POSITION PAPER

A generic architecture of an affective recommender system for e-learning environments

Juan Camilo Salazar[1] · Jose Aguilar[1,2,3] (ORCID) **· Julián Monsalve-Pulido[4] · Edwin Montoya[1]**

Abstract
Personalization of suggestions of contents plays a key role in adaptive virtual learning environments. Good recommendations can raise the interest of students in the learning process, while, on the other hand, bad recommendations can have catastrophic results for the learning process. The affective state of the student is a very influential factor in the learning process. In this work, a generic architecture of an affective recommender system for e-learning environments is developed, to serve as a guide for future implementations of this kind of recommender system. Here, the affective characteristics of students are represented by their personalities, learning styles, emotional states, and expertise levels. Thus, the main contribution is the proposition of a generic architecture of an affective recommendation system for the educational field. The architecture is completely modular, which gives it great flexibility because the emotion engine is separated from the personal characteristics engine, and is not based on specific models of emotions. This work finishes with examples of use cases of the architecture. According to the results in these examples, our architecture is capable of analyzing the polarity of academic documents based on their content, determining the personal characteristics of students (including their emotions), and from there, recommending learning resources to students considering emotions as the main element of the process.

Keywords Affective recommendation systems · Virtual learning environments · Emotion recognition · Sentiment analysis

1 Introduction

Every day a high amount of data is produced, and the problem of information overload is becoming increasingly common as the quantity of available information grows [8]. With new data being created every day, it becomes really difficult to search for relevant information. But recommendation systems have been created for addressing this problem, to reduce the time waste for finding useful information [13].

In recent years, the research interest in recommender systems has grown [40], and as more research is done, more challenges appear [44]. One of the biggest issues for educative recommender systems is the personalization of suggestions of learning resources to students [44]. Recommender systems in education can be very influential modules since a good recommendation can increase the interest of learners in the learning process, which will be reflected in the performance of the evaluative activities of the students, while a bad recommendation can frustrate the student and taking him to abandon the learning process [17].

Personalization of content suggestions is very important for virtual learning environments (VLEs) because every student learns in a different way, learning styles are so variate, changes to the student context-aware variables, and recommendations that are very influential for some students can be disappointing for others [4, 44, 50, 53]. Also, the emotional state of the student is relevant for the recommendation process, the mood influences a lot of the way a person learns,

✉ Jose Aguilar
aguilarjos@gmail.com; aguilar@ula.ve

Juan Camilo Salazar
jhenaos@eafit.edu.co

Julián Monsalve-Pulido
jmsistemas@gmail.com; julian.monsalve@uptc.edu.co

Edwin Montoya
montoya@eafit.edu.co

[1] GIDITIC, Universidad EAFIT, Medellín, Colombia

[2] CEMISID, Universidad de Los Andes, Mérida, Venezuela

[3] IMDEA Networks Institute, Madrid, Spain

[4] GIMI, Universidad Pedagógica y Tecnológica de Colombia, Tunja, Colombia

and it is not the same as a student learns when he/she is happy than when he/she is angry[17, 25, 30, 32].

In general, emotions have demonstrated their value from the area of psychology due to their influence on decision-making processes [9, 15, 38]. In fact, the relationship between emotions and learning performance is evident in various works, like [49, 27]. Emotions play a key role in adaptive systems, due to that they are necessary and must be included in the design process of virtual adaptive learning systems [19]. In this paper, emotion will be considered as an intense feeling that we experience due to a contextual stimulus, which can be accompanied by organic changes [9]. On the other hand, affective states will be feelings that are less intense than emotions, often lack a contextual stimulus, and are of prolonged duration [14, 49].

Affective information is very influential in adaptive learning systems and represents the main characteristic of students for providing more personalized recommendations of content in virtual learning environments. Addressing affective issues has been proved to enhance the performance of recommender systems in fields different from education [48]. In our work for boosting the personalization of recommendations, we consider four personal characteristics that enrich the context of the student in a VLE for taking better decisions of suggestions. These four characteristics are personality, learning style, expertise level, and emotional state.

Research in the area of affective recommendation systems in the educational field has been scarce in recent years [44]. Some papers have been published on this topic, as can be seen in the related papers section; however, there is evidence of a research opportunity in this domain [30, 44]. Also, little research has been done about a generic architecture to follow as a mold or guide for the implementation of this type of recommender system. The main contribution of this work is to propose a generic architecture, to motivate the investigation of this topic, which will lead to more students being more interested in the learning process, and therefore, to increase their performance [44]. This work describes the architecture in detail, with case studies and comparisons to study its behavior.

In an initial work [35], general ideas are presented to develop affective recommender systems, without giving details of their components, and even less, considering the specific aspects of an educational environment. For the development of the architecture, exploratory, descriptive and applied research was used, since our work seeks to solve a specific problem through techniques and previous research used to solve the identification problem of affective states during teaching–learning processes in virtual environments. Specifically, the work was based on the results of the systematic review of the literature on affective recommendation systems for educational settings carried out in [44], whose results highlight the lack of systems of this type. Based on

the above, a qualitative analysis of existing affective recommendation systems was carried out, and from there, the proposal for an educational context was elaborated and presented in detail in this work.

The rest of the paper is structured as follows: In Sect. 2, the state of the art on affective recommender systems in the educational field is explored. Section 3 presents our proposal of generic architecture, and in Sect. 4, every component of the architecture is analyzed in detail. In Sect. 5, some use cases of the architecture are explained, and finally, in Sect. 6 analysis, conclusions and future works are provided.

2 Related works

Currently, various architectures of emotions have been proposed in the educational context. In [28], a generic architecture for emotion-based recommender systems in cloud environments is proposed. In this approach, two main components are exposed: a service layer and a client layer. The service layer is in charge of the storage and the recommendation tasks, and the client layer is the module to present the recommendations to the users in the VLE. In this architecture, storages for learner affective state and metadata from learners, activities and learning resources are proposed, but they do not specify further details about the implementation, and only the general services provided by those storages are detailed. Particularly, this approach considers the emotional state of the students, but does not consider their learning styles, personalities, or experience levels to make content recommendations. They focus on cloud capabilities, such as high availability and scalability. In the architecture, they propose redundancy for the affective recommendation nodes, but neglect other important characteristics of education recommendation systems, such as records of interactions between students and the VLE, or records of recommendations to track the quality of the recommendation system. In the work of Ali et al. [7], they present an architecture of semantic recommendations through virtual agents based on user requirements and preferences, through the extraction of academic courses in a personalized way.

In the investigation of [47], an architecture for an affective recommender system in the educational field is proposed, where multimodality for recognizing emotions is considered. Also, the student profile considers interests and preferences, but these are not specified on how to extract these preferences and interests. They present interesting contextual information that is the device model, where the characteristics of the device that the user is using are considered. This system does not generate automatic recommendations built by the system, but is manually constructed by experts for specific scenarios, such that in each case the recommendations that most fit

is delivered. Also, it is not specified whether they consider important information about the students, like their personalities, learning styles, or expertise levels. In [31], Marcos-Pablos et al. propose an approach for multimodal emotion recognition using Kalman filters for the fusion of available discrete emotion recognition tools. In the paper, they describe an evolutionary approach to the integration of digital ecosystems into new sources of emotion recognition.

In other research, [36] presents a generic architecture for emotion-aware content-based recommender systems. They do not focus on the educational field, but it can be fitted because the architecture is flexible enough. In fact, the architecture is very general, with 4 main components: the content analyzer that analyzes the texts of contents for preprocessing and extracting features from them, the emotion analyzer that is in charge of assigning emotional labels to the contents, the learner profile that uses data mining for extracting preferences for the users, and the recommender that uses all the information obtained in the other components for making recommendations with content-based recommendation strategies.

However, the architecture only supports content-based recommender systems, and other kinds of systems cannot be implemented with that architecture. Additionally, they only consider emotional state and do not take into account other important personal information, like personality, learning style (this is specific to the educational field), and expertise level. Finally, the emotional state of users is extracted only from the historic rating of users to items, though other strategies for assessing the emotional state like sensors or dynamics in the interaction with the virtual platform are not included.

On the other hand, the review of [26] describes some recent implementations of affective recommendation systems for various contexts or objectives, identifying a low number of investigations in the educational area, where only three works are referenced that have applied affective recommendation systems in the teaching–learning processes. Also, a detailed analysis of deep learning-based recommendation systems for e-learning environments is conducted in [29]. They summarize how recurrent neural networks, convolutional neural networks, and deep reinforcement learning, among other techniques, have been used by recommendation systems in e-learning environments. Rahayu et al. [42] analyze ontology-based recommender systems in e-learning, that is, the utilization of ontologies in the recommendation process. These systems combined ontologies with other artificial intelligence techniques in the educational context. The main utilization is for student and learning object modeling, but learning path, feedback, context data, and learning devices could be future domains for investigation. According to their conclusions, ontology-based recommender systems

seldom use ontology methodologies, or ontology evaluation methodologies.

In [37], an architecture for an information retrieval system (IRS) is presented that considers the affective state of the users and their profiles for retrieving the most relevant documents to a query specified by the users. The process is as follows: The user specifies a query, and the system searches the documents in its database and returns the most relevant to the query using data mining and analytics techniques. These most relevant documents are presented to the user, and if the user is not satisfied with the answer, then the query is reformulated by the system making use of preferences and indicators based on the emotional state and the profile of the user, which are added as keywords to the query. A main disadvantage of information retrieval systems over recommendation systems is that a query is always needed, and the results are always related to that query. Also, the automatic building of that query can be computationally expensive; on the other hand, if it is built by the user, then it may not find the results the user is looking for if it is not properly specified. In addition, users are not always experts, and sometimes, they do not know what they are looking for. Moreover, the historical information of the emotional state is not considered, just the emotional state at the moment of reformulating the query. Finally, that architecture is specific for an IRS and cannot be fitted for a recommender system.

The work by [40] focuses on building an emotion-aware recommendation system that extracts information from multiple sources, like social networks, ratings, and reviews from users. They do not focus on the educational field, and they fuse the extracted information with analytic and machine learning models, and then use this fused information for making recommendations. One part of the information fused is the emotional information extracted from reviews posted by users. In the system, they do not consider important factors to the educational field, as the learning style and expertise level; also, they do not consider personality that is very influential for the affective state. Neither do they include sensors for assessing the emotional state, and do not take into account historical data captured from the virtual platform for extracting valuable information. They consider three types of recommendation strategies.

