

Document downloaded from:

<http://hdl.handle.net/10251/65557>

This paper must be cited as:

Rodríguez, P.; Heras Barberá, SM.; Palanca Cámara, J.; Duque, N.; Julian Inglada, VJ. (2016). Argumentation-based hybrid recommender system for recommending learning objects. En Multi-Agent Systems and Agreement Technologies. Springer. 234-248. doi:10.1007/978-3-319-33509-4 19.



The final publication is available at

http://link.springer.com/chapter/10.1007/978-3-319-33509-4_19

Copyright Springer

Additional Information

The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-319-33509-4_19

Argumentation-based Hybrid Recommender System for Recommending Learning Objects

Paula Rodríguez^{*1}, Stella Heras², Javier Palanca², Néstor Duque¹, and Vicente Julián²

¹ Universidad Nacional de Colombia (Colombia)

{parodriguezma, ndduqueme}@unal.edu.co

² Universitat Politècnica de València (Spain)

{sheras, jpalanca, vinglada}@dsic.upv.es

Abstract. Recommender Systems aim to provide users with search results close to their needs, making predictions of their preferences. In virtual learning environments, Educational Recommender Systems deliver learning objects according to the student's characteristics, preferences and learning needs. A learning object is an educational content unit, which once found and retrieved may assist students in their learning process. In previous work, authors have designed and evaluated several recommendation techniques for delivering the most appropriate learning object for each specific student. Also, they have combined these techniques by using hybridization methods, improving the performance of isolated techniques. However, traditional hybridization methods fail when the learning objects delivered by each recommendation technique are very different from those selected by the other techniques (there is no agreement about the best learning object to recommend). In this paper, we present a hybrid recommendation method based on argumentation theory that combines content-based, collaborative and knowledge-based recommendation techniques and provides the students with those objects for which the system is able to generate more arguments to justify their suitability. This method has been tested by using a database with real data about students and learning objects, getting promising results.

1 Motivation

According to the IEEE, a learning object (LO) can be defined as a digital entity involving educational design characteristics. Each LO can be used, reused or referenced during computer-supported learning processes, aiming at generating knowledge and competences based on student's needs [1]. LOs have functional requirements such as accessibility, reuse, and interoperability. The concept of LO requires understanding of how people learn, since this issue directly affects the LO design in each of its three dimensions: pedagogical, didactic, and technological [2]. In addition, LOs have metadata that describe and identify the educational resources involved and facilitate their searching and retrieval. Learning Objects

* Corresponding author.

Repositories (LORs), composed of thousands of LOs, can be defined as specialized digital libraries storing several types of heterogeneous resources. LORs are currently being used in various e-learning environments and belong mainly to educational institutions [2, 3]. Also, federations of LORs provide educational applications to search, retrieve and access specific LO contents available in any LOR [4].

Recommender Systems aim to provide users with search results close to their needs, making predictions of their preferences [5]. In virtual learning environments, Educational Recommender Systems (ERS) deliver LOs according to the student's characteristics, preferences and learning needs. In order to improve recommendations, ERS must perform feedback processes and implement mechanisms that enable them to obtain a large amount of information about users and how they use the LOs. ERS can be classified into several types [6]:

- Content-based ERS: in this kind of systems, recommendations are performed based on the user's profile and created from the content analysis of the LOs that the user has already assessed in the past. The content-based systems use "item-by-item" algorithms generated through the association of correlation rules among those items.
- Collaborative ERS: these systems hold great promise for education, not only for their purposes of helping learners and educators to find useful educational resources, but also as a means of bringing together people with similar interests and beliefs, and possibly as an aid to the learning process itself. In this case, the recommendations are based on a similarity degree among users. Collaborative filtering algorithms aim at suggesting new items or predicting the utility of a certain item for a particular user profile based on the choices of other similar user profiles.
- Knowledge-based ERS: these systems attempt to suggest LOs based on inferences about the user's needs and preferences. Knowledge-based approaches use knowledge about how a particular item meets a particular user need, and can therefore reason about the relationship between a need and a possible recommendation. In addition, these systems are based on the user's browsing history and his/her previously selected LOs.
- Hybrid Recommender Systems: the hybrid approach combines several ERS techniques in order to maximize the advantages of each one and, thus, make better recommendations. To make the hybridization of recommendation techniques –using at least two of them– Burke [6] describes different methods that could be applied (e.g. weighted, switching, mixed, cascade, feature combination, feature augmentation, and meta-level).

