannot be considered reliable.
- Methodological issues: Some works exhibit experimental flaws that produce non-existent gains over state-of-the-art approaches; examples are the adoption of weak baselines, or the leakage of testing information into the training process.
- Theoretical flaws: A few works describe new methods based on assumptions that are not verified, neither theoretically nor empirically.

Lack of Reproducibility

Reproducibility in research on recommendation algorithms is important because it is the only tool for the research community to ensure the correctness of an empirical study. A reviewer cannot guarantee that the gains reported in an empirical paper are correct. Exceptions are, for instance, theoretical studies, where

mathematical properties of new models can be formally proved. Reproducibility ensures transparency and creates the possibility for the research community to confirm or refute what is stated in an empirical paper. Reproducibility can only work as a tool for verifying the correctness of claimed empirical results if two requirements are met: *(i)* papers must be reproducible and *(ii)* researchers must reproduce previous empirical results before relying on them for further progress.

According to the study by [Gundersen et al. 2018], the vast majority of empirical works in AI in general is not well documented and thus not reproducible. Given the inconsistent use of the terms reproducibility and replication, [Gundersen and Kjensmo 2018] define reproducibility as the ''*[...] ability of an independent research team to produce the same results using the same AI method based on the documentation made by the original research team*.'' Moreover, they define three levels of reproducibility, where level R1, termed Experiment Reproducibility, refers to a situation when ''*[...] the execution of the same implementation of an AI method produces the same results when executed on the same data.*'' In the other two levels of reproducibility, R2 and R3, independent researchers can either *(i*) obtain the same results for the same data using an alternative implementation, or *(ii)* obtain the same results using an alternative implementation on different data. R1 is thus the weakest level, as it does not assess the sensitivity of the AI method to implementation details, nor does it assess the ability of the method to generalize under different experimental conditions. In the rest of this discussion, whenever we talk about reproducibility, we refer to level R1 as defined by [Gundersen and Kjensmo 2018], unless differently specified.

Research in recommender systems suffers from the same problems as general AI. In our previous studies [Ferrari Dacrema et al. 2019, Ferrari Dacrema et al. 2021], we systematically scanned top-level conference series such as KDD, IJCAI or SIGIR for algorithmic works on the traditional top-N recommendation problem [Cremonesi et al. 2010]. As mentioned above, less than 50% of the works were reproducible despite the high quality expectations for such conferences. On a positive note, the observed level of R1 reproducibility, which requires that code and data are shared by researchers, is much higher than in the study on general AI research, where [Gundersen and Kjensmo 2018] for example found that the method code was only shared in 8% of the cases.

Note that sharing the method code, i.e., the implementation of a recommendation machine learning model, and the used data is not necessarily sufficient to ensure R1 reproducibility. In our studies, we found that sometimes researchers shared some code, but the code was limited to a skeleton of the method and not executable. Furthermore, in the majority of the cases, no code was provided for data preprocessing, hyperparameter tuning, or for running the experiment including the compared baselines. The code of the baseline methods was actually missing in the majority of the cases. As a result, it remains difficult to understand what was done *exactly* in the experiment. Ultimately, any gains that are observed for a new method might, at worst, be the result of an incorrect implementation of the baseline methods. Often, also important pieces of information regarding the used hyperparameters, hyperparameter ranges or random seeds are missing in the provided documentation, see also [Henderson et al. 2018].

In our own studies, we also observed that almost all researchers implement their own code for the experimental evaluation. Before deep learning became popular, a number of libraries for the reproducible evaluation were frequently used, such as LIBREC or MYMEDIALITE [Guo et al. 2015, Gantner et al. 2011], which ensured that the evaluation procedures were inspected and quality-assured by more than one researcher. As a result of the constant re-implementation of evaluation code, technical mistakes and inconsistencies among evaluation methodologies are probably now more common.

As a result of their analysis of more then 400 papers, [Gundersen 2020] points out that AI, like other scientific disciplines such as psychology, is affected by a reproducibility *crisis*. When interpreting a crisis positively, i.e., as a turning point, there is hope that the situation improves. Actually, [Gundersen and Kjensmo 2018] found some patterns of improvement between 2013 and 2016 in terms of R1 reproducibility (but not in the other dimensions). This indicates that researchers increasingly share their code, and the findings of our own studies corroborate this development.

Methodological Issues

The second important dimension that may prevent us from obtaining true progress is related to methodological issues. We identified three main types of issues. The first set of problems is related to bad evaluation practices and other technical mistakes. The second set of problems is due to the choice of weak baselines for the empirical comparison with the proposed methods. The third set of problems can be attributed to the fact that the majority of published research works has no explicit research questions or hypotheses [Gundersen and Kjensmo 2018].

Information Leakage

In machine learning, information leakage happens whenever information from the testing set is used in the training process. This type of information is not expected to be available at prediction time, causing evaluation metrics to overestimate the predictive accuracy of the method. Information leakage can occur at different levels, some obvious and easy to avoid, others more devious and difficult to detect.

