Towards Better Evaluation of Recommendation Algorithms with Bi-directional Item Response Theory

Ziqi Xu*
RMIT University
Melbourne, Victoria, Australia
ziqi.xu@rmit.edu.au

Chenglong Ma
RMIT University
Melbourne, Victoria, Australia
chenglong.ma@rmit.edu.au

Yongli Ren
RMIT University
Melbourne, Victoria, Australia
yongli.ren@rmit.edu.au

Jeffrey Chan
RMIT University
Melbourne, Victoria, Australia
jeffrey.chan@rmit.edu.au

Wei Shao
Data61, CSIRO
Melbourne, Victoria, Australia
wei.shao@data61.csiro.au

Feng Xia
RMIT University
Melbourne, Victoria, Australia
feng.xia@rmit.edu.au

Abstract

Recommendation algorithms are typically evaluated on various datasets and compared against other algorithms employing diverse strategies. However, current evaluation practices predominantly rely on rank-based metrics, focusing solely on performance outcomes while overlooking the latent traits of datasets and recommendation algorithms. In this paper, we propose a bi-directional Item Response Theory (Bi-ReIRT[1]) framework, which offers a fine-grained evaluation by simultaneously modelling the latent traits of recommendation algorithms (i.e., their ability) and datasets (i.e., their inherent challenges). This is the first work to apply the IRT framework for evaluating recommendation algorithms on the dataset level. The Bi-ReIRT framework enables visualisations of algorithms' performance across datasets with varying levels of inherent challenge. We conduct extensive experiments across a portfolio of recommendation algorithms and datasets, exploring the implications of key IRT parameters such as discrimination, difficulty, and ability. Moreover, the interpretability of these parameters provides deeper insights into the characteristics of both recommendation algorithms and datasets.

CCS Concepts

• **Information systems → Recommender systems**.

Keywords

Recommender systems, Item Response Theory.

ACM Reference Format:
Ziqi Xu, Chenglong Ma, Yongli Ren, Jeffrey Chan, Wei Shao, and Feng Xia. 2018. Towards Better Evaluation of Recommendation Algorithms with Bi-directional Item Response Theory. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

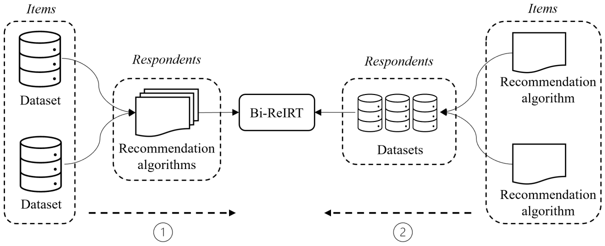

Figure 1: The architecture of Bi-ReIRT comprises two directions. The first direction, depicted on the left side, treats each dataset as an *item* and each recommendation algorithm as a *respondent*. The second direction, shown on the right side, reverses this arrangement by treating each recommendation algorithm as an *item* and each dataset as a *respondent*. Please refer to Section 2 for an introduction to the concepts of *item* and *respondent* within Bi-ReIRT.

1 Introduction

Experimental evaluation is vital for recommender systems, especially for tasks where theoretical evaluation is impractical. Current evaluation practices typically rely on diverse datasets and rank-based metrics (e.g., nDCG@k) [2]. However, such comparisons only reveal that algorithm A outperforms B, without providing insights into the underlying reasons for success or failure. To develop and enhance recommendation algorithms, it is essential to understand where and why algorithms fall short and identify datasets that present significant challenges to state-of-the-art algorithms.

Item Response Theory (IRT) was originally developed in the field of psychometrics to model the interaction between respondents and test items, providing insights into the respondents' abilities and the characteristics of test items [3]. In recent years, IRT has been proposed as a tool to evaluate performance in machine learning (ML) models [4, 6]. By treating datasets as items and ML models as respondents, IRT allows us to reinterpret the ability of an ML model in terms of the difficulty and discrimination levels of the datasets.

