Computers & Education 53 (2009) 1147–1154

Contents lists available at ScienceDirect

Computers & Education

journal homepage: www.elsevier.com/locate/compedu

Intelligent assistance for teachers in collaborative e-learning environments

Agustin Casamayor *, Analia Amandi, Marcelo Campo

ISISTAN Research Institute, UNICEN University, Campus Universitario, Paraje Arroyo Seco, B7001BBO Tandil, Bs. As., Argentina
CONICET, National Council for Scientific and Technical Research, C1033AAJ, Bs. As., Argentina

ARTICLE INFO

Article history:
Received 10 October 2008
Received in revised form 25 May 2009
Accepted 26 May 2009

Keywords:
E-learning
Distance learning
Collaborative work
Intelligent assistance
Software agents

ABSTRACT

Collaborative learning environments provide a set of tools for students acting in groups to interact and accomplish an assigned task. In this kind of systems, students are free to express and communicate with each other, which usually lead to collaboration and communication problems that may require the intervention of a teacher. In this article, we introduce an intelligent agent approach to assist teachers through monitoring participations made by students within a collaborative distance learning environment, detecting conflictive situations in which a teacher's intervention may be necessary. High precision rates achieved on conflict detection scenarios suggest great potential for the application of the proposed rule-based approach for providing personalized assistance to teachers during the development of group works.

1. Introduction

The great progress achieved in communication possibilities through information networks like the Internet, along with the development of numerous distance learning systems (Gilbert & Han, 1999; Papanikolaou & Grigoriadou, 2003; Peña, Marzo, & de la Rosa, 2003), has given place to the improvement of teaching and learning processes (Akhtar & Wyne, 2006). The basic idea of these systems is that teachers and students placed in different geographic places, can interact and achieve the learning process in a non-present way (Offir, Lev, & Bezalel, 2008), offering the continuous education of people who, for diverse reasons, are not able to attend to regular courses.

The technological development and the efforts addressed to improve education have contributed to convert distance learning platforms into a very interesting field for computer researchers and software engineers (Littlejohn, Falconer, & Mcgill, 2008). As these systems are used by different kinds of students with diverse abilities and preferences, the main desired characteristics for them are adaptability and personalization (Alrifai, Dolog, & Nejdl, 2006; Schiaffino & Amandi, 2004).

Besides the individual work a student can carry out using an e-learning platform, it is important to take into account the development of group works carried out in collaboration with other peers (Wessner & Pfister, 2001). Collaboration stimulates learning, increases motivation, promotes feelings of belonging to a team, encourages creativity, eases communication and increases the achieved personal satisfaction for the educative process (Plantamura, Roselli, & Rossano, 2004).

When using a virtual collaborative work environment, the basic interaction mechanism students have to collaborate with their peers is through proposals and counter-proposals within a structured discussion. A structured discussion consists in a workspace similar to a forum in which students can make proposals and vote based on fixed topics. In this case, these topics are the tasks proposed by the teacher for the collaborative work. Communication is asynchronous, that is to say, users communicate through messages that can differ considerably in time and space. Once the discussion about the way tasks are going to be performed is finished, students can edit their collaborative work in a Wiki-like[1] environment.

A student that proposes a way to face the solution of a problem usually finds three kinds of answers: (1) the acceptance from their peers, implying the development of the activity in the way that was suggested; (2) the counterpart is rejection, a member of the group does not agree with the suggestion enabling the possibility of presenting another alternative or continuing with the original proposal; and (3) a third possibility, that usually comes attached to a rejection, is the counter-proposal, meaning that there is a disagreement with the original idea, but an alternate way to face the problem is suggested. This last alternative is the most broadening, since it encourages communication and

* Corresponding author. Tel.: +54 9223 5625968.
E-mail addresses: acasamay@exa.unicen.edu.ar (A. Casamayor), amandi@exa.unicen.edu.ar (A. Amandi), mcampo@exa.unicen.edu.ar (M. Campo).

[1] A Wiki is a piece of software that allows multiple authors to edit web page content.

0360-1315/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compedu.2009.05.025

enables cross fertilization of ideas (Constantino Gonzalez, Suthers, & Escamila de los Santos, 2003). In this way, students in the same group can interact until they arrive to a solution for the proposed problem. It can also appear a fourth possibility known as indifference, that is, the non-participation in discussions.

Even though students have freedom to express and communicate without restrictions, the idea of supervision emerges naturally (Kosba, Dimitrova, & Boyle, 2007). Thus, teachers have the possibility to intervene whenever they consider it necessary, helping the regular course of the activities. Particularly, the approach discussed in this article focuses on providing assistance to teachers for fostering the development of the activities within groups (starting from the student's behavior in that group as well as in other group works with different mates, alerting them about conflictive situations) so that teachers can intervene in the cases they consider it necessary. In this work we used an approach based on intelligent agents, which combines both conflict detection and assistance to the human teacher through alerts.

The remaining of the article is organized as follows: the following section focuses on our approach of intelligent assistance for teachers on e-learning environments, explaining some basic concepts related to Computer Supported Collaborative Work (CSCW) applied to distance learning platforms and intelligent assistance. In Section 3 we explain how we materialized our approach, focusing on students' profiles, conflict detection and alert notifications. Then, in Section 4 we show the experimental results obtained using this approach. Section 5 summarizes some related works. Finally, concluding remarks are stated in Section 6.