One of the most common information leakage we have encountered in our analysis concerns early stopping
[Ferrari Dacrema et al. 2019, Ferrari Dacrema et al. 2021]. Early stopping should be implemented by evaluating the stopping criteria on a validation set different from the test set. However, 50% of the works we have analyzed evaluate the early stopping criteria on the test set itself, thus overfitting the test set on the stopping criteria. Even more serious, some works report results in which, for each metric and cutoff, the best value is computed on the testing set and each value corresponds to a different epoch. In all the affected papers, this information leakage occurs only for the newly proposed methods but not for the baselines.

An equally problematic issue was recently reported in the analysis by
[Sun et al. 2020] who found that 37% of their examined works tuned the

hyperparameters of their new method on the test set. If such a procedure was admissible, it is trivial to come up with an algorithm for implicit feedback that uses a parameter for each user-item combination and then ''learns'' the best setting for this variable by looking at the test set.

These two examples of information leakage are easy to detect and control, and authors should be able to take care of them without too much effort. Other types of information leakage are more devious and concern the splitting of datasets into training, validation and test sets.

Probably the most often used experimental design in the entire literature on recommendation algorithms is to predict held-out ratings on one of the many publicly available datasets containing implicit or explicit ratings. For the evaluation, the rating dataset is typically randomly split into a 80% training and 20% test split. Such a random data split however leads to the effect that the training data contains future data to predict a past event[1], see also [Ji et al. 2020]. Consider, for instance, the popular MovieLens dataset. If the training data by chance contains a highly positive ratings by a user for ''Rocky II'' and ''Rocky III'', the model might quite easily learn from the data that the user will also rate ''Rocky I'' highly. As a result, the observed prediction performance in such an experiment might be higher than when applied in practice, where it is probably much more difficult to guess if a user will like ''Rocky I'' when it is released.

The above problems concerning information leakage are part of a wider set of problems relating to evaluation methodologies, problems which we have identified in many works and which are sometimes significant, sometimes of little relevance. In several cases, the selection of negative samples was problematic, leading to a situation in which a different number of negative samples was used for different users. In other cases, we found that researchers implemented accuracy measures in uncommon ways. In one of the major cases, we found that the train-test split provided by the researchers was not documented, not reproducible and not consistent with any of the best practices about dataset partitioning. In all of these cases, the reported gains over baselines vanish when replicating the experiments with more conventional approaches.

Weak baselines

One of the most common problems that leads to an illusion of progress in research on recommender systems is related to the use of weak baselines. In some works the baselines are inherently weak, in other works the baselines are potentially stronger than reported, but are not adequately tuned for the problem under consideration. [Lin 2019] and [Yang et al. 2019], while recognizing the progress made by transformer models for certain tasks of document retrieval, observed that it is not rare to find articles comparing neural ranking algorithms only against weak baselines, raising questions regarding the empirical rigor in the field.
[Rendle et al. 2019] found that many algorithms that were published between 2015 and 2019 actually did not outperform longer-existing baselines, when these were properly tuned.

The apparent lack of progress can be attributed to two factors: the choice of the

baselines and the lack of proper optimization of the baselines. Ultimately, these phenomena can lead to a *cascade effect*, where only the most recent models are considered as the state-of-the-art even though they actually do not outperform existing models.

Choice of baselines. Hundreds of recommendation algorithms may have been designed over the past decade, and determining what represents the state-of-the-art is inherently difficult, because there is actually no ''best'' algorithm in general, as will be discussed below. Another observation is that, in many works, only the latest complex machine learning algorithms are considered as relevant baselines. For instance, in [Ferrari Dacrema et al. 2021] it was observed that more than 80% of the papers on deep-learning recommender systems use other deep learning algorithms as the only baselines. In that context, a key question is whether a specific family of algorithms is the best available to solve the problem addressed in the paper, or whether there are other families of better algorithms to use as stronger baselines. Simpler and slightly older methods are often ignored as possible baselines, although some of them have been published in top-level venues and often lead to strong results. To some extent, this phenomenon might also be a result of our publication process, where reviewers might mainly ask for the consideration of the latest models as baselines, thereby implicitly suggesting that older approaches are outdated.

Lack of proper tuning of baselines. Even in cases when suitable baseline algorithms are selected in a comparison, a frequently occurring issue is that the baselines used in the experimental evaluation are not properly tuned. This is probably the most common issue observed in our and other studies, and is not specifically tied to recommender algorithms. Researchers apparently invest significant efforts in tuning the hyperparameters of their own new method but do not always pay the same attention to the tuning of baselines. One commonly found mistake is that of authors using hyperparameter settings reported as optimal in a previous article, although these settings refer to experimental conditions other than those currently considered.

Lack of Explicit Research Questions

[Gundersen and Kjensmo 2018] report that less than 10% of the AI papers analyzed in their study contained an explicit statement regarding the addressed research goals and objectives. Similar trends exist in the recommender systems literature, where the research goal is only stated implicitly. Usually, the implicit goal is to create better recommender systems by making better relevance predictions, i.e., to obtain higher accuracy.