*Corresponding Author
[1] The source code can be found at https://github.com/IRON13/Bi-ReIRT.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

Furthermore, Liu et al. [5] extend this concept to recommender systems by proposing an evaluation framework based on IRT. In their framework, recommendation algorithms are treated as respondents to evaluate their latent abilities, while the (user, positive item, negative item) triples are used as responses to estimate item characteristics, such as difficulty and discrimination, which reflect user preferences. However, this framework operates at the instance level, relying on binary responses and the 3PL (3-parameter) IRT model. This approach inherently limits its ability to leverage rank-based metrics across a variety of datasets, as these metrics yield continuous outcomes rather than binary results.

To address the limitations of binary responses, we propose Bi-ReIRT, a framework based on beta-IRT that offers greater flexibility in modelling continuous responses across diverse items. Additionally, our framework introduces a bi-directional evaluation process on the dataset level, enabling the simultaneous learning of the latent traits of recommendation algorithms (i.e., their ability) and datasets (i.e., their inherent challenges). In summary, this paper makes the following contributions:

- To the best of our knowledge, this is the first work to apply the IRT framework for evaluating recommendation algorithms on the dataset level. This framework addresses the limitations of instance level evaluations by providing a bi-directional process to simultaneously learn the latent traits of recommendation algorithms and datasets.
- We introduce visualisations of recommendation algorithms' performance across datasets with varying levels of inherent challenge. These visualisations offer deeper insights into the interaction between recommendation algorithms and datasets, uncovering patterns and revealing underlying relationships.
- We evaluate the effectiveness of the Bi-ReIRT framework using a portfolio of datasets and a variety of recommendation algorithms. Our experimental results demonstrate that Bi-ReIRT provides comprehensive and interpretable evaluations of utility.

2 Methodology

In this section, we introduce the proposed Bi-ReIRT framework. We begin by presenting the backbone beta IRT model. Next, we briefly interpret the learned parameters from the two directions. Finally, we outline the workflow of the proposed framework.

In Item Response Theory (IRT), an item is a test question or measurement unit with characteristics like difficulty (i.e., $\boldsymbol{\delta}$) and discrimination (i.e., $\boldsymbol{a}$). A response is a respondent's reaction to an item. IRT models the relationship between a respondent's ability (i.e., $\boldsymbol{\theta}$) and the corresponding response, offering insights into both item quality and respondent performance.

We adopt the beta IRT, which has been demonstrated to capture a broader range of item characteristic curve (ICC) shapes compared to the logistic IRT [1]. In beta IRT, p_{ij} is the observed response of i-th respondent to j-th item, which is drawn from the Beta distribution,

$$\begin{aligned} p_{ij} &\sim \mathrm{Beta}(\alpha_{ij}, \beta_{ij}), \quad \alpha_{ij} = f_\alpha(\boldsymbol{\theta}_i, \boldsymbol{\delta}_j, \boldsymbol{a}_j) = \left(\frac{\boldsymbol{\theta}_i}{\boldsymbol{\delta}_j}\right)^{\boldsymbol{a}_j}, \\ \beta_{ij} &= f_\beta(\boldsymbol{\theta}_i, \boldsymbol{\delta}_j, \boldsymbol{a}_j) = \left(\frac{1-\boldsymbol{\theta}_i}{1-\boldsymbol{\delta}_j}\right)^{\boldsymbol{a}_j}, \end{aligned} \tag{1}$$

where the parameters α_{ij} and β_{ij} are computed by $\boldsymbol{\theta}_i$, $\boldsymbol{\delta}_j$, and $\boldsymbol{a}_j$.

Table 1: The parameter setting in Bi-ReIRT and original IRT.

