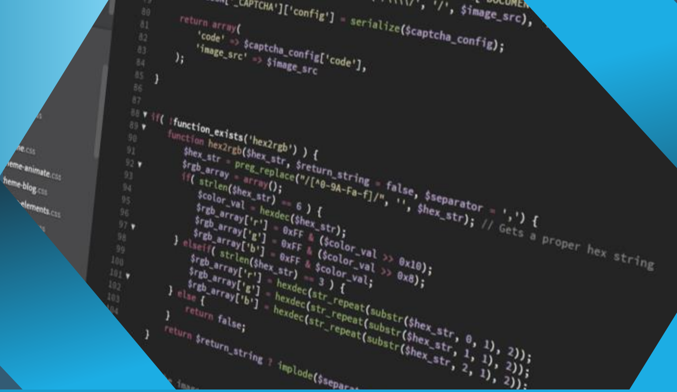
Clean CaDET
Clean Code and Design Educational Tool



REPORT ON THE STATE-OF-THE-ART KNOWLEDGE-BASED RECOMMENDER SYSTEM

Executive Summary

A shift from the traditional classroom to the e-learning system brought benefits to students. Research shows that educational Recommender Systems efficiently support and facilitate learning in the e-learning environments. As part of the Clean CaDET platform, we are developing the Smart Tutor module, an educational Recommender System focused on code smells. We aim to develop and evaluate a system that recommends learning materials for the detected target smell, initially envisioned as a Knowledge-Based recommender that will use ontology for knowledge representation. Thus, in this document, we provide an overview of recommenders in the e-learning and code smell domain. Firstly, we give an overview of Recommender Systems in e-learning, where we focus on recent systematic studies that help us identify components and best practices needed to develop an effective recommender in an adaptive e-learning environment. Secondly, we present the current research related to the Knowledge-Based recommendation in e-learning. Finally, we give an overview of the existing recommenders in the code smells domain.

Key Takeaways

1. *Collaborative Filtering is a popular technique for educational Recommender Systems, while Hybrid techniques have a competitive edge but lower popularity.*
2. *Ontologies have benefits of reusability, reasoning ability, and support for inference mechanisms, which helps to provide enhanced recommendations.*
3. *Knowledge-Based recommenders do not face cold-start problems since the requirements are directly elicited within a recommendation session, thus tend to work better than others at the beginning of their deployment.*

Contact
Jelena Slivka
slivkaje@uns.ac.rs
https://clean-cadet.github.io/

1 Introduction

Recent years brought a drastic shift from the traditional classroom to the novel e-learning system. The benefits of usage of e-learning systems are mainly flexibility in time, taking into consideration the interests of the learners, and the availability of quality resources [1]. Experiments conducted at different times and in various educational settings showed that the students who received computer-aided intelligent tutoring outperformed students from conventional cases in the majority of the observed cases [2].

The biggest problem faced by e-learning systems is information overload. To address this problem, Recommender Systems (RS) are introduced into the e-learning systems [1]. RS have shown promise in supporting learning activities by providing personalized recommendations, content, and services [3]. Educational RS helped students deal with information overload, enhanced information retrieval, and facilitated their collaborative learning [3][4].

Despite these benefits, there are several limitations of the research related to the use of educational RS. Firstly, there's limited information on whether conventional educational environments embraced and applied RS when the experiments finished. Secondly, the use of educational RS in the classroom is solely evaluated through empirical methods [4]. It is unclear how educators use the feedback from the RS to improve the learning environment and material. Furthermore, there are traditional RS problems such as cold-start and rating sparsity [1].

Our RS shall be a Hybrid RS that will combine Knowledge-Based (KB) and Collaborative Filtering (CF) approaches. The KB component will be developed first, to exploit explicit and deep knowledge about the educational domain. This component will enable the collection of user ratings of the recommendation material, and set the basis for the development of the CF component by providing necessary ratings and helping address traditional RS problems of cold-start and rating sparsity. To the best of our knowledge, this is the first educational RS in the domain of code smells. With the development and evaluation of an educational RS that recommends learning materials for detected target smell, we aim to address the limitations of the research related to the use of educational RS.

In this document, we first give an overview of the RS in e-learning in Section 2. Then, in Section 3 we present applications of KB RS in e-learning. Finally, in Section 4 we give an overview of the RS in the context of detecting code smells. While they focus on providing refactoring recommendations rather than recommendation educational materials, they serve as a valuable input for our solution.

