Integrating Expert Knowledge in Matrix Factorization

Changsheng Chen[1, 2, *], Robbe D'hondt[2, 3], Alireza Gharahighehi[2, 3], Celine Vens[2, 3, †] and Wim Van den Noortgate[1, 2, †]

[1] *Faculty of Psychology and Educational Sciences, KU Leuven, Campus KULAK, Kortrijk, Belgium*
[2] *imec research group itec, KU Leuven, Kortrijk, Belgium*
[3] *Department of Public Health and Primary Care, KU Leuven, Campus KULAK, Kortrijk, Belgium*

Abstract

Matrix Factorization (MF) has been widely used to build recommender systems that play an important role in adaptive lifelong learning systems. Yet, as a data-driven approach, the optimization of MF estimation mainly focuses on the improvement of prediction errors in a black-box way, which leads to a lack of interpretability regarding latent factors and validity examination of the resulting estimates and predictions. To address this problem, we proposed a revised version of MF to integrate expert knowledge. Specifically, the item-factor matrix is constrained based on expert-defined item-factor information in the estimation and those constraints deliver the interpretation to the resulting latent factors. We illustrate this method with an empirical data set with 60 items and 4,645 students from Trends in International Mathematics and Science Study (TIMSS). The results show that the revised MF has slightly lower prediction performance compared to the traditional MF but provides interpretable latent factors and validated user-factor estimates and accelerates the hyperparameter tuning operation.

Keywords

matrix factorization, expert knowledge, interpretability, validity, recommender system

1. Introduction

Recommending personalized learning materials or exercises to facilitate continuous learning in learners or trainees constitutes a pivotal component in the development of adaptive systems in a lifelong learning context [1-4]. Personalization entails the selection or design of relevant materials based on the unique characteristics of each user, whereas adaptivity signifies the system's ability to adjust to evolving needs and circumstances of users over time. In the past years, several techniques have been proposed to make personalized and adaptive recommendations within learning systems, and one of those is Matrix Factorization (MF) [5–9].

MF was originally proposed to recommend commercial products for online merchants (such as Netflix) based on a given user-item score matrix (rows: users; columns: items; entries: scores

Joint Proceedings of LAK 2025 Workshops, co-located with 15th International Conference on Learning Analytics and Knowledge (LAK 2025), Dublin, Ireland, March 03-07, 2025.

* Corresponding author.
† These authors jointly supervised this work as the co-final authors.
changsheng.chen@kuleuven.be (C. Chen); robbe.dhondt@kuleuven.be (R. D'hondt); alireza.gharahighehi@kuleuven.be (A. Gharahighehi);celine.vens@kuleuven.be (C. Vens); wim.vandennoortgate@kuleuven.be (W. V. D. Noortgate)
0000-0001-6092-6655 (C. Chen); 0000-0001-7843-2178 (R. D'hondt); 0000-0003-1453-1155 (A. Gharahighehi); 0000-0003-0983-256X (C. Vens); 0000-0003-4011-219X (W. V. d. Noortgate)

given by users to the items) [10]. The scores can be binary, indicating whether users watched or liked this movie, polytomous or continuous, referring to levels of interest of users. The input scores provide personalized information on users' historical interaction with the given items. The principle behind the basic MF is that it decomposes a score matrix into two low-rank matrices, i.e., the user-factor matrix and the item-factor matrix. The user-factor matrix provides the relationship information between users and latent factors, while the item-factor matrix offers information on the link between items and latent factors. The multiplication of two low rank matrices constitutes reconstructed scores, which offers prediction information. With the prediction, the system can personalize item recommendations. For example, items with higher predicted scores can be prioritized in movie recommendations. In addition, the relevant estimates can be evolved across time to realize adaptivity.