In previous work, authors have proposed a Student-Centered Hybrid ERS, designing and evaluating several recommendation techniques for delivering the most appropriate LO for each specific student [7, 8]. Also, they have combined these techniques by using hybridization methods, improving the performance of isolated techniques. The ERS proposed follows a hybrid recommendation technique that combines content-based, collaborative and knowledge-based approaches. In the system, LOs are retrieved from LORs and federations of LORs,

using the stored descriptive *metadata* for these objects. Concretely, our ERS follows the *IEEE-LOM*³ standard to represent the metadata about the LOs. This is a hierarchical data model that defines around 50 metadata fields clustered into 9 categories. Figure 1 shows the fields used in our ERS (highlighted in green). Also, *student profiles*, including their personal information, language, topic and LO's format preferences, educational level, and learning style (aural, kinesthetic, reader, or visual), are used by the system to generate recommendations.

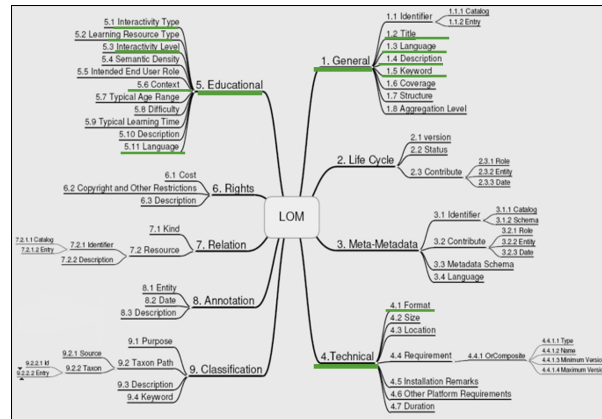


Fig. 1: IEEE-LOM metadata used in the ERS.

Therefore, as shown in Figure 2, the ERS is composed by six modules: three recommendation modules (one for each recommendation technique); a module that performs the hybridization (integration) process, which follows a cascade method to provide recommendations results in strict priority⁴; and, finally, two modules that handle information about student profiles and LOs metadata.

The content-based recommendation module generates its recommendations by applying inference rules among LOs metadata and the student's learning style. The collaborative recommendation module seeks similar user profiles to deliver items that have been assessed by students with similar profiles. The knowledge-based recommendation module searches some LOs similar to those that the student has previously assessed. Then, the integration module performs the hybridization process to provide the student with the most relevant and appropriate LOs. This is done by selecting LOs that have been proposed by 2

³ 1484.12.1-2002 - Institute of Electrical and Electronics Engineers (IEEE) Standard for Learning Object Metadata: <https://standards.ieee.org/findstds/standard/1484.12.1-2002.html>

⁴ Several hybridization methods, as proposed in [6], were tested, and the cascade approach achieved the best recommendation results [8].

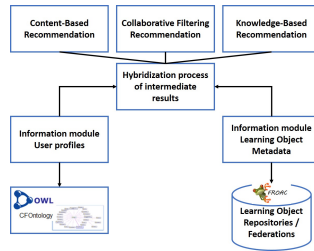


Fig. 2: Student-Centered Hybrid ERS.

or 3 of the recommendation modules. Figure 3 shows the specific LOs metadata and students' profile data that each recommendation modules uses.

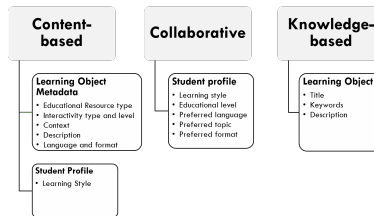


Fig. 3: LOs metadata and students' profile data used by the recommendation modules.

However, this hybridization method has several disadvantages. On the one hand, it does not take the relevance of the LOs into account to encourage the use of a specific LO over another (considering that a LO is relevant for a student if it matches his/her learning objectives and profile). On the other hand, it fails when the LOs delivered by each recommendation technique are very different from those selected by the other techniques (there is no agreement about the best LO to recommend). To analyze the incidence of this problem, we performed some experiments to determine the dispersion degree between the LOs proposed by each recommendation technique (to determine how different are the top 5 or the top 10 LOs proposed by the three recommendation modules). Dispersion tests were performed as follows:

1. A student with a visual learning style was selected.
2. A search on the federation of repositories was performed to retrieve LOs about the topic (keyword) *Algorithms*.
3. The top 5 and top 10 results provided by each recommendation module (content, collaborative and knowledge-based) were saved for analysis.

4. The process was repeated with other keywords (*Programming* and *Audit*).
5. The process was repeated with other students with auditory and kinesthetic learning styles.

Finally, the amount of LOs that overlap between the three recommenders for each iteration of the tests was computed. The average of the results are shown in Figure 4.

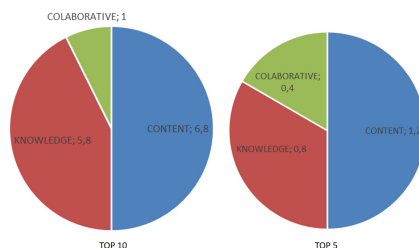


Fig. 4: Results of the dispersion tests.

