An Adaptive Learning Path Builder based on a Context Aware Recommender System

Marianna Carbone
PAA
Accademia delle Belle Arti Napoli
Napoli, Italy
mariannacarbone@abana.it

Francesco Colace
DIIn
Università degli Studi di Salerno
Fisciano (SA), Italy
fcolace@unisa.it

Marco Lombardi
DIIn
Università degli Studi di Salerno
Fisciano (SA), Italy
malombardi@unisa.it

Francesco Marongiu
DI
Università degli Studi di Salerno
Fisciano (SA), Italy
f.marongiu@studenti.unisa.it

Domenico Santaniello
DIIn
Università degli Studi di Salerno
Fisciano (SA), Italy
dsantaniello@unisa.it

Carmine Valentino
DIPMAT
Università degli Studi di Salerno
Fisciano (SA), Italy
cvalentino@unisa.it

***Abstract*—The world of distance education is constantly expanding, enriching itself with tools and services to increase the ability to provide training content. Due to the new technologies, the training paths take on new appealing features; however, it remains complex to suggest the appropriate training path to the right student. In this scenario, the use of Recommender Systems (RSs) could be helpful. RSs could allow recommending personalized learning paths to students in order to improve their abilities and their knowledge. In particular, among Recommender Systems, some of them consider contextual information. This paper aims to describe a new approach that suggests learning paths to users taking advantage of recommendation techniques and introducing them through multimedia content. Moreover, the proposed approach aims to provide recommendations when ratings are unknown through the knowledge of profiles of users and items. The proposed approach has been tested through students of two courses with diverse characteristics.**

***Index Terms*—e-learning, Context Aware, Recommender System.**

I. Introduction

The world of distance learning is continually enriched with new tools and services to increase its ability to convey training content. One of the most interesting tools is the one related to the ability to provide students with training paths adapted to their training needs [1]. An approach of this kind is particularly significant in the case of training programs that include a strong exercise and laboratory component. This objective is achieved through the adoption of the so-called Recommender Systems [2]–[5].

Recommender Systems are applied in different sectors but have one goal: to help people make choices based on an analysis of their behavior or of that of users similar in terms of characteristics or interests. In other words, these systems are widely used to analyze and filter information in support of an individual's choice of a service or a specific object. The field of use of Recommender Systems is very varied. Its use is known above all in e-commerce, in streaming services and dissemination sites, or in cases where many services or objects are made available. It is because only a part of them is interesting or relevant to the user. Among the main RS types, there are content-based systems [6]–[8], collaborative systems [9]–[11] and hybrid systems [12]. Added to these are also advanced recommendation services that manage and use the entire context [13] in which a user is located: Recommender Systems based on context-aware technologies [14]–[18].

One of the main characteristics of Recommender Systems is to predict the consideration that an individual may have about an item that has not yet been evaluated. How this forecast is made is one of the distinguishing criteria for the RS.

The main elements on which a Recommender System operates are user, item, and transaction [2].

The user represents the recommender phase's target, which can be identified through its needs and characteristics. The item represents an element that Recommender System suggests to the user. It can be classified according to its features. In Recommender Systems, it is fundamental to understand how an item feature influences its usefulness for a given user. The transaction represents the interaction between the system and the user. In particular, during the system's use, useful information is stored to generate recommendations.

In this paper, we introduce a Recommender System based on a Context-Aware Approach able to support students by selecting exercise paths to be carried out in the lab or remotely. The path is chosen dynamically based on the results obtained and the characteristics of the students. The Recommender System, moreover, is able, on teacher's solicitation and based on the actions put in place by the students, to continuously adapt the exercise path through the proposal of different combinations of exercises. The proposed approach has been introduced in two courses with very different training characteristics: Fundamentals of Informatics held at the University of Salerno and Costume Design held at the Academy of Fine Arts in Naples. The results in both cases have been very interesting.

The structure of the paper is the follow: in Section 2 Context

Aware Recommender Systems of the state of Art are discussed, in Section 3 the proposed approach is described, in Section 4 the experimental phase is discussed and in Section 5 there are conclusions and future works.

II. Related works

Context Aware Recommender Systems (CARS) allow to introduce contextual information in order to improve provided suggestions. There are three paradigms to introduce context in a Recommender System [15]: Contextual Pre-Filtering that introduce contextual information before the recommendation phase, Contextual Post-Filtering that introduce contextual information after the recommendation phase and Contextual Modeling that include contextual information inside the recommendation phase.

