Exploring Learning Resource Recommendation Approaches for Secondary Education

Christoph Brandstetter
Research Studios Austria FG
Linz, Austria
christoph.brandstetter@researchstudio.at

Fabian Suda
Research Studios Austria FG
Vienna, Austria
fabian.suda@researchstudio.at

Luca Papariello
Research Studios Austria FG
Vienna, Austria
luca.papariello@researchstudio.at

Fabian Dopler
Research Studios Austria FG
Linz, Austria
fabian.dopler@researchstudio.at

Bernhard Göschlberger
Research Studios Austria FG
Department of Telecooperation,
Johannes Kepler University, Linz
Linz, Austria
goeschlberger@researchstudio.at

ABSTRACT

Recommender Systems are a well researched area and there are many approaches and algorithms solving different kind of problems. In TEL recommending appropriate learning resources is a common problem that cannot be generalized well due to different didactic strategies, educational needs and heterogeneous data sources. We therefore argue that a design science process is best suited to apply, combine and improve different established recommendation system approaches to TEL systems. In this paper, we report our architecture to support our design process and the lessons learned from our specific TEL use case. Finally, we conclude discussing the open problems, advantages of our architectural approach and directions for future research.

CCS CONCEPTS

• **Applied computing** → **E-learning**; **Interactive learning environments**; • **Information systems** → **Content ranking**.

KEYWORDS

Technology Enhanced Learning, Recommender Systems, Design Science, Microservice Architecture

ACM Reference Format:
Christoph Brandstetter, Fabian Suda, Luca Papariello, Fabian Dopler, and Bernhard Göschlberger. 2021. Exploring Learning Resource Recommendation Approaches for Secondary Education. In *The 23rd International Conference on Information Integration and Web Intelligence (iiWAS2021), November 29-December 1, 2021, Linz, Austria.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3487664.3487774

iiWAS2021, November 29-December 1, 2021, Linz, Austria

ACM ISBN 978-1-4503-9556-4/21/11…$15.00
https://doi.org/10.1145/3487664.3487774

1 INTRODUCTION

Recommender systems are among the most influential technologies, when it comes to information systems. They provide means to overcome information overload and help users to find relevant content and products. Learning resource recommendation in e-learning differs from traditional recommender system approaches in its overall goal. Overall goals, such as buying a product in e-commerce or watching a movie on a streaming platform are directly linked to a subsequent interaction with an object, that is typically used as a ground truth for evaluation. When it comes to learning, the goal is to optimize learning outcomes efficiently. This often requires the use of metadata, didactic and sequence adaption. Our paper presents ongoing project work for a secondary education e-learning platform that requires two different recommender systems:

(1) learning resource recommendation for course creation (teachers select and sequence learning resources from a large repository),
(2) learning resource recommendation for self-regulated autonomous learning (students select and consume learning resources).

We build upon a metadata model for secondary education reported in [8] to refine content based recommendation approaches and apply didactic strategies. Throughout the project we employed a design science approach in accordance with [12, 13] following the guidelines of [27]. The identified design problem was to tailor the recommendation approach to the task at hand and integrate it with the actively developed, evolving e-learning platform. This paper reports our lessons learned and focuses on the architectural level rather than on the details of the incorporated recommendation algorithms, which have to be left to future work. The remainder of the paper is structured as follows: In section 2 we situate our paper within the context of data driven approaches in technology enhanced learning, general recommender systems and finally recommender systems in technology enhanced learning. We subsequently present our architectural design results in section 3 before concluding and providing an outlook on future work in section 4.

2 RELATED WORK

In the realm of Technology Enhanced Learning (TEL), recommender systems have to be adapted to fit the heterogeneous needs of the individual stakeholders (namely teachers and students) and the involved setting (informal/formal context [3, 20], or even blended learning[9]). The application of recommender systems in TEL Settings, has been widely discussed and evaluated in past research [5, 22, 23] and [17]. The following subsections provide a better understanding of recommender systems in general and within an educational setting, as well as the challenges that arise for these particular fields.