Results show that, in many cases, the traditional hybridization method cannot deliver any recommendation, since there is no intersection between the recommendations provided by the three recommendation modules. For instance, knowledge-based recommendations (coloured in red) on the Top 5 tests, result in an average dispersion of 0.8 LOs. This means that for each 5 LOs delivered by this recommendation module, on average, only 0.8 overlap with the results of the other techniques (there is no agreement among them).

To overcome this problem, in this paper we present a new hybrid recommendation method based on argumentation theory. Among the wide range of agreement technologies proposed in the last years [9], argumentation provides a natural means of dealing with conflicts and knowledge inconsistencies with a high resemblance with the way in which humans reach agreements. Our method combines content-based, collaborative and knowledge-based recommendation techniques, and provides students with those LOs for which the system is able to generate more arguments to justify their suitability.

The rest of the paper is structured as follows: section 2 reviews related work, section 3 presents our argumentation-based hybrid recommendation method, in section 4 we provide a validation proof for our proposal, and finally, section 5 presents conclusions and future work.

2 Related work

Over the last years, the literature on ERS reports a growing interest in the area. In [5, Chapter 12], discuss the need of support tools for learners based on contextualised recommender systems. According to the authors, it is very important

to take into account pedagogical aspects, like prior knowledge, learning goals or study time in the recommendation process. In addition, they argue that the development of concrete evaluation frameworks that follow a layered approach is still an open research issue. These frameworks may focus on incorporating as many evaluation dimensions as possible, on addressing pedagogical dimensions, or on combining a variety of evaluation methods, metrics, and instruments.

In this regard, in [10] a recommendation system based on genetic algorithms that performs two recommendation processes was proposed. The first one uses explicit characteristics represented in a matrix of student's preferences, while the second assigns implicit weights to educational resources that are considered as chromosomes in a genetic algorithm that optimises them by using historical values. However, compared with our proposal, this work does not perform hybrid recommendation, but combines the characteristics of the student profile. Following a hybrid approach similar to ours, Zapata et al. deliver educational materials adapted to the user profile by combining several types of filtering methods with the available information about LOs and users [11]. However, although this work combines several filtering criteria (content-based, collaborative activity, and demographics), it is aimed at helping teachers rather than students. By contrast, the research presented by Sikka et al. [3], which presents an e-learning environment to recommend learning materials by using web mining techniques and software agents, implements just a unique collaborative recommendation filter rather than using a hybrid approach. However, in [12] a review of some hybrid recommendation systems was performed, concluding that the hybrid filter obtained by integrating collaborative and content-based filtering approaches improves the predictions made by the recommender. We share this view and extend it to recommend educational materials recovered from LORs.

Traditional recommender systems base their recommendations on quantitative measures of similarity, but fail at using the qualitative data available to empower recommendations [13]. Usually, recommender systems do not provide an explanation about the reasoning process that has been followed to come up with specific recommendations. However, people rely more on recommendations when the system can also show the reasons behind the recommendations [14], and when they can understand the reasons why these recommendations are presented [15]. Moreover, even when users already know the recommendations presented, the latter work demonstrated that they prefer recommender systems that are able to justify their suggestions. Thus, what is understood as a good recommendation is changing from the one that minimises some error evaluation to the one that is really able to *persuade* people and make them happier.

Recently, some argument-based recommender systems and recommendation techniques have been proposed to recommend music [16], news [17], movies [18], or restaurants [19], to perform content-based web search [20] or to formalize and structure user opinions in online recommender systems [21]. Among them, we share the approach of the movie recommender system based on defeasible logic programming proposed in [22]. In this work, authors define a preset preference criteria between rules to resolve argument attacks. However, as will be

explained in section 3, we use a probabilistic method to compute the likelihood that an argument prevails over another, which makes the system more adaptive. In educational domains, argumentation theory and tools have a large history of successful applications, specially to teach critical thinking skills in law courses [23]. However, to the best of our knowledge, the application of argumentation theory to enhance ERS is a new area of research.

There are a number of open challenges for the application of argumentation theory to recommender systems [20], such as *exposing underlying assumptions behind recommendations*, *approaching trust and trustworthiness* from the perspective of backing recommendations and *providing rationally compelling arguments for recommendations*. Our work involves a contribution in this latter area.

3 Formal Framework

In this section, we provide an overview on the argumentation formalism used for our proposal. As pointed out in section 1, the original Student-Centered Hybrid ERS proposed uses several sources of knowledge to generate LOs recommendations for the students, namely information about the *student profile* and *metadata* about the LOs to recommend. In this paper, we present a hybrid recommendation method based on argumentation theory that uses these sources of knowledge and provides the students with those LOs for which the system is able to generate more arguments to justify their suitability. Concretely, we use a defeasible argumentation formalism based on logic programming (*DeLP*, see [24] for details) to encode information about the *facts* (metadata and profiles data) and the *rules* that determine the allowed inferences that can be done in our system.

