Comparison of Recommender Systems in an Ed-Tech Application

Alejandro Baldominos 1,2* and David Quintana 1

1 Computer Science Department, Universidad Carlos III de Madrid; {abaldomi,dquintan}@inf.uc3m.es
2 Smile and Learn Digital Creations; alejandro.baldominos@smileandlearn.com
* Correspondence: alejandro.baldominos@smileandlearn.com; Tel.: +34-91-624-6258

Version December 18, 2018

Abstract: Smile and Learn is an Ed-Tech company that runs a smart library with more than 100 applications, games and interactive stories, aimed at children aged 2 to 10 and their families. Given the complexity of navigating all the content, the library implements a recommender system. The purpose of this paper is to evaluate two aspects of such system: the influence of the order of recommendations on user exploratory behavior, and the impact of the choice of the recommendation algorithm on engagement. The assessment, based on data collected between 2018/10/15 and 2018/12/01, required the analysis of the number of clicks performed on the recommendations depending on their ordering, and an A/B/C testing where two recommender algorithms were compared with a random recommendation that served as baseline. The results suggest a direct connection between the order of the recommendation and the interest raised, and the superiority of recommendations based on popularity against other alternatives.

Keywords: recommender systems; collaborative filtering; artificial intelligence; ed-tech

1. Introduction and Background

Smile and Learn (S&L) is a mobile application in the educational technology (Ed-Tech) field, which is aimed at children and displays a large set of educational games. As of December 2018, there is a total of 107 games, which are grouped based on Gardner’s theory of multiple intelligences [1]. The games included in S&L cover the following intelligences: visual-spatial, logical-mathematical, verbal-linguistic, naturalistic, artistic, emotional-intrapersonal and group-interpersonal.

The application is intended for parents and educators to first create an account and then several profiles, one per child. When the application is then started, children can select a profile (if more than one exists) with their name and customized avatar, and then the main screen is displayed.

In the main menu, whose aspect can be seen in Figure 1, the different games are grouped in so-called “worlds”, with each world corresponding to an intelligence: Science (naturalistic), Spatial (visual-spatial), Multiplayer (group-interpersonal), Logic (logical-mathematical), Literacy (verbal-linguistic), Emotions (emotional-intrapersonal) and Arts (artistic). There is one additional world named as the child, which consists on a virtual village where the child must interact with characters in order to improve their wealth, requiring abilities from a combination of all intelligences.

Once a world is chosen, apps are displayed grouped in categories, with one category per line. The children can use the vertical scroll (swipe pattern) to navigate through categories, and the horizontal scroll to navigate between apps within a category. An example screenshot from the Science world is shown in Figure 2.

Because of the increase in the number of games, which is still growing month after month, navigation through the application is becoming more tedious, meaning that it can take more time and effort to search for a certain game within the application. In order to alleviate this issue, we have introduced a navigation ribbon at the bottom of the main menu which allows a fast access to some chosen applications. As it can be seen at the bottom of Figure 1, this navigation pattern consists of two
Figure 1. Screenshot of the main menu of Smile and Learn, showing the different worlds available in the application, each world corresponding to an intelligence.

Figure 2. Screenshot of a world menu (in this case, the Science world) in Smile and Learn, where apps belonging to the same world (or intelligence) are listed grouped by category.

tabs: “Recommended” and “Most played”. The latter displays a list of the five top-played apps for the child using the app, turning into a useful mechanism for providing a fast access to those frequent apps.

Regarding the former tab, it displays a list of up to seven apps recommended to children based on their usage behavior. The design motif behind this recommender system is not only to enable fast access to those apps which might be of interest for a child, but also to enhance exploration: letting children discover games which they might not reach if they had to find them by navigating through the app.

For generating recommendations, there are several strategies that can be followed. Currently, two different strategies are implemented in Smile and Learn:

- Popular: this approach recommends the most popular applications among other users. The motivation behind this approach is that it is likely that there are some games that, because of their quality, design or playability, are specially enjoyable for most children, and therefore it can be safe to recommend these games to children, since there is a high chance that they will find them engaging.
• Collaborative filtering: in this approach, the historical records of usage of all children are used in order to generate recommendations. The idea behind collaborative filtering is to find children similar to the one whose recommendations are being generated, so that these recommendations consist on games that have been played by these similar children, but not the child being targeted. The specifics of the implementation of these algorithms are discussed in relation to a prior version of the system by Ruiz-Iniesta et al. [2].