A common claim in most algorithmic research works in recommender systems is that ''our method significantly outperforms the state-of-the-art.'' Usually, these improvements are attributed to a novel model that better captures certain patterns in the data and/or the creative consideration of existing or additional types of

information. These improvements are then evidenced by reporting empirical results in which the new model is benchmarked against a set of baselines algorithms. Unfortunately, the way the experiments are designed cannot inform us if a method improves the state-of-the-art as claimed. The observed performance and ranking of algorithms can depend on a multitude of factors, including the characteristics of the used datasets, the chosen performance measures, the cut-off length for the measures, data pre-processing, specifics of the evaluation protocol like the data splitting procedure, or even the metric sampling approach [Krichene and Rendle 2020, Rossetti et al. 2016].[2] For all these aspects, no standards exist, and recent papers highlight that researchers use all sorts of experimental configurations in their experiments, often without a theory-guided justification for their choices [Sun et al. 2020, Ferrari Dacrema et al. 2021]. Therefore, what can be shown in a paper is that some proposed model outperforms existing approaches in very specific experimental settings, many of which can be arbitrarily chosen by the researcher. One possible consequence of this freedom is that researchers may be subject to psychological biases, most importantly a confirmation bias. In such a situation, researchers may unconsciously seek for evidence that supports their hoped-for outcomes, i.e., that their new model works, and this psychological phenomenon might guide their search for corresponding experimental configurations.

Another problem related to the lack of explicit research questions is the failure to identify the sources of empirical gains, as reported by [Lipton and Steinhardt 2019]. In many works, together with a new sophisticated algorithm, the authors propose a number of other empirical and apparently minor contributions, such as optimization heuristics, clever data-preprocessing, or extensive hyperparameter tuning. Too often, just one of the minor contributions is actually responsible for the performance gain, but the lack of proper ablation studies give the false impression that all of the proposed changes are necessary.

Theoretical Flaws

Besides reproducibility and methodological problems, our field is sometimes also plagued by theoretical issues.

Using Unsuited Models for the Data

Research on novel recommendation algorithms sometime involves speculations that do not properly undergo scientific scrutiny. For instance, in recent years, researchers have explored the use of convolutional neural networks (CNNs) for recommendation problems. The application of CNNs relies on the assumption that there is some form of relationship between neighboring or nearby data elements. One specific idea that was explored in at least three recent papers---all presented at IJCAI conferences---is to combine latent factor models (embeddings) with the power of CNNs. The underlying assumption in these papers is that there are correlations in the latent factor models. To consider these correlations in the models, the authors propose to apply convolutions over user-item interaction maps

that are obtained from the outer product of their embeddings. According to the results presented in the respective papers, CNNs have helped to significantly improve recommendation accuracy.

When looking closer at how the user-item interaction maps are usually created, one can however observe that the order of the elements in the embeddings carries no semantic meaning. Therefore, any performance gain observed through the application of CNNs cannot be attributed to correlations between nearby elements, but only to the fact that neural networks act as universal approximators. As shown by [Ferrari Dacrema et al. 2020b], perturbing the order of the elements in the embeddings, as expected, has no effect on recommendation accuracy. These findings do not rule out that CNNs can be useful for various recommendation tasks. In the mentioned cases, however, the claims made in research papers are not theoretically valid. Overall, this points to a more general issue in today's research practice in applied machine learning, where the goal of improving prediction accuracy is very often not accompanied by the questions ''What worked?'' and ''Why?'', see also [Lipton and Steinhardt 2019]. In general, while it might often be difficult to theoretically *prove* how exactly a particular model contributes to the overall performance, suitable ablation studies may usually help to discover potential issues empirically.

Unsuited Experiment Designs for the Research Goals

The experimental analyses reported in many papers deviate significantly from the claimed or implicit research goals. The probably most typical example are research papers that propose new algorithms for implicit-feedback recommendation scenarios. The goal of these research works is to demonstrate that a new model is better than previous ones at predicting whether a user will like an item or not. When it comes to the evaluation, it is however not uncommon that researchers take an explicit-feedback dataset from MovieLens and transform *all* ratings, including the negative ones, to positive signals. As a result, what is measured here is how good an algorithm is at predicting who will rate which item next. This, however, is almost never the research goal. Likewise, many sequential recommendation approaches are also based on MovieLens data. Here, the sequence of events is determined by the timestamp of the rating, which can be days, months, or years after a movie was watched. Again, in these approaches, researchers predict who will rate what next.

One could argue that such subtleties are not important because what matters is the ranking of the algorithms, which would probably be the same when only the very positive ratings, e.g., those with 4 or 5 stars, would be considered as positive signals. While the ranking might indeed be the same in some cases, we believe that it is unimaginable in other scientific disciplines to make claims regarding the power of a prediction model for a given research question when the underlying experiment is actually not suited to answer the question. The fact that this seems to be acceptable in our domain somehow indicates problems regarding rigor and our scientific method.