2.1 Grand Challenges for Data Driven TEL

As mentioned, recommender systems in the field of technology enhanced learning face specific challenges, that are distinct to this application area. The *1st dataTEL workshop on Datasets for Enhanced Learning* was specifically devoted to challenges of data and information in the field of TEL and reported four Grand Challenges for data-supported education [4]:

GC 1: a generic framework to share, analyse and reuse
GC 2: improve course completion and drop-outs
GC 3: accurate handling of educational data
GC 4: make data supported information systems an effective tool for educational practice

Although these are not specific to recommender systems, they deliver valuable thoughts and questions regarding recommender systems in TEL.

2.1.1 Grand Challenge 1: A generic framework to share, analyse, and reuse educational datasets. Even though there is an ever increasing amount of applications for TEL in schools or higher education, and thus a large set of produced data, data exploitation or the public availability of that data is still very limited. This leads to an unused opportunity for evaluating learning theories, didactical concepts or development of future learning applications. The need for a generic framework to share educational dataset for research purposes is still to be met.

2.1.2 Grand Challenge 2: Improve course completion and reduce dropouts through data-driven technologies. A general problem within educational institutions is the dropout rate. This is especially true when it comes to a distance or online learning setting. Isolated studying causes a significant amount of student that withdraw from it altogether. Research on TEL tools can help decrease the dropout rate by disseminating its research outcomes to develop novel ways to keep students engaged. These could be in the form of drop-out-analyzers reflection tools or content recommender systems.

2.1.3 Grand Challenge 3: Accurate handling of educational data. Educational Stakeholders could profit tremendously from the application of information retrieval technologies known as Learning Analytics (LA). The technological advantage could be used to mitigate the problem that arises from the increasing gap between the number of students and the number of teachers in the education systems (especially in Europe). LA could reduce delivery costs, create more effective learning environments and increase collaboration between students and teachers. On the other hand, LA faces certain barriers and limitations that arise from privacy and data protection, surveillance concerns that will have to be addressed in policy guidelines. Also, social and ethical implications must be kept in mind.

2.1.4 Grand Challenge 4: Make data supported information systems an effective tool for educational practice. In order to make TEL tools effective within educational practice, limitations and hurdles concerning privacy and education need to be addressed. Realising that data supported tools and their computed results are not easy to understand, it is important to present and visualize the necessary information in a clear and comprehensible way. Presenting or visualizing outcomes in this way is crucial for the right follow-up activities that might lead to improved learning. New competencies for educational stakeholders are required in order to deal with presented outcomes properly. These include statistical knowledge, critical thinking, privacy awareness as well as ethical abilities.

2.2 Generic Recommender System Challenges

Khusro et. al. [16] summarize issues and challenges in recommender systems research. Their list encompasses

(1) cold start problem
(2) synonymy
(3) shilling attacks
(4) privacy
(5) limited content analysis and overspecialization
(6) grey sheep
(7) scalability
(8) latency problem
(9) evaluation and availability of online datasets
(10) context-awareness.

While most work in the recommender systems community is dedicated to algorithmic improvements, certain problems are more structural and need to be solved on a different level. As our work reports the architectural design, we hereinafter focus on the aspects that especially informed our design decisions.

2.2.1 Evaluation and Availability of Online Datasets. The quality of a recommender system can only be inferred by its evaluation. Selecting the appropriate criteria/metrics is a key problem. Traditionally, offline testing takes place by using a portion of a dataset that has not been used for training (aka test set) and computing certain metrics, e.g. the Mean Absolute Error (MAE) or the Root Mean Squared Error (RMSE) [10, 31]. These metrics are not applicable across domains, particularly when it comes to evaluating context-aware recommenders, for these instances contextual precision and contextual ROC could be applied [36]. More time-consuming and costly methods include questionnaires, interviews, and user studies. The bigger issue at hand is that in many cases there are simply no valid or applicable datasets available for a proper evaluation outside of the real data that the recommender system will be integrated into/for, this is especially true within the context of TEL as is described within [17, 23].