	Original IRT	Bi-ReIRT - Direction ①	Bi-ReIRT - Direction ②
Setting	N respondents answering M test items	N recommendation algorithms evaluated on M datasets	M datasets evaluated by N recommendation algorithms
Item Parameters	$\boldsymbol{\delta}_j$: Test item difficulty	$\boldsymbol{\delta}_j$: Dataset difficulty	$\boldsymbol{\delta}_i$: Recommendation algorithm difficulty limit
	$\boldsymbol{a}_j$: Test item discrimination	$\boldsymbol{a}_j$: Dataset discrimination	$\boldsymbol{a}_i$: Recommendation algorithm consistency
Respondent Parameters	$\boldsymbol{\theta}_i$: Respondent ability	$\boldsymbol{\theta}_i$: Recommendation algorithm ability	$\boldsymbol{\theta}_j$: The inherent challenge of the dataset

The beta distribution allows us to generate non-logistic ICCs. The ICC is defined as follows,

$$\mathbb{E}[p_{ij}|\boldsymbol{\theta}_i, \boldsymbol{\delta}_j, \boldsymbol{a}_j] = \frac{\alpha_{ij}}{\alpha_{ij} + \beta_{ij}} = \frac{1}{1 + \left(\frac{\boldsymbol{\delta}_j}{1-\boldsymbol{\delta}_j}\right)^{\boldsymbol{a}_j} \left(\frac{\boldsymbol{\theta}_i}{1-\boldsymbol{\theta}_i}\right)^{-\boldsymbol{a}_j}}. \tag{2}$$

IRT has been applied to performance evaluation in the ML domain, with responses adapted to suit different types of tasks. For instance, in multi-class classification tasks, responses are represented by the probabilities that classifiers assign to the correct class for each instance. In our framework, responses are derived from the recommendation algorithm's rank-based metric results (e.g., nDCG@k) on each dataset. Notably, our framework does not impose restrictions on the choice of rank-based metrics. In section 3.3, we demonstrate the generalisability of our framework with different rank-based metrics.

The architecture of Bi-ReIRT is shown in Figure 1. Given N recommendation algorithms $R = (R_1, R_2, \ldots, R_N)$ and M datasets $D = (D_1, D_2, \ldots, D_M)$, our goal is to evaluate recommendation algorithms on the dataset level in two directions. Here, i represents the index of the recommendation algorithm ($i = 1, \ldots, N$), with each R_i corresponding to a specific algorithm in the set R. Similarly, j represents the index of the dataset ($j = 1, \ldots, M$), with each D_j corresponding to a specific dataset in the set D. We map the response p_{ij} to a rank-based metric (e.g., $nDCG@10_{ij}$), which captures the performance of the i-th recommendation algorithm on the j-th dataset. Through bi-directional analysis, we define the two ICC functions as follows:

$$nDCG@10_{ij} = \underbrace{\frac{1}{1 + \left(\frac{\boldsymbol{\delta}_j}{1-\boldsymbol{\delta}_j}\right)^{\boldsymbol{a}_j} \left(\frac{\boldsymbol{\theta}_i}{1-\boldsymbol{\theta}_i}\right)^{-\boldsymbol{a}_j}}}_{(3)} = \underbrace{\frac{1}{1 + \left(\frac{\boldsymbol{\delta}_i}{1-\boldsymbol{\delta}_i}\right)^{\boldsymbol{a}_i} \left(\frac{\boldsymbol{\theta}_j}{1-\boldsymbol{\theta}_j}\right)^{-\boldsymbol{a}_i}}}_{(4)},$$

where $\boldsymbol{\theta}$ and $\boldsymbol{\delta}$ are drawn from Beta distributions, with their priors set to $\mathrm{Beta}(1,1)$ in a general setting. $\boldsymbol{a}$ is drawn from a normal distribution with a prior mean of 1 and variance σ^2, where σ^2 is a hyperparameter. The default prior mean of $\boldsymbol{a}$ is set to 1 instead of 0 because $\boldsymbol{a}$ functions as a power factor in this context.

Equation (3) represents Direction ① (left side of Figure 1), where each dataset is treated as an item and each recommendation algorithm as a respondent. Conversely, Equation (4) represents Direction ② (right side of Figure 1), where each recommendation algorithm is treated as an item and each dataset is treated as a respondent. We observe that while the differences in the subscript are minor, they imply significant variations in the underlying meaning. The interpretation of each parameter in the two directions and compared with their counterparts in the original IRT setting are shown in Table 1.