Unreliable Conclusions from Sampled Metrics

Offline evaluation of modern machine learning algorithms can be computationally demanding when there are many items in the databases. This is in particular the case when ranking or classification measures are used. To determine such measures, one has to rank all items in the catalog for each user, which usually means that relevance scores for potentially tens of thousands of items have to be computed per user. To avoid this problem, researchers commonly apply an evaluation procedure where the *positive* items of a user in the test set---these are typically not many---are ranked together with N randomly sampled *negative* items, i.e., items where no preference (or a negative one) is available for the user. [Koren 2008] is an early example where this approach was taken, using $N = 1000$.

This evaluation approach is very common today. However, often much smaller values for N are chosen, e.g., 50 or 100. Only recently, the question was raised if the results obtained through such *sampled metrics* actually correspond to the algorithm ranking that one would obtain when all items were considered, i.e., when the exact metric is used. In their analysis, [Krichene and Rendle 2020] showed that the sampled metrics are *inconsistent* with their exact version, which means that statements like ''*Recommender A is better than Recommender B*'' cannot be reliably derived when using such metrics. This finding therefore may have major implications on what we believed to know about the relative ranking of algorithms, which is the main focus of almost all published algorithms research.

How Could This Happen?

To recap, other researchers and ourselves have found that in many cases the latest and more complex machine learning models presented at prestigious conferences are often not actually better than previously existing methods. Similar observations were also made for non-neural recommendation methods by [Rendle et al. 2019], for information retrieval research by [Yang et al. 2019] and by [Armstrong et al. 2009], and for other areas like time-series forecasting or certain clinical prediction tasks by [Makridakis et al. 2018] and [Bellamy et al. 2020]. Many factors might contribute to these phenomena, which are seemingly not tied to applied machine learning research in recommender systems or to deep learning. Here, we provide some thoughts and speculations regarding what might have put us in this unsatisfactory situation.

One quite obvious reason for limited reproducibility is that providing reproducible artifacts is work-intensive and there is little reward. The code that is written during the process might have been developed while the researcher is still learning about the technology, it may contain code and method variants that are no longer used as they did not lead to success, or contain programming hacks that were introduced under time pressure before a deadline. Preparing and re-working all artifacts into a sharable form, including the creation of instructions for installation and execution, might therefore seem to be a time investment that does not pay off for the researcher. Furthermore, when the provision of source code is

strongly encouraged for paper submissions at a conference or even a reviewing criterion, researchers might only prepare a minimal set of artifacts for sharing, i.e., the core code of their method but not the other material that would be needed to reproduce the experiments. Not being able or willing to invest more time in reproducibility or in systematically optimizing the baselines may also have to do with our academic incentive system and a corresponding publication pressure. An additional reason for not sharing the code might lie in a certain fear by researchers that their code contains mistakes or methodological issues. If such a mistake is detected, this may harm the reputation of the researcher, in particular if a found mistake invalidates the claimed findings. Finally, experimental activities, in particular programming, might often be carried out by young researchers who may not be fully aware of the possible impact on the results when they do not meticulously follow best evaluation practices. For instance, they might not be aware of subtleties that can lead to information leakage.

Considering the above-mentioned methodological issues and bad research practices, it seems that there are too few incentives (or too little pressure) in our community, e.g., from reviewers, program chairs, and journal editors, to improve these aspects. The general research methodology seems agreed-upon, and it appears that much more focus is put on the details of a fancy new technical contribution than on double-checking that the research methodology is appropriate to support a claim. For example, not reporting if and how the baseline methods were optimized is common; not reporting results of statistical significance tests is not an issue; not justifying why certain datasets, measure or cutoffs are used for the evaluation is acceptable. Finally, formulating a specific research question or laying out hypothesis, which are key pillars in scientific research elsewhere, is less than common.

In some ways, our research approach is sometimes seemingly boiling down to a leaderboard-chasing culture, where the goal is to obtain an accuracy improvement on some dataset, no matter where the improvements come from or whether they would actually matter in practice. In particular this latter aspect was discussed years ago by [Wagstaff 2012] who advocated that we should more often focus on ''machine learning that matters''. Symptoms of machine learning research that has limited impact in practice include a hyper-focus on benchmark datasets and machine learning competitions or a hyper-focus on abstract accuracy measures, and a limited connection to real-world problems. A recent analysis of the (often limited) correspondence of the offline and online performance of recommendation algorithms can be found in [Krauth et al. 2020], see also [Rossetti et al. 2016] and [Cremonesi et al. 2013].

Furthermore, the academic incentive system and publication culture focuses on a ''machine learning contribution'', but does not reward researchers enough that they perform all other activities that are needed to develop something that has impact in the real world. Similar observations were made more recently by [Kerner 2020], who found that AI research too often considers that real-world applications are not relevant. While this is probably a general issue in AI---see [D'Amour et al. 2020] for a discussion of the often poor performance of machine learning models when deployed in practice---it is particularly troubling in the context of recommender

systems research, which by definition is oriented towards a particular and very successful application of machine learning in practice.