2. Intelligent assistance approach

Computer Supported Collaborative Work (CSCW) systems provide the necessary support in the use of communication services for sharing information and finding appropriate users to collaborate (Mandviwalla & Grillo, 1994). E-learning technologies are effective if they are conceived and used with the express purpose of promoting learning and collaboration (Kligyte & Leinonen, 2001), focusing in groups' dynamics and in group-based activities to facilitate learning (Soller, Goodman, Linton, & Gaimari, 1998). However, building a group is not enough for students to collaborate (Qu & Nejdl, 2001). Effective collaboration involves a set of conversational skills to motivate, inform and require knowledge (Soller, Lesgold, Linton, & Goodwin, 1999). These abilities acquire special importance in virtual learning environments where students are not able to meet face to face (Ayala & Yano, 1996). Hence, to achieve an effective collaboration, it is necessary to know which of these abilities students intervening in a learning community have and which they have not, as a means to personalize learning experiences in collaborative environments.

Web based e-learning platforms provide the appropriate support to develop such collaboration approach, especially through shared workspaces, Wiki edition and other communication tools, such as forums and chats, where group members can exchange ideas and make progresses in developing activities collaboratively (Sheremetov & Guzmán Arenas, 2002). In the context of collaborative environments explained before, teachers play a fundamental role in the learning process, having to guide the groups in their activities and in the communication among members. In this scenario, the idea of assistance comes up naturally, not only for the students, but also for the teachers in charge of courses and works (Kosba et al., 2007).

In traditional distance learning platforms, a teacher may be able to control the activities and collaboration of a particular group or student by manually browsing their interactions. This approach is known as "control board", after its similarity with a human operator monitoring a machine. This is an extremely difficult, repetitive and time consuming task, since teachers have to considerate all individual groups and students, being particularly careful with some cases in which their expertise and intuition tell them about a possible collaboration problem. Following the same analogy as before, this problem might be tackled with the existence of some kind of "secretary" or "assistant" helping them to perform this task. That is to say, if the system could be able to learn the teachers' knowledge on detecting problems instead of having to inspect them manually, it can automatically warn them about conflictive cases. Needless to say, the second approach would be dramatically more convenient for teachers, relieving them from a lot of tedious and monotonous work.

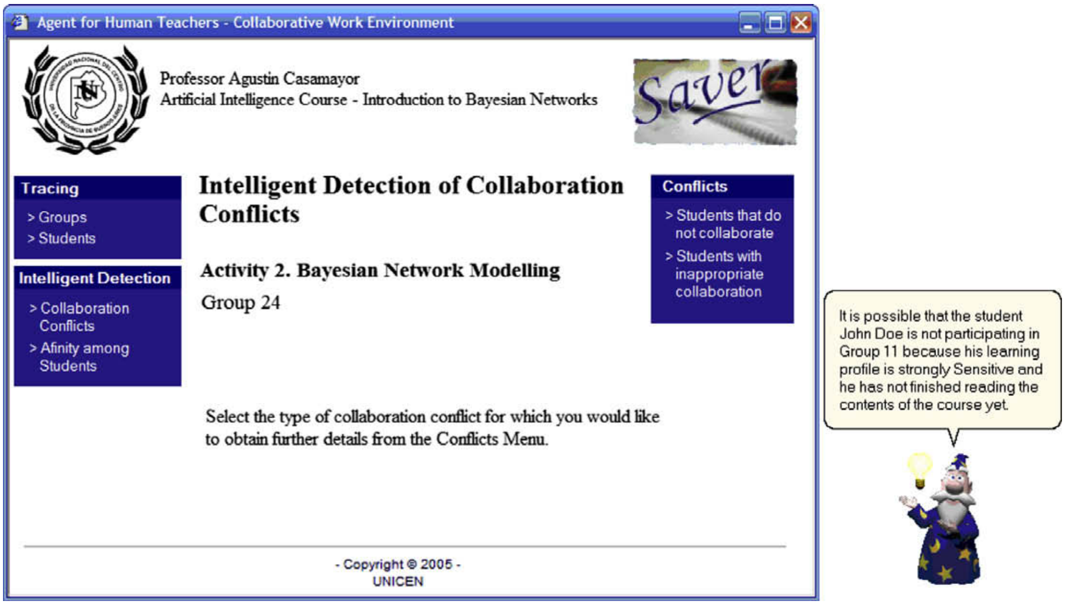
Intelligent assistance is a very promising research field of the Artificial Intelligence (AI) area (Lieberman, 1997; Schiaffino & Amandi, 2004), that is being quickly adopted by commercial software applications. Interface agents are computer programs that provide personalized assistance to users dealing with computer based applications (Jennings, Sycara, & Wooldridge, 1998). Regular tasks or control board interfaces can be successfully replaced by enhanced assistant-like user interfaces that politely warn the user about something that is going on in the application or suggest a possible way to proceed given a certain state in the execution.

The goal of our approach is the materialization of an intelligent interface agent, capable of providing the teacher with the necessary assistance to supervise and detect conflictive cases in collaborative works that are being carried out during the development of a course, giving an appropriate alert, so that the teacher can intervene when he or she considers it convenient. Fig. 1 shows a sample snapshot of the distance learning system uses in this work and the agent alerting the teacher about a collaboration conflict within the interface.