In [52], a framework for emotion-based recommender systems is presented. The authors propose assessing emotional state at three moments: at entry, during the utilization of the resources, and at the exit. This is not a proper recommendation system architecture, though they show the way how emotions can be included in the recommendation process, and present an interesting idea for the temporal component of recognizing emotional state. The work is very general, it does not focus on the educational domain, and more contextual information is necessary for better recommendations because only emotional states are considered.

In [6], adaptive learning activity selection algorithms to learner personality and competence are defined. Three algorithms were created to adapt the learning activities' knowledge complexity to learners' personality and competence. Ezaldeen et al. [18] propose a framework, namely enhanced e-learning hybrid recommender system, to provide an e-content corresponding to the learner's particular needs. To do that, they developed a model to estimate the semantic learner profile. The recommendation depends on the learner's preferences, other similar learners' experience, and background. Finally, in [44] a systematic literature review of affective recommender systems in learning environments is carried out. The goal of the paper is to explore the state of the art of the influence of emotions in the educational field, especially in content recommender systems.

3 Architecture proposal

The proposed architecture for affective recommender system is presented in Fig. 1. The flow is as follows: a student enters the VLE, the first time must register it. During this process, personal information is captured and personal traits and the learning style questionnaire must be filled out. Additionally, the expertise level of students can be obtained using quizzes or questionnaires, and these processes are executed by the personal characteristics engine. All this information is stored in the user profile, except the expertise level that is stored in the VLE database.

When a user is registered, he/she can login to the platform and interact with different contents. While the student is using the contents, several logs are captured by the VLE logger and stored in the VLE database. The emotion engine is capturing emotional information about the student before, during, and after using the contents through multiple

Fig. 1 Generic architecture

sources, such as a camera, microphone, and questionnaires, which are low invasive sources of emotional information for not disturbing the learning process. The emotional information gathered is stored in the resources module, in a special database of logs of emotional data. This information stored is the emotions felt by a student using a resource in a specific course, for a unit or activity of that course, together with some metadata of the interaction, such as the time stamp.

The emotion engine is also in charge of extracting emotional information from contents, assigning them an emotional tag that is used later for the recommender algorithm. The resources module takes control of storing all the resource information, including the emotional tags previously described, and the emotional logs from students using the resources. Since the learning style, the expertise level, and in some cases the personality traits are dynamic, the personal characteristics engine periodically assesses these characteristics implicitly through logs, or explicitly applying questionnaires to the student in the VLE.

Finally, all information obtained from resources, users, and its interactions, are passed to a recommender algorithm that, depending on the implementation, considers some information or another to suggest potential useful contents to students, which are given to the VLE so they can be displayed to the student and he/she can choose whether he/she can use them or not. This decision serves as feedback for the recommender system for improving recommendations. Various algorithms of recommendation are exposed in the use cases section. The components of this architecture are explained in detail in the following section.

4 Components of the architecture

The architecture is made up of six main components: a user component, a personal characteristics engine, the VLE, an emotion engine, a resource component, and a recommendation algorithm. The output is the final recommendation of the recommendation algorithm. Each component is described below, and the architecture with all components, subcomponents, and their relationships is presented.

4.1 The user component

The user component models the student considering its profile, and stores all the personal information about it. This component (the physical user) interacts with the VLE. It is composed of two subcomponents: a physical user and a database for storing user profiles, as it is shown in Fig. 2.

In the user profile database, the personal information is stored, together with the personality traits and learning styles that are extracted using the personal characteristics engine component. Later, this information stored in the user profile
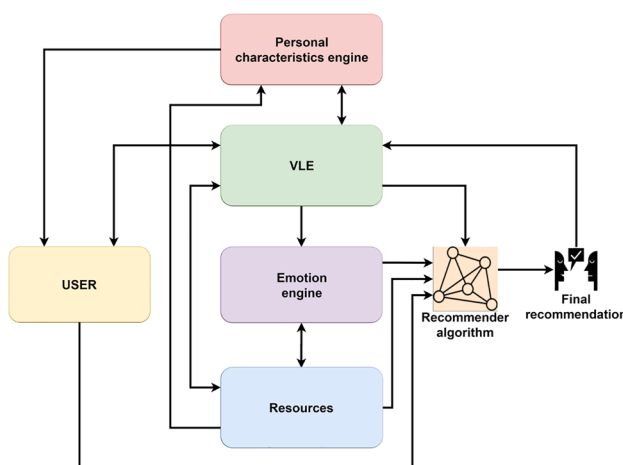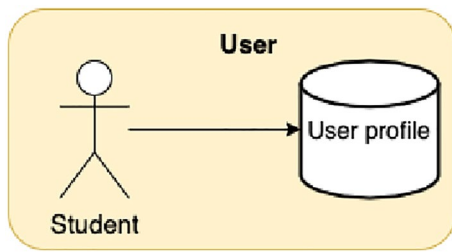
Fig. 2 User component

database is sent to the recommendation algorithm for custom recommendations of learning resources for each student.

4.2 The personal characteristics engine

This component is in charge of calculating and extracting the additional personal information, except the emotional state, which the emotion engine is in charge of. It has three subcomponents that work as calculators: a personal traits questionnaire, a learning style calculator, and an expertise-level calculator (see Fig. 3).

Personality is estimated through different questionnaires in psychometrics. For example, NEO-PI-R [16], BFI or 16PF5 [11] are used, among others. Personality is very static, but can change, and its changes are reflected over long periods of time. As a result, the personality questionnaire is applied only once per semester for each student.

The learning style is initially measured in the registry through a questionnaire, such as [21], to measure the initial values, whereby the questionnaire is applied every time an academic period begins. Learning style can change more frequently than personality, and estimation needs to be made more regularly. Therefore, the authors in [12] [46] propose to use dynamic methods, such that they assign a new learning style to students when their performance is poor. These methods use the information implicit in the student's records to assess their learning style.

The experience level changes with a relatively high frequency and must be constantly measured. To do so, data mining techniques are used on the records of the user who interacts with the VLE. This allows it to be measured as
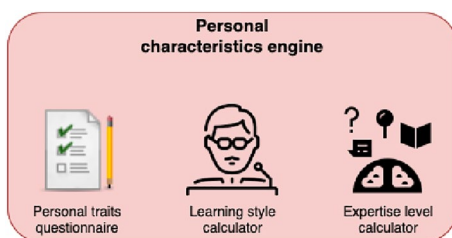
soon as the student interacts with the platform and does so in a transparent way without affecting the learning process. Some influential characteristics to measure the level of experience are the grades, the time it takes the student to complete the activity, and the emotional state.

4.3 VLE component

In this work, we use the term learning management system (LMS) as a synonym of VLE based on what is expressed in the works [22, 23]. The VLE component is made up of three subcomponents: LMS interface, LMS logger, and LMS database (see Fig. 4). The student interacts with the LMS provided by the resource component. The interactions are constantly captured by the LMS logger and then used to calculate the level of experience and learning style. These records include the qualification of a user to a resource. Additionally, these records are used by the recommendation algorithm to have contextual information and historical data to make more personalized recommendations of learning resources for students. The LMS database stores the records produced by the LMS interface, the student information, such as experience level, and the recommendations made by the recommendation algorithm, along with their records (information on whether the student used the recommendations or not).

4.4 Emotion engines

The main objective of the architecture is the affective recommendation. The emotion engines are the key component, where the emotional information is captured from the users and the contents (see Fig. 5). This component is composed of two emotional engines: the resource emotion engine and the human emotion engine.

4.4.1 The resource emotion engine

The resource emotion engine is in charge of extracting emotional information about the content, and its implementation varies according to the context of the application and the type of content (video, text, audio). Therefore, a specific implementation of this subcomponent is
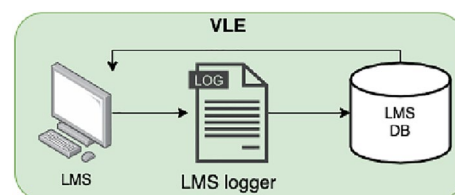

Fig. 3 Personal characteristics engine component
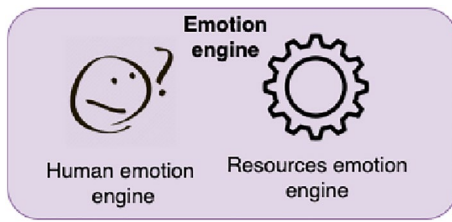

Fig. 4 VLE component

Fig. 5 Emotion engine component

not presented, and it is left free for the specific needs of each case. The emotion engine interacts with the resource component to obtain information about the content, process it, extract the characteristics of emotions, and return these data to the resource component to be stored.