Theoretically, MF can be seen as a data-driven black-box approach and it has been criticized because it is difficult to interpret resulting latent factors and hence predictions or recommendations [11–13]. From the perspective of the application domain, traditional MF produces predictions only based on a score matrix without considering any other human-defined information, which might cause concern about the validity and interpretability of results. Some studies have proposed approaches to account for those issues. For example, Abdollahi and Nasraoui [14] and Vlachos [15] proposed co-cluster approaches with different methodological designs to give explanations for resulting recommendations. These co-cluster approaches quantify similarities within items or users based on the given user-item interaction patterns, and the recommendations can be justified by identifying certain clusters that the corresponding scores belong to [14,15]. While these similarities can be seen as a form of trusted information regarding reasoning recommendations, it remains challenging to comprehend the resulting factors and predictions for both approaches. In addition, the two co-cluster approaches are based on business scenarios that show different properties compared to learning systems. In educational sciences, learning materials or test items are usually designed for improving or measuring students' skills (considered as latent factors), so the interpretation of latent factors is always important for learners and teachers. For example, an item designed for measuring or improving students' English verbal skill cannot provide information about students' geometry skill, and recommendations need to consider the improvement on targeted skills. Furthermore, in business applications, missing rating scores are typically coded as zero. In learning systems, a zero usually indicates an incorrect answer to a test item or a failed learning task. Those domain-specific differences cause special considerations in the development of analysis techniques.

To address the aforementioned concern, this study proposes a revised version of MF to integrate expert knowledge to interpret resulting latent factors and estimates. Specifically, in the traditional MF, a given response matrix is approximated by a user-factor matrix and an item-factor matrix. In the revised MF, we add constraints to the item-factor matrix with the help of expert knowledge by fixing certain entries to zero and skipping them in the optimization routine. In other words, we only optimize the full user-factor matrix and the non-zero entries of item-factor matrix. These constraints match expert-defined factor tags with latent factors to give interpretation to relevant estimates, which is the central idea in the revised MF. With the revised MF, the follow-up research questions include how integrating domain experts' opinion will affect the prediction performance and the user-factor matrix compared to the original MF. To answer these questions, an empirical study is presented below.

2. Method

The following sections start with an introduction to the empirical data set. After that, the technical details of the proposed revised MF are introduced. Then, the analysis design for comparing the traditional MF and the revised MF is explained.

2.1. Data

An empirical data set from Trends in International Mathematics and Science Study (TIMSS) was used in this study. The selected data was collected from the 4th grade students for mathematics test in TIMSS 2019 in Flanders, Belgium [16]. In total, there were 213 items designed for 4,655 students. The response matrix contained a large number of missing values because of the item administration design [17]. In particular, 86.73% of entries in the response matrix were missing. Each student and each item had 28 and 617 responses on average respectively. Furthermore, we followed relevant instructions of the codebook from the corresponding database in TIMSS 2019 to convert original multi-categorical responses to binary responses (i.e., right or wrong) [17,18]. Specifically, "Correct Response" and "Partially Correct Response" were all coded as 1 and "Incorrect Response" was coded as 0. Additionally, to consider the implementation of cross validation for tuning hyperparameters, we excluded 10 students and 4 items with less than five responses. Then, for the sake of illustration, we randomly selected 60 items. After that, the final response matrix contained 4,645 students as rows and 60 items as columns.

Table 1
Item Examples With Latent Factor Tags

Item ID	Topic Area
MP61261	Expressions, Simple Equations, and Relationships (ESER)
MP61182	Fractions and Decimals (FD)
MP61266	Geometry
MP71079	Measurement
MP51100	Reading, Interpreting, and Representing (RIR)
MP61084	Using Data to Solve Problems (UDSP)
MP61240	Whole Numbers (WN)

In TIMSS, each item is designed and labelled for assessing certain latent factors defined by domain experts, and we adopted latent factor tags under the "*Topic Area*" label system provided by the online codebook [16,17]. Table 1 provides examples of items with those latent factor tags. There are seven factor tags in total and each item is linked to one of seven factor tags. We used the tag information to construct an expert-defined item-factor matrix where the entry 1 indicated that the item can contribute information to the corresponding factor and the entry 0 referred to the opposite.

