Evaluation of Requirements Collection Strategies for a Constraint-based Recommender System in a Social e-Learning Platform

Francesco Epifania[1,2,3] and Riccardo Porrini[3]

[1]*Department of Computer Science, University of Milano, Via Comelico 39, Milano, Italy*

[2]*Social Things s.r.l., Via De Rolandi 1, Milano, Italy*

[3]*Department of Informatics, Systems and Communication, University of Milano-Bicocca, Viale Sarca 336, Milano, Italy*

Keywords: Recommender System, Learning Resources, Social Network, e-Learning, User-centric Evaluation.

Abstract: The NETT Recommender System (NETT-RS) is a constraint-based recommender system that recommends learning resources to teachers who want to design courses. As for many state-of-the-art constraint-based recommender systems, the NETT-RS bases its recommendation process on the collection of requirements to which items must adhere in order to be recommended. In this paper we study the effects of two different requirement collection strategies on the perceived overall recommendation quality of the NETT-RS. In the first strategy users are not allowed to refine and change the requirements once chosen, while in the second strategy the system allows the users to modify the requirements (we refer to this strategy as backtracking). We run the study following the well established ResQue methodology for user-centric evaluation of RS. Our experimental results indicate that backtracking has a strong positive impact on the perceived recommendation quality of the NETT-RS.

1 INTRODUCTION

Recommender Systems (RSs) are information filtering algorithms that generate meaningful recommendations to a set of users over a collection of items that might be of their interest (Jannach et al., 2010). In its basic incarnation, a RS takes in input a user profile and possibly some situational context and computes a ranking over a collection of recommendable items (Adomavicius and Tuzhilin, 2005). The user profile can possibly include explicit information, such as feedback or ratings of items and/or implicit information, such as items visited and time spent on them. RSs leverage this information to predict the relevance score for a given, typically unseen, item.

RS have been adopted in many disparate fields, ranging from movies, music, books, to financial services and live insurances (Jannach et al., 2010). In the e- Learning context, the NETT Recommender System (NETT-RS) (Mesiti et al., 2014) is a RS that recommends learning resources (e.g., slides, tutorials, papers etc.) to teachers who want to design a course. The NETT-RS is a component of the NETT platform, one of the main outcomes of the NETT European project.

The NETT project aims at gathering a networked social community of teachers to improve the entrepreneurship teaching in the European educational system. Among the other things, the platform allows teachers to design courses. The NETT-RS supports the teachers in the design of courses by recommending adequate and high quality learning resources (resources, for brevity). In order to finalize the design, teachers go through three sequential steps: they specify (1) a set of rules and (2) keywords for a course (e.g., required skill = statistics) and (3) the system recommends a set of resources such that they fit the rules and the keywords specified by the teacher (e.g., no differential calculus for a basic math course) and have an high rating.

The characteristics of a NETT-RS closely match the ones proper of constraint-based RS (Felfernig et al., 2011) as the teacher specifies a set of requirements (in the form of rules and keywords) to which resources must adhere in order to be recommended. The multi-phased process allows the teacher to incrementally explore the resource space in order to find the most suitable ones for her/his course, in the vein of conversational RS (Pu et al., 2011b; Chen and Pu, 2012). However, this interaction with the user required by the NETT-RS entails several challenges. The teacher must be put within an interactive loop

Epifania, F. and Porrini, R.
Evaluation of Requirements Collection Strategies for a Constraint-based Recommender System in a Social e-Learning Platform.
In *Proceedings of the 8th International Conference on Computer Supported Education (CSEDU 2016) - Volume 1*, pages 376-382
ISBN: 978-989-758-179-3

with the system, with the possibility to revise the rules and keywords previously specified. We refer to this feature as backtracking.

In this paper, we study the effect of the backtracking feature on the NETT-RS. We argue that providing a backtracking feature to the NETT-RS strongly influences the perceived recommendation quality. In order to answer this research question, we set up a user-centric evaluation of the NETT-RS following the ResQue methodology (Pu et al., 2011a). We compare two versions of the NETT-RS (with and without backtracking) over many different user- centric quality dimensions. Evidence gathered from this study substantiates our intuitions: the presence of backtracking has a strong impact on many different quality measures, such as control, perceived ease of use and overall satisfaction.

The reminder of the paper is organized as follows. In Section 2 we sketch the main components of the NETT-RS. In Section 3 we describe the user study that we conducted and discuss the results. We compare the NETT-RS with related work in Section 4 and end the paper with conclusions and highlight future work in Section 5.