Finally, some of the observed problems might also stem from the current situation, where conferences receive thousands of submissions, and where it might be very challenging to recruit a sufficient number of senior reviewers that would be required to critically evaluate the research methodology. In our own experience, early-stage researchers who act as reviewers more often focus strongly on technical innovation and novel models, and put less thought on evaluation aspects, as the standards for evaluation seem agreed-upon and of secondary relevance.

Recommended Action

What follows are a number of suggestions based on our personal experiences on what we, as a community, could do to counter these trends, besides simply suggesting that authors should refrain from today's bad practices.

Best Practices for Reproducibility

Regarding reproducibility in AI in general, [Gundersen et al. 2018] provide a set of best practices, recommendations, and a detailed checklist for authors. In the following, we summarize the main points and add specific aspects that are relevant in the context of recommender systems research.

According to the checklist of [Gundersen et al. 2018], the data used in experiments should be publicly available, be documented, maybe include license information, and should have a DOI or persistent URL attached. Since in recommender systems research the raw data is very often preprocessed, e.g., to reduce the sparsity, authors should also publicly share the preprocessed data. Furthermore, sharing the used data splits and the negative samples used in the experiments can be very helpful to ensure reproducibility even when the splitting and sampling procedures are described in the paper.

The requirements of public accessibility, documentation, licences and permanent reference hold for the method code as well. Moreover, the code (and the data) should be provided in a way that re-running the experiments reported in the paper is as easy as possible. This requires that *all* code is shared: the code of the new method, the baselines code, the code used for data preprocessing, the one used for hyperparameter optimization, and the code used for evaluation. Moreover, the code should be accompanied with relevant instructions regarding software or hardware requirements, installation scripts, and prepared scripts and configuration files to run the experiments. In case a complex software environment is required to run the experiments, the provision of a Docker image or similar formats is advisable. In terms of additional documentation that might not fit into a research paper, authors should provide their optimal hyperparameter settings and the ranges they have explored on a validation set.

Note that (standardized) *benchmarks* are often considered a possible cure to many of the mentioned problems. In the best case, the existence of such benchmarks

could relieve researchers from the burden of running all previous baselines, and would just have to report their new results. Making a dataset publicly available is, however, not enough to achieve this goal. It requires that a specific train-test split is provided and that the *exact* evaluation procedure, including the metrics, is known. In the best case, the code to be used for the evaluation should be standardized as well, because small implementation details can make a difference. Besides these challenges, the use of benchmark problems can in general be a double-edged sword, where researchers focus too much on a very specific known problem setting and hyperparameter tuning, which might prevent them to explore other and probably more important research questions.

(1) Data:	
Share well documented data, include	
1.1	Raw dataset, with meta-data, license information and permanent link
1.2	Processed dataset, including train, validation and test sets, as used to obtain the reported results
1.3	Other data samples used in the evaluation, e.g., negative samples
(2) Code:	
Share documented code, include	
2.1	Code for the proposed method
2.2	Code for the baselines
2.3	Code used for data pre-processing
2.4	Code for hyperparameter optimization
2.5	Evaluation code
(3) Execution Instructions:	
Share additional instructions for re-running the experiment	
3.1	Hardware and software requirements, including all external libraries necessary to run the codes
3.2	Installation instructions
3.3	Scripts for installation and experiment execution
(4) Experiment Documentation:	
Share additional details of experiment	
4.1	Final hyperparameter settings and explored ranges (new method and baselines)
4.2	Detailed experimental results
4.3	Any additional required artifacts, e.g., configuration files

Table 1: Reproducibility Checklist

On a positive note, we can observe increased awareness in recent years in the community regarding the potential helpfulness of best practices, research guidelines and reproducibility checklists, see e.g. [Pineau et al. 2020]. Besides more

general guidelines, as summarized in Table 1, also more specific best practices for certain aspects of machine learning research were recently proposed, e.g., in [Lindauer and Hutter 2019].

Recommendations for Chairs, Editors and Reviewers

With this work, our goal is to make conference chairs, journal editors, and reviewers better aware of the severity of today's issues with respect to reproducibility and methodology. As a first step forward, we believe that more emphasis on these topics should be placed both in the call for contributions and in the reviewing process. In particular reproducibility should be a key criterion in every evaluation of algorithmic contributions. Conference chairs and journal editors should therefore provide clear guidelines to reviewers on what level of reproducibility is expected and on how much weight should be given to this aspect when evaluating a paper.

Certainly, there might be reasons why only parts of the code and the data can be shared, but these reasons should be well explained. In particular researchers from industry often face the problem that they are not allowed to share the materials that they used in their experiments. The involvement of industry in recommender systems research, with no doubt, is highly important and absolutely necessary to move our field forward. It might therefore be advisable to create dedicated opportunities for contributions from industry, where it is clear that these works do not necessarily meet the needed reproducibility standards.