2.2. Method Implementation

Suppose that an observed user-item rating matrix R_{UI} is given, where r_{ui} refers to the rating of user u for item i. The rating matrix can be approximated by a product of two decomposed matrices P_{UK} and Q_{IK} with a defined rank K (equal to the number of latent factors), which is

$R_{UI} \approx \hat{R}_{UI} = P_{UK}Q_{IK}^T$ for non-missing entries in R_{UI}. It can be further expressed as $r_{ui} \approx \hat{r}_{ui} = p_{uK}q_{iK}^T$. Here, $\hat{r}_{ui}$ refers to the predicted rating that is the product of the user-factor vector $p_{uK} = (p_{u1}, p_{u2}, \dots, p_{uK})$ for user u and the transpose of the item-factor vector $q_{iK} = (q_{i1}, q_{i2}, \dots, q_{iK})$ for item i [10]. To obtain these two decomposed matrices P_{UK} and Q_{IK}, a loss function is constructed to minimize the reconstruction error, which is:

$$\sum_{i=1}^{I}\sum_{u=1}^{U}\left(r_{ui} - p_{uK}q_{iK}^T\right)^2 + \lambda\left(\|p_{uK}\|^2 + \|q_{iK}\|^2\right) \tag{1}$$

In the above equation, λ is the regularization parameter for controlling the overfitting. To minimize the defined loss function, we slightly revised the stochastic gradient descent optimization method proposed by Simon Funk [19], which is:

$$\begin{aligned} p_{uk} &\leftarrow p_{uk} + \gamma(e_{ui}q_{ik} - \lambda p_{uk}) \\ q_{ik} &\leftarrow q_{ik} + \gamma(e_{ui}p_{uk} - \lambda q_{ik})\ when\ q_{ik} \neq 0 \end{aligned} \tag{2}$$

where the prediction error is defined as $e_{ui} = r_{ui} - p_{uK}q_{iK}^T$ and γ denotes the learning rate. The user-factor and item-factor entries denoted as p_{uk} and q_{ik} are repeatedly computed with γ and λ, and q_{ik} is only computed when it is not equal to zero. In other words, when q_{ik} is equal to zero, those entries will be skipped in the optimization routine. The iteration for p_{uk} and q_{ik} is stopped when the defined error threshold or the defined number of iteration steps is reached. Furthermore, P_{UK} and Q_{IK} are initialized with random numbers. In order to integrate expert knowledge, certain entries of Q_{IK} are constrained to zero based on a given expert-defined binary item-factor matrix for mapping between items and expert-defined factors and delivering factor tags to corresponding factors. The zero constraint means that certain items cannot contribute any information to the defined factors.

In the illustration analysis, specifically, P_{UK} and Q_{IK} were initialized with numbers generated from a uniform distribution $U(0,1)$. Then, for the revised MF, certain entries in Q_{IK} were constrained to zero based on the aforementioned expert-defined item-factor matrix for the TIMSS data set. The number of factors (or the defined rank *K*) is fixed to seven. The error threshold was defined as $e_{ui} < 0.01$ and the iteration steps followed hyperparameter settings (see Table 2).

2.3. Design and Analysis

The evaluation of two versions of MF (with and without adaptation) was based on R 4.3.2 [20] in the Flemish Supercomputer Center (Vlaams Supercomputer Centrum; VSC) by an Intel Xeon 8360Y processor with 72 cores. In particular, 71 cores were used to do parallel computation for tuning relevant hyperparameters, and 1 core was used for training the final model and the result analysis.

For the evaluation operation, first, 80% of entries from an input response matrix were selected as training data, which is to estimate the model parameters, and the rest 20% of entries were used as test data, which is to compare the observed and predicted scores. The selection was operated in a random way but under a condition that each item and each student had at least one available response for both training and test data. This is because MF cannot make predictions for completely new items or users, which is denoted as the cold-start problem in the literature [13,21].

Table 2
Hyperparameter Consideration

Hyperparameters	Search Range
Iteration steps	From 100 to 200 with increments of 20
Learning rate: γ	From 0.001 to 0.02 with increments of 0.001
Regularization parameter λ	From 0.01 to 0.2 with increments of 0.01

Second, a 2-folds cross validation was implemented within the training step to tune hyperparameters. In particular, training data was further divided into training and validation data with the same random selection operation. The average of Mean Squared Error (MSE) across folds was used to compute the performance of the input set of hyperparameters. Table 2 presents defined ranges for three hyperparameters, including the iteration steps, the learning rate and the regularization parameter. Those defined values were based on recommended values from relevant software [22] and computation power consumption. It is worth noting that the number of factors is fixed to seven as known information. In total, *6 (iteration steps) × 20 (learning rates) × 10 (regularization parameters) = 1,200* scenarios were created, and the grid search was implemented in tuning procedures.