2.2.2 Privacy. Recommender system algorithm performance can be improved by including personal information of a user. This might lead to issues of data privacy and security. If there is reduced trust in the system due to intransparent data protection and usage information, there is a certain reluctance when it comes to adding additional data into a recommender system. Strategies to mitigate problems arising from data breaches/leaks include cryptographic mechanisms, randomized perturbation techniques [28]. Other approaches include allowing users to publish private data without revealing their identities, use Semantic web technologies mainly ontologies in combination with Natural language processing techniques [11].

2.2.3 Context-Awareness. Operationally speaking, context-awareness represents information about the setting in which a recommender system is used in (e.g. current location/activity, time). This contextual information can have a big impact on the performance of a recommender system [6]. Performance can be improved further by gathering user context-related information in an unobtrusive way such as detecting facial expressions [35], recording speech interpretation and physiological signals analysis [14].

2.2.4 Scalability. The growth rate of nearest-neighbour (CF) algorithms shows a linear relation between the number of items and the number of users. For typical recommender systems it is becoming increasingly difficult to process such large-scale data. Although this is not (yet) a problem for recommender systems in the TEL context, due to the limited data available, a system should still be able to scale in the future. Proposed techniques include clustering, reducing dimensionality and Bayesian Networks [33].

2.3 Recommender Systems in TEL

Beyond these more general challenges, research on recommender systems in technology enhanced learning points to further, more specific problems within this application area. Kopeinik et al. [17] compared multiple recommender algorithms in the context of learning environments, by applying them to TEL related datasets[1] from different application areas.

The recommendation problem is frequently recast as a regression task, in which one tries to predict the ratings that users will give to some items. An emblematic example can be found in movie recommendations. Commonly used metrics are thus the Mean Absolute Error (MAE) or the Root Mean Squared Error (RMSE). While measuring the predictive capabilities of a model in such a way might suit some use-cases, this is not the case if one is interested in the actual use of the recommendations. In this case, the system can be evaluated using precision- and recall-based metrics. It is indeed common practice to perform an evaluation up to a given level k and report results in terms of precision, recall, or F_1-score at k ($P@k$, $R@k$, and $F_1@k$, respectively).

While in some scenarios one could argue that the maximization of a formal evaluation metric (such as RMSE for a rating prediction task) is tightly linked to the maximization of a more important utility (e.g. increasing profit or user time), it is arguable whether this applies to the TEL field. The complexity in measuring success for TEL recommendations stems not only from the two very different roles (teacher and student), but also from the inherent difficulty in defining how to evaluate a successful recommendation, especially for students. To answer this question, one could for instance, go all the way down to the consideration of different educational learning theories.

The study [17] included some more generic recommendation algorithms as well as some very specific ones, that were closely tied to the respective learning environment. Specifically they investigated the following approaches:

- Content-based Filtering (CB) [1],
- Collaborative Filtering (CF) [24, 30],
- Most Popular (MP) [15],
- Usage Context-based Similarity (UCbSim) [7, 25, 26],
- Base Level Learning Equation with Associative Component (BLL_{AC}) [19],
- SUSTAIN [21, 32],

The authors acknowledge that the state and availability of specific datasets for TEL applications is rather scarce. They accumulated multiple sources in order to make a proficient assessment of the above mentioned algorithms, yet the data range in each individual source did not contain a high number of users (15236 users in KDD15) nor a high number of resources (42320 Resources in CiteULike dataset) [17]. These numbers pale in comparison to traditional recommender system datasets, such as the dataset for the Netflix Prize Competition, that provides 100 million anonymous movie ratings [2].

Kopeinik et al. [17] conclude that most standard resource recommendation algorithms (MP, CF, CB) are not well suited for the application in TEL. They attribute this to the fact that these more general recommendation algorithms originated in a data-rich domain, which do not fit well with the needs of sparse-data learning environments. The best results were achieved by a combination of different algorithms, such as CF and CB. Consequently, we included to our design objectives, to allow for creating hybrid recommendation services.

3 ARCHITECTURE

One of the conclusions of [23] is that there is currently no evaluation framework for recommender systems within TEL. But rather than aiming for such a unified evaluation framework, we've learned that socio-technical systems—such as the e-learning system subject to our project—ultimately need to be evaluated in the realm of the social subsystem. In other words, the impact of changes in the technical subsystem on the social subsystem in terms of the overall evaluation goal is more important than formal evaluation metrics like RMSE, precision or recall.