2 Overview of Recommender Systems in E-learning

Personalization of learning gets even more important with the increased use of digital learning environments. As the RS are information processing systems that actively gather various kinds of data to build their recommendations, data is primarily about the items to suggest and the users who will receive these recommendations. The uptake of personalized learning approaches and especially RS nowadays is reasonable due to the high demand for interpreting data that is stored in educational institutions. [5]

Data used by RS refers to three kinds of objects:

1. *Items* - objects that are recommended. The value of an item may be positive if the item is useful to the user, or negative if the item is not appropriate and the user made the wrong decision when selecting it.
2. *Users* - since no personalization is possible without a convenient user model, the user model will always play a central role in RS.
3. *Transactions* - recorded interaction between a user and the RS. Ratings are the most popular form of transaction data that an RS collects [5].

RS in e-learning is designed by mapping learners/students as Users, lessons/courses/programs/learning objects as Items, and learners/students' feedback about the Items as Transactions.

A systematic review of RS in education [6] shows that the majority of educational RS recommend courses or learning resources. The most common techniques of RS are CF, followed by Hybrid, and Constraint-based, a special case of the KB approach. Various input parameters pretraining to the requirements of the learners are used in the design of RS. Predominantly, the performance-based parameters (test scores, grades, etc.) followed by learners' interest and gender. The majority of the research work is focused on virtual classrooms and online students. The most common modeling strategies are user profiles as graphs and ontologies.

The recent systematic study of Machine Learning (ML) based RS in e-learning [9] developed a taxonomy that accounts for components required to develop an effective RS. It was found that the necessary components are: ML techniques, algorithms, datasets, evaluation, valuation, and output. The study revealed that CF is a popular technique, while Hybrid techniques have a competitive edge but lower popularity. Major issues of ML-based RS for e-learning are scalability, latency with the newly emerging issue of privacy.

We envision our KB component as an ontology-based RS. A recent systematic study [1] concluded that the ontology-based RS are being used to overcome the cold-start problem and to bring in personalization. As ontology is a way to model learners and learning resources, which helps to retrieve details, this generates more relevant materials for learners. Ontologies have benefits of reusability, reasoning ability, and support for inference mechanisms, which helps to provide enhanced recommendations. As the main challenge, while designing ontologies, research mentions knowledge engineering as mandatory for ontology design and maintenance. One of the major issues is the extremely low number of publicly available datasets needed for the implementation of RS for e-learning.

We see our RS as an adaptive e-learning system that will be able to offer its users a personal and unique experience, with the goal of maximizing their performance. In his systematic study, Troung [8] showed that the Felder-Silverman is the most popular theory applied in adaptive learning systems. It was also pointed out that the integration of different learning styles is recommended. Additionally, evaluation of applications using learning styles is required, especially statistical one.

Another systematic study [7] was conducted to identify the most commonly used personal traits in modeling the learner and the existing techniques suitable for identifying personal traits in an adaptive learning environment. The study showed that the most frequently used traits category is the cognition learning domain category, and the most commonly used personal trait is learning style. The most common technique for identifying personal traits in an adaptive learning environment is computer-based detection, which was significantly more used than the questionnaire. From computer-based detection techniques, hybrid techniques (the combination of two or more Machine Learning (ML) or non-ML techniques) were mostly used. The most common issues in adaptive learning environments are small sample sizes, the limited use of a single dimension of personal traits, and the limitations of existing techniques for identifying personal traits.

3 Knowledge-Based Recommender Systems in E-learning

KB Systems recommend items based on the specific domain knowledge about how certain item features meet users' needs and preferences, and ultimately, how the item is useful for the user. Compared to other recommendation approaches, KB recommenders do not face cold-start problems since the requirements are directly elicited within a recommendation session, thus tend to work better than others at the beginning of their deployment. Even though the work of knowledge engineers is required to explicitly encode the knowledge domain experts into a formal and executable representation [5], KB RS are widely present in e-learning research.

Nitchot et al. [10] proposed a Web-based system for constructing knowledge structures and suggesting study material links. Besides pedagogically-informed knowledge structures and associated applications, researchers proposed a tool for designing and building such structures, a tool for navigating the structures for particular purposes, and a tool for recommending appropriate materials. In this research, the knowledge structures are derived from the subject matter within a targeted knowledge domain using task analysis. Experimental studies which explored the expert reaction ratings showed that the proposed research tool is overall acceptable for learners. However, the system's dependent on search engine API affects the usability of the application.