To assist teachers effectively, the proposed collaborative work interface agent has to be capable of providing them with a summary of the individual progress of each group member, which could be an indicator of the level of knowledge about topics of the course, and the type of participations students have had in their work groups, that is to say, the amount and quality of their proposes, acceptances, rejections, counter-proposals, or simply, an indifference attitude before ongoing discussions. Additionally, the agent counts with information about the Learning Style test of each student (Felder & Brent, 2005; Felder & Silverman, 1988), which defines their learning mechanism in four categories, considering both extremes for each case: intuitive/sensitive, sequential/global, active/reflexive and visual/verbal. Student's learning style can be defined by a standard test[2] taken by each student before starting their activities, or detected by the system based on the student's interactions (Garcia, Amandi, Schiaffino, & Campo, 2007).

If the characteristics of the action plan defined for a work and the tasks being held in a group do not match with the learning style of one or more students, this can influence negatively and has to be taken into account to notify the teacher. Considering this information about the particular learning style of each student, the global performance of a group, the attitude of each group member before the proposed work, the student's profile of behavior in that particular group and the history of attitudes of the student in other collaborative works, human teachers may receive alert messages from the agent, having to decide if they want to intervene and in which fashion, always trying to improve the accomplishment of the proposed activities as well as changing negative attitudes from one or more group members.

[2] http://www.engr.ncsu.edu/learningstyles/ilsweb.html.


Agent for Human Teachers - Collaborative Work Environment
Saver
Tracing
> Groups
> Students
Intelligent Detection
> Collaboration Conflicts
> Afinity among Students
Conflicts
> Students that do not collaborate
> Students with inappropriate collaboration

Professor Agustin Casamayor
Artificial Intelligence Course - Introduction to Bayesian Networks

Intelligent Detection of Collaboration Conflicts

Activity 2. Bayesian Network Modelling
Group 24

Select the type of collaboration conflict for which you would like to obtain further details from the Conflicts Menu.

- Copyright © 2005 -
UNICEN

It is possible that the student John Doe is not participating in Group 11 because his learning profile is strongly Sensitive and he has not finished reading the contents of the course yet.

Fig. 1. Sample snapshot of conflict detection and notification.

For instance, if in a group, a student whose learning style has a notorious sequential tendency does not take part in the ongoing discussions, considering the action plan defined for the work and the activity logs on the development of the tasks, the agent can detect that the group is not following the established order to solve the exercise, and that is the reason why the sequential student does not intervene. In that case, the agent sends an alert to the teacher, notifying about this situation. In this work we used an alert notification system that displays an animated character with warning messages of possible conflicts while the human teacher browses a certain group work. The final decision about the way to solve or at least lighten the problem has to be taken by the teacher, who will have to communicate with the involved groups or students.

3. Intelligent conflict detection and notification

The goal of this work is the materialization of an intelligent agent to assist teachers in a collaborative work environment. From an intelligent agent's system point of view, this assistance can be seen as the combination of different tasks. First, the agent has to gather information from each student's interaction with the system to build their profile, which models their relevant features for the system. Then, it processes this information, together with the profile and the student's learning style, in order to detect conflicts and elaborate a summary. Finally, the interface agent interacts with the teacher, notifying him or her about possible conflicts. A general schema of the information processed by the agent and the interaction with the teacher is shown in Fig. 2.

The gathering and the processing of information is done every time students interact with the system as a background task. The summary view and the notification of alerts are activated by the teacher on demand, either both at the same time or separately. In other words, the teacher may just want to use the summary tool to manually review the status of a group, work or student, deactivating or without paying attention to the notifying agent in the graphical user's interface. Also, the teacher may not be interested in the summary and just wait for the agent to alert about a conflictive situation.

The usual mechanism consists of receiving an alert from the agent and, therefore, inspecting manually the particular situation. Considering this information, teachers may decide to intervene, the same way they would do in a regular presence course when students are not making any progress with their activities. Details about each of these tasks are explained in the following subsections.

3.1. Logging interactions and building students' profiles

In the context of intelligent information agents, building user profiles is not a trivial problem (Godoy & Amandi, 2005). Our student's profile was built based on information about interactions with the distance learning environment. Every time students use the system, the agent silently logs their actions, that is, it observes each student's interaction with the system and registers them in an appropriate format to be processed later using Web Usage Mining (WUM) techniques (Srivastava, Cooley, Deshpande, & Tan, 2000).

This task has been successfully approached by student modeling techniques in some previous works (Kay, Maisonneuve, Yacef, & Zanane, 2006; Talavera & Gaudioso, 2004). Based on these works, Table 1 shows the attributes and values selected for the student's model. With this information, the agent builds and updates a profile of each student to recognize and track the activities carried out in the e-learning system.

It is very important to remark that these data is collected every time a student interacts with the system and the profile is built in advance so that the agent can access this information and process it to make inferences about a student's behavior.