In the navigation bar displayed at the bottom of the main menu, recommendations from different recommenders could be combined. Also, it is noticeable that while there is a maximum of seven recommended games, in some screen sizes or formats it can occur that less than seven are displayed and then the list becomes scrollable. Based on our experience, most screens are able to show five recommendations, with the last two being hidden and accessible only upon scroll (through a swipe pattern or by pressing the side arrows, an example can be seen in Figure 1).

In this paper, we want to design a sound experiment in order to determine which recommender strategy is working best, i.e., which one is translating into a higher engagement of children with the recommended games. We are also interested in getting insights about the interaction patterns taken by children when dealing with the recommendations. With this information, we could improve the system thus resulting in a higher success of the recommender system and an enhanced user experience.

The rest of the document is structured as follows: In Section 2 we study related works of recommender systems and interaction patterns within the context of an Ed-Tech application. Later, in Section 3 we thoroughly describe the experimental methodology, whose results are later described in Section 4. Finally, conclusive remarks are presented in Section 5 along with suggested lines of future work.

2. Related Work

Applications of technology to educational processes (a field known under the term “Ed-Tech”) are not particularly new. However, it has only been in the last decade that the availability and ubiquity of technology devices (such as smart-phones or tables) has reached a point in which these applications can be deployed in large-scale settings.

Additionally, the implementation of data science or machine learning techniques are allowing to extend Ed-Tech far beyond the classical definition of using technology to deliver contents to an audience (e.g., slides, interactive videos, etc). Instead, novel developments are able to track the progress of individual users, detect strengths and weaknesses, and provide a customized experience to enhance the learning process, among other possibilities.

Many examples of such novels applications can be found in the literature in recent years. For example, Charleer et al. [3] presented a learning analytics dashboard aimed at improving the communication in advising sessions, helping to increase students’ motivation. Käser et al. [4] have proposed an approach to student modelling to represent and predict students’ knowledge and skills. Also, Al-Saleem et al. [5] and Bydžovská [6] have proposed alternative methods to predict students’ academic performance, and Uneno and Miyazawa [7] introduced a system designed to scaffold learning in programming. Finally, it is worth mentioning the efforts to develop virtual worlds for education, reviewed by Nunes et al. [8], and the new possibilities in the introduction of gamification dynamics in the space favored by the adoption of information technologies discussed by Dichev and Dicheva [9].

An important subset of Ed-Tech applications are those aiming at providing a customized experience by suggesting users a certain topic or learning process. This problem can generally be regarded as a recommendation problem. Recommender systems, the instruments used to tackle the problem, have a three-decade long history and have been the subject of a sizable amount of research both in general [10,11] and specifically focused on education [12,13].

The range of strategies developed to assign users to items is broad. Some well established alternatives would be the recommendation of the most popular items [14]; collaborative filtering [15];
content-based approaches [16] or usage context-based similarity [17]. In addition to the basic possibilities, there is room for hybrid strategies, like the one currently implemented at S&L, that combine the output of several canonical strategies to generate their output.

The design and implementation of recommender systems in the education space pose a number of difficulties, as discussed by Tarus and Niu [18], and so does their evaluation [19]. Among these challenges, we can mention the selection of the right algorithm (or set of algorithms), and the design of the interface. Even though the first aspect is not settled yet, there have been valuable efforts to contribute evidence in this regard like a recent work by Kopeinik et al. [20].

The second aspect is specially complicated when it involves children, as the intuitions by adult designers might not be correct [21,22]. Even though there have been advances in this aspect, like the contribution of Wu et al. on interface design for children [23] there is still a significant amount of work to be done.

If we focus on recommendation for children, even though we could mention some relevant works like [24] the volume is still very scarce. To illustrate this, it is worth mentioning that the first specialized workshop, KidRec, took place in 2017 [25]. As Deldjoo et al. [26] explain, recommender systems have been traditionally focused on adults and, when it comes to children, the field is still in its infancy.

3. Materials and Methods

In this study we are interested in comparing different recommender strategies that can be used for suggesting potential games of interest to a child. As it was described earlier, the two alternatives that we are interested in comparing are the popular one, which provides recommendations based on apps commonly played by most children; and the collaborative filtering whose recommendations are based on those games played by similar users, i.e. those who have a similar record of played games.