The accuracy of the Bi-ReIRT framework's analysis depends on the quality of the data [3]. To ensure the effectiveness of the framework, we make the following assumption:

Assumption 1. *Given a set of recommendation algorithms $R = \{R_1, R_2, \ldots, R_N\}$, these algorithms are expected to perform differently across datasets. Performance is measured using rank-based metrics (e.g., nDCG@k), with sufficient diversity required to highlight differences among the algorithms.*

In this context, *diversity* implies that the performance of recommendation algorithms should vary notably across datasets. Some recommendation algorithms should excel on certain datasets while others perform poorly, ensuring a wide range of performance.

This assumption is both practical and critical for the success of the Bi-ReIRT framework. It is easily satisfied in real-world scenarios since recommendation algorithms often exhibit varied levels of effectiveness due to their inherent design choices, hyper-parameters, and the specific characteristics of the datasets they process. The Bi-ReIRT framework cannot function effectively if all recommendation algorithms achieve identical performance on all datasets. Evaluating a set of algorithms is meaningless under such circumstances, as there would be no meaningful variability to analyse.

The workflow of Bi-ReIRT is as follows:

i Preparing the input matrix. Applying N recommendation algorithms to M datasets and recording each performance result using the selected rank-based metrics produces an $N \times M$ matrix. Each entry in the matrix represents the performance of a specific recommendation algorithm on a particular dataset.

ii Fitting the beta-IRT model in a bi-directional manner. The fitting process is shown in two directions. Equation (3) represents Direction ①, while Equation (4) represents Direction ②. The interpretation of the learned parameters from each direction is illustrated in Figure 1.

iii Analysing the evaluation results. The evaluation results are analysed using the learned parameters. The dataset's ICC is generated using $\boldsymbol{\delta}_j$, $\boldsymbol{a}_j$, and $\boldsymbol{\theta}_i$, while the recommendation algorithm's ICC is generated using $\boldsymbol{\delta}_i$, $\boldsymbol{a}_i$, and $\boldsymbol{\theta}_j$. Based on the learned traits, recommendation algorithms are analysed by their ability, and datasets are analysed by their inherent challenges.

3 Experiments

3.1 Experimental Setup

We use RecBole [7] as the framework to train 17 recommendation algorithms across 11 datasets, 7 of which are from the Amazon 2023 dataset collection. The algorithms and datasets used in our experiments are listed in Table 2. Due to space constraints, references for the selected recommendation algorithms and datasets are not included; detailed descriptions can be found in the RecBole documentation[2]. It is worth mentioning that training on large and complex datasets demands substantial time and computational resources. Following RecBole's efficiency analysis[3], we prioritise recommendation algorithms with relatively fast training speeds to ensure computational feasibility.

Table 2: The selected 17 recommendation algorithms and 11 datasets.

Recommendation algorithm	Dataset
Random, Pop, ItemKNN (2004), BPR (2009), ENMF (2020), DMF (2017), NNCF (2017), MultiDAE (2018), MultiVAE (2018), NCEPLRec (2019), RecVAE (2020), CDAE (2020), LINE (2015), SGL (2021), DiffRec (2023), RaCT (2020), SimpleX (2021)	Subscription_Boxes, Magazine_Subscriptions, Digital_Music, Gift_Cards, Health_and_Personal_Care, Handmade_Products, All_Beauty, ml-100k, ml-1m, epinions, ModCloth

To manage the training of 17 recommendation algorithms across 11 datasets, we use the default parameters suggested by RecBole and train each algorithm for 100 epochs. All recommendation algorithms are trained on two NVIDIA Tesla P40 GPUs, each with a maximum memory capacity of 24 GB. We use nDCG@10 as the primary rank-based metric in the proposed Bi-ReIRT framework. To demonstrate the generalisability of our framework, we also include four additional popular rank-based evaluation metrics: Precision@10, Recall@10, Hit@10, and MRR@10.