Definition 1 (DeLP). *A defeasible logic program (DeLP) $P = (\Pi, \Delta)$, models strict (Π) and defeasible (Δ) knowledge about the application domain. In our system, the set Π includes strict inference rules with empty body that represent facts. Correspondingly, the set Δ includes defeasible rules of the form $P \leftarrow Q_1, \dots, Q_k$, which represent the defeasible inference that literals Q_1, \dots, Q_k may provide reasons to believe P .*

For instance, *auditory(jose)* represent the fact that a student named 'jose' has an auditory learning style and prefers materials with sounds, and auditory formats such as mp3, mp4, or avi. Facts are assumed to be non-contradictory (e.g., if \sim represents default logic negation, *auditory(jose)* and \sim *auditory(jose)* cannot be inferred). Also, we show below the main defeasible rules of our argumentative framework⁵. These rules are divided on 4 groups, 3 to represent the knowledge used by each recommendation technique (content-based, collaborative or knowledge-based), and 1 to represent general domain knowledge. Section 4 provides an example to clarify their meaning and use.

⁵ The complete rule set is not provided due to space limitations.

GENERAL RULES

$G1: \sim\text{recommend}(\text{user}, LO) \leftarrow \text{cost}(LO) > 0$

$G2: \sim\text{recommend}(\text{user}, LO) \leftarrow \text{quality_metric}(LO) < 0.7$

CONTENT-BASED RULES

$C1: \text{recommend}(\text{user}, LO) \leftarrow$

$\text{educationally_appropriate}(\text{user}, LO) \wedge \text{generally_appropriate}(LO)$

$C1.1: \text{educationally_appropriate}(\text{user}, LO) \leftarrow$

$\text{appropriate_resource}(\text{user}, LO) \wedge \text{appropriate_interactivity}(\text{user}, LO)$

$C1.1.1: \text{appropriate_resource}(\text{user}, LO) \leftarrow \text{user_type}(\text{user}, \text{type}) \wedge \text{resource_type}(LO, \text{type})$

$C1.1.2: \text{appropriate_interactivity}(\text{user}, LO) \leftarrow \text{user_type}(\text{user}, \text{type}) \wedge \text{interactivity_type}(LO, \text{type})$

$C1.2: \text{generally_appropriate}(LO) \leftarrow \text{structure}(OA, \text{atomic}) \wedge \text{state}(LO, \text{final})$

$C2: \text{recommend}(\text{user}, LO) \leftarrow$

$\text{educationally_appropriate}(\text{user}, LO) \wedge \text{generally_appropriate}(LO) \wedge \text{technically_appropriate}(\text{user}, LO)$

$C2.1: \text{technically_appropriate}(\text{user}, LO) \leftarrow \text{appropriate_language}(\text{user}, LO) \wedge \text{appropriate_format}(LO)$

$C2.1.1: \text{appropriate_language}(\text{user}, LO) \leftarrow \text{language_preference}(\text{user}, \text{language}) \wedge \text{object_language}(LO, \text{language})$

$C2.1.2: \text{appropriate_format}(LO) \leftarrow \text{format_preference}(\text{user}, \text{format}) \wedge \text{object_format}(LO, \text{format})$

$C3: \text{recommend}(\text{user}, LO) \leftarrow$

$\text{educationally_appropriate}(\text{user}, LO) \wedge \text{generally_appropriate}(LO) \wedge \text{updated}(OA)$

$C3.1: \text{updated}(LO) \leftarrow \text{date}(LO, \text{date}) < 5 \text{ years}$

$C4: \text{recommend}(\text{user}, LO) \leftarrow$

$\text{educationally_appropriate}(\text{user}, LO) \wedge \text{generally_appropriate}(LO) \wedge \text{learning_time_appropriate}(LO)$

$C4.1: \text{learning_time_appropriate}(LO) \leftarrow \text{hours}(LO) < \gamma$

COLLABORATIVE RULES

$O1: \text{recommend}(\text{user1}, LO) \leftarrow \text{similarity}(\text{user1}, \text{user2}) > \alpha \wedge \text{vote}(\text{user2}, LO) \geq 4$

KNOWLEDGE-BASED RULES

$K1: \text{recommend}(\text{user1}, LO) \leftarrow \text{similarity}(LO1, LO2) > \beta \wedge \text{vote}(\text{user1}, LO2) \geq 4$

Given a DeLP, the program can be queried to resolve if a ground literal can be derived from the program, and hence supported by an argument(s) based on the rules of Δ . Concretely, for our hybrid recommendation method to recommend a

LO to a specific user, we need to be able to derive any of the $recommend(user, LO)$ defeasible rules from our DeLP.