In order to get a better understanding of which strategy is working best, we will test the performance of both recommender strategies and, additionally, we will compare them against a baseline random recommender, which just recommends random alternatives.

Therefore, an A/B/C testing is performed when running the recommender system. During the whole period in which the experiment will be running, each child will be assigned one group (either A, B or C) randomly following a uniform distribution. When running the recommender, children will receive recommendations coming from a different strategy based on their group:

(A) The popular recommendation strategy is used.
(B) The collaborative filtering recommendation strategy is used.
(C) The random recommendation strategy is used.

Instead of generating all the recommendations (a maximum of seven) from the resulting strategy, only the first three recommendations will be computed using the recommender corresponding to the child’s group, and the remaining recommendations will be chosen randomly.

In all cases, some filters are applied in order to opt out some recommendations which might not be suitable for a given child:

- Some games are blacklisted, meaning that they might not have enough quality as to be recommended (e.g., they are in a beta stage).
- Games are filtered out if they are not available in the version of the app owned by the user (e.g., a game was introduced in version 4 and the child is using version 3).
- Games are filtered out if they are not designed for the range of age of the target child (e.g., a game is aimed at children 4–6 years old, but the child is 3 years old).

It is worth mentioning that these filters are applied in all cases, even in the random recommendation strategy.

Regarding the collaborative filtering recommender, it is based on implicit rather than explicit feedback. This decision is motivated by the fact that small children have their own interacting patterns,
and might be unable to properly give credit or score to a game after playing it [26]. For this reason, a system based on explicit feedback could be unreliable. Instead, we are basing recommendations on implicit patterns of interaction, and in particular in how much and how long does a child play with a game.

3.1. Dataset

During the period of the experiment, 2018/10/15–2018/12/01, we recorded the following information, which we will use to later evaluate the system and report the results:

- The recommender (popular, collaborative filtering, or random) assigned to each child.
- The recommendations generated, and also which recommender provided each of these recommendations.
- The date and time at which each recommendation is generated.
- The games usage per child, including the times at which they play and the duration for each game.

It is worth noting that regarding the second aspect, recommendations might not always be generated using the desired recommender. For example, a child might be assigned the collaborative filtering strategy, but this recommender might not have enough information about usage as to generate useful recommendations, or that all of these recommendations are filtered out. In that case, random recommendations are provided instead.

With this data, we can explore the patterns of engagement of children with the different games depending on whether those games were recommended or not.

3.2. Performance metrics

The impact of the position in the ribbon of recommendations will be analyzed using click-through frequencies. That way, we will determine whether the first five visible items get more clicks than the last two, and whether the ordering within the visible and invisible ones matters.

The assessment of the recommendation algorithms will be made according to two engagement metrics. One based on the number of games and another one on game time.

The first one, the average number of games per user (ANG), is formally defined as:

$$\text{ANG} = \frac{\sum_{i=1}^{\text{NumUsers}_R} \text{Games}_{RI}}{\text{NumUsers}_R}$$  \hspace{1cm} (1)$$

Where $\text{Games}_{RI}$ is the number of games played by user $i$ on apps recommended by algorithm $R$ (either Collaborative Filtering, Popular or Random, and $\text{NumUsers}_R$ is the total number of users who were recommended the apps by algorithm $R$ and acted on it.

The second aspect of engagement to be measured is the average game time (AGT) by users who acted on the recommendations. The expression used to compute the indicator is:

$$\text{AGT} = \frac{\sum_{i=1}^{\text{NumUsers}_R} \text{GameTime}_{RI}}{\text{NumUsers}_R}$$  \hspace{1cm} (2)$$

Where $\text{GameTime}_{RI}$ is the total time spent by user $i$ playing apps recommended by algorithm $R$ (either Collaborative Filtering, Popular or Random, and $\text{NumUsers}_R$ is the total number of users who were recommended the apps by algorithm $R$ and used them.

We should note that, for the computation of these two metrics, we define “game” as the event where the user interacts with the application for 10 seconds or more. Given the presence of some outliers, we also filtered out games of more than 3000 seconds and the instances where a game was played more than 60 times by the same user. These accounted for less than 0.5% of the sample.
4. Results and Discussion

As it was mentioned before, the first part of the study has to do with the analysis of the impact of the position in the recommendation ribbon on the click-through. The experimental results on this aspect are reported in Table 1. There, we can see the total number of clicks over the relevant period by position for the apps that were recommended in all the possible slots.