The emotional information that will be extracted from the academic resources is the implicit emotion that the content tries to induce in the users who use the resource, or the true emotion that it generates in the users. Academic content has an implicit emotion that can be of great importance for any recommendation system, where it improves the effectiveness of classification due to the induction of the content in the student according to his/her mood. Figures 6 and 7 illustrate the proposal for this subcomponent. The implementation of the resource emotion engine is separated into two phases:

- Firstly, the information implicit in the content of resources is used for extracting the implicit emotion wanted to be generated by the resource on users. From the metadata of a resource, some important features are extracted (emotionally relevant words, the polarity of the content, and special embeddings of the content). Then, these features are used by an emotional detector, which is in charge of deciding whether the content has a strong emotional component that can generate emotions in the user that interacts with it. In the affirmative case, the features are then sent to an emotional recognizer, which recognizes the emotion that the resource is most likely to generate in the students.
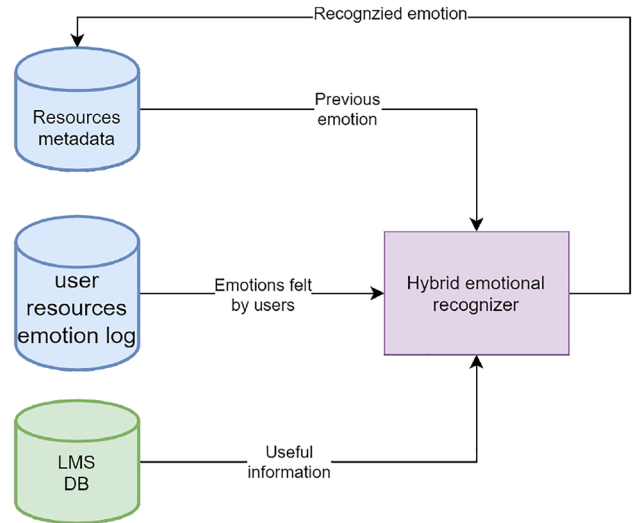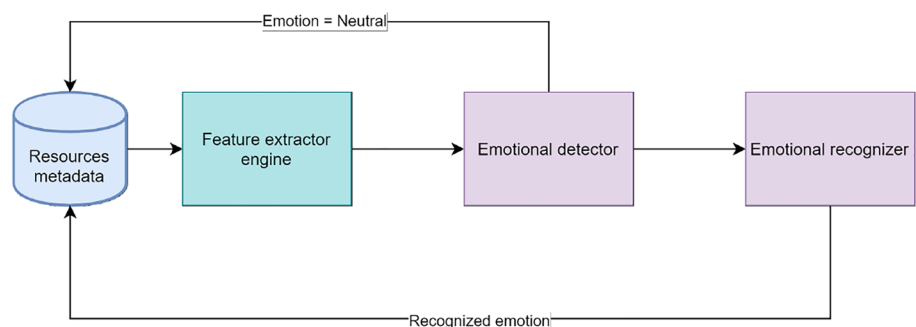

**Fig. 7** Resource emotion engine subcomponent at the second phase

- Second, this subcomponent is executed periodically and consists of obtaining the user's emotions through the use of a specific resource. The information to be analyzed is the student comments, and interactions by clicking on resources, among others. All these data are sent to the hybrid emotional recognizer to recognize the most appropriate emotion for the resource. This most appropriate emotion means the emotion that a student interacting with the resource is most likely to feel. The architecture of this subcomponent is presented in Fig. 7.

4.4.2 The human emotion engine

The human emotion engine is the subcomponent that extracts emotional information from students. For this objective, multiple sources are used, and in the end, they are fused by a hybrid engine, as shown in Fig. 8.

In this subcomponent, six main sources of emotions are proposed: webcam, microphone, comments (student content reviews), keyboard, mouse, and questionnaires. They were chosen due to their low intrusion because high intrusion

Fig. 6 Resource emotion engine subcomponent at the first phase

Fig. 8 Human emotion engine subcomponent
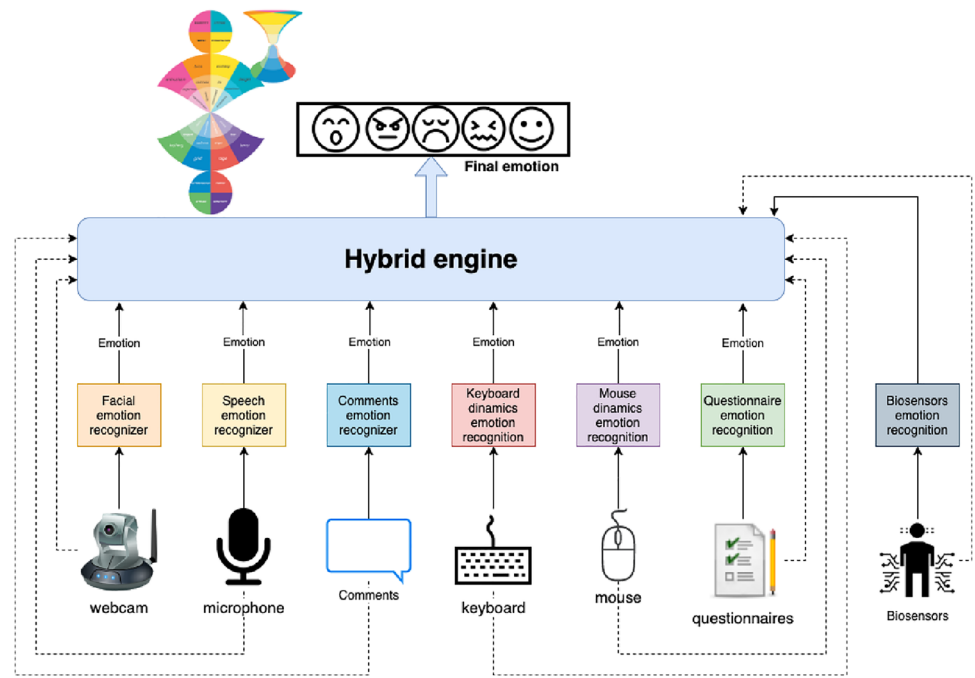
fonts can disrupt students' learning process, as demonstrated in [20]. For example, when the student is in a VLE at home, then learning is normal, but when the student has an electroencephalogram in his head, the entire learning process is affected by the intrusion of the sensor.

Other additional sources are biosensors, using portable devices that are not as invasive, such as smartwatches and heart bands, can be used to analyze student emotions.

Two strategies for fusing the different modalities are proposed, namely feature-level fusion and decision-level fusion, as proposed by [39]:

- The first is the fusion of characteristics directly extracted from the source, for example extracting acoustic features from audio and visual features from video, and putting them together in a single vector, and then, the hybrid engine recognizes the emotion based on the single vector that contains the characteristics extracted from multiple sources. This strategy is represented by the dashed lines in Fig. 8;
- The second is to fuse the different modalities at the decision level. For achieving this, an emotion recognizer specific to each source is needed, which is why it appears the different emotion engines in Fig. 8. In the end, the outputs from all these engines are fused on the hybrid engine. Many techniques for fusing at these levels exist, some of them are cascade, where a first engine recognizes one emotion, then the second takes the output from the first and uses a second engine, and select if leave the first output or change it due to the evidence provided by the second engine, and so on;

- Another approach is a linear combination of the different emotion recognizers or a weighted sum. This is possible only when output from engines are continuous values, for example, polarity or valence–arousal values. Another strategy is switching which turns on and off engines, this one fits well for this problem because it is able of turning off engines when data are not available. For example, if it is planned to retrieve emotional data from students at the beginning, during, and after using a content (as we propose in this work), comments are only available after using the content because a student comments a content after using it, not before. So, the comments engine should be turned off when retrieving emotional information before and during the use of the content. In brief, the hybrid engine combines the outputs from every source, and outputs a general emotional information, based on all the sources available at the moment.

The human emotion engine subcomponent receives information from the LMS DB subcomponent of the VLE component for assessing the emotional information (classified by the Hourglass Model [51]). Then, it sends the detected emotion to the resource components for storing them in the user resource emotional logs database, in order to analyze the historic emotional behavior of students. The gathered emotional data are also sent to the recommender algorithm for making recommendations of contents based on the emotion of the student at the moment.

4.5 Resource component

The resource component is in charge of storing all the information referred to content. Two storages are considered: one for metadata of contents and the second for logs of emotions recognized by the human emotion engine, as shown in Fig. 9.

The *first database* stores the metadata of contents including the emotional information extracted using the resource emotion engine subcomponent. These metadata are not changing in time, updating may be performed, but changes are not frequent, so a SQL database is proposed. This first storage only stores metadata including the location of the data and provides this location to LMS interface for displaying contents to students. Also, it sends information to the resource emotion engine for extracting emotional data from contents, and at the end, the metadata here stored are passed to the recommender algorithm for making recommendations.

The *second storage* is where the logs of the emotions extracted with the human emotion engine subcomponent
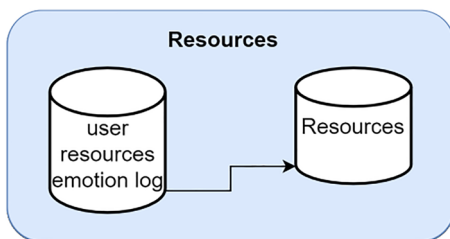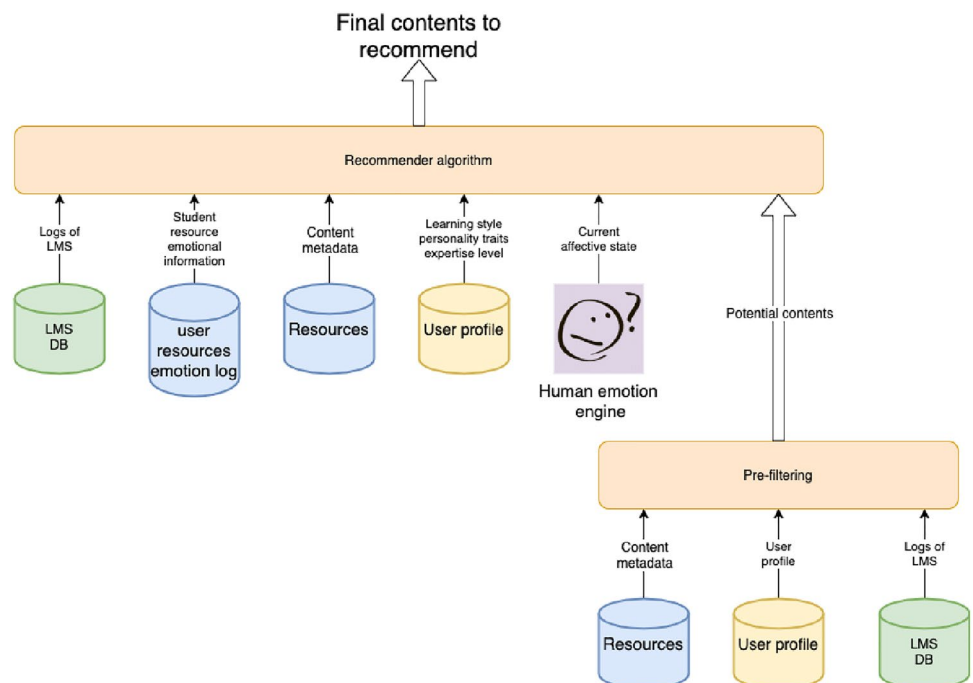
for a student are located, using a resource at a specific time, being part of a course or studying. As many records of this type can be retrieved for a pair (student–resource), we planned this database as one event database. In other words, the emotion tagged for a resource used by a student can be reconstructed using several observations of emotions during the use of content by the same student, and form only one general emotion tag for a pair (student–resource). This subcomponent receives information from the human emotion engine, provides and receives information from the resource emotion engine, and provides information to the other storage of resources, the learning style calculator, and the recommender algorithm.