Finally, the set of hyperparameters with the lowest MSE was selected as the final one, which was implemented to train the final MF model. After that, the final MF model made predictions based on test data and the obtained MSE was used for evaluating methods' performance. Apart from that, relevant estimates for three selected students were further investigated to examine the interpretability and validity performance.

3. Results

3.1. Methods Evaluation

Table 3 shows the results of the two versions of MF. For the revised MF (with integrating expert-defined latent factor information), the lowest MSE for tuning hyperparameters in the training stage was 0.272, which was 30% higher than the traditional MF (0.210). The MSE for the revised MF in test data was 0.262, which was 26% higher than the traditional MF (0.208). The difference for both was around 0.06. In terms of time usage, compared to the traditional MF, the revised MF saved 209 seconds for tuning hyperparameters. Regarding the time for training the final tuned model, due to the higher number of iteration steps, training the final revised MF model took 8 seconds longer than the traditional MF. Additionally, as item-factor estimates were constrained in the revised MF, which affected user-factor estimates, the overall differences in resulting user-factor estimates were investigated as well. Figure 1 presents the pairwise MSE based on user-factor estimates between two approaches, and the overall average MSE was around 0.099. It is worth noting that the order of the user-factor vectors in the traditional MF does not correspond to the order in the revised MF, so the pairwise MSE looped over all columns in the user-factor matrix.

Table 3
Results of Methods Comparison

	Iteration Steps	Gamma: γ	Lambda: λ	Train MSE	Test MSE	Tuning Time	Training Time
Revised MF	160	0.003	0.05	0.272	0.262	9986s	42s
Traditional MF	120	0.006	0.12	0.210	0.208	10195s	34s

Note. “Training Time” here refers to the time for training the final tuned model.

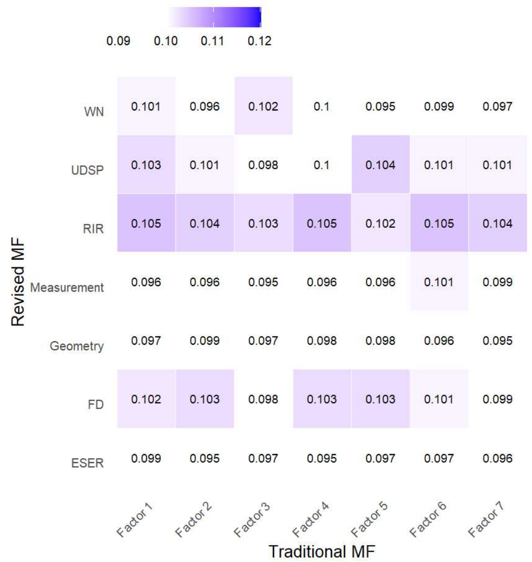

Figure 1: Pairwise MSE for User-Factor Estimates

3.2. Single Case Evaluation

To further evaluate the interpretability and validity performance of two approaches, 3 students were randomly selected: student No. 822, student No. 4091, and student No. 4396. Table 4 presents relevant information for selected students based on the final trained model. Generally, thanks to the item-factor constraints, namely the integration of expert knowledge, the resulting user-factor estimates in the revised MF were interpretable. When the constraints were added to each factor column in the item-factor matrix in the revised MF, factor columns with respective constraints corresponded to expert-defined tags. Specifically, for student No. 822 who answered 7 items, two items designed for measuring “*Expressions, Simple Equations, and Relationships*” were answered correctly and the rest of items for measuring “*Fractions and Decimals*”, “*Measurement*”, and “*Reading, Interpreting, and Representing*” were answered incorrectly. From corresponding estimated user-factor scores, it can be found that scores of giving correct answers were higher than scores of giving wrong answers in the revised MF. Furthermore, scores from the revised MF for indicating the relationship between student No. 822 and the defined factor “*Expressions, Simple Equations, and Relationships*” was 0.662, higher than scores of “*Fractions and Decimals*” (0.601), “*Measurement*” (0.491), and “*Reading, Interpreting, and Representing*” (0.294). This pattern existed for the student No. 4396 and No. 4091 as well.