2 THE RECOMMENDER SYSTEM

The NETT-RS recommendation process consists of three sequential steps: rule induction, keyword extraction and resource selection. In the rest of the Section we describe how items (i.e., learning resources) are represented within the NETT- RS, along with a sketch of the three phases. We also highlight one of the main issues that underlies this multi-step process: the need for backtracking.

2.1 Learning Resources

Learning resource (resource for brevity), which are suggested by using a set of metadata that adhere to the Learning Object Metadata standard (LOM). These metadata characterize resources in terms of, for example, format (e.g., text, slide etc.) or language (e.g., Italian, English etc.). More formally, resources are characterized by a fixed set of n metadata $\mu_1, \ldots, \mu_n$, which can be qualitative (nominal/ordinal) or quantitative (continuous/discrete), where the latter are suitably normalized in $[0,1]$. Table 1 presents some example metadata used within the NETT-RS. The resources are also characterized by a particular metadata: the keywords. The keywords ideally describe the topics that a resource is about. Each resource has a

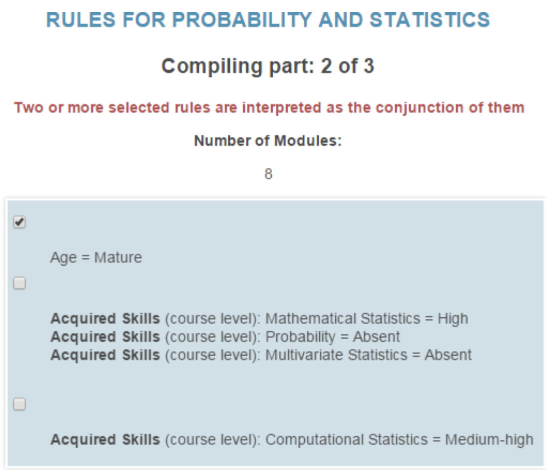

Figure 1: The rule selection step.

textual content π (e.g., the text extracted from a slide). Each resource is affected by a rating p, typically normalized in $[0,1]$.

2.2 Rule Induction

As a first step, the teacher is asked to select a set of constraints (i.e., *rules*) over the learning metadata. Those rules are computed automatically by the system leveraging a well known rule induction algorithm, as explained later on in this Section. An example of rules selection is depicted in Figure 1. Rules, which should in principle accurately describe the resources available in the system, are encoded as Horn clauses made of some antecedents and one consequent. The consequent is fixed: "π is good" (i.e., the content of a resource is good). The antecedents are Boolean conditions c_j (true/false) concerning sentences of two kinds: (1) "$\mu_i \lesseqgtr \theta$", where θ stands for any symbolic (for nominal metadata) or numeric constant (for quantitative variables) and (2) "$\mu_i \in A$", with A a suitable set of constants associated with qualitative metadata. A rule is hence formally defined as $c_1, \ldots, c_k \rightarrow \pi$ is good.

We may obtain these rules starting from one of the many algorithms generating decision trees dividing good from bad items, where the difference between the various methods stands in the entropic criteria and the stopping rules adopted to obtain a tree, and in the further pruning heuristics used to derive rules that are limited in number, short in length (number of antecedents), and efficient as for classification errors. In articular we use RIPPERk, a variant of the Incremental Reduced Error Pruning (IREP) proposed by Cohen (Cohen, 1995) to reduce the error rate, guaranteeing in the meanwhile a high efficiency on large samples, and in particular its Java version

Table 1: An excerpt of metadata that characterize a resource.

Metadata	Type	Values
Learning Resource Type	qualitative	Diagram, Figure, Graph, Index, Slides, Table, Narrative Text, Lecture, Exercise, Simulation, Questionnaire, Exam, Experiment, Problem Statement, Self Assessment
Format	qualitative	Video, Images, Slide, Text, Audio
Language	qualitative	English, Italian, Bulgarian, Turkish
Keywords	qualitative	entrepreneurship, negotiation, ...
Typical Learning Time	quantitative	30 minutes, 60 minutes, 90 minutes, +120 minutes

Table 2: A set of two candidate rules.