Manouselis et al. [23] also suggest that it is crucial to apply careful testing and parameterization before finally being able to deploy them to a real setting. Conversely, a lesson learned from our project is, that the lack of sufficient volume of data for evaluation effectively prevents such an approach for TEL environments.

Our approach is to circumvent this insurmountable obstacles by resorting to a more generic solution. The goal is not to find the best possible solution before the system is actually used, but rather start with a simple baseline approach and integrate it with the technical system such that other approaches can be added and evaluated against it, in both, technical and social subsystem. This aligns with Hevners idea of design as a search process [13].

[1] Bibsonomy, CiteULike, KDD15, MACE, TravelWell, Aposdle

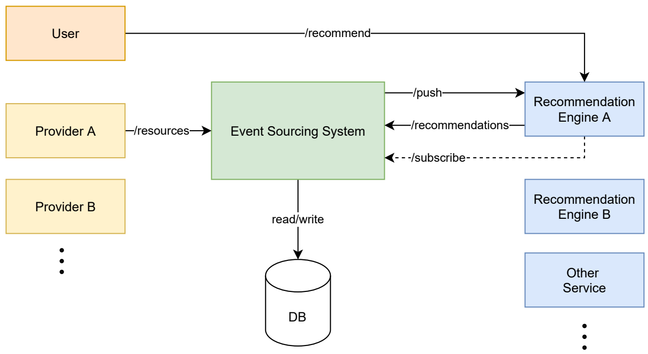

Figure 1: Service Architecture: the Event Sourcing Service serves as intermediary for different Data Providers (such as the e-learning system itself, or an xAPI-endpoint) and the data consuming Recommender Services to allow for loose coupling.

The following subsections describe our designed solution from a structural perspective and a data/process oriented perspective.

3.1 Structural Architecture

We use microservices to allow a loosely-coupled, service-oriented software design. This design principle also allows us to integrate with other components of the e-learning platform, as well as external services such as an xAPI-endpoint, used to track, store and provide learning records[2]. From a structural viewpoint our framework includes the following microservices:

- **Event Sourcing Service:** Service that receives and stores all the data needed/available for (future) recommendations, as well as previous recommendation results from the service and their fetch-timestamp.
- **Baseline Recommender Service:** Recommender System using proven/established general purpose algorithms or algorithms from other domains to provide a baseline and being integrated with the production environment.
- **Experimental Recommender Service:** Improved Recommendation Service, specifically developed for the system under test, and optimized for specifically defined evaluation criteria (e.g. from didactics or educational psychology).

3.1.1 Event Sourcing Service. This central component is responsible for providing different types of data to one or more consumers. This system denotes data as resources and can range from xAPI statements to all kinds of metadata describing a learning object (LO). The implementation of a recommendation engine is categorized as consumer and can use the provided resources for its predictions.

The architecture as depicted in figure 1 gives an overview of all components and their relations. Starting on the left side, one or more providers can use the interface made available by the Event Sourcing System to send resources into the system. The Event Sourcing System (cf. [29]) persists all incoming resources and their origin inside a database. On the other side, consumers can subscribe to the Event Sourcing System for all resources they need. There can be multiple recommendation engines (Baseline Recommender, Experimental Recommender, ...) where each of them is built as its own service. During the subscription process, a consumer also has to provide an endpoint for receiving resources. The Event Sourcing System is now able to push all new incoming resources as well as all historic data that was already stored in the database. In order to evaluate the recommendations of an engine, the Event Sourcing System provides an interface for logging all recommendations requested by a user. Acknowledging privacy concerns, metadata that is transmitted does not contain any sensible data.

Allowing multiple consumers and therefore different recommendation engines, it would be possible to collect the results from different algorithms and return them back to the user. It is worth mentioning that in the future other services than a recommendation engine can subscribe to the Event Sourcing System and use the underlying resource data for their specific calculations. For example, a service can use the existing xAPI statements for learning analytics.