In Collaborative Filtering Recommender Systems, Contextual Pre-Filtering approach allow to select known ratings before ratings forecasts calculation. An example of Contextual Pre-Filtering method is provided by Item -Splitting [19] that works on rating matrix. The contextual information allow to split matrix columns that refer to items. After the split phase the Recommendation one can be performed in order to obtain ratings forecasts.

In Content-Based Recommender Systems, Contextual Pre-Filtering method allow to evaluate contextual information through users and items profiles. An example of Contextual Pre-Filtering in this field is provided by Looker [20] where contextual information provides relevant items to a Content-Based Recommender System based on Kullback-Leibler divergence.

Contextual Modeling method in Collaborative Filtering methods introduces contextual information in the calculation of ratings forecasts. Indeed Context Aware Matrix Factorization [21] exploits factorization of ratings matrix and the ability to introduce contextual information through item bias. The rating forecast is obtained as in the following formula:

$$\hat{r}_{u,i,c} = \bar{r}_i + p_u q_i + b_u + b_{i,c} \tag{II.1}$$

where $\bar{r}$ is the rating mean of item i, $p_u q_i$ is obtained though known rating matrix factorization, b_u is the user bias and $b_{i,c}$ is the bias of item i in the context c.

Contextual Modeling approach in Content-Based Recommender Systems includes contextual information in similarity evaluation between users and items. Indeed Shin et al [22] provide a Content-Based method that integrates contextual information in vectors of items and users features in order to obtain contextualized suggestions. Through the contextual information evaluation this method allow to obtain better suggestions than ones obtained with Recommender Systems without integration of context.

Post-Filtering approach exploits contextual information in order to select the best ratings forecast provided by Recommendation phase. An example of Post-Filtering approach to introduce contextual information is provided by Xu at al [23] that exploit time and location contexts in order to improve ratings forecasts obtained through a Memory-Based Collaborative Filtering method [2].

III. The proposed approaches

In this section the developed Hybrid Recommender Systems is presented. This method exploits contextual information with Contextual Modelling and Contextual Post-Filtering methods. The Contextual Modelling approach can be used if there are enough known ratings in the system. The aim is to provide training programs to students in order to improve their knowledge.

The proposed approach can be divided in three phases: Information acquisition phase, Ratings Forecast phase and Execute phase.

The contextual information evaluated in this architecture are:

- Day of the Week: Monday, Tuesday, Wednesday, Thursday, Friday, Weekend. In Weekdays the system consider the course hours that students have in the specific day in order not to overload them.
- Location: Home, University, Others. The architecture use this information in order to evaluate the grade of attention that students can provide based on location information.
- Time that students want to engage in order to training themselves.

The details of the architecture are follow.

A. Information acquisition Phase

The Information Acquisition Phase aims to gather information about students in order to create their profiles. When students make the first access into the system a questionnaire is proposed in order to acquire information about their knowledge. In particular, the students evaluate the knowledge by themselves through a rating which value is among 1 and 10. This rating is given in each field of knowledge supported by the system. Moreover, if the student provides his consent, personal data are also acquired through social networks. Discretization of the data is done by Data Management Module which create numerical vectors that acts as features profiles of students. These discretized profiles are memorized in the Knowledge Base Module.

Items Profiles are generated by professors of the courses that provide exercise labeled according to difficulty and topics. Data Management Module aims to discretize data provided and create numerical vectors that describe items features. These items profiles are memorized in the Knowledge Based Module.

The numerical profiles created are modified in order to obtain vectors that have unitary euclidean norm. Let U the users set and I the item set, the user profile $u \in U$ and the item profile $i \in I$ have the property described by (III.2)

$$\begin{aligned} \|u\|_2 = 1 \quad \forall u \in U \\ \|i\|_2 = 1 \quad \forall i \in I \end{aligned} \tag{III.2}$$

Others properties of the profiles created by Data Management Module are:

- users and items profiles have same dimension n fixed in advance.

$$\begin{aligned} u \in \mathcal{R}^n \quad \forall u \in U \\ i \in \mathcal{R}^n \quad \forall i \in I \end{aligned} \tag{III.3}$$

- users and items profiles have non negative elements.

$$\begin{aligned} u_j \geq 0 \quad \forall j = 1, \dots, n \quad \forall u \in U \\ i_j \geq 0 \quad \forall j = 1, \dots, n \quad \forall i \in I \end{aligned} \tag{III.4}$$