4.6 Recommender algorithm

The recommendation algorithm component is variable and depends on the use case and the recommendation strategy (content-based, collaborative filtering, etc.), or may be a combination of multiple of them. We planned this component to be flexible enough to fit in every implementation, though something that is transversal to every implementation is the two-step filtering, as illustrated in Fig. 10. A two-step filter is used to improve the computational response, since a high number of resources and excessive iterations can have a computational overload.

For the system to be scalable, a first pre-filter is made considering the topics that the student is studying, to extract only those that are related to the topics. Finally, only a part of the entire resource batch is retrieved, to

Fig. 9 Resource component

Fig. 10 Recommender algorithm component

continue with a subsequent and more specialized filter that considers more information.

For the last filter that outputs the contents to be recommended to the student, additional information is considered, such as the user resource emotional logs, the emotional state of the student, the personal profile, and the LMS logs, for calculating the relevance of each content for the students. Implementation of this subcomponent can vary a lot, as commented previously. In the following section, some examples of implementations are given in specific use cases.

Emotions can be directly relevant to the recommendation algorithm, independent of the recommender strategy. For example, for content-based emotions, the similarity between resources can be calculated using affective features from them, as the induced emotion extracted by the resource emotions engine, or the average emotion that users feel when using that resource. For collaborative filtering based on users, the user's similarity can be computed using the emotions felt by those users with the same resources. The knowledge-based compensation of emotions strategy can be used for making recommendations, trying to compensate for the negative emotions with resources that produce positive emotions. Every recommender strategy was designed for the product

recommendation, though they are completely adaptable for affective recommendations of learning resources.

4.7 Interactions between the components

The final architecture, including all the connections between components and subcomponents described in this section, is shown in Fig. 11.

Each of the interactions is described below (there is no general order of interactions, they can be executed in different orders):

1. The student interacts with the LMS interface. Initially, the user is registered and must fill out some personal data and questionnaires; then, the user can interact with the resources of the LMS;
2. The personal information captured in the register is sent to the user profile database to be stored;
3. During the register, a questionnaire for extracting the personality traits of each user is performed, and the results are sent to the personality traits questionnaire subcomponent;
4. During the register, a questionnaire for extracting the learning style of each user is performed, and the results are sent to the learning style calculator subcomponent;

Fig. 11 Full components and subcomponents architecture
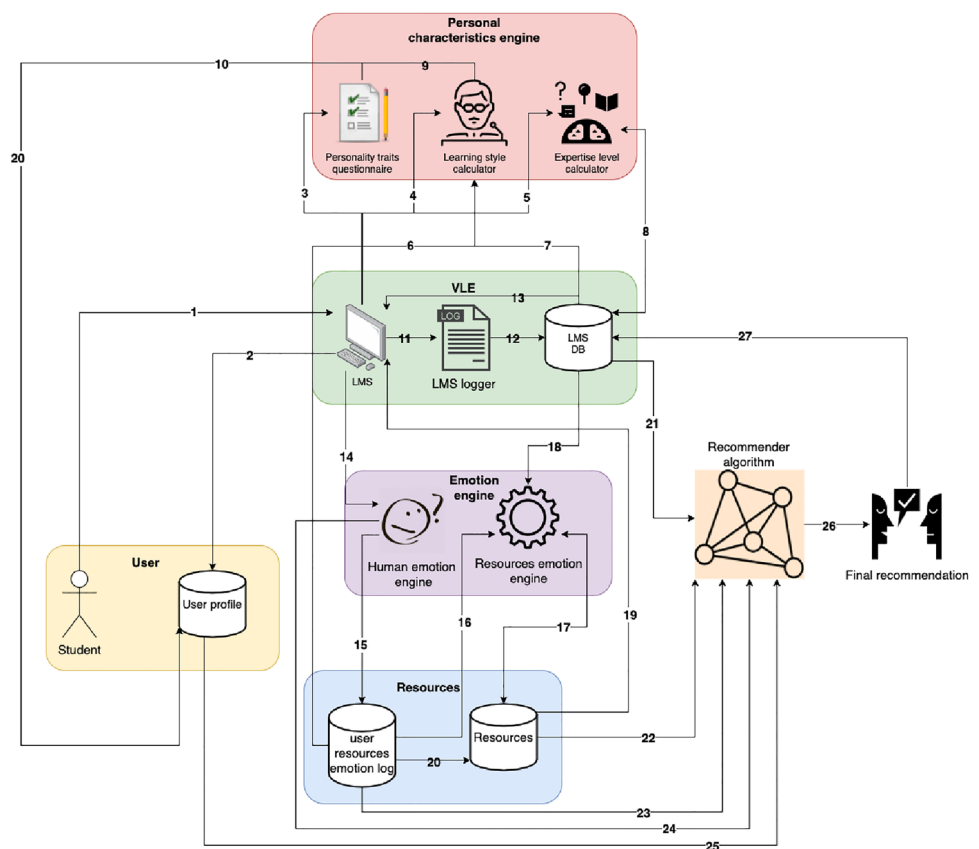
5. During the register, a quiz for extracting the expertise level of each user in different topics is performed, and the results are sent to the expertise-level calculator subcomponent;

6. Periodically, the learning style calculator extracts information from the emotional logs of the user to dynamically calculate the learning style;

7. Periodically, the learning style calculator extracts information from the LMS logs of the user to dynamically calculate the learning style;

8. Constantly, the expertise-level calculator extracts data from the LMS logs in order to calculate the expertise level, and return the expertise level to be stored in the LMS database;

9. The resulting learning style is sent to the user profile to be stored;

10. The resulting personality traits are sent to the user profile to be stored;

11. The LMS logger captures the interactions of users in the LMS interface;

12. The LMS logger sends the logs to the LMS database to be stored;

13. The LMS uses the information stored in the LMS database for recommending the resources that the recommender algorithm suggested, or for consulting information from users;

14. The human emotion engine is constantly capturing data from sensors during the interactions between users and contents, which are taking place in the LMS interface, in order to recognize emotional information from the user;

15. The emotional information extracted from the user is sent to the user resource emotion log to be stored in an event format;

16. Emotional logs are used by the resource emotion engine for calculating emotions induced by the contents;

17. Metadata from resources are used by the resource emotion engine for calculating emotions induced by the contents, and this emotional information is stored in the resource database;

18. Logs from LMS are sent to the resource emotion engine for extracting effective information from them;

19. The metadata from resources is consulted from LMS in order to offer them to the users;

20. The "average" emotion built by the user resource emotion log, described by an event pattern, is sent to the resource database for being stored;

21. The LMS database information is consulted by the recommender algorithm for making suggestions of contents;

22. The resource metadata are sent to the recommender algorithm for making recommendations of contents;

23. The user resource emotion logs are used by the recommender algorithm for making suggestions of resources;

24. The current emotional state of the user, recognized by the human emotion engine, is used by the recommender algorithm for making recommendations of resources;

25. The user profile is sent to the recommender algorithm in order to make suggestions of contents;

26. The final recommended resources are calculated by the recommender algorithm;

27. The metadata from the recommended resource are sent to the LMS database for being stored, carrying out a track of the recommendations for acquiring feedback for the recommender algorithm.

5 General analysis

In this section, we present use cases to analyze the behavior of each component of the architecture. Next, we carry out an analysis of the partial implementations of its components, and finally, we present a case study in an initial prototype that integrates all these components of the architecture.

5.1 Use cases

Four use cases of the proposed architecture are presented. The first use case is the extraction of emotional information from documents and focuses on the implementation of the resource emotion engine, the next steps for making recommendations are proposed but not specified. The other three use cases assume that the human emotion engine is already implemented, and use the information extracted for making recommendations.

In all use cases, the two-step filtering for recommendations is used. The first is used for retrieving contents that are coherent with what the learner is studying, so just a subset of all the contents that are potential suggestions is passed to the second filter. For implementing this first filter, various strategies can be used: One of them is filtering the resources by keywords related to the subject the learning is studying, and another approach is building an inverted index and retrieving those resources that contain the keyword of the subject. The first filter is general and implicit for all the use cases. The second filter is explained in each of them. This last filter ranks the filtered contents using the affective information and contextual information.

The first and second use cases are based on emotional techniques for making recommendations, such that if a student feels happy, then the recommender system will try to suggest contents according to this emotion. The third use case uses collaborative filtering techniques for making recommendations, and the fourth use case uses a content-based

strategy for making recommendations. In brief, the list of use cases discussed in this section includes behavior analysis of:

- The resource emotion engine based on the polarity of the documents;
- The resource emotion engine based on the user emotion;
- The content-based recommender strategies in our architecture;
- The collaborative filtering strategies in our architecture.

5.1.1 Analysis of the behavior of the resource emotion engine based on the polarity of the documents

The general idea behind the first use case is to extract the dominant emotion from a document and then make recommendations based on the emotion the user is currently feeling. In that way, contents with dominant emotion implicit in them similar to the context information are recommended. The instantiation of our architecture for this use case is shown in Fig. 12.

