An Argumentation-based Conversational Recommender System for Recommending Learning Objects

Extended Abstract

Javier Palanca
Universitat Politècnica de València
Valencia, Spain
jpalanca@dsic.upv.es

Stella Heras
Universitat Politècnica de València
Valencia, Spain
scheras@dsic.upv.es

Néstor Duque
Universidad Nacional de Colombia
Manizales, Caldas, Colombia
ndduqueme@unal.edu.co

Vicente Julián
Universitat Politècnica de València
Valencia, Spain
vinglada@dsic.upv.es

Paula Rodríguez Marín
Instituto Tecnológico Metropolitano
Medellín, Colombia
Paularodriguez@itm.edu.co

ABSTRACT

In this paper, we present an argumentation-based Conversational Educational Recommender System (C-ERS), which helps students to find the more suitable learning resources considering their learning objectives and profile. The recommendation process is based on an argumentation-based technique, which selects those learning objects (LOs) for which it is able to generate a greater number of arguments justifying their suitability.

CCS CONCEPTS

• Information systems → Recommender systems; • Computing methodologies → Intelligent agents;

KEYWORDS

Educational Recommender Systems; Argumentation; Explanations

ACM Reference Format:

1 INTRODUCTION

With the actual massive availability of online learning resources, Technology Enhanced Learning (TEL) systems that aim to develop socio-technical innovations for learning and education are gaining great popularity [3]. Many academic institutions offer Massive Online Open Courses (MOOCs), and most new methodologies for teaching-learning (e.g. flip teaching) rely on digital learning objects (LOs) shared on the Internet. This opens new challenges for the research community on recommender systems, where TEL (or Educational Recommender Systems, ERS) is now a hot topic. ERS can help people to find the best learning resources for their learning objectives, educational level, and learning style. The student’s learning style determines which LOs are more adequate for them.

2 CONVERSATIONAL EDUCATIONAL RECOMMENDER SYSTEM (C-ERS)

C-ERS follows a hybrid recommendation technique that uses LOs metadata and information about the student’s profile to compute the recommendations. Also, student profiles include their personal information, interactivity level, language and format preferences, learning style (auditory, kinaesthetic, reader, or visual), and usage history. This technique combines content-based, collaborative and knowledge-based recommendation styles [6] by modeling this expert knowledge in a logic program. We use a defeasible argumentation formalism based on logic programming [4] to implement the logic of the recommendation system and generate arguments to support the recommendation of specific LOs. Thus, our C-ERS provides the students with those LOs that are better supported by a greater number of arguments.
A defeasible logic program $P = (\Pi, \Lambda)$, models strict ($\Pi$) and defeasible ($\Lambda$) knowledge about the application domain. In our C-ERS, the set $\Pi$ represents facts (i.e. strict inference rules with empty body). The set $\Lambda$ represents defeasible rules that encode the defeasible inference that provide reasons to believe $P$. There are different types of defeasible rules that represent the underlying logic each of the system’s recommendation approaches. Content-based rules use information about the student profile to recommend suitable LOs. Collaborative rules use information about the students’ profile to compute a similarity degree among them and recommends a LO that was suitable for similar students. Knowledge-based rules use information about other LO that the user has already assessed in the past to recommend a similar LO.

The program that represents the logic of the C-ERS can be queried to resolve if an argument that supports a specific recommendation can be derived. Thus, when the C-ERS is requested to recommend LOs for a specific student, it tries to derive all possible defeasible rules by backward chaining facts and defeasible rules and following a similar mechanism to the Selective Linear Definite (SLD) derivation of standard logic programming. Therefore, since the system can generate an argument to support the literal that can be derived from each defeasible rule, arguments in this framework are defined as follows: An argument $A$ for $h$ (where $A$) is a minimal non-contradictory set of facts and defeasible rules that can be chained to derive the literal (or conclusion) $h$.

In our argumentation formalism, arguments can be attacked by other arguments that rebut them (i.e. propose the opposite conclusion) or undercut them (i.e. attack clauses of their body). Attacks between arguments are resolved by using a probability measure that estimates the probability that an argument succeeds based on the aggregated probability of the facts and clauses in the body of the rules used to generate the argument. Thus, our C-ERS uses a simplified probabilistic argumentation framework [5] that assigns probability values to arguments and aggregates these probabilities to compute a suitability value to rank and recommend LOs.

Our argumentation-based recommendation technique can also generate explanations from arguments as justification texts. These can be offered to the user to persuade them to try certain LOs and to explain why the system has proposed a specific LO. With each explanation, the user can interact with the system by selecting 1 of the 3 possible predefined responses (one to accept the explanation and hence the recommendation, one to ask for more justifications, and another one to reject the recommendation). The natural order to perform the backward chaining of rules and facts to derive arguments that support recommendations allows us to establish a conversational process between the C-ERS and the user. By this process, the system is able to elicit the actual preferences of the user and allows him/her to correct the system’s wrong assumptions.

3 EVALUATION

For the evaluation tests of our C-ERS, we implemented a prototype of the system that makes use of the LOs available at FROAC (Federation of Learning Objects Repositories of Colombia). Concretely, we used a database 75 LOs of different areas, and 50 students of a computer systems management course of the Universidad Nacional de Colombia. With the evaluation tests, we obtained a database of 472 ratings in total.

We evaluated the effectiveness of C-ERS to provide recommendations that suit the students’ profile and learning objectives. We compared the average ratings provided by the students, both when LOs are provided with and without explanations. Students provided higher ratings to those LOs that presented explanations (3.23 versus 2.6 on a scale of 0 to 5), which demonstrates the advantages of using explanations to offer effective recommendations. In addition, the quantity of objects that got high ratings (from 3 to 4) was greater when explanations were included.

To evaluate the persuasive power of C-ERS, we compared the initial rating provided by the student to LOs and the final one after a ‘conversation’ where the system tried to persuade the student to change the rating by providing her/him with explanations. A wide percentage of ratings were improved by the interchange of explanations (75% improved, 19% unchanged, and 6% decreased), which demonstrates the persuasive power of explanations to improve the opinion of students about the LOs recommended.

To evaluate the scrutability of C-ERS, we analysed the average percentage of students that rejected explanations because they changed any of their initially declared preferences. A significant percentage of students decided to change their preferences at any step of the recommendation process. In average over the total number of the recommendation processes where they participated, 38% of students decided to change any preference. Our C-ERS system is able to capture these changes and allow students to indicate that it has made a wrong inference about his/her preferences in a reasonable time (an average of 2.2 explanations to convince the students), which is crucial in this educational recommendation domain to provide them with recommendations that actually suit their learning objectives.

4 CONCLUDING REMARKS

Actually, the current system constitutes a proof of concept tested with a small set of 50 students of computer science. This could entail some bias in the students’ profile. As future work, we plan to extend the evaluation tests to a large number of students with more heterogeneous profiles. Moreover, we want to study to what extent the new user model acquired with the new preferences elicited during the conversation should be updated and stored permanently. Finally, we plan to enhance the interaction mode of the system with a new natural language interface, able to conduct dialogs in natural language with the students.

ACKNOWLEDGMENTS

This work was funded by the ‘Programa Nacional de Formación de Investigadores COLCIENCIAS’, and by the COLCIENCIAS project 1119-569-34172 from the Universidad Nacional de Colombia. It was also supported by the project TIN2015-65515-C4-1-R of the Spanish government, and by the grant program for the recruitment of doctors for the Spanish system of science and technology (PAID-10-14) of the Universitat Politècnica de València.

REFERENCES