This information is also accessible to the teacher through statistically grouped summaries and reports organized following two criteria: by groups or by students. The agent gathers and groups the information about the student's activities in order to present it with the highest possible level of detail. For instance, say we want to inspect the topics a student has read, in the "Read Topics" sections, topics' titles are

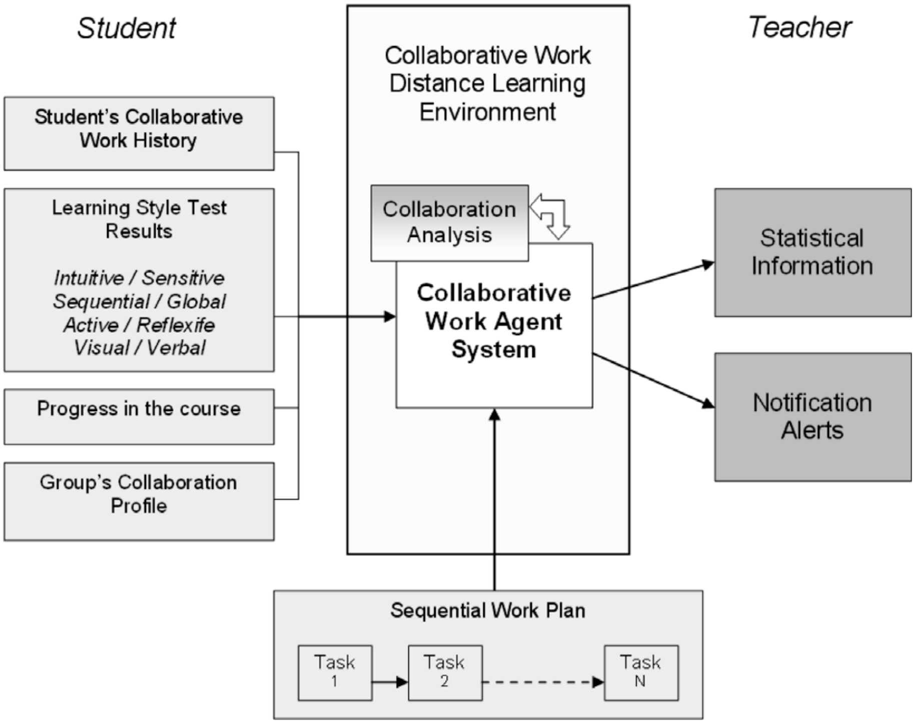


Fig. 2. General scheme of intelligent agent assistance.

Table 1
Student's profile.

Attribute	Values	Description
Read_curse	– All – Some – none	Amount of topics read from the content of the course
Tests_course	– All – Some – None	Amount of tests taken from the course
Selftests_course	– All – Some – None	Amount of self tests taken from the course
Part_group_discuss	– Yes – No	Participation of the student in structured discussions to develop the work. It includes all types of contributions, that is, proposals, counter-proposals, acceptances, rejects, comments and readings. If the student just reads the proposals but does not participate actively, this attribute is considered with a "no" value
Profile_sen_int	– Sensitive – Intuitive – Balanced	Tendency in the student's learning style. For each attribute, the learning styles test calculates a value between −11 and 11. The tendency was considered "strong" when the value was greater or equal to 7 (that is to say, between −11 and −7 and between 7 and 11). Intermediate values were considered balanced
Profile_act_ref	– Active – Reflective – Balanced	Same as above
Profile_sec_glo	– Sequential – Global – Balanced	Same as above
Profile_vis_ver	– Visual – Verbal – Balanced	Same as above

listed, along with the unit data and the type of explanation; when dealing with collaborations (as proposals or counter-proposals), a count is shown in conjunction with an optional detail of the text.

3.2. Detecting conflictive situations

The main goal of our agent is aiding the teacher during the difficult and time consuming task of manually tracking students' activities. While the system is still gathering information about each group and student in order to build their profile, the agent can speed up some queries grouping data and notifying some basic alerts about the existence of different problematic situations:

- students that do not participate or do not participate frequently (passive students);
- students with atypical participation (i.e. repeated behavior: accepting, rejecting or ignoring proposals most of the time and without proper justification);

- groups that have not finished any task;
- students that have not read any of the course's contents;
- groups that have not started any discussion.

Thus, the teacher avoids having to look over every particular case, obtaining the information quickly.

The second and most important function of the intelligent agent consists of a series of inferences it does based on context and behavioral data from the students' profile, which includes their learning style, their collaboration profile, the progress made in the course, and the group's collaboration profile (proposal voting order, acceptance or rejection time of a proposal, etc.).

In order to obtain domain knowledge to detect collaboration conflicts based on relationships that can exist within the interactions of students with the system, we produced association rules from actual use logs generated with previous usage experiences of the collaborative work environment in two real courses with more than 100 students working in groups (approximately 100,000 interactions per course).

Association rule mining (Agrawal, Imielinski, & Swami, 1993) is a well known AI technique used to find interesting associations and correlation relationships among large sets of data items, showing attribute-value conditions that occur frequently together in a given dataset. These rules provide probabilistic information in the form of "if–then" statements (Klemettinen, Mannila, Ronkainen, Toivonen, & Verkamo, 1994).

An association rule is an implication of the form $X \Rightarrow Y$, where $X \in I$ is a set of items, $Y \in I$ is a single item, and $X \cap Y = \emptyset$ (Agrawal et al., 1993). The rule $X \Rightarrow I_j$ is satisfied in the set of transactions T with the confidence factor $0 \leqslant c \leqslant 1$ if, and only if, at least $c\%$ of transactions in T that satisfy X also satisfy I_j. We will use the notation $X \Rightarrow I_j|c$ to specify that the rule $X \Rightarrow I_j$ has a confidence factor of c.