The central component for this use case is the resource emotion engine, where the dominant emotions in the documents are extracted. For extracting the dominant emotion, the following procedure is used:

1. If the document does not have associated keywords, then they are calculated using keyword extraction techniques. For this objective, four main types of approaches exist: based on the frequency of words, based on lexical approaches, based on graphs, and based on machine learning. Inside these approaches different techniques exist, like BM25 or LDA for frequency-based, or SVM, or conditional random fields for machine learning-based [5];

2. After keywords are calculated with any of those techniques, the sentences where keywords appear are obtained and the polarity of all these sentences is calculated using, for example, Senticnet 5 [10] as a knowledge base [43]. Senticnet provides polarity values for 100.000 terms in the English language; for fusing, we can define a simple mean over the polarity of each of the terms that are present in Senticnet 5 and the sentence. When all emotions are extracted for all the sentences that contain keywords, they are grouped by keyword, and then, a weighted mean polarity is calculated using as weights the number of keywords the sentence contains. The more keywords the sentence contains, the more relevant it is to the average. After this weighted average, a unique emotion by keyword is obtained, and finally, all of these emotions by keywords are averaged to obtain a

Fig. 12 Instantiation of our architecture in the first use case
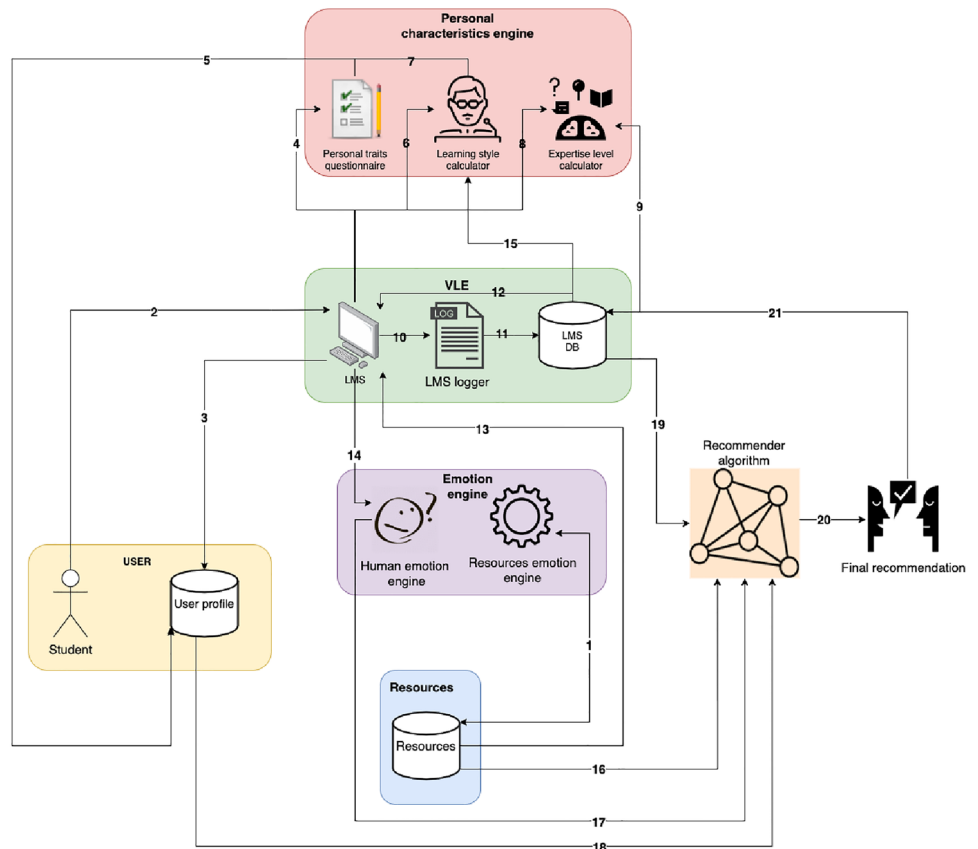
final emotion by document; this is the dominant emotion in the document;

3. This dominant emotion is stored in the resource database and then used for making recommendations together with the current emotion of the student, its profile, and the logs from LMS, by the recommender algorithm. As the main objective of this use case is to calculate the dominant emotion for documents, details on further implementation of the recommender algorithm are not given.

The process of extraction of emotions from contents is the first of our architecture (see arrow number 1 in Fig. 12). When this process has finished, then the rest of the processes are executed: The user registers (arrows: 2, 3), the personal characteristics engine extracts the personal features (arrows: 4, 5, 6, 7, 8, 9, 15), the user interacts with the resources through the VLE (arrows: 2, 13), the human emotion engine captures the emotion felt by the student (arrow: 14), the LMS logger captures the logs of interactions (arrows: 10, 11, 12), and lastly, the recommender algorithm makes suggestions of resources, which are tracked to give feedback to the recommender algorithm (arrows: 16, 17, 18, 19, 20, 21).

5.1.2 Analysis of the behavior of the resource emotion engine based on the user emotion

The second use case is similar to the first in the way that it looks for making recommendations based on the dominant emotion of each document, except that in this use case the dominant emotion is not extracted from the content of the resource, but from the emotions of the users when have used the resource. This dominant emotion can be thought of as the mean emotion felt by all the students that have used the resource.

The instantiation of our architecture for this use case is shown in Fig. 13. In this use case, the students do the register (arrows: 1, 2, 3, 4, 5, 7, 8) and then interact with the resources (arrows: 1, 9), and the LMS logger captures these interactions and stores them in the LMS database (arrows: 10, 11, 12). Additionally, the personal learning style and expertise level are periodically calculated (arrows: 5, 15, 6, 7, 8). While they are interacting, the human emotion engine is constantly capturing the emotion felt by a student at the different moments, and stores it in the user resources emotional log database (arrows: 13, 14). With these logs, a dominant emotion for each user to a resource can be built, fusing them in some way. This process of fusion of the different

Fig. 13 Instantiation of our architecture in the second use case
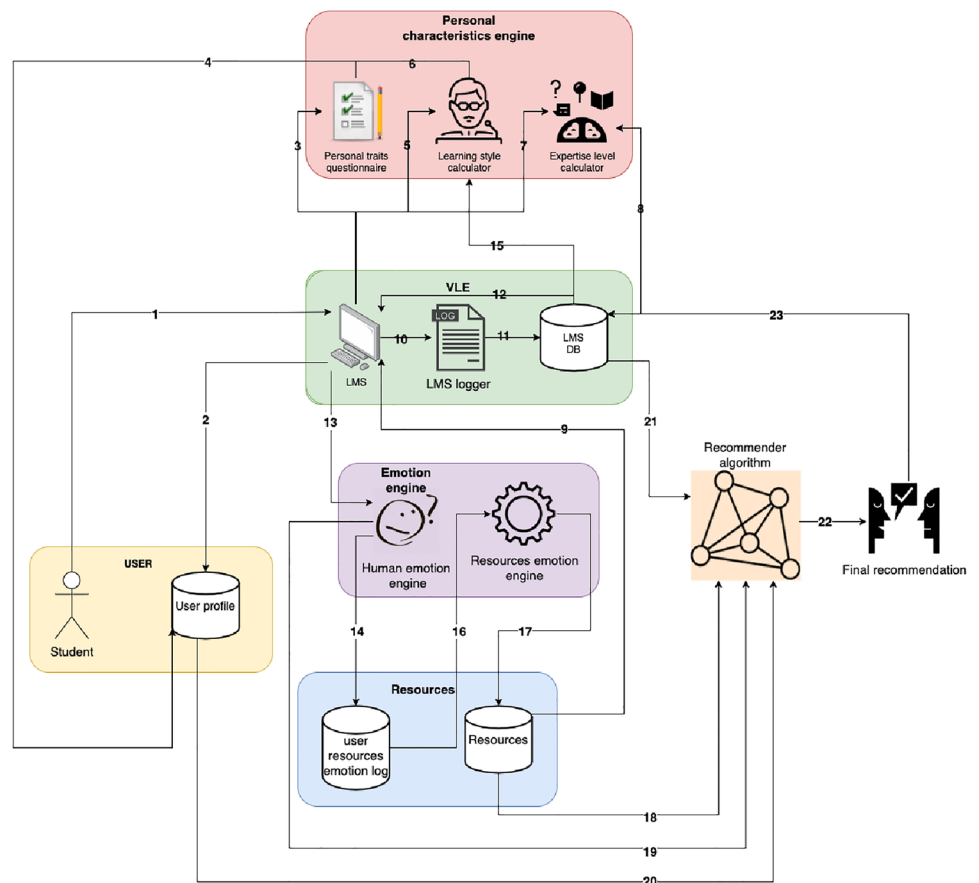
emotions felt by the users that have used the resource is performed by the resource emotion engine (arrow: 16). Finally, an average emotion for all users of a document is calculated, which is the dominant emotion for a document. This is stored in the resource database (arrow: 17), and the information is used for making recommendations of resources that are related to the emotion that the user is currently feeling, its personal profile, and its historical behavior (arrows: 18, 19, 20, 21, 22, 23).

5.1.3 Analysis of the behavior of the collaborative filtering strategies in our architecture

The third use case is based on collaborative filtering strategies, where the items recommended are those that are rated as good for users that are very similar to the user that is looking for recommendations. For achieving this, a user similarity measure is established.

In this use case, the user similarity score function is based on the logs, as major collaborative filtering-based recommender systems do, but the difference is that in this use case, the user similarity measure also considers the personal information: personality, learning style, expertise

level, and especially, the way the users feel. For calculating the users feel similarity, the logs from the user resources emotion log database are used. Two users are compared by the resources they have both used, then the emotional logs are compared, and as they feel similar emotions with the same resources, their similarity arises. The comparison between personalities and learning styles is more direct because they are defined using quantitative values [1]. Finally, the expertise level of a user to a resource is also a quantitative measure, so it can be easily compared. In this approach, the recommended items are not those that were rated as good for similar users, but those that were best rated when similar users were feeling the emotion that the user is currently feeling. The instantiation of our architecture for this use case is presented in Fig. 14.

The flow of execution in our architecture is very similar to the one presented in use cases one and two: The user registers (arrows: 1, 2, 3, 4, 5, 7, 8), personal features are periodically calculated (arrows: 5, 15, 6, 7, 8), the user interacts with resources (arrows: 1, 9), the logs of interactions are captured (arrows: 10, 11, 12) like the user emotional state (arrows: 13, 14), and the recommender algorithm makes recommendations based on the information available (arrows: 16, 17, 18, 19, 20, 21).