Regarding dedicated publication outlets, we furthermore encourage the creation of opportunities to publish reproducibility studies, e.g., in the form of a special conference track. Correspondingly, the papers submitted to such tracks should be evaluated differently as regular submissions, where, for example, novelty aspects are not in the focus.

For reviewers, once they are aware of the existing issues, we expect that they more often have a closer look at evaluation aspects when assessing a paper. We speculate that reviewers who work in recommendation algorithms themselves are more interested in the novelty and the particularities of the proposed technical approach than in the finer details of the experimental setup. With clear guidelines provided by the chairs and editors, they can be pointed to additional aspects that should be asked, e.g., if the work is reproducible or if the chosen experimental design is suited to answer the research question that is asked in a paper.

Finally, as our machine learning models increasingly become more complex, some reviewers might be misled into thinking that the complexity of a model is always positively correlated with its effectiveness. This has probably led to a certain level of ''mathiness'', defined as a ''*tangling of formal and informal claims*'', which is observed by [Lipton and Steinhardt 2019] for machine learning research in general. Instead, the principle of Occam's Razor should be kept in mind. Translated to our problem, this means that one should prefer simple over complex models, provided that they are performing equally well. Today, we unfortunately still too often observe a self-enforcing pattern, where researchers may intrinsically

have a certain fondness for complex models and reviewers at the same time ask for or appreciate these more fancy models.

Recommendations for Researchers and the Community

We believe that today's issues regarding the lack of reproducibility and progress observed in our previous studies are mainly related to a lack of awareness and attention on these topics in our community. In the future, when the community appropriately rewards and incentivizes the additional efforts by researchers, we hope that it will be ''natural'' for researchers trying to ensure that their work is reproducible and thus verifiable. Providing all artifacts for reproducibility should in the long run also serve today's scientific career mechanics, as reproducible work might be more often used as a basis for innovation and comparisons by other researchers.

However, to achieve true progress in our field, ensuring reproducibility of new algorithms is not enough. Accuracy optimization for top-N recommendation tasks has been done for about twenty-five years now, with a large fraction of research works using movie rating datasets for the evaluations. The question has been raised before, if there is a ''magic barrier'' where we cannot improve our algorithms anymore, e.g., because of the natural noise in the data [Said et al. 2012, Amatriain et al. 2009]. Moreover, there are various works that indicate that these usually slight improvements in accuracy obtained in offline experiments might not matter in practice, and a number of user studies suggest that higher accuracy does not correlate positively with the users' quality perceptions [Cremonesi et al. 2012, Krauth et al. 2020]. In recent years, we might have expanded our research operationalization from matrix completion to alternative scenarios like session-based or sequential recommendation, and we have explored questions of recommendation diversity and novelty. The problems of offline evaluations however remain the same.

To achieve true progress, we believe that researchers in recommender systems should more often focus on problems ''that matter'', in the sense of [Wagstaff 2012]. In our opinion, this means that, in principle, we should more often first work with specific problems in an application domain---ultimately recommender systems research is an applied discipline---and only then try to generalize. We can no longer try to solve the problem of finding ''the best algorithm'', because it does not exist. Whether an algorithm is preferred over another one basically cannot be judged in isolation, because we have to take into account what the purpose of the recommendations is in the first place [Jannach and Adomavicius 2016]. Today, we mainly measure what is easy to measure, i.e., accuracy in offline experiments, but this seems to be too limited to make most of our research impactful in practice. We therefore highly encourage researchers to leave established paths and focus on more interesting and relevant problems, see, e.g., [Jannach and Bauer 2020] for potential ways forward.

Conclusion

A number of recent studies indicate that research in recommendation algorithms has reached a certain level of stagnation. In this paper, we have reviewed the possible reasons that lead to the effect that many new algorithms are published, even at top-level outlets, for which it is ultimately not clear if they really improve the state-of-the-art. Among the most important causes, we have identified limited reproducibility, weak baselines, improper evaluation methodologies and lack of explicit research questions. Furthermore, we outlined a number of research best practices to avoid the ''phantom progress'' due to low levels of reproducibility and the use of improper methodologies. Ultimately, however, we do not only need better research practices, but we have to re-think how we do algorithms research in recommender systems, where we have to shift from a hyper-focus on accuracy in offline experiments or on already well-explored problems, to questions that really matter and have an impact in practice.

Notes

[1][Jannach et al. 2016] used the term ''postdict''.

[2]Recently, fundamental questions were also raised by [Ferrante et al. 2021] and others regarding the validity of experimental results that are obtained with common information retrieval measures like Precision and Recall, which are also commonly used in recommender systems evaluations.

References

[Amatriain et al. 2009] X. Amatriain, J. M. Pujol, N. Tintarev, and N. Oliver. Rate it again: Increasing recommendation accuracy by user re-rating. In *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys '09, page 173–180, 2009.