3.2 Experimental Results

The experimental results are presented in Figure 2 and Figure 3. The left plot in Figure 2 analyses dataset difficulty and discrimination. The vertical axis represents dataset difficulty, where higher values signify datasets that are more difficult for most recommendation algorithms (i.e., the ICCs mostly remain in the lower region). The horizontal axis represents dataset discrimination, where higher values indicate a stronger ability to differentiate between the performance of different algorithms (i.e., the ICCs are steeper). The right plot in Figure 2 illustrates the relationship between recommendation algorithm ability (i.e., the latent trait learned by Direction ①) and performance. Each ICC depicts how performance changes as algorithm ability increases. The horizontal axis represents algorithm ability, while the vertical axis shows performance, with higher values indicating better outcomes.

From Figure 2, we observe that the selected algorithms have limited ability to achieve high performance, with their ability range being $[0.004, 0.541]$ as shown in the shaded area. All ICCs show an increasing trend in performance as algorithm ability increases. The dataset Health_and_Personal_Care exhibits the highest discrimination value, reflected by its steep ICC. We can infer that if a recommendation algorithm has an ability value higher than 0.75, its performance will become more sensitive to increases in ability, yielding greater benefits.

The left plot in Figure 3 examines recommendation algorithm consistency and difficulty limit. The vertical axis represents the difficulty limit, where higher values signify more powerful algorithms for most datasets (i.e., ICCs remain higher). The horizontal axis represents algorithm consistency, where higher values indicate greater

[2] https://recbole.io/docs/user_guide/model_intro.html

[3] https://github.com/RUCAIBox/RecBole/blob/master/asset/time_test_result/General_recommendation.md

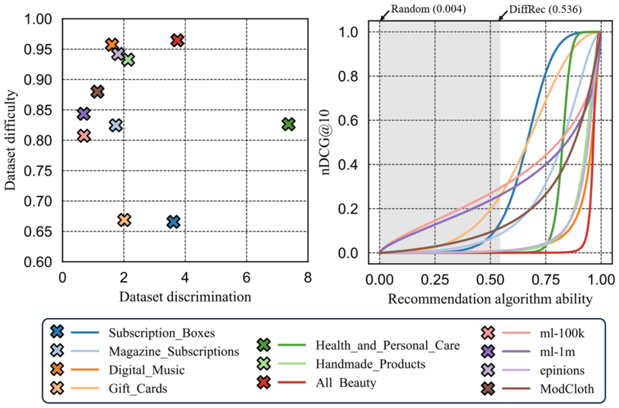

Figure 2: The results for Direction ①. The scatter plot (*left*) shows the dataset discrimination and difficulty, while the ICC (*right*) for each dataset is also presented. The shaded area indicates the ability range of the selected recommendation algorithms.

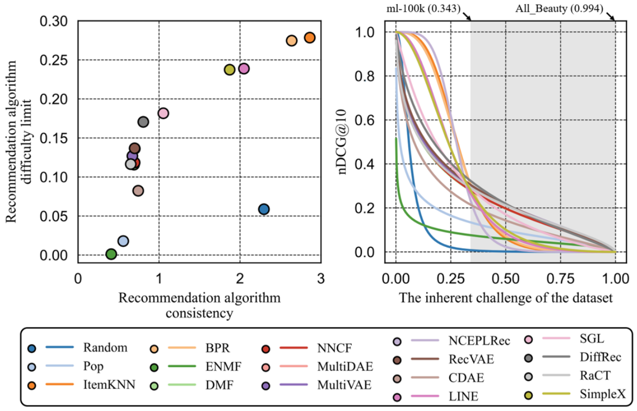

Figure 3: The results for Direction ②. The scatter plot (*left*) shows the recommendation algorithm consistency and difficulty limit, while the ICC (*right*) for each recommendation algorithm is also presented. The shaded area indicates the range of inherent challenges of the selected datasets.

sensitivity to dataset challenges (i.e., steeper ICCs). The right plot in Figure 3 visualises recommendation algorithms' performance across datasets with varying challenges (i.e., the latent trait learned by Direction ②). Each ICC shows how performance changes as dataset difficulty increases. The horizontal axis represents algorithm ability, and the vertical axis shows the inherent challenge of the dataset, with higher values indicating greater difficulty.