Data Management Module also works in order to cluster the users set U based on similarity metrics. In this way students that have similar knowledge and features can be place in the same group. Indeed Data Management Module creates the sets $U_1, \dots, U_z$ described in (III.5).

$$U_j = \{u \in U : \cos(u, u_j) \geq toll_{user}\} \quad j = 1, \dots, z \tag{III.5}$$

The profile u_j is the representative of the set U_j which contains users that have profiles similar to u_j. The similarity is calculated through the cosine formula (III.6):

$$cos(x, y) = \frac{< x, y >}{\|x\|_2 \|y\|_2} \tag{III.6}$$

where $< x, y >$ is the standard scalar product. The memorization of profiles with unitary euclidean norm allows to simplify the calculation (III.6) because the denominator is equal to 1.

If the profile of a new user is not part of one of the sets $U_1, \dots, U_z$, a new set U_{z+1} will be created and the new user will be the representative of the set. Moreover regularly the cardinality of these sets is evaluated. If the cardinality is too small the tollerance $toll_{user}$ will be reduced in order to avoid sparsity problem. Instead if the cardinality is too big, the tollerance will be increased in order to avoid a great computational cost to calculation of ratings forecasts. In both cases the representative of sets is the user that validates (III.5).

In the Knowledge Base Module there is also the tensor $\mathcal{R} \in \mathbb{R}^{|U| \times |I| \times k \times l}$where the indices $|U|$,$|I|$ are associated to cardinality of sets of users and items, the index k is associated to "Day of the Week" context and the index l is associated to location context.

B. Ratings Forecast Phase

In this phase the calculation of ratings forecast is executed and it is composed by three modules: Data Module, RS Module e CARS Module.

The Data Module identifies the specific user u and acquires the U_j in which user u is by Knowledge Base Module.

Through the Context Dimension Tree (CDT) [24], [25] the specific contexts are determined. In this way the CDT provides to Data Module the indices k_c and l_c that have to be selected in the tensor $\mathcal{R}$ and provides the known ratings in the selected contexts.

At this point the following checks are done:

- the check on cardinality of N_j is done. If there is only the user u in the set the check will fail;
- the check on known ratings is done. If $\mathcal{R}_{p,i,k_c,l_c} = 0, \ \forall p \in N_j, \ \forall i \in I$ the check will fail;

If one of checks fails the Data Module sends user profile u e items profiles to RS Module. In this case the ratings forecasts are calculated through the Cosine Similarity formula [2]:

$$\hat{r}_{ij} = cos(u, i) = \frac{< u, i >}{\|u\|_2 \|i\|_2} \quad \forall i \in I \tag{III.7}$$

The provided forecast does not take into account "day of the Week" and location contexts but it allows the system to provide ratings forecasts if enough information are not known.

Instead, if checks do not fail, contextual forecasts are done in CARS Module that take information from Data Module. These forecasts are calculated through the following formula:

$$\hat{\mathcal{R}}_{u,i,k_c,l_c} = \bar{R}_{k_c,l_c} + \frac{\sum_{p \in N_j} w_{up} \left(\mathcal{R}_{p,i,k_c,l_c} - \bar{R}_{k_c,l_c}\right)}{\sum_{p \in N_j} w_{up}} \qquad \forall i \in I \tag{III.8}$$

where $\bar{R}_{i,k_c,l_c}$ is the mean of known items ratings in the selected contexts and $w_{up} = cos(u, p)$.

The ratings forecasts are filtered by Time context e send to Paths Builder Module. The Post-Filtering phase happens if ratings forecasts are done by RS Module and also if they are done by CARS Module.

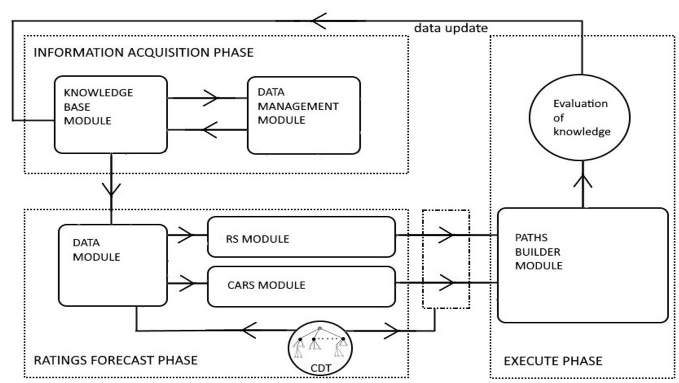


Figure 1. System Architecture

C. Execute Phase

In the Execute Phase the Paths Builder Module aims to provide a training program personalized to students. It suggests exercises to students in order to improve their knowledge in the fields where they have more difficulties in order to improve themselves. The built path proposes the exercises starting from the field where the student finds more difficulties. In order to reach this objective the CDT provides the contextual information associated to time context. In this way the suggested items are filtered and the proper training program can be recommended.