Id	Rule
R_1	*skill_required* Communication Skill in Marketing Information Management = low **and** *language* = Italian ⇒ *good_course*
R_2	*skill_acquired* Communication Skill in Marketing Information Management = medium-high **and** *skill_acquired* Communication Skill in Communications Basic = high **and** *age* = teenager-adult ⇒ *good_course*

JRip available in the WEKA environment (Hall et al., 2009). This choice was mainly addressed by computational complexity reasons, as we move from the cubic complexity in the number of items of the well known C4.5 (Quinlan, 1993) to the linear complexity of JRip. Rather, the distinguishing feature of our method is the use of these rules: not to exploit the classification results, rather to be used as hyper- metadata of the questioned items. In our favorite application field, the user, in search of didactic material for assembling a course on a given topic, will face rules like those reported in Table 2. Then, it is up to her/him to decide which rules characterize the material s/he's searching for.

2.3 Keywords Extraction and Resource Selection

As a second step, the system presents the teacher with a subset of the keywords extracted from the metadata of resources that satisfy the rules selected during the rule induction phase. An example of keywords is depicted in Figure 2.

In fact, even after applying the filtering capability provided by the selected rules, the number of resources that are to be suggested can still be very high. Thus, a meaningful subset of the keywords is presented to the teacher. The NETT-RS looks for the best subset of keywords in terms of the ones providing the highest entropy partition of the resource set selected by the rules Figure 4. With this strategy, the number of selected resources is guaranteed to reduce uniformly at an exponential rate for whatever keyword subset chosen by the teacher.

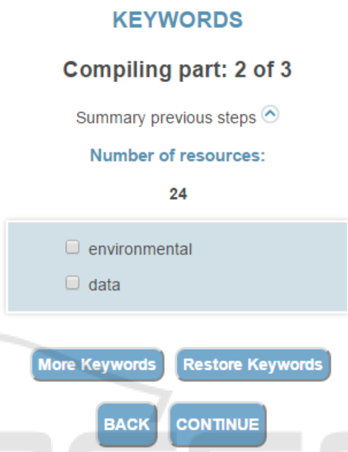

Figure 2: The keywords selection step.

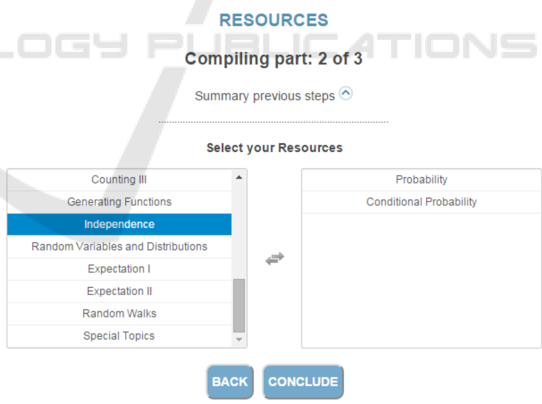

Figure 3: The resource selection step.

As the final step, the NETT-RS recommends a set the resources such that: (1) they satisfy the rules and (2) they are annotated with the selected keywords. The teacher then finalizes the design of the course by selecting the resources considered suitable. An example of suggested resources is depicted in Figure 3.

3 USER EXPERIENCE EVALUATION OF THE RECOMMENDER SYSTEM

3.1 The Backtracking Feature

The NETT-RS requires the teacher to go through all the three steps described above in order to finalize the design of a course. During each step the system provides the teacher with a set of automatically selected items, namely: rules, keywords or resources. The strong assumption we make on such a process is that the choices made by the teacher in one phase can potentially affect the result of the subsequent phases. For this reason we argue that allowing the teacher to go back and forth the phases, and possibly revising the selections, has a strong impact on the perceived quality of the resource suggestion in the final step (Figure 3). The need of such a backtracking feature was furthermore observed by alpha testers of the NETT-RS, which initially were not equipped with such feature.

3.2 Evaluating the Backtracking Feature

From the user interaction point of view we argue that the backtracking feature has a high impact on the overall perceived quality of the NETT-RS. We substantiate this claim with empirical evidence gathered from a user-centric evaluation of the NETT- RS. The remainder of this Section describes the experiment we conducted, starting from the research question and hypotheses, the experimental setting, and ending with the discussion of the experimental results.

3.3 Research Question and Hypotheses

Our research question is rather simple and pragmatic: Does providing a backtracking feature to teachers affect the perceived quality of the recommendation of the NETT System?

In order to provide an answer to this research question, we evaluate the NETT-RS and formulate the two following hypotheses:

H1: the possibility to revise the choices made during the course design process increases the perceived user control over the NETT-RS.

H2: the possibility to revise the choices made during the course design process increases the perceived overall quality of the NETT-RS.