Fig. 14 Instantiation of our architecture in the third use case
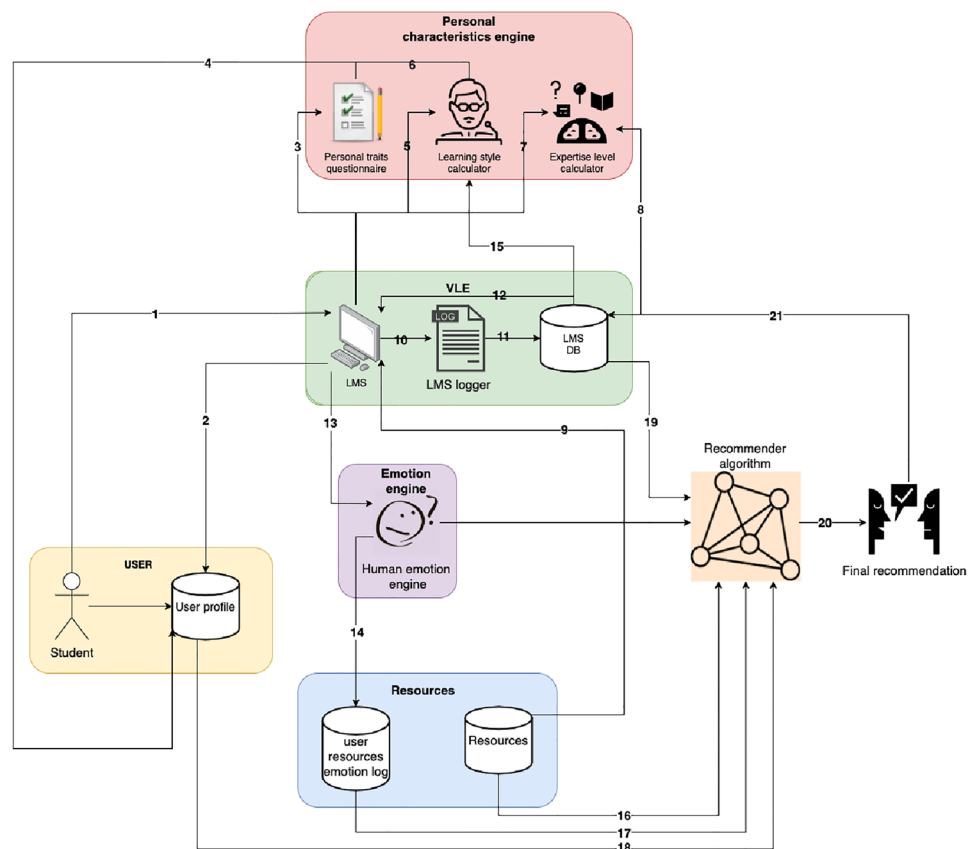
5.1.4 Analysis of the behavior of the content-based recommender strategies in our architecture

The fourth use case follows content-based strategies for making suggestions for items. In these strategies, the items that are recommended are those that are most similar to those that the user liked or rated as good in the past. For this objective, a content similarity measure is needed.

In this use case, the similarity measure can be given by the cosine similarity of vectors of quantitative features that represents the contents, and these vectors of features can be extracted using the Word2Vec technique [33], with a special change to consider the emotional information of each document. The Word2Vec algorithm is trained to predict a missing word inside a context, given the rest of the words of the context. Normally, this context is a sentence, and a word excluded from the sentence is wanted to be predicted using the rest of the words in the same sentence. This technique uses a neural network architecture of 3 layers: one for input, one hidden, and one for output. The Word2Vec algorithm can be trained in two ways: using CBOW and Skip-gram architecture (see [5] for more details). At the end of training with any of the two methods, the weights of the hidden layer are extracted, and those are the vector embeddings that represent the words.

For this use case, the context is not a sentence but a session of study where a student uses different resources to study for an evaluative activity in one course. Thus, instead of words, the elements of the context are resources with a special characteristic: They are labeled with the dominant emotion that the user felt using it. In this way, the same resource with different dominant emotions is treated as different elements, and this is the way the emotional information is considered in the algorithm.

From each session, multiple samples for training can be obtained, subtracting one resource from the session and trying to predict the missing resource using the other resources in the same session using Word2Vec. At the end of the training, the weights of the hidden layer are obtained, and those are the embedding vectors for each resource labeled with its dominant emotion. These vectors are used for calculating document similarity score using the cosine similarity, and the most similar resources to those that the user has used for the same evaluative activity, and rated as good when feeling the same current emotion, are recommended. The instantiation of our architecture for this use case is shown in Fig. 15.

The sequence of execution in our architecture is very similar to the three previous use cases, but in this case, there is a singularity: The personal characteristic engine is not needed because this use case does not use the personal features of users to make recommendations. Instead, the similarity of resources based on their content and their emotional label is used for making suggestions of content. Due to that, the sequence is as follows: The user registers into the system (arrows: 1, 2), the user interacts with the resources in the VLE (arrows: 1, 3), and the logs of these interactions are captured (arrows: 4, 5, 6), like the user emotional states (arrows: 7, 8). The emotional states of all users are sent to the resource emotion engine for training the model, and extracting the embedding vectors from resources (arrows: 9, 12, 11). Finally, the recommender algorithm receives all this information, including the embedding vectors, and makes recommendations of resources as explained in previous paragraphs (arrows: 12, 13, 14, 15, 16, 17, 18).

Fig. 15 Instantiation of our architecture in the fourth use case
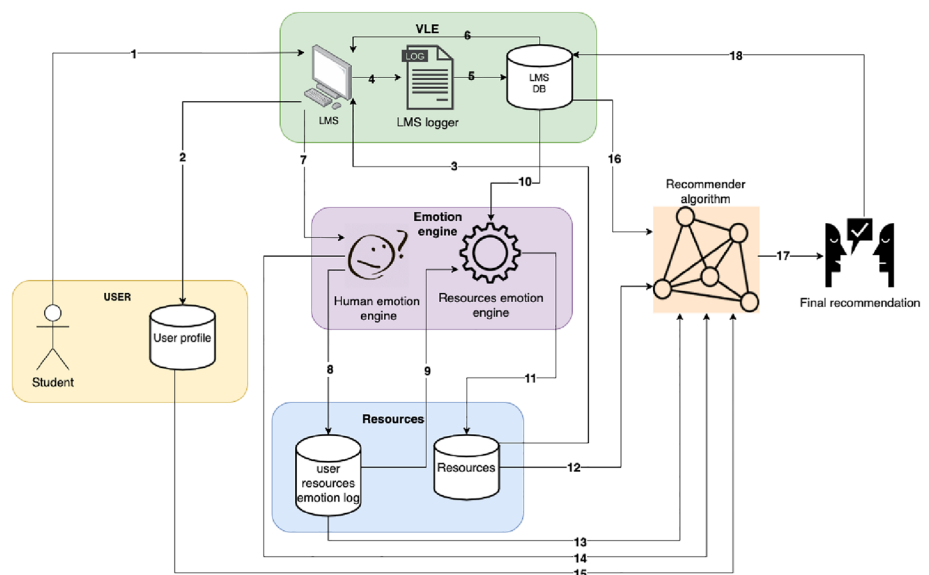
5.2 Preliminary advances in the architecture implementation

For the implementation of the proposed architecture, some of its components have been developed in different works. These components have been tested in various scenarios, to ensure their adaptive capabilities to the educational context.

In particular, the components of the proposed architecture that have been developed in other works are the personal characteristics engine, the emotion engine (composed of a feature extractor engine, an emotional detector and a hybrid emotional recognizer), the human emotion engine, and the recommender algorithm.

At the level of the hybrid emotional recognizer and the human emotion engine components, in [45], the affective state of users in virtual learning environments was evaluated in terms of continuous activation and valence values, making use of multimodal information (audio, text and video). Different approaches, using feature-level fusion and decision-level fusion, were used in this work for emotion multimodal recognition with missing data. This is a novel proposal because it represents emotions in the continuous space, which is not common in virtual education, and the use of the modalities available in a virtual environment at any moment of the teaching–learning process.

For the emotional detector component, [43] presented the sentiment classification problem in texts, and proposed a strategy to classify their polarity (positive or negative). To do this, three methods of extracting keywords from the text are analyzed, and a process for automatic identification of their polarity is defined. The extracted features/keywords were analyzed using the polarity analysis process, to determine the positive/negative connotation of the text.

With respect to the feature extractor engine component, Aguilar et al. [5] analyzed the capabilities of different techniques to build a semantic representation of educational digital resources. They extracted the features/characteristics from the digital resources, using the next feature extraction methods: the Best Matching 25, the latent semantic analysis, Doc2Vec, and the latent Dirichlet allocation. These features/descriptors were tested in three types of educational digital resources (scientific publications, learning objects, patents), a paraphrase corpus, and two use cases in an information retrieval context and in an educational recommendation system. For this analysis, unsupervised metrics were used to determine the feature quality proposed by each one, which are two similarity functions and the entropy. Jimenez et al. [24] analyzed several feature types in audio in a classroom from different points of view: time series, sound engineering, etc. They described the audio as a set of time series, which is not very common in the literature. Moreover, they proposed an automated method for feature engineering in

audios, to extract, analyze, and select the best features in a learning context.

With respect to the recommender algorithms, in [54] an adaptive hybrid recommendation architecture is proposed, which responds to the dynamic behavior of the environment through the use of metrics (meta-characteristics), from which the hybrid configuration for the recommendation is determined. In the experiments, in the context of a case study, its adaptive capacities were shown, exemplifying its operation, and evidencing its flexibility to be implemented in various ways and in multiple contexts.

Finally, in [35] a pilot test was applied in hybrid and virtual courses of the EAFIT University of Colombia, integrating the different components developed in the Moodle platform. The objective of the test was to evaluate the architecture in an integrated manner in several recommendation cases.

The first relevant conclusion of the implementation of the first components is that the architecture depends a lot on the decisions made in the implementation, at the level of mechanisms, libraries, strategies, among others, used. For example, in the development of the components proposed in [45] and [43], we see that their results depend on the emotion representation scheme used, or on the sentiment analysis tool used (in our case, Senticnet).

5.3 Case Study

This section presents an example of the functioning of the pilot of the architecture used in [35] that integrates the implemented components described in the previous section. For the case study, the proposed architecture was applied in an academic course on programming algorithms for the training process of software engineering students. The course is attended by 22 students, 14 men and 8 women.

The course is made up of 12 general topics to cover in 16 weeks of academic activity, using learning resources made up of guides, books, workshops, laboratories, and evaluation content. The case study was applied for an initial recommendation of learning resources, using a classification of emotions of the academic material stored in the LMS—Moodle, and the learning styles, personalities, and emotions of the students. In the first case, the emotion engine of academic resources is used [43], and in the second case, the engines of personal characteristics and human emotion are used [45].