Moreover Paths Builder Module also provide some theoretical tools that allows to simplify the repetition of proper topics. The results of performed exercises allows to update the student profile if he improves his knowledge of the topic.

Before the update, users can evaluate the proposed exercises. If they respond to questionnaire, obtained ratings, linked to suggested items, are memorized in tensor $\mathcal{R}$ in the Knowledge Base Module in order to update $\mathcal{R}$. If instead user don not answer to questionnaire the update phase is done by ratings forecasts generated by the system.

IV. Experimental Phase

In this section the online experiment that allows us to evaluate the proposed approach is described. To achieve this a prototype has been developed and has been submitted to 50 students recruited through the courses of Fundamentals of Informatics held at the University of Salerno and of Costume Design held at the Academy of Fine Arts in Naples. The students are heterogeneous and the age range is between 19 and 24 years.

Table I
Questionnaire answer

Section	TD	D	U	A	TA
A	0	1	2	10	37
B	1	3	4	8	34
C	0	1	2	12	35
D	0	2	4	14	30
E	1	3	5	13	28

Moreover the proposed prototype was trained before the submission. In this way the tensor $\mathcal{R}$ content in Knowledge Base Module does not contain only non zero elements, indeed the system can use both the recommendation modules RS and CARS in the proper case.

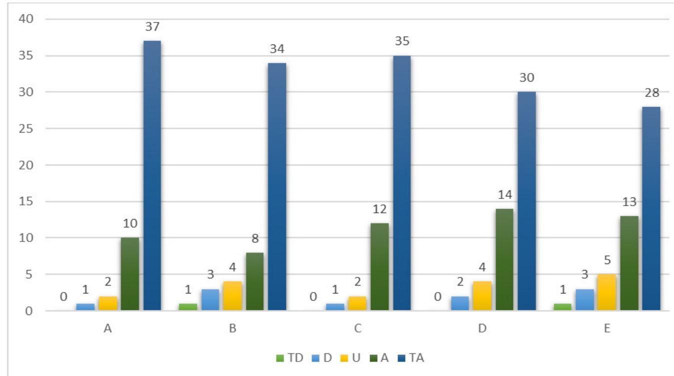
Through a questionnaire users had to evaluate Utility (A), Reliability (B), Recommendation (C), Performance (D), Usability (E) and they could choose between five options: I totally disagree (TA), I disagree (D), Undecided (U), I agree (A), I totally agree (TA).

Utility (A) aims to evaluate how much students benefits of this approach through the exercise done, Reliability (B) aims to evaluate the adequacy of the proposed exercises, Recommendation (C) allows to evaluate the correctness of field and difficulty of proposed exercises, Performance (D) allows to estimate the prototype and Usability (E) aims to evaluate the facility in the use of the prototype.

The questionnaire answers are in Table I and Figure 2 gives a graphic representation.

Results return that Utility section is rated as the best. This result confirms that the use of Context Aware Recommender Systems manage to provide useful training programs to students in order to improve their ability in the topics of the courses analyzed. Moreover Reliability and Recommendation sections are also rated as good. This results allow to assert that the use of a Classical Recommendation techniques, that only considers time context with Post-Filtering strategy when there are insufficient information, does not invalidate the quality of provided recommendations. In this way the reliability of the proposed recommendation approach is ensured.

Figure 2. Questionnaire answer trend



V. Conclusions

This article presents a new approach, which exploits the techniques of recommendation in order to provide a personalized training programs to students and in order to increase the their knowledge.

The good results obtained in experimentation phase are promising. In the future the proposed approach will be tested on a sample of users more significant and the integration of recommendation techniques and context will be improved.

When the reliability of each module will be tested, the next phase will be the integration of machine learning techniques in order to improve the stability when the number of users and items increases.

References

[1] Francesco Colace, Massimo De Santo, Marco Lombardi, Rosalba Mosca, and Domenico Santaniello. A multilayer approach for recommending contextual learning paths. *J. Internet Serv. Inf. Secur.*, 10(2):91–102, 2020.