[Armstrong et al. 2009] T. G. Armstrong, A. Moffat, W. Webber, and J. Zobel. Improvements that don't add up: Ad-hoc retrieval results since 1998. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM '09)*, pages 601--610, 2009.

[Beel et al. 2016] J. Beel, C. Breitinger, S. Langer, A. Lommatzsch, and B. Gipp. Towards reproducibility in recommender-systems research. *User Modeling and User-Adapted Interaction*, 26(1):69–101, 2016.

[Bellamy et al. 2020] D. Bellamy, L. Celi, and A. L. Beam. Evaluating progress on machine learning for longitudinal electronic healthcare data. *CoRR*, abs/2010.01149, 2020. URL https://arxiv.org/abs/2010.01149.

[Cremonesi et al. 2010] P. Cremonesi, Y. Koren, and R. 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] P. Cremonesi, F. Garzotto, and R. Turrin. Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study. *Transactions on Interactive Intelligent Systems*, 2(2):1--41, 2012.

[Cremonesi et al. 2013] P. Cremonesi, F. Garzotto, and R. Turrin. User-centric vs. system-centric evaluation of recommender systems. In *IFIP Conference on Human-Computer Interaction*, pages 334--351. Springer, 2013.

[D'Amour et al. 2020] D'Amour et al. Underspecification presents challenges for credibility in modern machine learning. *CoRR*, abs/2011.03395, 2020. URL https://arxiv.org/abs/2011.03395.

[Ekstrand et al. 2011] M. D. Ekstrand, M. Ludwig, J. A. Konstan, and J. T. Riedl. Rethinking the recommender research ecosystem: Reproducibility, openness, and lenskit. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, RecSys '11, page 133–140, 2011.

[Ferrante et al. 2021] M. Ferrante, N. Ferro, and N. Fuhr. Towards Meaningful Statements in IR Evaluation. Mapping Evaluation Measures to Interval Scales. *CoRR*, abs/2101.02668, 2021. URL https://arxiv.org/abs/2101.02668.

[Ferrari Dacrema et al. 2019] M. Ferrari Dacrema, P. Cremonesi, and D. Jannach. Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys '19)*, pages 101--109, 2019.

[Ferrari Dacrema et al. 2020a] M. Ferrari Dacrema, P. Cremonesi, and D. Jannach. Methodological issues in recommender systems research (extended abstract). In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4706--4710, 2020.

[Ferrari Dacrema et al. 2020b] M. Ferrari Dacrema, F. Parroni, P. Cremonesi, and D. Jannach. Critically examining the claimed value of convolutions over user-item embedding maps for recommender systems. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM 2020)*, 2020.

[Ferrari Dacrema et al. 2021] M. Ferrari Dacrema, S. Boglio, P. Cremonesi, and D. Jannach. A troubling analysis of reproducibility and progress in recommender systems research. *ACM Transactions on Information Systems (TOIS)*, 39(2), 2021.

[Gantner et al. 2011] Z. Gantner, S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme. MyMediaLite: A free recommender system library. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, pages 305--308, 2011.

[Gundersen 2020] O. E. Gundersen. The reproducibility crisis is real. *AI Magazine*, 41(3):103--106, Sep. 2020.

[Gundersen and Kjensmo 2018] O. E. Gundersen and S. Kjensmo. State of the art: Reproducibility in artificial intelligence. In *AAAI*, 2018.

[Gundersen et al. 2018] O. E. Gundersen, Y. Gil, and D. W. Aha. On Reproducible AI: Towards Reproducible Research, Open Science, and Digital Scholarship in AI Publications. *AI Magazine*, 39(3):56--68, Sep. 2018.

[Guo et al. 2015] G. Guo, J. Zhang, Z. Sun, and N. Yorke-Smith. Librec: A java library for recommender systems. In *Posters, Demos, Late-breaking Results and Workshop Proceedings of the 23rd Conference on User Modeling, Adaptation, and Personalization (UMAP 2015)*, 2015.

[He et al. 2017] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*, pages 173--182, 2017.

[Henderson et al. 2018] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger. Deep reinforcement learning that matters. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3207--3214, 2018.

[Jannach and Adomavicius 2016] D. Jannach and G. Adomavicius. Recommendations with a purpose. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys 2016)*, pages 7--10, Boston, Massachusetts, USA, 2016.

[Jannach and Bauer 2020] D. Jannach and C. Bauer. Escaping the McNamara Fallacy: Towards more Impactful Recommender Systems Research. *AI Magazine*, 40(4), 2020.

[Jannach et al. 2016] D. Jannach, P. Resnick, A. Tuzhilin, and M. Zanker. Recommender systems - beyond matrix completion. *Communications of the ACM*, 59(11):94--102, 2016.

[Ji et al. 2020] Y. Ji, A. Sun, J. Zhang, and C. Li. On offline evaluation of recommender systems, 2020.