Arguments in this framework are defined as follows:

Definition 2 (Argument). *An argument \mathcal{A} for h (represented as a pair $\langle \mathcal{A}, h \rangle$) is a minimal non-contradictory set of facts and defeasible rules that can be chained to derive the literal (or conclusion) h .*

Then, arguments are generated by backward chaining of both facts and defeasible rules, a mechanism similar to the *Selective Linear Definite (SLD)* derivation of standard logic programming. Therefore, recommendations are computed by chaining arguments in a recursive process that creates a *dialectical tree* (see [24]) whose root node is the original argument under discussion (i.e. whether to recommend or not a LO for a particular user), and whose children nodes are arguments that defeat their parents.

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).

Definition 3 (Attack). *An argument $\langle \mathcal{B}, q \rangle$ attacks another argument $\langle \mathcal{A}, h \rangle$ if we can derive $\sim h$ from \mathcal{B} or if q implies that one of the clauses of \mathcal{A} no longer holds (there are a sub-argument $\langle \mathcal{A}_1, h_1 \rangle$ from $\langle \mathcal{A}, h \rangle$ such that $\Pi \cup \{h_1, q\}$ is contradictory).*

Therefore, an argument for not recommending a LO can be generated if an argument for recommending is attacked. Note that we assume negation as failure, so an argument for not recommending a LO can be generated by chaining rules whose literals cannot be derived (we do not have information to resolve them). For instance, by using the rule *O1*, which recommends a LO for a *user1* if other similar *user2* likes that object (i.e. *user2* has voted the LO with a score greater than 4), we can derive an argument for not recommending the LO: 1) if the system cannot find a similar user (negation as failure); or 2) if there is a similar user and he/she does not like the LO (undercut).

To resolve attacks between arguments, each rule has an associated probability measure that estimates the probability that an argument (generated by using the rule) succeeds based on the aggregated probability of the clauses that form the body of the rule. In doing so, we use a simplified *probabilistic argumentation* framework [25] that assigns probability values to arguments and aggregates these probabilities to compute a suitability value to rank and recommend LOs.

Definition 4 (Argumentation Framework). *In our ERS, an argumentation framework is a tuple (Arg, P_{Arg}, D) where Arg is a set of arguments, $D \subseteq Arg \times Arg$ is a defeat relation, and $P_{Arg} : \rightarrow [0 : 1]$ is the probability that an argument holds.*

The probability of an argument $Arg = \langle \mathcal{A}, h \rangle$ is calculated as follows:

$$P_{Arg} = \begin{cases} 1, & \text{if } \mathcal{A} \subseteq \Pi \\ \frac{\sum_{i=1}^k P_{Q_i}}{k}, & \text{if } \mathcal{A} \subseteq \Delta \mid h \leftarrow Q_1, \dots, Q_k \end{cases} \quad (1)$$

Facts are assumed to have probability 1. The probability of defeasible rules is computed as the average of the probabilities of the literals Q_1, \dots, Q_k that form their body (i.e. 1 if they are facts, 0 if they cannot be resolved, or P_{Q_i} if they are derived from other defeasible rules).

Definition 5 (Defeat). *In our ERS, an argument $\langle \mathcal{B}, q \rangle$ defeats another argument $\langle \mathcal{A}, h \rangle$ if \mathcal{B} attacks \mathcal{A} and $P_B > P_A$.*

4 Validation

Students query our ERS to get LO recommendations that may fit their learning objectives and preferences. With this aim, the system has a search engine that allows a student to find LOs by using keywords that express the educational skills that they want to achieve. This search results in a list of LOs that match the keywords. After that, our ERS starts the recommendations process to rank and deliver LOs of this list: the content-based recommendation module triggers its inference rules by using the LOs metadata and the student’s learning style; the collaborative recommendation module seeks similar user profiles to deliver items that have been evaluated by similar students; and the knowledge-based recommendation module determines whether any LO in the list is similar to another LO that the student has already used and assessed positively. Then, the new argumentation-based hybridization method is used to combine these three sets of LOs and deliver those for which the system can generate better arguments to justify their suitability for the search performed by the student.