In contrast, scores estimated by the traditional MF were distributed differently. In the traditional MF, no item-factor constraints were used to deliver interpretable information for each factor, so it was difficult to interpret resulting factors and relevant user-factor scores In other words, the current corresponding position for user-factor scores in the traditional MF in Table 4 can be rearranged freely. In addition, we also randomly selected other single cases to examine the robustness of detected patterns. We found some exceptional cases in the results of revised MF, and they were usually associated with high prediction errors.

Table 4
User-Factor Estimates for Selected Students

Student	Item	Original Response	Predicted Response: Revised MF	Predicted Response: Traditional MF	Factor Tag	User-Factor Score: Revised MF	User-Factor Score: Traditional MF
No. 822	MP61232	1	0.628	0.405	ESER	0.662	0.427
	MP71216AA	1	0.769	0.485	ESER	0.662	0.427
	MP61182	0	0.201	0.127	FD	0.601	0.526
	MP71071	0	0.535	0.433	Measurement	0.491	0.024
	MP71098	0	0.414	0.341	Measurement	0.491	0.024
	MP61211A	0	0.120	0.157	RIR	0.294	0.023
	MP71202	0	0.373	0.51	RIR	0.294	0.023
No. 4396	MP71045	1	0.814	0.49	ESER	0.912	0.617
	MP71179C	1	0.825	0.75	Geometry	0.654	0.545
	MP71067	0	0.067	0.329	Measurement	0.126	0.262
	MP71070	0	0.139	0.602	Measurement	0.126	0.262
No. 4091	MP51103	1	1.011	0.572	FD	0.739	0.174
	MP51079	0	0.117	0.408	Geometry	0.119	0.071
	MP61266	0	0.064	0.203	Geometry	0.119	0.071
	MP51100	0	0.16	0.456	RIR	0.151	0.563
	MP61018	1	0.664	0.476	WN	0.633	0.107
	MP61018B	1	0.815	0.562	WN	0.633	0.107

4. Discussion and Conclusion

In the present study, we proposed a revised version of MF to integrate expert knowledge in estimation procedures. Specifically, in the revised MF the item-factor matrix was constrained in the estimation based on the expert-defined item-factor relationship. The added constraints delivered interpretable information to resulting latent factors with relevant estimates and further provides possibilities for the theoretical validity examination. We illustrated the revised MF and compared it with the traditional MF based on empirical assessment data from the TIMSS 2019. The empirical analyses show that using the revised MF generally makes the estimation faster than the traditional MF while the prediction performance of the former slightly goes down compared to the latter. This is expected somehow because adding constraints to the item-factor matrix means that the available parameter space for optimization is shrunken, which reduces models' expressiveness. At the same time, thanks to the constraints, the time usage for tuning hyperparameters decreases because certain entries in the item-factor matrix are fixed to zero and those are skipped in the optimization routine.

Except for the results of prediction performance and time usage, the analysis of individual students suggested that integrating expert-defined item-factor information produced different user-factor estimates. In detail, regarding the interpretability, adding constraints from an expert-defined item-factor matrix helps match resulting factors with defined factor tags. In terms of validity, user-factor scores of giving correct answers are higher than the case of giving wrong answers in the revised MF. Both improvements cannot be observed in the traditional MF, which is a key reason for proposing the revised MF. As mentioned before, the traditional MF is a purely data-driven method, and the only consideration of estimation optimization is to minimize a defined loss function (mainly related to the prediction error). From the perspective of application domains, it is difficult to interpret and validate resulting item-factor and user-factor estimates. In contrast, using the revised MF largely improves this issue. In practice, the latent factor is usually interpreted as skills in the context of educational science. It is much more logical that when students give wrong answers to certain questions, corresponding skills are lower than when students give right answers, which is in line with results from the revised MF.