The hypothesis **H1** focuses on a specific quality of the NETT-RS (i.e., the user control over the recommendation process), which is only one of the possible dimensions that contribute to the perceived overall quality of system (**H2**).

3.4 Experimental Design

Two versions of the NETT-RS were evaluated: the first one without the backtracking feature enabled (i.e., NETT-RS) and the second one with backtracking (i.e., NETT-RS-b). As for testing our hypotheses, we adopted the ResQue methodology (Pu et al., 2011a), which is a well-established technique for the user-centric evaluation of RSs. We selected 40 participants, mainly university professors, and asked them to design a course on Probability and Statistics, choosing from 1170 different learning resources. We selected such resources from the MIT Open CourseWare website. The participants were equally partitioned into two disjoint subsets (20 + 20). Participants from the first subset were asked to design a course using NETT-RS, while participants from the second subset used NETT-RS-b. Finally, participants were presented with a questionnaire (Table 3)

3.5 The Adapted ResQue Questionnaire

The ResQue questionnaire (Pu et al., 2011a) defines a wide set of user-centric quality metrics to evaluate the perceived qualities of RSs and to predict users' behavioral intentions as a result of these evaluations. The original version of the questionnaire included 43 questions, evaluating 15 different qualities, such as recommendation accuracy or control. Participants' responses to each question are characterized by using a 5-point Likert scale from strongly disagree (1) to strongly agree (5). Two versions of the questionnaire have been proposed (Pu, et al., 2011): a longer version (43 questions) and a shorter version (15 questions). In our study we adopted the short version in order to reduce the cognitive load required to participants. A modified version of the questionnaire, tailored for a system that recommends learning resources, was presented to the participants (Table 3).

3.6 Experimental Results and Discussion

Table 4 reports the mean grades for all the issued questions. We got a Cronbach's α (Peterson, 1994) equal to 0.919 and 0.887 for grades given by participants who evaluated the NETT-RS and the NETT-RS-b, respectively. Thus, we consider the questioned

Table 3: The adapted version of the ResQue questionnaire used in our study.

	Quality	Question
Q1	*recommendation accuracy*	The teaching material recommended to me match my interests
Q2	*recommendation novelty*	The recommender system helped me discover new teaching material
Q3	*recommendation diversity*	The items recommended to me show a great variety of options
Q4	*interface adequacy*	The layout and labels of the recommender interface are adequate
Q5	*explanation*	The recommender explains why the single teaching materials are recommended to me
Q6	*information sufficiency*	The information provided for the recommended teaching material is sufficient for me to take a decision
Q7	*interaction adequacy*	I found it easy to tell the system what I like/dislike
Q8	*perceived ease of use*	I became familiar with the recommender system very quickly
Q9	*control*	I feel in control of modifying my requests
Q10	*transparency*	I understood why the learning material was recommended to me
Q11	*perceived usefulness*	The recommender helped me find the ideal learning material
Q12	*overall satisfaction*	Overall, I am satisfied with the recommender
Q13	*confidence and trust*	The recommender can be trusted
Q14	*use intentions*	I will use this recommender again
Q15	*purchase intention*	I would adopt the learning materials recommended, given the opportunity

participants to be reliable. NETT-RS-b achieves the most noticeable result on the control quality (Q9) showing that the presence of the backtracking lifts the mean judgment up from 1.30 to 4.45 (342% of improvement). The difference is significant with a p-value < 0.0001, providing strong experimental evidence for the hypothesis H1: the possibility to revise the choices made during the course design process increases the perceived user control over the NETT-RS. As far as the overall quality is concerned (hypothesis H2) we observe strong significant improvements ($p < 0.0001$) in the perceived ease of use, perceived usefulness, overall satisfaction, confidence and trust, use intentions and purchase intention qualities. This evidence allows us to correlate the presence of the backtracking feature with a higher perceived overall quality of the NETT-RS in terms of the above features.

The presence of the backtracking feature does not lead to a significant improvement of the recommendation accuracy. However, we observe significant improvements ($p \approx 0.012$ and $p \approx 0.002$) on recommendation novelty and recommendation diversity. Our interpretation is that enabling the users to go back and forth the steps allows them to better explore the resource space, thus leading to novel and diverse recommendations.

Finally, we observe that the presence of the backtracking feature has no significant impact on the interface adequacy, explanation and transparency qualities. We furthermore observe that participants assigned a relatively low grade, especially for the interface adequacy. Such results may come from the difficulty to understand the meaning of rules presented by the NETT-RS. We consider it as a stimulus for a future improvement of the system.