Harrathi et al. [12] proposed a hybrid KB RS based on ontology for a recommendation of e-learning activities to learners in the context of MOOCs. The main focus was on supporting and personalizing learners' needs to address the problem of losing motivation and leaving the learning process. Ontology is used to model and represent the knowledge about the domain model, learners, and learning activities. Rules are used in an inference process to recommend suitable learning activities based on the learner's profile. In preparation for the proposed RS, authors [11] outlined a set of dimensions that distinguish, describe and categorize learning activities. They proposed a classification of these recommended learning activities according to Bloom's taxonomy. They identified three main learners' dimensions for the learning profile: level of knowledge, learning style, and preferences in terms of pedagogical methods. The proposed RS is not yet integrated and evaluated with a learning platform that supports MOOCs.

Bhaskaran et al. [13] developed an intelligent recommender using split and conquer strategy-based clustering that can adapt automatically to the requirements, interests, and levels of knowledge of learners. Recommender analyzes and learns the styles and characteristics of learners automatically. It was experimentally concluded that the proposed cluster-based recommender improved performance, as indicated by learners in the simulation cluster completing more lessons than those in the no-recommender cluster category. Groups of less than 1000 learners produced better metrics, while the groups with more than 1000 learners produced significant differences in learning metrics.

Fraihat and Shambour [14] proposed a semantic RS for e-learning for assisting learners to find and select relevant learning objects (LO) to their field of interest. The proposed framework utilized various semantic relationships between LO and the learner's needs to provide personalized recommendations for learners. The semantic recommendation algorithm is based on the query keywords by using semantic relations, concepts and reasoning mean in the domain ontology. The proposed RS was not validated and implemented on a prototype.

Obeid et al. [15] proposed an ontology-based RS improved with ML techniques to orient students in higher education. The main objective was to identify the student requirements, interests preferences, and capabilities to recommend the appropriate major and university for each one. Ontologies in the system are used to model higher education institutions, employment, and student. ML techniques are used to learn the profiles of students in order to make recommendations. The system is yet to be implemented and evaluated.

As seen from the presented, the major shortcoming of the research of KB RS in e-learning is the lack of evaluation. The majority of the produced research represents prototypes that aren’t fully implemented and integrated into the real e-learning system. Lack of formal evaluation threatens the validity of the proposed KB RS.

4 Recommender Systems in Code Smells domain

To the best of our knowledge, our RS shall be the first educational RS in the domain of code smells. Nevertheless, in this section we're giving an overview of the existing RS in the code smells domain. They are a valuable input for our solution, even though they focus on providing refactoring recommendations rather than educational materials.

Mkaouer et al. [16] proposed a novel recommendation tool for software refactoring that dynamically adapts and suggests refactorings to developers interactively based on their feedback and introduced code changes. A developer can approve, modify or reject each suggested refactoring. This feedback is then used to update the ranking of the suggested refactorings. The system used rules for code smell detection. The tool was evaluated on four large open-source systems and industrial projects. Statistical analysis of experiments over 31 runs showed that the dynamic refactoring approach performed significantly better than three other search-based refactoring techniques, manual refactorings, and one refactoring tool not based on heuristic search. The limited number of subjects and evaluated systems externally threatens the generalizability of results. The authors only focused on the recommendation of refactorings.

Ouni et al. [17] proposed a refactoring recommendation approach that was driven by three objectives: improvement of design quality defined by software quality metrics, fixing code smells, and introduction of design patterns. The authors employed a non-dominated sorting genetic algorithm to find the best trade-off between these three objectives. Evaluation of the approach was conducted using a benchmark of seven medium and large open-source systems, seven commonly-occurring code smells, and four common design patterns. The approach is empirically evaluated through a quantitative and qualitative study to compare it against three different state-of-the-art approaches, two popular multi-objective search algorithms, as well as random search. The statistical analysis of the results showed that the approach successfully fixed 84% of code smells and introduced an average of six design patterns. The qualitative evaluation showed that 69% of the suggested refactorings are considered by developers to be relevant and meaningful.