Table 1. Accumulated click-through by position in the recommendation ribbon and mean for the visible and invisible positions for the period 2018/10/15 - 2018/12/01.

| Position | Visible | Hidden |
|----------|---------|--------|
|          | 1       | 2      | 3      | 4      | 5      | 6  |
| Clicks   | 6329    | 5834   | 4516   | 2639   | 3960   | 3150| 2937|
| Mean     | 4830    | 3044   |        |        |        |    |
| Rank     | 1 2 3 7 | 4 5 6 1 |

It is apparent that the first three positions grabbed much more interest than the rest. The difference between the first two and the other five is specially sizable. While it is true that the 4th received fewer clicks than the ones that follow, other than that, the results are consistent with the existence of a direct relationship between the order and the number of times that users acted on the recommendations.

If we consider visibility, the ribbon only shows five recommendations at a time. We expected that to be a relevant factor, as getting to the last two requires a supplementary effort from the user. Interestingly, even though the average number of clicks on the visible slots was 4830, higher than the 3044 average clicks on the hidden ones, the role of friction as an element that drags click-through down could be questioned. If we consider the effect of position, where apps on the left hand side gather more interest that those on the right, and we infer the trend line, the recommendations on the 6th and 7th positions get more clicks than expected.

Regarding engagement, the recommender algorithm based on popularity was superior across metrics. As we can see in Table 2, it outperformed the other two both in terms of number of games and accumulated use time by user. Unexpectedly, the proposed collaborative filtering algorithm resulted in a slightly lower mean number games vs. the random alternative. The sign of this difference, however, was the opposite for game time.

Table 2. Engagement metrics for game apps by recommender for the period 2018/10/15 - 2018/12/01.

| Engagement Metric | Mean | Median | Variance |
|-------------------|------|--------|----------|
| Games             |      |        |          |
| NG<sub>Random</sub> | 4.79 | 3      | 34.22    |
| NG<sub>Popular</sub> | 6.72 | 4      | 53.89    |
| NG<sub>CF</sub> | 4.22 | 2      | 24.75    |
| Time              |      |        |          |
| GT<sub>Random</sub> | 740.98 | 405.26 | 1084308.47 |
| GT<sub>Popular</sub> | 1202.76 | 639.79 | 3286849.96 |
| GT<sub>CF</sub> | 776.01 | 295.37 | 5020886.62 |

The statistical significance of the differences reported in Table 2 was assessed according to the protocol that follows. First, we started testing the normality of the distribution of the engagement metrics with Kolmogorov-Smirnov test with Lilliefors correction. In case normality was rejected, we applied Wilcoxon’s test. Otherwise, we tested for the presence homoskedasticity using Levene test and, based on the result, we relied either on Welch test, or the traditional t-test. The results were analogous for the metrics. The superiority of the Popular algorithm over the other two was significant at 1%. Regarding the comparison of the baseline vs the implementation of collaborative filtering, the null
hypothesis of equality could not be rejected at the 5% conventional level for neither the number of games nor the total game time.

5. Summary and Conclusions

In this paper we have described and evaluated the behavior of a recommender system in the scope of an Ed-Tech application. In particular, Smile and Learn is a company that runs a smart library (available in most relevant mobile markets) with over 100 applications, games and tales; aimed at children aged 2–10 and their families. The implemented recommender system served for the purpose of easing navigation through the library and enhancing exploration.

The two aspects that constituted the subject of study of the recommender system were the influence of the order of recommendation on user exploratory behavior and the impact of the choice of a recommendation algorithm on engagement. In order to be able to evaluate such aspects, we acquired data of real interaction with the application between 2018/10/15 and 2018/12/01, running an A/B/C test with three different implementations of a recommender system (including a random baseline) and measuring clicks made on recommendations.

The following conclusions are drawn from the analysis of these data: First, there is a direct association between the order in the recommendation ribbon and the number of clicks. Apps on the left gather more interest that those on the right hand side. The friction introduced by the fact that reaching the last recommendations requires either swiping or pressing on the arrows on the side does not seem to have a negative impact, which was unexpected.