According to the emotion engine of academic resources [43], for each of the documents, the keywords were identified with the tf–idf algorithm (from term frequency–inverse document frequency), which were used to obtain the polarity of these documents using Senticnet 5 [10]. Particularly, all emotions are extracted for all sentences containing keywords, grouped by keyword, and then, a weighted mean polarity is calculated using as a

Table 1 Example of classification of academic documents

Course topic	Resource	Emotion
Introduction to programming algorithms	Fun1.pdf	Neutral
	Intro1Al.pdf	Happy
	Casos.pdf	Worried
Conditionals	Cond1.pds	Happy
	Example	Confused
	Workshop	Neutral

variable the number of keywords the sentence contains. The more keywords the sentence contains, the more relevant it is to the average. After this weighted average, a unique emotion per keyword is obtained, and finally, all these emotions per keyword are averaged to obtain a final emotion per document. This is the dominant emotion in the document, stored in the database for later use. Table 1 provides an example of the emotion classification for some learning resources of the teaching–learning process that make up the selected course.

For the determination of personal characteristics, the following procedure is used. For the registration of the course, the students filled out a form for the identification of the learning style, applying the learning style model proposed by Felder–Silverman [21] (learning style: active/reflective–sensory/intuitive–visual/verbal–sequential/global). After identifying the learning style, the psychometric test 16PF5 [11] was applied, for the identification of the students' personality, with the aim of using it in the recommendation process. Table 2 provides an example of

the results of the learning styles and the personalities (Ex: extraversion; Ax: anxiety; Tm: directness; In: independence; SC: self-control).

As already mentioned, the recommendation of the learning resources will only be made for the beginning of the course, taking into account the initial emotion of the student during the teaching–learning process detected by the human emotion engine [45]. Thus, after storing the previous information, the recommendation system used the content filter algorithm with the aim of determining the learning resources to be recommended according to the student's learning style, emotion, and personality [54].

Figure 16 shows an example of the results obtained in the recommendation process in the case study. For each of the course topics, there are several content options that the recommendation system assigns to the student according to their learning style, personality and emotion. The learning resources are represented by a color, depending on its emotion classification by using the colors represented by the hourglass (see Fig. 8) proposed by [51].

A first general conclusion based on the results of the case study is the ability of our approach to analyze the polarity of academic documents based on their content. The emotion resource engine is able to determine the emotion conveyed/implicit in the document (see Table 1). A second general conclusion of our approach is the ability to define the personal characteristics of students, which include their emotions and learning styles 2. To do this, our approach uses different engines, such as the recognition of human emotions.

Finally, using all this information, our approach is capable of recommending learning resources to students for each

Table 2 Examples of student characteristics (personality and learning styles)

Student	Learning style	Personality				
		Ex	Ax	Tm	In	Sc
Student 1	Sequential/global	4,9	5,8	8,6	4,4	5,4
Student 2	Active/reflective	4,6	5,2	5,3	5,1	4,9
Student 3	Sensory/intuitive	5,1	5,6	6,3	4,3	4,8
Student 4	Active/reflective	6,2	6,1	7,2	5,5	5,1
Student 5	Active/reflective	4,6	6,6	8,0	5,3	5,0
Student 6	Sensory/intuitive	4,2	5,0	6,1	5,5	5,3

Fig. 16 Example of recommendation results

	Topic 1						Topic 2					Topic 3				Topic 4			
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16	R17	R18	R19
Student1		X									X			X			X		
Student2			X						X			X						X	
Student3				X				X					X					X	
Student4		X					X					X					X		
Student5	X					X			X					X					X
Student6		X								X									X
Student7					X			X							X				

topic of a course (see Fig. 16). It recommends them, considering emotions as a fundamental element of the process. It consistently recommends learning resources with similar emotions, for different topics, for a given student (see, for example, in Fig. 16, student 1). This even leads it not to give recommendations when emotionally adequate resources for the student's profile are not found (see, for example, in Fig. 16, students 6 and 7 and topic 3).

6 Comparison with previous works

In order to compare our work with similar previous works, a set of qualitative criteria of interest were used, identified in [44]:

- **A**: the work considers emotions from users,
- **B**: the work considers emotions from contents (i.e., the emotion the resource is generating in the users that interact with it),
- **C**: the work considers the personality traits for characterizing users,
- **D**: the work considers the learning style or preferences of users for making recommendations,
- **E**: the work considers the expertise level for characterizing users,
- **F**: the work uses the logs or interactions of users with the VLE for making suggestions of resources,
- **G**: the work uses the logs of recommendations for improving them (i.e., it considers whether the user liked or used the recommendations previously given, for improving future recommendations),
- **H**: the work considers information from multiple modalities for the recognition of the emotional state of users,
- **I**: the work provides automatic recommendations, that is, recommendations are built by the recommender system,
- **J**: the work does not only use the current emotional state, but also the historic emotional state for making recommendations, and finally,
- **K**: the work is flexible and can consider multiple recommendation strategies, like content-based, collaborative filtering, knowledge-based, among others.

Table 3 presents the comparison of our architecture with related works.

As given in Table 1, the only architecture that meets all criteria is the one proposed in the present paper. In general, all architectures consider the emotions of the users, but only our proposed approach and [12] also consider the emotion in the contents. The works [28, 37] use a discrete modeling approach to emotions, while the rest use a continuous one, among which those based on Sentic computing stand out ([18, 36] and the current study).

Moreover, it is not usual for papers to use the level of expertise or personal traits of users, nor do they consider multimodal approaches to recognize an emotion. For example, some of the few approaches that try to define a user profile mainly use approaches such as the Felder–Silverman model [47], or schemes based on behavioral metadata [37], without considering emotions. However, our work is the only one that considers the behavior (emotion), learning styles, and personality of users when using the recommendations as feedback, to improve future recommendations.

On the other hand, few works use the users' emotional history to self-adjust. For example, the work [37] uses the logs of Moodle. Our architecture is the only one that integrates different sources (for example, the interactions in the VLE) to enrich the recommendation process, and in this way, improve the personalization of the recommendations considering the emotions in the context (in users and resources).

For the recognition of the emotional state of users, normally, the face is used, but some also use written content on social media ([40, 47] and our work). On the other hand, some works use multiple recommendation strategies such as content-based and collaborative filtering in [28], or several types of collaborative filtering in [36], but our approach is the only one to mix different recommendation strategies with the current and the historic emotional state for making recommendations.

Finally, this architecture is easy to model using the multiagent systems paradigm [2, 3], which facilitates the modular development and subsequent integration of its components. This is what has allowed the development of some of the components described in the previous sections.

Table 3 Comparison with related works

	A	B	C	D	E	F	G	H	I	J	K
[28]	X								X	X	X
[47]	X			X				X	X		
[37]	X		X	X	X	X			X		
[36]	X	X		X		X			X		
[40]	X					X		X	X		X
[18]	X		X	X	X				X		
this work	X	X	X	X	X	X	X	X	X	X	X

7 Conclusions

The architecture proposed in this work is flexible enough for supporting multiple implementation ideas of affective recommender systems, even for implementations where a recommendation is not necessary, like in the first use case where the final objective was extracting the dominant emotion in the contents of resources.

In this architecture, the emotion engine is separated from the personal characteristics engine since although emotions are personal, they present a lot of variance and the learning process is very sensitive to the emotional state of the learner, as it has been shown in related works [41]. In addition, the emotional state of a person changes much more frequently than expertise level, learning style, and personality.

In the first use case, the user resource emotion log database is not considered because the historic information of the user is not taken into account. In the second use case, the cold start can be a serious problem at the beginning of the system for making recommendations, but it can be attacked using the information generated by the first use case at the beginning. For the fourth use case, a lot of tagged information is needed for training the Doc2Vec model. Also, additional personal information could be considered, like personality traits, learning styles, and expertise level of a student on the documents.

Aspects that are not addressed in this work, and are important for the implementation of the architecture, include the communication between components and subcomponents, which can follow the principles of service-oriented design for a low coupling. Another aspect is the representation and storage of the information used by our architecture, like the emotional states, learning style, among others. Just as an example, for representing the emotional state many models exist and there is no consensus on which is better, or what emotions to use. In each use case, the model for representing emotions can be different, for example for use cases one and two, the model must be continuous to determine the average, but for use cases three and four, the model can be discrete. Future works should analyze in depth which emotional models are most appropriate for each component of our architecture.

Future works must also consider the implementation of the human emotion engine, dealing with the multimodality problem of recognizing emotions. Moreover, future work must prove the real performance of the architecture in a real scenario, evaluating how much the student's performance raises when considering affective characteristics for making recommendations of contents. Another suggestion for work is the analysis of the feedback mechanisms based on emotions to improve the recommendations, and the analysis of our architecture to support students with disabilities, such as students with dyslexia, autism, and blindness. Finally, other works must consider how to integrate this architecture in more advanced recommendation systems, like intelligent recommender systems or autonomous recommender systems [34].

References

1. Aguilar, J., Buendia, O., Pinto, A., Gutiérrez, J.: Social learning analytics for determining learning styles in a smart classroom. Interactive Learn. Environ. **30**(2), 245–261 (2022). https://doi.org/10.1080/10494820.2019.1651745

2. Aguilar, J., Narciso, F.and Hidrobo, F., Bessembel, I., Cerrada, M.: Una metodología para el modelado de sistemas de ingeniería orientado a agentes. Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial **12**, 39–60 (2008). https://www.redalyc.org/articulo.oa?id=92513102003

3. Aguilar, J., Cerrada, M., Hidrobo, F.: A methodology to specify multiagent systems. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) Agent and Multi-Agent Systems: Technologies and Applications. pp. 92–101 (2007)

4. Aguilar, J., Jerez, M., Exposito, E., Villemur, T.: Carmicloc: Context awareness middleware in cloud computing. In: 2015 Latin American Computing Conference (CLEI). pp. 1–10 (2015). 10.1109/CLEI.2015.7360013

5. Aguilar, J., Salazar, C., Velasco, H., Monsalve-Pulido, J., Montoya, E.: Comparison and evaluation of different methods for the feature extraction from educational contents. Computation **8**(2), 30 (2020). https://doi.org/10.3390/computation8020030

6. Alhathli, M., Masthoff, J., Beacham, N.: Adapting learning activity selection to emotional stability and competence. Front. Artif. Intell. **3** (2020). https://doi.org/10.3389/frai.2020.00011

7. Ali, S., Hafeez, Y., Humayun, M., Jamail, N.S.M., Aqib, M., Nawaz, A.: Enabling recommendation system architecture in virtualized environment for e-learning. Egypt. Inf. J. **23**(1), 33–45 (2022)