[Kerner 2020] H. Kerner. Too many AI researchers think real-world problems are not relevant. MIT Technology Review, August 2020, 2020. URL https://www.technologyreview.com/2020/08/18/1007196/ai-research-machine-learning-applications-problems-opinion/.

[Konstan and Adomavicius 2013] J. A. Konstan and G. Adomavicius. Toward identification and adoption of best practices in algorithmic recommender systems research. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*, page 23–28, 2013.

[Koren 2008] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '08)*, pages 426--434, 2008.

[Kouki et al. 2020] P. Kouki, I. Fountalis, N. Vasiloglou, X. Cui, E. Liberty, and K. Al Jadda. From the lab to production: A case study of session-based recommendations in the home-improvement domain. In *Fourteenth ACM Conference on Recommender Systems (RecSys '20)*, RecSys '20, page 140–149, 2020.

[Krauth et al. 2020] K. Krauth, S. Dean, A. Zhao, W. Guo, M. Curmei, B. Recht, and M. I. Jordan. Do offline metrics predict online performance in recommender systems? *CoRR*, abs/2011.07931, 2020. URL https://arxiv.org/abs/2011.07931.

[Krichene and Rendle 2020] W. Krichene and S. Rendle. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 1748–1757, 2020.

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

[Lindauer and Hutter 2019] M. Lindauer and F. Hutter. Best practices for scientific research on neural architecture search. *CoRR*, abs/1909.02453, 2019. URL http://arxiv.org/abs/1909.02453.

[Lipton and Steinhardt 2019] Z. C. Lipton and J. Steinhardt. Research for practice: troubling trends in machine-learning scholarship. *Communications of the ACM*, 62(6):45--53, 2019.

[Ludewig et al. 2020] M. Ludewig, S. Latifi, N. Mauro, and D. Jannach. Empirical analysis of session-based recommendation algorithms. *User Modeling and User-Adapted Interaction*, 31(3), pages 49–181, 2020.

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

[Ning and Karypis 2011] X. Ning and G. Karypis. SLIM: Sparse linear methods for top-n recommender systems. In *Proceedings of the 11th IEEE International Conference on Data Mining (ICDM '11)*, pages 497--506, 2011.

[Pineau et al. 2020] J. Pineau, P. Vincent-Lamarre, K. Sinha, V. Larivière, A. Beygelzimer, F. d'Alché Buc, E. Fox, and H. Larochelle. Improving Reproducibility in Machine Learning Research (A Report from the NeurIPS 2019 Reproducibility Program). *CoRR*, abs/2003.12206, 2020. URL https://arxiv.org/abs/2003.122069.

[Rendle et al. 2019] S. Rendle, L. Zhang, and Y. Koren. On the difficulty of evaluating baselines: A study on recommender systems. *CoRR*, abs/1905.01395, 2019. URL http://arxiv.org/abs/1905.01395.

[Rendle et al. 2020] S. Rendle, W. Krichene, L. Zhang, and J. Anderson. Neural collaborative filtering vs. matrix factorization revisited. In *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys '20)*, 2020.

[Resnick et al 1994] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer-Supported Cooperative Work (CSCW '94)*, pages 175--186, 1994.

[Rossetti et al. 2016] M. Rossetti, F. Stella, and M. Zanker. Contrasting offline and online results when evaluating recommendation algorithms. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys '16)*, pages 31--34, 2016.

[Said et al. 2012] A. Said, B. J. Jain, S. Narr, and T. Plumbaum. Users and noise: The magic barrier of recommender systems. In *User Modeling, Adaptation, and Personalization*, pages 237--248. Springer Berlin Heidelberg, 2012.

[Sun et al. 2020] Z. Sun, D. Yu, H. Fang, J. Yang, X. Qu, J. Zhang, and C. Geng. Are We Evaluating Rigorously? Benchmarking Recommendation for Reproducible Evaluation and Fair Comparison. In *Fourteenth ACM Conference on Recommender Systems*, RecSys '20, page 23–32, 2020.

[Wagstaff 2012] K. Wagstaff. Machine learning that matters. In *Proceedings of the 29th International Conference on Machine Learning (ICML '12)*, pages 529--536, 2012.

[Wu et al. 2019] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan. Session-based recommendation with graph neural networks. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI*, pages 346--353, 2019.

[Yang et al. 2019] W. Yang, K. Lu, P. Yang, and J. Lin. Critically Examining the ''Neural Hype'': Weak Baselines and the Additivity of Effectiveness Gains from Neural Ranking Models. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1129--1132, 2019.

Paolo Cremonesi (paolo.cremonesi@polimi.it) is professor of Computer Science Engineering at Politecnico di Milano, Italy. His current research interests focus on recommender systems and quantum computing. He is the coordinator of the new-born Quantum Computing lab at Politecnico di Milano.

Dietmar Jannach (dietmar.jannach@aau.at) is a professor of computer science at the University of Klagenfurt, Austria. His research is focused on practical

applications of artificial intelligence, with a focus on recommender systems. He is also the leading author of the first textbook on recommender systems published with Cambridge University Press.