From a probability point of view, support can be seen as a measure of significance or importance of an itemset, represented by the joint probability of items in the consequent and the antecedent, as follows,

$$support(R) = p(X \cup I_j) \quad (1)$$

Confidence can interpreted as the probability of seeing the rule's consequent under the condition that the transactions also contain the antecedent, and it is represented by the conditional probability of I_j having occurred X in the transaction, as follows,

$$confidence(R) = p(Y \subseteq T | X \subseteq T) \quad (2)$$

Using an implementation of the Apriori algorithm (Agrawal & Srikant, 1994) included in the collection of Machine Learning (ML) algorithms provided by a tool called Weka[3], we generated a large set of association rules from our use logs. The attributes and possible values used as input to the association algorithm were intentionally the same ones we considered for the student's profile (Table 1).

After discarding repeated and redundant rules, we used a tool called ARTool[4] to apply filtering algorithms (Cristofor & Simovici, 2002; Liu, Hsu, Chen, & Ma, 2000) in order to obtain the most relevant rules, with both support and confidence values equal or grater than 90%. Taking into account that a student is not participating in the ongoing discussions (consequent part of the rule) as the triggering mechanism for the agent to find conflicts, we identified the following as possible reasons of collaboration conflicts:

- the student is strongly sequential and tasks are not being performed according to the order established in the task plan (Type 1);
- the student is strongly reflexive and tasks are performed very quickly, before she can actually read them (Type 2);
- the student is strongly sensitive and did not finish reading the contents of the course (Type 3).

Even though the AI technique used to perform conflict detection could have been applied dynamically for each particular case, we chose to use a selected number of association rules generated automatically from actual use logs, mainly due to efficiency and run-time issues. The obtained association rules actually represent very valuable domain knowledge, similar to what could have been provided by a teacher based on her expertise. The mechanism used to develop the detection feature can be easily extended, allowing the incorporation of new rules or modification of the existing ones, enhancing the potentially detectable conflicts. New rules could be generated either automatically or semi-automatically using other AI techniques as well as consulting an expert in the area.

When the teacher is inspecting a given a work and a group, the agent is able to provide an alert, notifying about possible conflicts by applying these rules. If certain students are not taking part in the assignment, the agent matches the antecedent of the rules with the corresponding values in their profiles, trying to find one or more collaboration conflicts. The corresponding notification is performed through an alert similar to "It is possible that the student John Doe is not participating because..." (Fig. 1), followed by the rule or rules that apply in each particular case. With this alert, the teacher is notified about the existing conflict the agent detected, having to take the decision of what action needs to be taken (e.g. sending an email to the student).

If the agent has enough historical information about students working in different groups, it can additionally advise the teacher about students that work well together or students that do not, considering that a group worked "right" or "wrong" not only if the obtained grade was a good or a bad one, respectively, but also taking into account individual quality of participations in group discussions.

3.3. Notifying alerts

In order to alert the teacher about any of the situations mentioned before in a user-friendly fashion, we used an animated character integrated with the web environment of the system, similar to the ones used by commercial applications. This character can interact with the user displaying custom messages on her screen, receiving input data, and even playing sounds and speaking.

The technology used to incorporate the animated character to the user's interface is *Microsoft Agent*,[5] which can be used in personal or academic projects without restrictions, following their end-user license agreement. It consists of a set of programmable software services that

[3] http://www.cs.waikato.ac.nz/ml/weka/.

[4] http://www.cs.umb.edu/~laur/ARtool/.

[5] http://www.microsoft.com/msagent/default.asp.

Table 2
Summary of conflict detection experiments.

Conflict type	Existent	Detected		Undetected	Precision %	Recall %
		Existent	Non-existent			
1. Sequential student and tasks are performed unsorted	200	198	3	2	98.5	99
2. Reflexive student and tasks are performed very quickly	200	195	1	5	99.5	97.5
3. Sensitive student and did not finish reading the contents of the course	200	197	4	3	98	98.5
Total	600	590	8	10	98.7	98.3

enable the use of interactive animated characters within a web or desktop environment. This software provides what is called "conversational interfaces", which take very good advantage of the natural aspects of human communication. Several characters are available, we chose a magician called Merlin, but it is possible to change it very easily because all of the characters provide a predefined set of actions and animations, defined in the *Ms. Agent* public interface. This conversational interface approach does not intend to replace in anyway traditional user interfaces. On the contrary, user's interaction with the animated character can be easily blended with the rest of the web or desktop environment.

Microsoft Agent's platform provides an *ActiveX* control that allows accessing its services through any programming language with *ActiveX* support, for instance, *JavaScript*. This means that the interaction can be even embedded in HTML web pages. The character downloads automatically into the user's computer via the application's web server, running as an *ActiveX* control, as long as it has permission. If we would like to add another communication mechanism to replace *Microsoft Agent*, it would just be necessary to re-implement the corresponding *JavaScript* that obtains the generated messages from the server.

In short, if the agent has something to tell the teacher and the animated character is visible and enabled, a message is displayed integrated into the system's graphical user interface. It is important to mention that the agent is visible and enabled by default, but the teacher can hide it, move it across the screen or even disable it at anytime.