To illustrate the operation of our method, in this section we show the results of a validation experiment that we have performed using the LOs stored in the FROAC⁶ repository (the Federation of Learning Objects Repositories of Colombia) [2]. FROAC has 637 LOs indexed, stored in different repositories. The main topics of the LOs stored are: Analysis and design of algorithms and information systems, audit, databases, software engineering, artificial intelligence, programming, natural sciences, social sciences, computing, and mathematics. FROAC was developed at the *Universidad Nacional de Colombia*, as a result of a research project entitled *ROAC, Creación de un modelo para la Federación de OA en Colombia que permita su integración a confederaciones internacionales* of *COL-CIENCIAS*. FROAC also stores information about its user’s profiles (students). For each student, FROAC stores explicit features such as personal information

⁶ FROAC: <http://froac.manizales.unal.edu.co>

(e.g. full name, date of birth, email, gender, and language), LO preferences (language, topic, and format), and psycho-pedagogical information (learning style). The students' learning style is obtained through a test with 24 questions that determine how the student processes the information that he/she receives and turns it into knowledge. The students of the National University of Colombia make an intensive use of FROAC. However, they have difficulties in specifying a query string that meets what they really want to find. Therefore, our ERS was implemented to help those students to find materials to support their learning. Furthermore, students also reported difficulties to understand why the system selects a specific LO over the list of potential candidates as the one that best fits their learning objectives. Thus, we have designed the new argumentation-based hybridization module not only with the objective of improving the quality of recommendations, but also with the aim of being able to provide the students with justifications for those recommendations.

In what follows, we report the results of one of the validation experiments that we performed. We selected a student with an *auditory* learning style (he prefers auditory LOs with formats such as mp3, mp4, avi, etc.), has queried the ERS to find LOs that can help him to improve his *programming* skills (he has used the keyword 'programming'). After retrieving the list of LOs that match this query, the ERS executed its recommendations process and got the following results⁷: the content-based recommendation module delivered the LO with ID *LO262*; the collaborative recommendation module proposed a different LO, with ID *LO269*; and finally, the knowledge-based recommendation module delivered again the LO with ID *LO269*.

The ERS selected from these three proposals the LO that should be more relevant for the student learning objectives. The relevance is understood as the suitability of a LO in view of the student's preferences and profile. Therefore, a LO delivered by our ERS can be considered as 'relevant' if it matches the student's learning objectives (determined by the keywords) and profile (his/her learning style, format, language, and learning time preferences). For this example, the traditional hybridization method that our ERS used to date [8] will select and provide the *LO269* to *Jose*, since it has been recommended by two out of the three recommendation modules.

To evaluate recommendation results according to their relevance for the student, we can use the usual precision formula:

$$Precision = \frac{RelevantLOs \cap RetrievedLOs}{RetrievedLOs} \quad (2)$$

Therefore, according to our relevance definition, we get the following results:

- LO262 Precision = 1 content-based recommendation
- LO269 Precision = 0 collaborative recommendation and knowledge-based recommendation

⁷ For the sake of simplicity, we only provide the top 1 recommendation results of each module.

which shows how the traditional hybridization method failed to deliver the most relevant LO in this case. In fact, although LO269 is *educationally appropriated* (its type fits the user's learning style) and it is *updated* (it has been updated within the last 5 years), it does not meet other user's preferences. It is not *generally appropriated* (its structure is not atomic and its state is not final, which means that it can be a LO under review), not *technically appropriated* (its language and format do not match the preferences of the user), and not *learning time appropriated* (it exceeds the maximum learning time preferred by the user).

Alternatively, our new argumentation-based hybridization method will trigger the following rules⁸ for LO262 and LO269 with their associated probabilities:

CONTENT-BASED RULES

- $C1_{LO262} P=1: recommend(user, LO262) \leftarrow$
 $educationally_appropriate(user, LO262) \wedge generally_appropriate(LO262)$
- $C1_{LO269} P=0.5: recommend(user, LO269) \leftarrow$
 $educationally_appropriate(user, LO269) \wedge generally_appropriate(LO269)$
- $C2_{LO262} P=1: recommend(user, LO262) \leftarrow$
 $educationally_appropriate(user, LO262) \wedge generally_appropriate(LO262) \wedge$
 $technically_appropriate(user, LO262)$
- $C2_{LO269} P=0.33: recommend(user, LO269) \leftarrow$
 $educationally_appropriate(user, LO269) \wedge generally_appropriate(LO269) \wedge$
 $technically_appropriate(user, LO269)$
- $C3_{LO262} P=1: recommend(user, LO262) \leftarrow$
 $educationally_appropriate(user, LO262) \wedge generally_appropriate(LO262) \wedge$
 $updated(LO262)$
- $C3_{LO269} P=0.66: recommend(user, LO269) \leftarrow$
 $educationally_appropriate(user, LO269) \wedge generally_appropriate(LO269) \wedge$
 $updated(LO269)$
- $C4_{LO262} P=1: recommend(user, LO262) \leftarrow$
 $educationally_appropriate(user, LO262) \wedge generally_appropriate(LO262) \wedge$
 $learning_time_appropriate(LO262)$
- $C4_{LO269} P=0.33: recommend(user, LO269) \leftarrow$
 $educationally_appropriate(user, LO269) \wedge generally_appropriate(LO269) \wedge$
 $learning_time_appropriate(LO269)$