Table 3: The top 2 and bottom 2 ability value of recommendation algorithms and inherent challenge of datasets based on different rank-based metrics.

#	ndcg@10	hit@10	mrr@10	recall@10	precision@10
1	DiffRec (0.536)	RaCT (0.644)	DiffRec (0.610)	RaCT (0.523)	DiffRec (0.374)
2	RaCT (0.532)	DiffRec (0.630)	RecVAE (0.602)	MultiVAE (0.523)	RecVAE (0.365)
...	...	...	...	...	...
16	ENMF (0.036)	ENMF (0.039)	ENMF (0.010)	NCEPLRec (0.169)	ENMF (0.123)
17	Random (0.004)	Random (0.003)	Random (0.001)	Random (0.005)	Random (0.032)
1	All_Beauty (0.994)	Handmade (0.993)	All_Beauty (0.992)	All_Beauty (0.989)	All_Beauty (0.959)
2	Handmade (0.993)	All_Beauty (0.992)	Handmade (0.992)	Handmade (0.983)	Handmade (0.959)
...	...	...	...	...	...
10	ml-1m (0.350)	ml-1m (0.089)	ml-1m (0.207)	ml-1m (0.492)	ml-1m (0.378)
11	ml-100k (0.343)	ml-100k (0.082)	ml-100k (0.204)	ml-100k (0.492)	ml-100k (0.361)

From Figure 3, we observe that the selected datasets have a high level of inherent challenge, with a range of [0.343, 0.994], as shown in the shaded area. While ItemKNN appears to be the best algorithm with the highest difficulty limit, it performs well only on datasets with low inherent challenges (i.e., those less challenging than ml-100k). DiffRec, on the other hand, is the most suitable candidate for the given portfolio of datasets, as its ICC outperforms other algorithms within most of the shaded range.

Overall, Directions ① and ② provide complementary analyses. The recommendation algorithm ability learned by Direction ① is the representation of algorithm difficulty limits and consistency, while the inherent challenge of the dataset learned by Direction ② is the representation of dataset difficulty and discrimination.

3.3 Generalisability Analysis

We use different rank-based metrics to demonstrate the generalisability of Bi-ReIRT, as shown in Table 3. The results show slight variations in the rankings of recommendation algorithms by ability values and datasets by inherent challenge values, which reflect the different performance aspects emphasised by each rank-based metric. We note that the choice of metrics should align with the specific task and context, which is beyond the scope of our work.

4 Conclusion

In this paper, we propose Bi-ReIRT, a beta-IRT-based framework for evaluating recommendation algorithms at the dataset level. By modelling continuous responses and incorporating rank-based metrics, Bi-ReIRT enables bi-directional analysis of recommendation algorithm ability and dataset inherent challenge, providing deeper insights into their interactions. Experimental results confirm its effectiveness in delivering interpretable and fine-grained evaluations across diverse datasets and algorithms.

References

[1] Yu Chen, Telmo Silva Filho, Ricardo B Prudencio, Tom Diethe, and Peter Flach. 2019. β^3-IRT: A New Item Response Model and its Applications. In *AISTATS*. 1013–1021.

[2] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *RecSys*. 39–46.

[3] Susan E Embretson and Steven P Reise. 2013. *Item response theory.*

[4] Sevvandi Kandanaarachchi and Kate Smith-Miles. 2023. Comprehensive algorithm portfolio evaluation using item response theory. *JMLR* (2023), 177:1–177:52.

[5] Yang Liu, Alan Medlar, and Dorota Glowacka. 2023. What We Evaluate When We Evaluate Recommender Systems: Understanding Recommender Systems' Performance using Item Response Theory. In *RecSys*. 658–670.

[6] Fernando Martínez-Plumed, Ricardo BC Prudêncio, Adolfo Martínez-Usó, and José Hernández-Orallo. 2019. Item response theory in AI: Analysing machine learning classifiers at the instance level. *Artificial intelligence* (2019), 18–42.

[7] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, et al. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *CIKM*. 4653–4664.