Table 4: Mean grades to questionnaire's questions. p-values are computed by means of a two-tailed t-test. Statistically significant improvements are marked in bold.

	Quality	NETT-RS	NETT-RS-b	p-value
Q1	*recommendation accuracy*	3.80	3.95	0.481
Q2	*recommendation novelty*	3.50	**4.05**	0.012
Q3	*recommendation diversity*	3.50	**4.10**	0.002
Q4	*interface adequacy*	2.90	3.30	0.088
Q5	*explanation*	3.40	3.60	0.162
Q6	*information sufficiency*	3.35	**4.25**	< 0.0006
Q7	*interaction adequacy*	3.10	**3.60**	< 0.002
Q8	*perceived ease of use*	3.45	**4.60**	< 0.0001
Q9	*control*	1.30	**4.45**	< 0.0001
Q10	*transparency*	3.45	3.75	0.110
Q11	*perceived usefulness*	3.00	**4.00**	< 0.0004
Q12	*overall satisfaction*	2.80	**3.90**	< 0.0001
Q13	*confidence and trust*	3.15	**3.80**	< 0.001
Q14	*use intentions*	2.70	**3.70**	< 0.0001
Q15	*purchase intention*	3.30	**4.10**	< 0.0001

4 RELATED WORK

A widely accepted classification of RSs divides them into four main families (Jannach et al., 2010): *content-based* (CB), *collaborative filtering* (CF), *knowledge-based* (KB) and *hybrid*. The basic idea behind CB RSs is to recommend items that are similar to those that the user liked in the past (see e.g., (Balabanovic and Shoham, 1997; Pazzani and Billsus, 1997; Mooney and Roy, 2000)). CF RSs recommend items based on the past ratings of all users collectively (see e.g., (Resnick et al., 1994; Sarwar et al., 2001; Lemire and Maclachlan, 2005)). KB RSs suggest items based on inferences about users' needs: domain knowledge is modeled and leveraged during the recommendation process (see e.g., (Burke, 2000; Felfernig and Burke, 2008; Felfernig and Kiener, 2005)). Hybrid RSs usually combine two or more recommendation strategies together in order to leverage the strengths of them in a principled way (see e.g., (de Campos et al., 2010; Shinde and Kulkarni, 2012; Ren et al., 2008)).

The NETT-RS falls into the KB RSs family, and more precisely into the *constraint-based* category. For a more exhaustive and complete description of constraint-based RSs we point the reader to (Felfernig et al., 2011). The typical features of such RSs are: (1) the presence of a *knowledge base* which models both the items to be recommended and the explicit rules about how to relate user requirements to items, (2) the collection of user requirements, (3) the repairment of possibly inconsistent requirements, and (4) the explanation of recommendation results.

We recall from Section 2 that learning resources in the NETT-RS are characterized by metadata. This characterization provides the basic building block for the construction of a knowledge base (e.g., using Semantic Web practices tailored for the education domain (Dietze et al., 2013)). As for the collection of user requirements, the NETT-RS collects them during the rule and keywords selection phases. The NETT-RS does not provide any kind of repairment for inconsistent requirements (i.e., rules and keywords), in contrast with most state-of-the-art constraint-based RSs (Felfernig and Kiener, 2005; Felfernig and Burke, 2008; Felfernig et al., 2009). However, we notice that the interaction that the NETT-RS requires to the teachers is different: rules and keywords are not directly specified. Instead, teachers specify the requirements by choosing from a suggested set of available rules and keywords, ensuring the specification of consistent requirements only. Finally, the NETT-RS currently does not provide any explanation of recommendation results. However, as pointed out by our experiments in Section 3, the system would benefit from the application of such explanation techniques (Friedrich and Zanker, 2011).

In KB RSs literature, special attention has been devoted to requirements collection, being it a mandatory prerequisite for recommendations to be made (Felfernig et al., 2011). Requirements can be collected using different strategies, each one leading to different interaction mechanisms with the user. Such mechanisms can be relatively simple as *static fill-out* forms filled each time a user accesses the RS, but also more sophisticated like the *interactive conversational dialogs*, where the user specifies and refines the requirements incrementally by interacting with the system (Pu et al., 2011b; Chen and Pu, 2012). The backtracking feature added to the NETT-RS goes exactly towards this direction.