[2] Francesco Ricci, Lior Rokach, and Bracha Shapira. Recommender systems: introduction and challenges. In *Recommender systems handbook*, pages 1–34. Springer, 2015.

[3] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. *Knowledge-based systems*, 46:109–132, 2013.

[4] Lara Quijano-Sánchez, Iván Cantador, María E Cortés-Cediel, and Olga Gil. Recommender systems for smart cities. *Information systems*, 92:101545, 2020.

[5] Mario Casillo, Francesco Colace, Massimo De Santo, Marco Lombardi, Rosalba Mosca, and Domenico Santaniello. A recommender system for enhancing coastal tourism. In *Research and Innovation Forum 2020: Disruptive Technologies in Times of Change*, pages 113–122. Springer International Publishing, 2021.

[6] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. *Recommender systems handbook*, pages 73–105, 2011.

[7] Mario Casillo, Dajana Conte, Marco Lombardi, Domenico Santaniello, and Carmine Valentino. Recommender system for digital storytelling: A novel approach to enhance cultural heritage. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10-15, 2021, Proceedings, Part VII*, pages 304–317. Springer International Publishing, 2021.

[8] Raymond J Mooney and Loriene Roy. Content-based book recommending using learning for text categorization. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 195–204, 2000.

[9] Dheeraj Bokde, Sheetal Girase, and Debajyoti Mukhopadhyay. Matrix factorization model in collaborative filtering algorithms: A survey. *Procedia Computer Science*, 49:136–146, 2015.

[10] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[11] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1235–1244, 2015.

[12] Robin Burke. Hybrid web recommender systems. *The adaptive web*, pages 377–408, 2007.

[13] Gregory D Abowd, Anind K Dey, Peter J Brown, Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *International symposium on handheld and ubiquitous computing*, pages 304–307. Springer, 1999.

[14] Xiaomei Huang, Guoqiong Liao, Naixue Xiong, Athanasios V Vasilakos, and Tianming Lan. A survey of context-aware recommendation schemes in event-based social networks. *Electronics*, 9(10):1583, 2020.

[15] Norha M Villegas, Cristian Sánchez, Javier Díaz-Cely, and Gabriel Tamura. Characterizing context-aware recommender systems: A systematic literature review. *Knowledge-Based Systems*, 140:173–200, 2018.

[16] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer, 2011.

[17] Gediminas Adomavicius, Ramesh Sankaranarayanan, Shahana Sen, and Alexander Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1):103–145, 2005.

[18] Mario Casillo, Francesco Colace, Massimo De Santo, Marco Lombardi, and Domenico Santaniello. A chatbot for training employees in industry 4.0. In *Research and Innovation Forum 2020: Disruptive Technologies in Times of Change*, pages 397–409. Springer International Publishing, 2021.

[19] Linas Baltrunas and Francesco Ricci. Experimental evaluation of context-dependent collaborative filtering using item splitting. *User Modeling and User-Adapted Interaction*, 24(1-2):7–34, 2014.

[20] Sondess Missaoui, Faten Kassem, Marco Viviani, Alessandra Agostini, Rim Faiz, and Gabriella Pasi. Looker: a mobile, personalized recommender system in the tourism domain based on social media user-generated content. *Personal and Ubiquitous Computing*, 23(2):181–197, 2019.

[21] Linas Baltrunas, Bernd Ludwig, and Francesco Ricci. Matrix factorization techniques for context aware recommendation. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 301–304, 2011.

[22] Dongmin Shin, Jae-won Lee, Jongheum Yeon, and Sang-goo Lee. Context-aware recommendation by aggregating user context. In *2009 IEEE Conference on Commerce and Enterprise Computing*, pages 423–430. IEEE, 2009.

[23] Zhenxing Xu, Ling Chen, and Gencai Chen. Topic based context-aware travel recommendation method exploiting geotagged photos. *Neurocomputing*, 155:99–107, 2015.

[24] Cristiana Bolchini, Carlo Curino, Fabio Alberto Schreiber, and Letizia Tanca. Context integration for mobile data tailoring. In *7th International Conference on Mobile Data Management (MDM'06)*, pages 5–5. IEEE, 2006.

[25] Mario Casillo, Francesco Colace, Massimo De Santo, Saverio Lemma, and Marco Lombardi. Cat: a context aware teller for supporting tourist experiences. *International Journal of Computational Science and Engineering*, 20(1):69–87, 2019.