Compared to previous approaches [14,15], our approach provides an easy way to integrate external information to improve the interpretability in the MF. Previous methods focus on giving explainable information on resulting recommendations in business applications by quantifying similarities to link predicted recommendations to given ratings. In contrast, our approach concentrates on giving interpretable information on latent factors and estimates, and this kind of information can also be used to provide explainable information for possible recommendations. For example, student No. 4396 had lower values for "*Measurement*" and learning systems could provide materials related to "*Measurement*" for that student. Apart from that, learning systems can also examine predicted responses to items that students do not try and focus on items that students cannot answer correctly. Overall, materials in learning systems are always designed to reach certain learning goals, such as improving students' math skills, which is significantly different from materials in commercial systems. This crucial difference calls for the information that can be interpreted and validated based on educational theories regarding technologies applied in learning systems.

Several limitations of this study need to be acknowledged. First, MF has evolved into different versions in the past years. The proposed version of MF is based on the basic MF without considering newly developed features, which can be improved in the future. Second, the illustration analysis assumed that each item had the same difficulty and was designed for measuring one skill. When items have different difficulty levels and are developed to measure multiple skills, patterns may change. In addition, we only considered binary responses, which could be extended to graded responses. Third, expert-defined item-factor information might be biased or limited, which could be detected by comparing or validating estimates based on different sets of constraints in the revised MF, and this operation is not studied in the above analysis. For example, some studies have explored data-driven refinement methods for Q-matrices based on the performance of defined index [24]. Furthermore, constraints, such as fixing some item-factor entries to zero, might be too strict to reflect realistic complex situations, which can be further relaxed somehow. Fourth, the study is mainly based on empirical data rather than simulation or synthetic data. In a simulation study, true models behind data are known or controlled by researchers, which offers grounds for more comprehensive comparison.

Acknowledgements

This work was partly supported by Research Fund Flanders (FWO) 1S38023N (Robbe D'hondt). We also acknowledge the Flemish Government (AI Research Program). Additionally, the computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation-Flanders (FWO) and the Flemish Government-department EWI.

References

[1] S.I. Ramírez Luelmo, N. El Mawas, J. Heutte, Learner models for MOOC in a lifelong learning context: a systematic literature review, in: H.C. Lane, S. Zvacek, J. Uhomoibhi (Eds.), Computer Supported Education, Springer International Publishing, Cham, 2021: pp. 392–415. https://doi.org/10.1007/978-3-030-86439-2_20.

[2] J. Kay, Lifelong Learner Modeling for Lifelong Personalized Pervasive Learning, IEEE Trans. Learning Technol. 1 (2008) 215–228. https://doi.org/10.1109/TLT.2009.9.

[3] A. Gharahighehi, R. Van Schoors, P. Topali, J. Ooge, Adaptive Lifelong Learning (ALL), in: International Conference on Artificial Intelligence in Education, Springer Nature Switzerland, Cham, 2024: pp. 452–459.

[4] J. Conesa, J.-M. Batalla-Busquets, D. Bañeres, C. Carrion, I. Conejero-Arto, M. Del Carmen Cruz Gil, M. Garcia-Alsina, B. Gómez-Zúñiga, M.J. Martinez-Argüelles, X. Mas, T. Monjo, E. Mor, Towards an educational model for lifelong learning, in: L. Barolli, P. Hellinckx, J. Natwichai (Eds.), Advances on P2P, Parallel, Grid, Cloud and Internet Computing, Springer International Publishing, Cham, 2020: pp. 537–546. https://doi.org/10.1007/978-3-030-33509-0_50.

[5] A. Gharahighehi, Recommender systems for personalized applications, Doctoral dissertation, KU Leuven, 2022. https://lirias.kuleuven.be/retrieve/675420.

[6] K. Mouri, F. Suzuki, A. Shimada, N. Uosaki, C. Yin, K. Kaneko, H. Ogata, Educational data mining for discovering hidden browsing patterns using non-negative matrix factorization, Interactive Learning Environments 29 (2021) 1176–1188. https://doi.org/10.1080/10494820.2019.1619594.