A/B/C testing was used to assess two recommendation approaches, one that recommends apps based on popularity and an implementation of collaborative filtering, both compared against a random recommender used as baseline. The experiments offered two main results: The first one is that the popular algorithm beats the other two both in terms of number of games and the total time spent playing the recommended contents. The second is that the current implementation of collaborative filtering does not seem to be add add value, as it offers the same performance as the baseline.

Future works would include thorough studies on alternatives to the current implementation of collaborative filtering or the optimization of its parameters; studying new possibilities based on the interface to drive the attention of the user to the recommended applications; or the development of recommendation strategies based on competences aimed at fostering the development of the user according to predefined preferences established by parents and educators.

Acknowledgments: This research has received funding from the European Union’s Horizon 2020 Research and Innovation Programme under grant agreement No 756826.

Bibliography

1. Gardner, H. *Multiple Intelligences*; Basic Books: New York, NY, USA, 1983.
2. Ruiz-Iniesta, A.; Melgar, L.; Baldominos, A.; Quintana, D. Improving Children’s Experience on a Mobile EdTech Platform through a Recommender System. *Mobile Information Systems* 2018, 2018, 1–8. doi:10.1155/2018/1374017.
3. Charleer, S.; Moere, A.V.; Klerkx, J.; Verbert, K.; De Laet, T. Learning Analytics Dashboards to Support Adviser-Student Dialogue. *IEEE Transactions on Learning Technologies* 2017, 11, 389–399.
4. Käser, T.; Klingler, S.; Schwing, A.G.; Gross, M. Dynamic Bayesian Networks for Student Modeling. *IEEE Transactions on Learning Technologies* 2017, 10, 450–462.
5. Al-Saleem, M.; Al-Kathiry, N.; Al-Osimi, S.; Badr, G. Mining educational data to predict students’ academic performance. In *Lecture Notes in Computer Science*; Springer, 2015; Vol. 9166, pp. 403–414.
6. Bydžovská, H. Student performance prediction using collaborative filtering methods. In *Lecture Notes in Computer Science*; Springer, 2015; Vol. 9112, pp. 550–553.
7. Ueno, M.; Miyazawa, Y. IRT-based adaptive hints to scaffold learning in programming. *IEEE Transactions on Learning Technologies* 2017, PP, 1–1. doi:10.1109/TLT.2017.2741960.
8. Nunes, F.B.; Herpich, F.; Paschoal, L.N.; De Lima, J.V.; Tarouco, L.M.R. Systematic review of virtual worlds applied in education. Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE), 2016, Vol. 27, p. 657.

9. Dichev, C.; Dicheva, D. Gamifying education: what is known, what is believed and what remains uncertain: a critical review. International Journal of Educational Technology in Higher Education 2017, 14, 1–36.

10. Bobadilla, J.; Ortega, F.; Hernando, A.; Gutiérrez, A. Recommender Systems Survey. Knowledge-Based Systems 2013, 46, 109–132. doi:10.1016/j.knosys.2013.03.012.

11. Lu, J.; Wu, D.; Mao, M.; Wang, W.; Zhang, G. Recommender system application developments: A survey. Decision Support Systems 2015, 74, 12–32. doi:10.1016/j.dss.2015.03.008.

12. Drachsler, H.; Verbert, K.; Santos, O.C.; Manouselis, N., Panorama of Recommender Systems to Support Learning. In Recommender Systems Handbook; Ricci, F.; Rokach, L.; Shapira, B., Eds.; Springer US: Boston, MA, 2015; pp. 421–451. doi:10.1007/978-1-4899-7637-6_12.

13. Erdt, M.; Fernández, A.; Rensing, C. Evaluating Recommender Systems for Technology Enhanced Learning: A Quantitative Survey. IEEE Transactions on Learning Technologies 2015, 8, 326–344. doi:10.1109/TLT.2015.2438867.

14. Jäschke, R.; Marinho, L.; Hotho, A.; Schmidt-Thieme, L.; Stumme, G., Tag Recommendations in Folksonomies. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladeníč, D., Skowron A. In Knowledge Discovery in Databases: PKDD 2007: 11th European Conference on Principles and Practice of Knowledge Discovery in Databases, Warsaw, Poland, September 17-21, 2007. Proceedings; Kok, J.N.; Koronacki, J.; Lopez de Mantaras, R.; Matwin, S.; Mladeníč, D.; Skowron, A., Eds.; Lecture Notes in Computer Science, vol 4702. Springer Berlin Heidelberg: Berlin, Heidelberg, 2007; pp. 506–514. doi:10.1007/978-3-540-74976-9_52.