3.1.2 Baseline Recommender Service. In the initial phase of our project, we only had access to user-item interactions—in the context of TEL, these are user-LO interactions. No profitable information was available to characterize users or LOs. The possibility of using a CB approach was therefore excluded, forcing us to opt for a CF approach. For our (CF) recommender system, we have chosen a model-based approach, and in particular low-rank matrix factorization (i.e. a latent space method). Reasons behind this choice are that (i) it scales better with the number of users and (ii) it deals well with the sparsity of our data. More specifically, we opted for the CF-based method first made popular by Simon Funk during the Netflix Prize[3] (cf. [18]).

3.1.3 Experimental Recommender Service. Depending on the data from future user-LO interactions, LO metadata and any other information, one or more experimental recommendation engines will be developed. As mentioned in the Baseline Recommender Service, a CB method is a potential approach which can make use of the LO's metadata. Another possibility would be a combination of these two methods and therefore implementing a hybrid between CF and CB filtering. The Baseline Recommender Service will help to evaluate the new experimental services. Results from newer approaches can be compared to the result of the baseline recommender.

3.2 Data Processing

3.2.1 Scheduled Pre-Calculation. Certain calculations are time consuming and have to be performed ahead of time. Depending on the used algorithm for the recommendation engines, it is possible that different kinds of data will be used. Although new data is retrieved and stored in real time, some algorithms might need a longer time to include the new data and to support immediate recommendation results. For example, xAPI statements are produced quite frequently whenever a user interacts with the learning platform. It wouldn't be feasible to retrain the model every time a new statement is retrieved. A solution to this problem is to retrain the model in off-peak times, like once a day at 2 a.m.

[2] https://adlnet.gov/projects/xapi/

[3] https://sifter.org/~simon/journal/20061211.html

3.2.2 Prior/Historical Data Retrieval. Due to the Event Sourcing Service's ability of storing all previous resource data, a consumer like the recommendation service can make use of the historic data. Whenever a new service is registered, all prior data can be requested and used for further processing. Another use case is requesting data in between a given time frame. This is especially interesting when it comes to the evaluation of the recommendation engine and also the comparison between different recommendation engines. In secondary education the students will usually only learn in certain months of the year. This means that different recommendation systems should be compared with data from the same time period.

3.2.3 Unification and Combination of Results. Our architecture allows multiple recommendation engines. Therefore, the system needs a function for combining results from different engines. One possibility is to have a single service in between the user and all available recommendation services. The user can then specify which engines they want to use and then get presented with all recommendation from each engine. The Event Sourcing System already provides a feature for logging all recommendations. This can be extended to support the collection of additional evaluation data. Feedback can be returned to the Event Sourcing System containing the favoured recommendation engine.

4 CONCLUSION AND FUTURE WORK

For many information systems recommender systems and more specifically, the quality of recommendations is a crucial factor for system success. Consumer recommender systems are generally evaluated using formal metrics, such as RMSE, precision, or recall. For most TEL systems these metrics are often not correlated to the overall objectives. Measures to evaluate the success of recommender systems in the TEL context are critically important, but standardised evaluation metrics are to date unavailable.

Based on the literature and our own lessons learned, we conclude that system's success in educational settings seems to be best captured by non-traditional metrics such as efficiency, effectiveness, or the usefulness perceived by its users [34]. As a consequence, upfront fine-tuning of algorithms or generalization of evaluation results from one learning environment to others seems unrealistic. Conversely, smart architectures that allow continuous improvement and a *design as a search process* approach are needed to customize, scale and integrate evaluation into the learning ecosystem.

For our project, a clear distinction needs to be drawn between different user roles, i.e. teachers and students. For instance, a performance indicator that might make sense to measure for teachers is the average time it takes to build a course with and without the help of the recommender system. For students, it would be useful to consider the average exam performance, possibly while minimising the learning time. Additionally, other success metrics might need to be collected through other methods, such as qualitative research. The historical data retrieval capabilities of our architectural solution, facilitates a qualitative post-hoc investigation with system users, as well as quantitative comparisons of algorithms or quantitative verification against predefined criteria from an educational theory or model. The unification and combination approach also allows to employ A/B-testing in production without additional implementation effort. Through the loose coupling, this even holds true for changing data needs of different algorithms.