COLLABORATIVE RULES

- $O1_{LO262} P=1: recommend(user, LO262) \leftarrow similarity(user, 'juan') > \alpha \wedge vote('juan',$
 $LO262) \geq 4$
- $O1_{LO269} P=1: recommend(user, LO269) \leftarrow similarity(user, 'pablo') > \alpha \wedge$
 $vote('pablo', LO269) \geq 4$

KNOWLEDGE-BASED RULES

- $K1_{LO262} P=1: recommend(user, LO262) \leftarrow similarity(LO262, LO258) > \beta \wedge$
 $vote(user, LO258) \geq 4$

⁸ Only a selection of these rules are presented due to space restrictions.

$K1_{LO269} P=1: recommend(user, LO269) \leftarrow similarity(LO269, LO274) > \beta \wedge vote(user, LO274) \geq 4$

Rule	Explanation	Description
$C1$	$E1$	The learning object LO fits the topic T, is suitable for your LS learning style, and it is atomic and stable.
$C2$	$E2$	The learning object LO fits the topic T, is suitable for your LS learning style, and fits your L language and F format preferences.
$C3$	$E3$	The learning object LO fits the topic T, is suitable for your LS learning style, fits your L language and F format preferences, and it is updated.
$C4$	$E4$	The learning object LO fits the topic T, is suitable for your LS learning style, and fits your L language, F format preferences and learning time $< T$ preferences.
$O1$	$E5$	The system has found a user that whose profile is similar to yours who liked LO
$K1$	$E6$	The system has found that you liked LOx, which is similar to LOy.

Table 1: Explanation schemes.

The collaborative recommendation module was able to find two similar users 'juan' that liked LO262, and 'pablo' that liked LO269, but recommended LO269 since 'pablo' is more similar to the actual user than 'juan'. These inferences are also encoded in rules $O1_{LO262}$ and $O1_{LO269}$. Similarly, the knowledge-based recommendation module was able to find a LO258 similar to LO269 and another LO274 similar to LO269 that were successfully recommended in the past to the actual user, but LO274 received a highest vote, and hence, LO269 was recommended. These inferences are also encoded in rules $K1_{LO262}$ and $K1_{LO269}$. All these requirements were also met by LO262. However, while for LO262 all literals hold and all rules have an associated probability of 1, some literals do not hold for LO269 (those that represent the unfulfilled user preferences encoded in the content-based rules), which decreases the probability associated with their rules.

Therefore, as the new argumentation-based hybridization method is able to generate more arguments to justify the recommendation of LO262, the system would succeed in selecting the most relevant LO for this specific user. Furthermore, we have also designed a module for constructing explanations (arguments) based on these rules. Since the number of rules of our ERS is finite and small, this is a simple module that associates each rule with a scheme of explanation (see table 1).

For instance, with the rule $C1_{LO262}$ the ERS can use the explanation scheme $E1$ and provide the user with an argument to justify the recommendation of LO262: 'The learning object LO262 fits the topic 'Programming', is suitable for your 'auditory' learning style, and it is atomic and stable'.

5 Conclusions and future work

This paper has proposed the employment of an argumentation-based formalism for modeling a hybrid recommender system which recommends LOs for

specific students. In addition, an initial validation using real data from a LO repository of the *Universidad Nacional de Colombia* has been done with better results than previously implemented approaches. The proposed argument-based hybridization method is able to select the most relevant and suitable LOs to recommend, among those delivered previously by three recommendation modules (content-based, collaborative and knowledge-based). Also, by using this method, the recommender system can generate arguments to justify its recommendations. The whole system is still being implemented to be integrated in the Federation of Learning Objects Repositories of Colombia. As future work, we plan to enhance the simple explanation module with an advanced human-computing interaction module integrated in a conversational agent. Also, comprehensive evaluation tests will be performed.

Acknowledgements

This work was partially developed with the aid of the doctoral grant offered to Paula A. Rodríguez by 'Programa Nacional de Formación de Investigadores – COLCIENCIAS', Colombia and partially funded by the COLCIENCIAS project 1119-569-34172 from the Universidad Nacional de Colombia. It was also supported by the projects TIN2012-36586-C03-01 and TIN2014-55206-R of the Spanish government and by the Spanish Ministry of Education, Culture and Sports under the Program for R&D Valorisation and Joint Resources VLC/CAMPUS, as part of the Campus of International Excellence Program (Ref. SP20140788).