Carvalho et al [18] introduced refactoring RS which is capable of binding ontologies for code smell analysis and software refactoring. Refactorings are automatically chosen and semantically linked to their respective code smells. Each refactoring technique has been mapped onto a specific set of six code smells that the authors' smell mining tool is capable of detecting. During the evaluation of this study and previous work related to the development of ontologies, the authors did not conduct any validation with the help of software developers. A limited evaluation was performed on the 4 software projects. Evaluation under a formal experiment will provide applicability of the RS to a broader context of use.

5 References

[1] George, G. and Lal, A.M., 2019. Review of ontology-based recommender systems in e-learning. Computers & Education, vol. 142.

[2] Kulik, J.A. and Fletcher, J.D., 2016. Effectiveness of Intelligent Tutoring Systems: A Meta-Analytic Review. Review of Educational Research, vol. 86, pp.42-78.

[3] Garcia-Martinez S. and Hamou-Lhadj A., 2013. Educational Recommender Systems: A Pedagogical-Focused Perspective. Multimedia Services in Intelligent Environments, vol. 25, pp.113-124.

[4] Rivera, A.C., Tapia-Leon, M. and Lujan-Mora S., 2018. Recommendation Systems in Education: A Systematic Mapping Study. In Proceedings of the International Conference on Information Technology & Systems (ICITS 2018), pp. 937-947

[5] Ricci, F., Rokach, L. and Shapira B., 2015. Recommender System Handbook. Springer US.

[6] Vaidhehi, V. and Suchithra, R., 2018. A Systematic Review of Recommender Systems in Education. International Journal of Engineering & Technology, vol. 7, pp. 188.

[7] Afini Normadhi, N.B., Shuib, L., Md Nasir, H.N., Bimba, A., Idris, N. and Balakrishnan, V., 2018. Identification of personal traits in adaptive learning environment: Systematic literature review. Computers & Education, vol. 130, pp. 168-190.

[8] Truong H.M., 2016. Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities. Computers in Human Behavior, vol. 55, pp. 1185-1193.

[9] Khanal, S.S., Prasad, P.W.C., Alsadoon, A. and Maag, A., 2020. A systematic review: machine learning based recommendation systems for e-learning. Education and Information Technologies, vol. 25, pp 2635-2664.

[10] Nitchot, A., Wettayaprasit, W. and Gilbert L., 2019. Assistive tool for constructing knowledge structures and suggesting related study materials links. Education and Information Technologies, vol. 24, pp. 219-230.

[11] Harrathi, M., Touzani, N. and Braham, R., 2017. Towards a Personalized Recommender System for Learning Activities in the Context of MOOCs. Intelligent Interactive Multimedia Systems and Services, vol. 76, pp. 575-583.

[12] Harrathi, M., Touzani, N. and Braham. R., 2017. A Hybrid Knowledge-Based Approach for Recommending Massive Learning Activities. 14th International Conference on Computer Systems and Applications (AICCSA), pp. 49-54.

[13] Baskaran, S., Marappan, R. and Santi, B., 2021. Design and Analysis of a Cluster-Based Intelligent Hybrid Recommendation System for E-Learning Applications. Mathematics, vol. 9, pp. 197.

[14] Fraihat, S. and Shambour, Q., 2015. A Framework of Semantic Recommender Systems for e-Learning. Journal of Software, vol. 10, pp. 317-330.

[15] Obeid, C., Lahoud, I., El Khoury, H. and Champin, P.A., 2018. Ontology-based Recommender System in Higher Education. Companion of the Web Conference 2018, pp. 1031-1034.

[16] Mkaouer, M.W., Kessentini, M., Bechikh, S., Deb, K. and O Cinneide, M., 2014. Recommendation system for software refactoring using innovization and interactive dynamic optimization. In Proceedings of the 29th ACM/IEEE international conference on Automated software engineering (ASE '14), pp. 331-336.

[17] Ouni, A., Kessentini, M., O Cienneide, M., Sharaoui, H., Deb, K. and Inoue K,. 2017. MORE: A multi-objective refactoring recommendation approach to introducing design patterns and fixing code smells. Journal of Software: Evolution and Process, vol. 29, pp. e1843.

[18] Carvalho, L.P.S., Novais, R.L., Salvador, L.N. and Neto, M.G.M., 2018. An Approach for Semantically-Enriched Recommendation of Refactorings Based on the Incidence of Code Smells. Enterprise Information Systems, vol. 321, pp. 313-335.

The Project is funded by the Science Fund of the Republic of Serbia, Grant No. 6521051, AI - Clean CaDET

Science Fund
of the Republic of Serbia