In future work, we plan to use our architectural framework to design, deploy and evaluate different recommendation strategies for the project use cases (teacher and student).

ACKNOWLEDGMENTS

The work presented in this paper is part of the project *Adaptive Learning Community Information System for Education* which is funded by the Austrian Research Promotion Agency (FFG) under the general programme.

REFERENCES

[1] Justin Basilico and Thomas Hofmann. 2004. Unifying Collaborative and Content-Based Filtering. *Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004* (09 2004). https://doi.org/10.1145/1015330.1015394
[2] James Bennett and Stan Lanning. 2006. The Netflix Prize. *Proceedings of KDD Cup and Workshop* Vol. 2007 (11 2006).
[3] Helen Colley, Phil Hodkinson, and Janice Malcolm. 2019. *Non-formal learning: mapping the conceptual terrain. A Consultation Report.* https://infed.org/mobi/non-formal-learning-mapping-the-conceptual-terrain-a-consultation-report/ Accessed: 2021-07-12.
[4] Hendrik Drachsler, Katrien Verbert, Nikos Manouselis, Riina Vuorikari, Martin Wolpers, and Stefanie Lindstaedt. 2012. Data Supported Research in Technology-Enhanced Learning. *International Journal Technology Enhanced Learning* 4 (01 2012).
[5] Hendrik Drachsler, Katrien Verbert, Olga C. Santos, and Nikos Manouselis. 2015. *Panorama of Recommender Systems to Support Learning.* 421–451. https://doi.org/10.1007/978-1-4899-7637-6_12
[6] Alexander Felfernig, M. Jeran, Gerald Ninaus, Florian Reinfrank, and Stefan Reiterer. 2013. *Toward the Next Generation of Recommender Systems: Applications and Research Challenges.* 81–98.
[7] Martin Friedrich, Katja Niemann, Maren Scheffel, Hans-Christian Schmitz, and Martin Wolpers. 2007. *Object Recommendation based on Usage Context.* 106–121.
[8] Bernhard Göschlberger, Fabian Dopler, and Christoph Brandstetter. 2020. Managing Learning Resource Metadata for Secondary Education. *ACM International Conference Proceeding Series* (11 2020), 462–466. https://doi.org/10.1145/3428757.3429148
[9] Charles Graham. 2006. *Blended learning systems: Definition, current trends, and future directions.* 3–21.
[10] Jon Herlocker, Joseph Konstan, Loren Terveen, John C.s Lui, and T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems* 22 (01 2004), 5–53. https://doi.org/10.1145/963770.963772
[11] Marcel Heupel, Lars Fischer, Mohamed Bourimi, and Simon Scerri. 2014. *Ontology-Enabled Access Control and Privacy Recommendations.* Vol. 8940. https://doi.org/10.1007/978-3-319-14723-9_3
[12] Alan R. Hevner and Samir Chatterjee. 2010. Design science research in information systems. In *Design research in information systems.* Springer, 9–22.
[13] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. 2004. Design science in information systems research. *MIS quarterly* 28, 1 (2004), 75–105.
[14] Joris Janssen, Egon L. van den Broek, and Joyce Westerink. 2012. Tune in to your emotions: A robust personalized affective music player. *User Modeling and User-Adapted Interaction* 22 (07 2012), 255–279. https://doi.org/10.1007/s11257-011-9107-7
[15] Robert Jäschke, Leandro Marinho, Andreas Hotho, Lars Schmidt-Thieme, and Gerd Stumme. 2007. Tag Recommendations in Folksonomies. 506–514. https://doi.org/10.1007/978-3-540-74976-9_52
[16] Shah Khusro, Zafar Ali, and Irfan Ullah. 2016. *Recommender Systems: Issues, Challenges, and Research Opportunities.* 1179–1189. https://doi.org/10.1007/978-981-10-0557-2_112
[17] Simone Kopeinik, Dominik Kowald, and Elisabeth Lex. 2016. Which Algorithms Suit Which Learning Environments? A Comparative Study of Recommender Systems in TEL, Vol. 9891. https://doi.org/10.1007/978-3-319-45153-4_10
[18] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (2009), 30–37. https://doi.org/10.1109/MC.2009.263
[19] Dominik Kowald and Elisabeth Lex. 2015. Evaluating Tag Recommender Algorithms in Real-World Folksonomies: A Comparative Study. https://doi.org/10.1145/2792838.2799664
[20] Norman Longworth. 2003. *Lifelong Learning in Action: Transforming Education in the 21st Century.* 1–194 pages. https://doi.org/10.4324/9780203465684