5 CONCLUSION AND FUTURE WORK

We conducted a user-centric evaluation of the constraint-based NETT-RS, a RS that recommends resources to teachers who want to design a course. Our goal was to study the effect on the overall perceived recommendation quality of a backtracking feature, that is to give the possibility to teachers to revise the constraints (i.e., rules and keywords) over the resources specified within the recommendation process. Our study reveals a strong correlation between the presence of the backtracking feature and an higher perceived quality.

We foresee at least two main future lines of work. From the experimental point of view, we would like to run the experiment on learning resources from different domains and include more participants. From the point of view of the NETT-RS itself, we plan to take advantage of the insights that we got from this user study and include in the system also the explanation of the recommendation results inspired by related work in this area (Felfernig and Kiener, 2005; Felfernig and Burke, 2008; Felfernig et al., 2009).

REFERENCES

Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, 17(6):734–749.

Balabanovic, M. and Shoham, Y. (1997). Content-based, collaborative recommendation. *Commun. ACM*, 40(3):66–72.

Burke, R. (2000). Knowledge-based recommender systems. *Encyclopedia of Library and Information Systems*, vol. 69, Supplement 32.

Chen, L. and Pu, P. (2012). Critiquing-based recommenders: survey and emerging trends. *User Model. User-Adapt. Interact.*, 22(1-2):125–150.

Cohen, W. W. (1995). Fast effective rule induction. In *ICML*, pages 115–123.

de Campos, L. M., Fernández-Luna, J. M., Huete, J. F., and Rueda-Morales, M. A. (2010). Combining content-based and collaborative recommendations: A hybrid approach based on bayesian networks. *Int. J. Approx. Reasoning*, 51(7):785–799.

Dietze, S., SanchezAlonso, S., Ebner, H., Yu, H. Q., Giordano, D., Marenzi, I., and Nunes, B. P. (2013). Interlinking educational resources and the web of data: A survey of challenges and approaches. *Program*, 47(1):60–91.

Felfernig, A. and Burke, R. (2008). Constraint-based recommender systems: Technologies and research issues. In *ICEC*, pages 3:1–3:10.

Felfernig, A., Friedrich, G., Isak, K., Shchekotykhin, K. M., Teppan, E., and Jannach, D. (2009). Automated debugging of recommender user interface descriptions. *Appl. Intell.*, 31(1):1–14.

Felfernig, A., Friedrich, G., Jannach, D., and Zanker, M. (2011). Developing constraint-based recommenders. In *Recommender Systems Handbook*, pages 187–215. Springer.

Felfernig, A. and Kiener, A. (2005). Knowledge-based interactive selling of financial services with fsadvisor. In *AAAI*, pages 1475–1482.

Friedrich, G. and Zanker, M. (2011). A taxonomy for generating explanations in recommender systems. *AI Mag.*, 32(3):90–98.

Hall, M. A., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: an update. *SIGKDD Expl.*, 11(1):10–18.

Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender Systems - An Introduction*. Cambridge University Press.

Lemire, D. and Maclachlan, A. (2005). Slope one predictors for online rating-based collaborative filtering. In *SDM*, pages 471–475.

Mesiti, M., Valtolina, S., Bassis, S., Epifania, F., and Apolloni, B. (2014). e-teaching assistant - A social intelligent platform supporting teachers in the collaborative creation of courses. In *CSEDU*, pages 569–575.

Mooney, R. J. and Roy, L. (2000). Content-based book recommending using learning for text categorization. In *DL*, pages 195–204.

Pazzani, M. J. and Billsus, D. (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27(3):313–331.

Peterson, R. A. (1994). A meta-analysis of cronbach's coefficient alpha. *Journal of Consumer Research*, 21(2):381–91.

Pu, P., Chen, L., and Hu, R. (2011a). A user-centric evaluation framework for recommender systems. In *RecSys*, pages 157–164.

Pu, P., Faltings, B., Chen, L., Zhang, J., and Viappiani, P. (2011b). Usability guidelines for product recommenders based on example critiquing research. In *Recommender Systems Handbook*, pages 511–545. Springer.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann.

Ren, L., He, L., Gu, J., Xia, W., and Wu, F. (2008). A hybrid recommender approach based on widrow-hoff learning. In *FGCN (1)*, pages 40–45.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In *CSCW*, pages 175–186.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *WWW*, pages 285–295.

Shinde, S. K. and Kulkarni, U. (2012). Hybrid personalized recommender system using centering-bunching based clustering algorithm. *Expert Syst. Appl.*, 39(1):1381–1387.