4. Empirical evaluation

The agent that observes the behavior of the students and assists human teachers to perform collaborative activities was integrated in a distance learning web platform with support for collaborative work called SAVER.[6] It provides two separate sets of functionality, considering if the user is a teacher or a student. On the one hand, teachers can perform different actions. For instance, see the works' status, create new works, create ane edit groups for a work, modify existing works (add or delete files and tasks) and correct finished works. On the other hand, students have the possibility to participate in structured discussions, a collaborative chat room and edit ongoing works. The system was modified so that each student can have a related profile built from the attributes detailed before, such as participation in the course, performance in group works and the information about learning styles.

4.1. Validation of rule-based conflict detection

In order to analyze the accuracy of the agent with the alert notification system, we ran several tests with a set of artificially generated data-sets, based on real use logs. For preliminary experimental results, we chose to test the agent in this way with the purpose of testing only the detection without interference of possible input mistakes. This is a common methodology for evaluating and self-validating adaptive systems since it prevents from soiling the results with possible user-caused errors, considering a more controlled environment (Castillo, Gama, & Breda, 2003; Sunderam, 2002). In this case, we generated these test logs with hypothetical user profiles to simulate real user behavior.

The generated log data-sets were included 200 of each possible predefined conflict the agent can detect and 100 that it cannot. The evaluation of conflict detection rates of our agent was performed in terms of *precision* and *recall*, which are two standard measures used in Information Retrieval (IR) problems (Baeza-Yates & Ribeiro-Neto, 1999). In this particular case, we considered precision as the number of correctly detected existing conflicts divided by the total number of conflicts that were classified as existent. Similarly, recall was the number of correctly detected conflicts divided by the total number of actual conflicts in the test set. The experimental results are shown in Table 2.

As expected, regular non-participation conflicts were detected successfully almost every time. Regarding the cases that were not originally taken into account by the conflict detection agent, they were skipped successfully, but with the possibility to add the corresponding rules to consider them in the future.

4.2. Validation with real users

After having tested the conflict detection capabilities of the proposed rules mechanism, a second study was conducted with real users during a course on Object Oriented Programming. Approximately 80 students, randomly split into 20 groups, were assigned the same work and they were given one week to solve it collaboratively and hand it in using the software web platform mentioned before. The same human teacher supervised the development of these works, using the proposed agent for conflict detection with 10 of these groups, and not using it with the remaining 10 groups. Afterwards, a second work was assigned to the same groups, but this time, the teacher only used the intelligent agent for supervising those groups that had been previously supervised without it, so that each group could be evaluated both with and without intelligent conflict detection.

[6] SAVER stands for "Software de Asistencia Virtual para Educación Remota" in Spanish. Copyrighted to UNICEN, Tandil, Buenos Aires, Argentina (2005). Available at http://www.e-unicen.edu.ar/.

Table 3
Summary of experiments with real users.

	Using intelligent agent	Without using intelligent agent
Collaboration conflicts detected	23	6
Teacher's intervention for solving collaboration conflicts	83% (19)	100% (6)
Works handed in (in a timely fashion)	100% (20)	85% (17)
Approved works	90% (18)	75% (15)

The same teacher evaluated both sets of works for every group. Results were analyzed in terms of the number of collaboration conflicts detected and solved with the teacher's intervention and the amount of handed in and approved works. A summary of the obtained results is shown in Table 3.

The number of collaboration conflicts detected using the intelligent agent is remarkably higher than the one without using it, and even though the teacher's intervention for solving the conflict was not needed in every case, it was above 80%, which can be understood as a valuable and highly precise help for the teacher. Additionally, the percentages of success in terms of handed in and approved works for the groups that were supervised using the intelligent agent outperform the ones without the agent.

It is also worth mentioning that the teachers involved in the course felt comfortable with the agent, pointing out that they were not interrupted very often and they were pleased with the results of the assignments.

5. Related work

At the moment, diverse e-learning platforms similar to SAVER are being developed. The main characteristic concerning this work, is the support for collaborative work provided to teachers, which is available in the mentioned tool used to develop and perform experimental tests in this work. Some extra functionality has been added to perform the logging and control over the activities of the groups. Most tools just monitor students' performance in their groups and each group itself. Some examples are Amadeus (Tadesco & Gomes, 2002), HabiPro (Vizcafno, Contreras, Favela, &Prieto, 2000) and CollabSS (Borges & Baranauskas, 2003). These tools maintain a profile of the student and the groups, providing statistical information about the students' participation and their behavior within a group. Another similar example is ALFANET (Santos, Barrera, Gaudioso, & Boticario, 2003; Santos, Rodriguez, Gaudioso, & Boticario, 2003), which uses a multi-agent architecture that allows the tool to build and update several models such as the student, the group and the collaboration models. Personalization appears in different sections of the tool and it is customized by calculating collaboration indicators related to the performance shown by the students during the collaborative work tasks.

Other tools advise both the student and the teacher during the development of the activities and at the end of them. This is the case of DEGREE (Barros & Verdejo, 2000), a system that provides support for the development of learning tasks by small groups of students through different collaboration techniques. User interventions, called contributions, describe the process of argumentative discussion that enables to reach the construction of a collaborative solution to a problem. The system manages and stores all user interventions, that can be used lately to perform quantitative and qualitative analysis, not only about individual but also group performance. DEGREE offers the possibility of sending messages that, depending on the calculated collaborative indicators, can help the student to reflect and improve his way of working. This information is also available for teachers, who can therefore decide whether to intervene or not.

Compared to the tools mentioned before, the original contribution of our proposal consists of the agent's ability of capturing and considering a contextual behavior to make inferences as well as the incorporation of the results from the learning styles test to determine the collaborative profile of the student and the detection of conflictive situations, in which the teacher should intervene.