15. Schafer, J.B.; Frankowski, D.; Herlocker, J.; Sen, S., Collaborative Filtering Recommender Systems. In: Brusilovsky P., Kobza A., Nejdl W. In The Adaptive Web: Methods and Strategies of Web Personalization. Lecture Notes in Computer Science; Brusilovsky, P.; Kobza, A.; Nejdl, W., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2007; Vol. 4321, pp. 291–324. doi:10.1007/978-3-540-72079-9_9.

16. Basilico, J.; Hofmann, T. Unifying Collaborative and Content-based Filtering. Proceedings of the Twenty-first International Conference on Machine Learning; ACM: New York, NY, USA, 2004; ICML ‘04, p. 9. doi:10.1145/1015330.1015394.

17. Niemann, K.; Wolpers, M., Usage Context-Boosted Filtering for Recommender Systems in TEL . In: Hernández-Leo D., Ley T., Klamma R., Harrer A. In Scaling up Learning for Sustained Impact: 8th European Conference, on Technology Enhanced Learning, EC-TEL 2013, Paphos, Cyprus, September 17-21, 2013. Proceedings; Hernández-Leo, D.; Ley, T.; Klamma, R.; Harrer, A., Eds.; Springer Berlin Heidelberg: Berlin, Heidelberg, 2013; Vol. 8095, pp. 246–259. doi:10.1007/978-3-642-40814-4_20.

18. Tarsus, J.; Niu, Z. A survey of learner and researcher related challenges in e-learning recommender systems. Communications in Computer and Information Science 2017, 734, 122–132. Conference of 6th International Workshop on Learning Technology for Education Challenges, LTEC 2017; Conference Date: 21 August 2017 Through 24 August 2017, doi:10.1007/978-3-319-62743-4_11.

19. Lombardi, M.; Marani, A. A comparative framework to evaluate recommender systems in technology enhanced learning: A case study. Lecture Notes in Computer Science 2015, 9414, 155–170. Conference of 14th Mexican International Conference on Artificial Intelligence, MICAI 2015; Conference Date: 25 October 2015 Through 31 October 2015, doi:10.1007/978-3-319-27101-9_11.

20. Kopeinik, S.; Kowald, D.; Lex, E. Which Algorithms Suit Which Learning Environments? A Comparative Study of Recommender Systems in TEL. Adaptive and Adaptable Learning: 11th European Conference on Technology Enhanced Learning, EC-TEL 2016, Lyon, France, September 13-16, 2016, Proceedings; Verbert, K.; Sharples, M., Klobučar, T., Eds. Springer, 2016, Vol. 9891, Lecture Notes in Computer Science, pp. 124–138. doi:10.1007/978-3-319-45153-4_10.

21. Sears, A.; Jacko, J.A. Human-computer interaction: Designing for diverse users and domains; CRC Press, 2009.

22. Bjorklund, D.; Causey, K. Children’s Thinking: Cognitive Development and Individual Differences; SAGE Publications, 2017.

23. Wu, K.c.; Tang, Y.m.; Tsai, C.y. Graphical interface design for children seeking information in a digital library. Visualization in Engineering 2014, 2, 5. doi:10.1186/2213-7459-2-5.
24. Pera, M.S.; Ng, Y.K. Automating Readers’ Advisory to Make Book Recommendations for K-12 Readers. Proceedings of the 8th ACM Conference on Recommender Systems; ACM: New York, NY, USA, 2014; RecSys ’14, pp. 9–16. doi:10.1145/2645710.2645721.

25. Pera, M.S.; Fails, J.A.; Gelsomini, M.; Garzotto, F. Building Community: Report on KidRec Workshop on Children and Recommender Systems at RecSys 2017. SIGIR Forum 2018, 52, 153–161. doi:10.1145/3274784.3274803.

26. Deldjoo, Y.; Frà, C.; Valla, M.; Paladini, A.; Anghileri, D.; Tuncil, M.A.; Garzotta, F.; Cremonesi, P.; others. Enhancing Children’s Experience with Recommendation Systems. Proceedings of the 11th ACM Conference of Recommender Systems, Workshop on Children and Recommender Systems (KidRec’17), 2017.