8. Bawden, D., Holtham, C., Courtney, N.: Perspectives on information overload. In: Aslib Proceedings. MCB UP Ltd (1999)

9. Bechara, A., Damasio, H., Damasio, A.R.: Emotion, decision making and the orbitofrontal cortex. Cerebral Cortex **10**(3), 295–307 (2000)

10. Cambria, E., Poria, S., Hazarika, D., Kwok, K.: Senticnet 5: Discovering conceptual primitives for sentiment analysis by means

of context embeddings. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)

11. Cattell, R.B., Russell, M.T., Karol, D.L., Cattell, A.K.S., Cattell, H.E.: 16 PF-5. Tea (2000)

12. Chamba-Eras, L., Aguilar, J.: Augmented reality in a smart classroom-case study: Saci. IEEE Revista Iberoamericana de Tecnologias del Aprendizaje **12**(4), 165–172 (2017). https://doi.org/10.1109/RITA.2017.2776419

13. Chen, M., Yang, J., Zhou, J., Hao, Y., Zhang, J., Youn, C.H.: 5g-smart diabetes: toward personalized diabetes diagnosis with healthcare big data clouds. IEEE Commun. Mag. **56**(4), 16–23 (2018)

14. Clore, G.L., Schwarz, N., Conway, M.: Affective Causes and Consequences of Social Information Processing, vol. 1, pp. 323–417. Lawrence Erlbaum Associates, Inc (1994)

15. Cordero, J., Aguilar, J., Aguilar, K., Chávez, D., Puerto, E.: Recognition of the driving style in vehicle drivers. Sensors **20**(9), 2597 (2020). https://doi.org/10.3390/s20092597

16. Costa Jr, P.T., McCrae, R.R.: The Revised NEO Personality Inventory (NEO-PI-R). Sage Publications, Inc (2008)

17. Dwivedi, P., Kant, V., Bharadwaj, K.K.: Learning path recommendation based on modified variable length genetic algorithm. Educ. Inf. Technol. **23**(2), 819–836 (2018)

18. Ezaldeen, H., Misra, R., Bisoy, S.K., Alatrash, R., Priyadarshini, R.: A hybrid e-learning recommendation integrating adaptive profiling and sentiment analysis. Web Semant. **72**(C) (apr 2022)

19. Fatahi, S.: An experimental study on an adaptive e-learning environment based on learner's personality and emotion. Educ. Inf. Technol. **24**(4), 2225–2241 (2019)

20. Feidakis, M., Daradoumis, T., Caballé, S.: Emotion measurement in intelligent tutoring systems: what, when and how to measure. In: 2011 Third International Conference on Intelligent Networking and Collaborative Systems. pp. 807–812. IEEE (2011)

21. Felder, R.M., Silverman, L.K., et al.: Learning and teaching styles in engineering education. Eng. Educ. **78**(7), 674–681 (1988)

22. García-Peñalvo, F.J., Seoane Pardo, A.M.: Una revisión actualizada del concepto de elearning. décimo aniversario. Educ. Knowl. Soc. (EKS) **16**(1), 119-144 (mar 2015). https://doi.org/10.14201/eks2015161119144

23. Gros, B., García-Peñalvo, F.J.: Future trends in the design strategies and technological affordances of e-learning. In: Learning, Design, and Technology. An International Compendium of Theory, Research, Practice, and Policies (2016)

24. Jiménez, M., Aguilar, J., Monsalve-Pulido, J., Montoya, E.: An automatic approach of audio feature engineering for the extraction, analysis and selection of descriptors. Int. J. Multimed. Inf. Retr. **10**, 33–42 (2021)

26. Katarya, R., Verma, O.P.: Recent developments in affective recommender systems. Phys. A Stat. Mech. Appl. **461**, 182–190 (2016)

27. Le, Y., Liu, J., Deng, C., Dai, D.Y.: Heart rate variability reflects the effects of emotional design principle on mental effort in multimedia learning. Comput. Human Behav. **89**, 40–47 (2018)

28. Leony, D., Parada Gélvez, H.A., Munoz-Merino, P.J., Pardo Sánchez, A., Delgado Kloos, C.: A generic architecture for emotion-based recommender systems in cloud learning environments. J-jucs (2013)

29. Liu, T., Wu, Q., Chang, L., Gu, T.: A review of deep learning-based recommender system in e-learning environments. Artif. Intell. Rev. **55**(8), 5953–5980 (2022). https://doi.org/10.1007/s10462-022-10135-2

30. Louvigné, S., Uto, M., Kato, Y., Ishii, T.: Social constructivist approach of motivation: social media messages recommendation system. Behaviormetrika **45**(1), 133–155 (2018)

31. Marcos-Pablos, S., Lobato, F., García-Peñalvo, F.: Integrating emotion recognition tools for developing emotionally intelligent agents. Int. J. Interactive Multimed. Artif. Intell. **7**(6), 69–76 (2022)

32. Méndez, N.D.D., Zapata, Á.M.P., Collazos, C.A.: A learning object recommendation model with user mood characteristics. In: International Workshop on Social Computing in Digital Education. pp. 39–48. Springer (2015)

33. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)

34. Monsalve-Pulido, J., Aguilar, J., Montoya, E., Salazar, C.: Autonomous recommender system architecture for virtual learning environments. Appl. Comput. Inf. (2020)

35. Montoya-Múnera, E., Aguilar, J., Monsalve-Pulido, J.A., Salazar, C., Varela-Tabares, D., Jiménez-Narváez, M., Montoya-Jaramillo, E.: Toward the application of artificial intelligence in academic content: An autonomous recommendation system. Education 4.0 A view from different digital proposals, p. 12 (2020)

36. Narducci, F., De Gemmis, M., Lops, P.: A general architecture for an emotion-aware content-based recommender system. In: Proceedings of the 3rd Workshop on Emotions and Personality in Personalized Systems 2015, pp. 3–6 (2015)

37. Neji, M., Ammar, M.B., Alimi, A.M.: Real-time affective learner profile analysis using an emaspel framework. In: 2011 IEEE Global Engineering Education Conference (EDUCON). pp. 664–670. IEEE (2011)

38. Perozo, N., Aguilar, J., Teran, O., Molina, H.: An affective model for the multiagent architecture for self-organizing and emergent systems (MASOES). Revista Tecnica de la Facultad de Ingenieria **35**, 80–90 (2012)

39. Poria, S., Cambria, E., Hussain, A., Huang, G.B.: Towards an intelligent framework for multimodal affective data analysis. Neural Netw. **63**, 104–116 (2015)

40. Qian, Y., Zhang, Y., Ma, X., Yu, H., Peng, L.: Ears: emotion-aware recommender system based on hybrid information fusion. Inf. Fusion **46**, 141–146 (2019)

41. Qin, J., Zheng, Q., Li, H.: A study of learner-oriented negative emotion compensation in e-learning. J. Educ. Technol. Soc. **17**(4), 420–431 (2014)

42. Rahayu, N.W., Ferdiana, R., Kusumawardani, S.S.: A systematic review of ontology use in e-learning recommender system. Comput. Educ. Artif. Intell. **3**, 100047 (2022). https://doi.org/10.1016/j.caeai.2022.100047

43. Salazar, C., Aguilar, J., Monsalve-Pulido, J., Montoya, E.: Análisis de sentimientos/polaridad en diferentes tipos de documentos. Revista Ibérica de Sistemas y Tecnologías de la Información **E41**, 344–357 (2021)

44. Salazar, C., Aguilar, J., Monsalve-Pulido, J., Montoya, E.: Affective recommender systems in the educational field. A systematic literature review. Comput. Sci. Rev. **40**, 100377 (2021)

45. Salazar, C., Montoya-Múnera, E., Aguilar, J.: Analysis of different affective state multimodal recognition approaches with missing data-oriented to virtual learning environments. Heliyon **7**(6), e07253 (2021)

46. Sánchez, M., Aguilar, J., Cordero, J., Valdiviezo-Díaz, P., Barba-Guamán, L., Chamba-Eras, L.: Cloud computing in smart educational environments: Application in learning analytics as service. In: Rocha, Á., Correia, A.M., Adeli, H., Reis, L.P., Mendonça Teixeira, M. (eds.) New Advances in Information Systems and Technologies, pp. 993–1002. Springer International Publishing, Cham (2016)

47. Santos, O.C., Boticario, J.: Affective issues in semantic educational recommender systems. In: RecSysTEL@ EC-TEL. pp. 71–82 (2012)

48. Santos, O.C., Boticario, J.G., Manjarrés-Riesco, Á.: An approach for an affective educational recommendation model. In: Recommender Systems for Technology Enhanced Learning, pp. 123–143. Springer (2014)

49. Shen, L., Wang, M., Shen, R.: Affective e-learning: Using "emotional'' data to improve learning in pervasive learning environment. J. Educ. Technol. Soc. **12**(2), 176–189 (2009)

50. Sánchez, H., Aguilar, J., Terán, O., Gutiérrez de Mesa, J.: Modeling the process of shaping the public opinion through multilevel fuzzy cognitive maps. Appl. Soft Comput. **85**, 105756 (2019). https://doi.org/10.1016/j.asoc.2019.105756

25. Suk Jung, K., suk Choi, Y.: Brain wave and user profile based learning content type recom-mendation in interactive e-learning environment. Int. J. Smart Home **6**(3), 33–40 (2012)

51. Susanto, Y., Livingstone, A., Ng, B.C., Cambria, E.: The hourglass model revisited. IEEE Intell. Syst. **35**(5) (2020)

52. Tkalčič, M., Burnik, U., Odić, A., Košir, A., Tasič, J.: Emotion-aware recommender systems–a framework and a case study. In: International Conference on ICT Innovations. pp. 141–150. Springer (2012)

53. Vallejo-Correa, P., Monsalve-Pulido, J., Tabares-Betancur, M.: A systematic mapping review of context-aware analysis and its approach to mobile learning and ubiquitous learning processes. Comput. Sci. Rev. **39**, 100335 (2021)

54. Varela, D., Aguilar, J., Monsalve-Pulido, J., Montoya, E.: Propuesta arquitectónica de un sistema de recomendación híbrido adaptativo. Revista Ibérica de Sistemas y Tecnologías de la Información **E38**, 171–184 (2020)