6. Conclusions

In this article we introduced an alert-based approach of intelligent assistance for teachers acting in collaborative work environments, with the addition of behavioral and context analysis for detecting conflictive collaboration situations. To fulfill this goal, we developed an intelligent agent integrated with a web-based distance learning platform for supporting collaborative work in distance learning education. The agent uses an extensible set of rules to detect collaboration conflicts among students and notify about them to teachers thorough on-screen alerts. Thus, the agent lightens this hard and time-consuming task that is carried out by teachers.

Log analysis for the generation of rules between the students' learning style and their profile enables the introduction of domain knowledge, pruning the obtained number of rules, and speeding up response time while performing rule-matching. Additionally, the mechanism used to detect conflicts allows the incorporation of new rules in a very easy way. New rules can be generated either automatically, semi-automatically or even provided by an expert in the area.

Conflict detection accuracy was validated both with artificial data (automatically generated logs according to pre-established profiles) and also with a controlled group of human users in a real course. Collaboration conflicts detected using the intelligent agent were remarkably higher than the ones without using it, showing that the intervention of a teacher to solve the conflict was needed more than 80% of the times. The high percentage of handed in and approved works for the groups that were supervised using the intelligent agent suggests that intelligent detection and intervention not only helps teachers while supervising groups, but also improves the final success of students in their assignments. Even though the results with real users are very promising, further experiments should be performed with more courses and larger groups of students in order to enhance conflict detection for different cases and provide more useful help to teachers.

References

Agrawal, R., Srikant, R. (1994). Fast algorithms for mining association rules in large databases. In Jorge B. Bocca, Matthias Jarke, & Carlo Zaniolo (Eds.), *VLDB'94: Proceedings of 20th international conference on very large data bases* (pp. 487-499). Santiago de Chile, Chile: Morgan Kaufmann.

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. In *SIGMOD'93: Proceedings of the 1993 ACM SIGMOD international conference on management of data* (pp. 207–516). New York, NY, USA: ACM Press [Vol. 2].

Akhtar, S., & Wyne, M. F. (2006). Distance learning education using the web. In *MATH'06: Proceedings of the 10th WSEAS international conference on applied mathematics* (pp. 565–571). Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS).

Alrifai, M., Dolog, P., & Nejdl, W. (2006). Learner profile management for collaborating adaptive elearning applications. In *APS'06: Proceedings of the joint international workshop on adaptivity, personalization & the semantic web* (pp. 31–34). New York, NY, USA: ACM.

Ayala, G., & Yano, Y. (1996). Learner models for supporting awareness and collaboration in a CSCL environment. In *Intelligent tutoring systems* (pp. 158–167).

Baeza-Yates, R. A., & Ribeiro-Neto, B. (1999). *Modern information retrieval*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.

Barros, M., & Verdejo, M. (2000). Analysing student interaction processes in order to improve collaboration. The DEGREE approach. *International Journal of Artificial Intelligence in Education, 11*, 221–241.

Borges, M. A. F., & Baranauskas, M. C. (2003). CollabSS: A tool to help the facilitator in promoting collaboration among learners. *Educational Technology & Society, 6*(1).

Castillo, G., Gama, J., & Breda, A. M. (2003). Adaptive bayes for a student modeling prediction task based on learning styles. In Peter Brusilovsky, Albert T. Corbett, & Fiorella de Rosis (Eds.), *User modeling. Lecture notes in computer science* (Vol. 2702, pp. 328–332). Springer.

Constantino Gonzalez, M., Suthers, D., & Escamila de los Santos, J. G. (2003). Coaching web-based collaborative learning based on problem solution differences and participation. *International Journal of Artificial Intelligence in Education, 13*.

Cristofor, L., & Simovici, D. (2002). Generating an informative cover for association rules. In *ICDM'02: Proceedings of the 2002 IEEE international conference on data mining (ICDM'02)* (pp. 597). Washington, DC, USA: IEEE Computer Society.

Felder, R. M., & Brent, R. (2005). Understanding student differences. *Journal of Engineering Education, 94*(1), 57–72.

Felder, R. M., & Silverman, L. K. (1988). Learning and teaching styles in engineering education. *Engineering Education, 78*(7), 674.

Garcia, P., Amandi, A., Schiaffino, S., & Campo, M. (2007). Evaluating bayesian networks precision for detecting students learning styles. *Computers and Education, 49*(3), 794–808.

Gilbert, J. E., & Han, C. Y. (1999). Arthur: Adapting instruction to accommodate learning style. In *World conference on the WWW and internet (WebNet 99)* (pp. 433–438).

Godoy, D., & Amandi, A. (2005). User profiling in personal information agents: A survey. *Knowledge Engineering Review, 20*(4), 329–361.

Jennings, N. R., Sycara, K., & Wooldridge, M. (1998). A roadmap of agent research and development. *Journal of Autonomous Agents and Multi-Agent Systems, 1*(1), 7–38.

Kay, J., Maisonneuve, N., Yacef, K., Zanane, O. (2006). Mining patterns of events in students' teamwork data. In *In educational data mining workshop, held in conjunction with intelligent tutoring systems (ITS)* (pp. 45–52).