References

1. Learning Technology Standards Committee. IEEE Standard for Learning Object Metadata. *Institute of Electrical and Electronics Engineers, New York*, 2002.
2. N.D. Duque, D.A. Ovalle, and J. Moreno. *Objetos de aprendizaje, repositorios y federaciones...conocimiento para todos*. Universidad Nacional de Colombia, 2015.
3. R. Sikka, A. Dhankhar, and C. Rana. A Survey Paper on E-Learning Recommender System. *International Journal of Computer Applications*, 47(9):27–30, June 2012.
4. H. Van de Sompel, R. Chute, and P. Hochstenbach. The adore federation architecture: digital repositories at scale. *International Journal on Digital Libraries*, 9(2):83–100, 2008.
5. N. Manouselis, H. Drachsler, R. Vourikari, H. Hummel, and R. Koper. Recommender Systems in Technology Enhanced Learning. *Recommender Systems Handbook*, 54:479–510, 2011.
6. R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 2002.
7. P.A. Rodríguez, N.D. Duque, and D.A. Ovalle. Multi-agent system for knowledge-based recommendation of learning objects using metadata clustering. In *Highlights of Practical Applications of Agents, Multi-Agent Systems, and Sustainability-The PAAMS Collection*, pages 356–364. Springer, 2015.
8. P.A. Rodríguez, D.A. Ovalle, and N.D. Duque. A student-centered hybrid recommender system to provide relevant learning objects from repositories. In Panayiotis Zaphiris and Andri Ioannou, editors, *Learning and Collaboration Technologies*,

- volume 9192 of *Lecture Notes in Computer Science*, pages 291–300. Springer International Publishing, 2015.
9. S. Ossowski. *Agreement technologies*, volume 8. Springer Science & Business Media, 2012.
 10. M. Salehi, M. Pourzaferani, and S.A. Razavi. Hybrid attribute-based recommender system for learning material using genetic algorithm and a multidimensional information model. *Egyptian Informatics Journal*, January 2013.
 11. A. Zapata, V.H. Menendez, M.E. Prieto, and C. Romero. A hybrid recommender method for learning objects. *IJCA proceedings on design and evaluation of digital content for education (DEDCE)*, 1:1–7, 2011.
 12. V. Vekariya and G.R. Kulkarni. Hybrid recommender systems: Survey and experiments. In *2012 Second International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, pages 469–473. Ieee, May 2012.
 13. J. Palanca, S. Heras, J. Jorge, and V. Julian. Towards persuasive social recommendation: Knowledge model. *SIGAPP Appl. Comput. Rev.*, 15(2):41–49, August 2015.
 14. G. Linden, J. Hong, M. Stonebraker, and M. Guzdial. Recommendation algorithms, online privacy, and more. *Communications of the ACM*, 52(5):10–11, May 2009.
 15. R. Sinha and K. Swearingen. The role of transparency in recommender systems. In *CHI'02 extended abstracts on Human factors in computing systems*, pages 830–831. ACM, 2002.
 16. C.E. Briguez, M. Budán, C. Deagustini, A.G. Maguitman, M. Capobianco, and G.R. Simari. Towards an argument-based music recommender system. *COMMA*, 245:83–90, 2012.
 17. C.E. Briguez, M. Capobianco, and A.G. Maguitman. A theoretical framework for trust-based news recommender systems and its implementation using defeasible argumentation. *International Journal on Artificial Intelligence Tools*, 22(04):1350021, 2013.
 18. J.A. Recio-García, L. Quijano, and B. Díaz-Agudo. Including social factors in an argumentative model for group decision support systems. *Decision Support Systems*, 56:48–55, 2013.
 19. S. Heras, M. Rebollo, and V. Julián. A dialogue game protocol for recommendation in social networks. In *Hybrid Artificial Intelligence Systems*, pages 515–522. Springer, 2008.
 20. C.I. Chesñevar, A.G. Maguitman, and M.P. González. Empowering recommendation technologies through argumentation. In *Argumentation in artificial intelligence*, pages 403–422. Springer, 2009.
 21. S. Heras, K. Atkinson, V. Botti, F. Grasso, V. Julián, and P. McBurney. Research opportunities for argumentation in social networks. *Artificial Intelligence Review*, 39(1):39–62, 2013.
 22. C.E. Briguez, M. Budán, C. Deagustini, A.G. Maguitman, M. Capobianco, and G.R. Simari. Argument-based mixed recommenders and their application to movie suggestion. *Expert Systems with Applications*, 41(14):6467–6482, 2014.
 23. P.A. Kirschner, S.J. Buckingham-Shum, and C.S. Carr. *Visualizing argumentation: Software tools for collaborative and educational sense-making*. Springer Science & Business Media, 2012.
 24. A.J. García and G.R. Simari. Defeasible logic programming: An argumentative approach. *Theory and practice of logic programming*, 4(1+ 2):95–138, 2004.
 25. H. Li, N. Oren, and T.J. Norman. Probabilistic argumentation frameworks. In *Theorie and Applications of Formal Argumentation*, pages 1–16. Springer, 2012.