[21] Bradley Love, Doug Medin, and Todd Gureckis. 2004. SUSTAIN: a network model of category learning. *Psychological review* 111 (05 2004), 309–32. https://doi.org/10.1037/0033-295X.111.2.309
[22] Nikos Manouselis, Hendrik Drachsler, Katrien Verbert, and erik duval. 2012. *Recommender Systems for Learning*.
[23] Nikos Manouselis, Hendrik Drachsler, Riina Vuorikari, Hans Hummel, and Rob Koper. 2011. *Recommender Systems in Technology Enhanced Learning*. https://doi.org/10.1007/978-0-387-85820-3_12
[24] Miquel Montaner, Beatriz López, and Josep Rosa. 2003. A Taxonomy of Recommender Agents on the Internet. *Artif. Intell. Rev.* 19 (06 2003), 285–330. https://doi.org/10.1023/A:1022850703159
[25] Katja Niemann and Martin Wolpers. 2013. Usage Context-Boosted Filtering for Recommender Systems in TEL, Vol. 8095. 246–259. https://doi.org/10.1007/978-3-642-40814-4_20
[26] Katja Niemann and Martin Wolpers. 2014. Creating Usage Context-Based Object Similarities to Boost Recommender Systems in Technology Enhanced Learning. *IEEE Transactions on Learning Technologies* 8 (12 2014), 1–1. https://doi.org/10.1109/TLT.2014.2379261
[27] Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. 2007. A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems* 24, 3 (12 2007), 45–77. https://doi.org/10.2753/MIS0742-1222240302
[28] H. Polat and Wenliang Du. 2003. Privacy-preserving collaborative filtering using randomized perturbation techniques. *Electrical Engineering and Computer Science*, 625 – 628. https://doi.org/10.1109/ICDM.2003.1250993
[29] Chris Richardson. 2018. *Microservices patterns: with examples in Java*. Simon and Schuster.
[30] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative filtering recommender systems. In *The adaptive web*. Springer, 291–324.
[31] Gunnar Schröder, Maik Thiele, and Wolfgang Lehner. 2011. Setting Goals and Choosing Metrics for Recommender System Evaluations. 811 (01 2011).
[32] Paul Seitlinger, Dominik Kowald, Simone Kopeinik, Ilire Hasani-Mavriqi, Elisabeth Lex, and Tobias Ley. 2015. Attention Please! A Hybrid Resource Recommender Mimicking Attention-Interpretation Dynamics. 339–345. https://doi.org/10.1145/2740908.2743057
[33] Cyrus Shahabi and Yi-Shin Chen. 2003. Web Information Personalization: Challenges and Approaches, Vol. 2822. 5–15. https://doi.org/10.1007/978-3-540-39845-5_2
[34] Katrien Verbert, Nikos Manouselis, Xavier Ochoa, Martin Wolpers, Hendrik Drachsler, Ivana Bosnic, and Erik Duval. 2012. Context-Aware Recommender Systems for Learning: A Survey and Future Challenges. *IEEE Transactions on Learning Technologies* 5, 4 (2012), 318–335. https://doi.org/10.1109/TLT.2012.11
[35] Songhua Xu, Hao Jiang, and Francis Lau. 2008. Personalized online document, image and video recommendation via commodity eye-tracking. *RecSys '08*, 83–90. https://doi.org/10.1145/1454008.1454023
[36] Yong Zheng, Robin Burke, and Bamshad Mobasher. 2014. Splitting Approaches for Context-Aware Recommendation: An Empirical Study. *Proceedings of the ACM Symposium on Applied Computing*. https://doi.org/10.1145/2554850.2554989