Klemettinen, M., Mannila, H., Ronkainen, P., Toivonen, H., & Verkamo, A. I. (1994). Finding interesting rules from large sets of discovered association rules. In *CIKM'94: Proceedings of the third international conference on information and knowledge management* (pp. 401–407). New York, NY, USA: ACM.

Kligyte, G., Leinonen, T. (2001). Study of functionality and interfaces of existing CSCL/CSCW systems. ITCOLE project.

Kosba, E., Dimitrova, V., & Boyle, R. (2007). Adaptive feedback generation to support teachers in web-based distance education. *User Modeling and User-Adapted Interaction, 17*(4), 379–413.

Lieberman, H. (1997). Autonomous interface agents. In *CHI'97: Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 67–74). New York, NY, USA: ACM Press.

Littlejohn, A., Falconer, I., & Mcgill, L. (2008). Characterising effective elearning resources. *Computers & Education, 50*(3), 757–771.

Liu, B., Hsu, W., Chen, S., & Ma, Y. (2000). Analyzing the subjective interestingness of association rules. *IEEE Intelligent Systems, 15*(5), 47–55.

Mandviwalla, M., Grillo, P. (1994). TeamBox: Researching the groupware interoperability problem. In *CSCW'94: Demonstrations supplement to the CSCW'94 proceedings*.

Offir, B., Lev, Y., & Bezalel, R. (2008). Surface and deep learning processes in distance education: Synchronous versus asynchronous systems. *Computer & Education, 51*(3), 1172–1183.

Papanikolaou, K. A., Grigoriadou, M. (2003). An instructional framework supporting personalized learning on the web. In *Advanced learning technologies, 2003. Proceedings. The 3rd IEEE international conference on* (pp. 120–124).

Peña, C. I., Marzo, J. L., & de la Rosa, J. L. (2003). Intelligent agents in a teaching and learning environment on the web. In *ICALT2002*. Kazan, Russia: IEEE.

Plantamura, P., Roselli, T., & Rossano, V. (2004). Can a CSCL environment promote effective interaction? In *ICALT'04: Proceedings of the IEEE international conference on advanced learning technologies (ICALT'04)* (pp. 675–677). Washington, DC, USA: IEEE Computer Society.

Qu, C., & Nejdl, W. (2001). Constructing a web-based asynchronous and synchronous collaboration environment using WebDAV and Lotus Sametime. In *Proceedings of the 29th annual ACM SIGUCCS conference on user services (SIGUCCS'01)* (pp. 142–149). New York, NY, USA: ACM Press.

Santos, O., Barrera, C., Gaudioso, E., Boticario, J. (2003). ALFANET: An adaptive e-learning platform. In *Proceedings of the 2nd international meeting on multimedia and information communication technologies in education*.

Santos, O., Rodriguez, A., Gaudioso, E., Boticario, J. G. (2003). Helping the tutor to manage a collaborative task in a web-based learning environment. In *Proceedings of AIED'03 workshop on towards intelligent learning management systems*.

Schiaffino, S., & Amandi, A. (2004). User–interface agent interaction: Personalization issues. *International Journal of Human–Computer Studies, 60*(1), 129–148.

Sheremetov, L., & Guzmán Arenas, A. (2002). EVA: An interactive web-based collaborative learning environment. *Computers & Education, 39*(2), 161–182.

Soller, A., Goodman, B. A., Linton, F., & Gaimari, R. (1998). Promoting effective peer interaction in an intelligent collaborative learning system. In *Proceedings of the 4th international conference on intelligent tutoring systems (ITS'98)* (pp. 186–195). London, UK: Springer-Verlag.

Soller, A., Lesgold, A., Linton, F., & Goodwin, B. (1999). What makes peer interaction effective? Modeling effective communication in an intelligent CSCL. In Susan E. Brennan, Alain Giboin, & David Traum (Eds.), *Working papers of the AAAI fall symposium on psychological models of communication in collaborative systems* (pp. 116–124). Menlo Park, California: American Association for Artificial Intelligence.

Srivastava, J., Cooley, R., Deshpande, M., & Tan, P. (2000). Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations Newsletter, 1*(2), 12–23.

Sunderam, A. V. (2002). Automated personalization of internet news. In *AH'02: Proceedings of the second international conference on adaptive hypermedia and adaptive web-based systems* (pp. 348–357). London, UK: Springer-Verlag.

Tadesco, P., Gomes, A. (2002). Amadeus: A framework to support multi-dimensional learner evaluation. In *Proceedings of the 1st international conference on web-based learning*, China.

Talavera, L., Gaudioso, E. (2004). Mining student data to characterize similar behavior groups in unstructured collaboration spaces. In *Workshop on artificial intelligence in CSCL* (pp. 17–23), Valencia, Spain.

Vizcafno, A., Contreras, J., Favela, J., & Prieto, M. (2000). An adaptive, collaborative environment to develop good habits in programming. In Gilles Gauthier, Claude Frasson, & Kurt VanLehn (Eds.), *Intelligent tutoring systems, 5th international conference, ITS 2000, June 19–23, 2000, proceedings. Lecture notes in computer science* (1839, pp. 262–271). Montréal, Canada: Springer.

Wessner, M., & Pfister, H. R. (2001). Group formation in computer-supported collaborative learning. In *Proceedings of the 2001 international ACM SIGGROUP conference on supporting group work (GROUP'01)* (pp. 24–31). New York, NY, USA: ACM Press.