[7] M.C. Desmarais, Mapping question items to skills with non-negative matrix factorization, ACM SIGKDD Explorations Newsletter 13 (2012) 30–36. https://doi.org/10.1145/2207243.2207248.

[8] O.C. Santos, J.G. Boticario, Educational recommender systems and technologies: practices and challenges, IGI Global, 2012. https://doi.org/10.4018/978-1-61350-489-5.

[9] J. Kim, J.Y. Park, W. Van den Noortgate, On the use of bayesian probabilistic matrix factorization for predicting student performance in online learning environments, in: 6th International Conference on Higher Education Advances (HEAd'20), Universitat Politècnica de València, 2020. https://doi.org/10.4995/HEAd20.2020.11137.

[10] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (2009) 30–37. https://doi.org/10.1109/MC.2009.263.

[11] M. Aleksandrova, A. Brun, A. Boyer, O.G. Chertov, What about interpreting features in matrix factorization-based recommender systems as users?, in: ACM Conference on Hypertext & Social Media, 2014. https://api.semanticscholar.org/CorpusID:1761880.

[12] M. Rossetti, F. Stella, M. Zanker, Towards explaining latent factors with topic models in collaborative recommender systems, in: 2013 24th International Workshop on Database and Expert Systems Applications, IEEE, Los Alamitos, CA, USA, 2013: pp. 162–167. https://doi.org/10.1109/DEXA.2013.26.

[13] A. Gharahighehi, K. Pliakos, C. Vens, Addressing the cold-start problem in collaborative filtering through positive-unlabeled learning and multi-target prediction, IEEE Access 10 (2022) 117189–117198. https://doi.org/10.1109/ACCESS.2022.3219071.
[14] B. Abdollahi, O. Nasraoui, Using explainability for constrained matrix factorization, in: Proceedings of the Eleventh ACM Conference on Recommender Systems, ACM, Como Italy, 2017: pp. 79–83. https://doi.org/10.1145/3109859.3109913.
[15] M. Vlachos, C. Dunner, R. Heckel, V.G. Vassiliadis, T. Parnell, K. Atasu, Addressing interpretability and cold-start in matrix factorization for recommender systems, IEEE Trans. Knowl. Data Eng. 31 (2019) 1253–1266. https://doi.org/10.1109/TKDE.2018.2829521.
[16] F. Bethany, F. Pierre, Y. Liqun, TIMSS 2019 user guide for the international database, 2nd ed., Retrieved from Boston College, TIMSS & PIRLS International Study Center website: https://timssandpirls.bc.edu/timss2019/international-database/, n.d.
[17] I.V.S. Mullis Ed, M.O. Martin Ed, TIMSS 2019 assessment frameworks, International Association for the Evaluation of Educational Achievement., 2017.
[18] M.O. Martin, M. von Davier, I.V.S. Mullis, Methods and procedures: TIMSS 2019 technical report, International Association for the Evaluation of Educational Achievement, 2020. https://eric.ed.gov/?id=ED610099 (accessed August 13, 2024).
[19] F. Simon, Netflix update: Try this at home, (2006). https://sifter.org/~simon/journal/20061211.html.
[20] R Core Team, R: A language and environment for statistical computing, (2024). https://www.R-project.org/.
[21] U. Ocepek, J. Rugelj, Z. Bosnić, Improving matrix factorization recommendations for examples in cold start, Expert Systems with Applications 42 (2015) 6784–6794. https://doi.org/10.1016/j.eswa.2015.04.071.
[22] N. Hug, Surprise: A Python library for recommender systems, JOSS 5 (2020) 2174. https://doi.org/10.21105/joss.02174.
[23] W.-S. Chin, Y. Zhuang, Y.-C. Juan, C.-J. Lin, A fast parallel stochastic gradient method for matrix factorization in shared memory systems, ACM Trans. Intell. Syst. Technol. 6 (2015) 2:1-2:24. https://doi.org/10.1145/2668133.
[24] J. Delafontaine, C. Chen, J.Y. Park, W. Van den Noortgate, Using country-specific Q-matrices for cognitive diagnostic assessments with international large-scale data, Large-Scale Assessments in Education 10 (2022) 19. https://doi.org/10.1186/s40536-022-00138-4.