Learning Faults Detection by AIS Techniques in CSCL Environments

Amina Zedadra* and Yacine Lafifi
LabSTIC Laboratory, University 8 May 1945 Guelma, Algeria // BP 401, Guelma 24000, Algeria // zedadra_a@yahoo.fr // laf_yac@yahoo.fr

*Corresponding author

(Submitted August 22, 2013; Revised January 8, 2014; Accepted February 3, 2014)

ABSTRACT

By the increase of e-learning platforms, huge data sets are made from different kinds of the collected traces. These traces differ from one learner to another according to their characteristics (learning styles, preferences, performed actions, etc.). Learners’ traces are very heterogeneous and voluminous, so their treatments and exploitations are difficult, that make hard the tutors’ tasks. This paper introduces one of the bio-inspired computing techniques to improve the learning quality. In fact, Artificial Immune System (AIS) is a technique which was adapted for designing an assistant system that detects the wrong scenarios made by learners. Furthermore, this assistant system assists the learners in their activities. The main aim is to present the basic concepts of a new approach that aims at providing learners with relevant traces to improve their learning in order to minimize the tutor’s tasks. A novel algorithm is proposed to design the assistant system based on the two mechanisms of the AIS techniques (negative and clonal selection). The proposed algorithm was applied on a collaborative learning system called LETline 2.0 (http://www.labstic.com/letline/). An experiment was conducted in an Algerian University. The obtained results from this experiment were good and very efficient. The proposed approach enhances the cognitive and behavioral profiles of learners. In fact, the results show that the cognitive profiles of most students were improved. Also, it minimizes the tutor’s tasks.

Keywords

CSCL, AIS, Negative selection, Clonal selection, Assistant system

Introduction

In distance learning context, learning systems provide an environment where the learners can learn and interact with each other. These systems cultivate the abilities and the comportments of learners. However, this doesn’t mean that distance learning is perfect since it has many drawbacks, whereby a big number of learners are not successful at guiding their own learning process.

Every system has its own interface with heterogeneous content and different presentations of information. This fact leads to the emergence of a big problem in making the balance between the abilities of learners and the sight of the designers when developing the predictive scenarios. According to the heterogeneity of the interfaces, many learning systems cannot respond to the learners’ need and may waste time in wrong scenarios. Many studies (e.g., Djouad, 2011; Lafifi, Halimi, Herkas, Ghodabani, & Salhi, 2009; Sehaba, 2012) have revealed the use of learner’s traces (individual and collaborative activities among the users) to solve the problems mentioned above in order to boost the effectiveness of the collaborative learning systems. These traces are the results of the interaction between the actors, and the human actors and the system. However, traces are very voluminous and heterogeneous to be exploited manually by a human actor who is generally a tutor or a teacher. So, can the human actor manipulate all learners’ traces?

Our work context is the collaborative learning systems, where learners can learn and interact with each other to reformulate, test, refine, and repair their mental models (Hmelo-Silver, Jordan, Liu, & Chernobilsy, 2011). Also, it provides excellent opportunities for students to propose, support, evaluate, critique, and refine ideas in a more productive manner (Jeong, Clark, Sampson, & Menekse, 2011). In collaborative learning, learners need an additional support from human actors (teachers or tutors) to promote opportunities for effective collaboration. Besides, the collaboration in groups is difficult (Gweon, Jun, Lee, Finger, & Rosé, 2011), which confirms that an effective monitor and support is necessary to success the collaborative learning.

This research aims at providing answers to the following questions: Does the learner need an effective support and monitoring to improve his/her learning process? Is the tutor present usually when learners need monitoring? Is the
development of an assistant system for supporting and monitoring students can replace the human actor and can it success the collaborative learning?

We try to answer these questions in this research work. The latter belongs to the context of analysis and assistance of the learning situations to encourage students based on their traces. It aims at implementing an assistant system and integrating it in a CSCL (Computer-Supported Collaborative Learning) environment to improve the profiles of learners. For doing this task, the wrong scenarios must be found. For that, an algorithm of filtering traces is proposed to filter the irrelevant traces. Due to the problem context which aims at detecting the wrong scenario, bio-inspired techniques can be used. Indeed, the bio-inspired techniques are used in different fields: the anomaly detection in the computer security, detection of the application faults, optimization, pattern recognition, approximation functions, etc. More specifically, the Artificial Immune Systems (AIS) have important characteristics: detection, learning, memorization and cloning. So why not use the AIS techniques for the detection of the wrong scenarios. But, how can we adopt the AIS techniques to our problem? And is its integration gives good results?

The paper is organized in six sections. The section two presents the literature review. In section three, we give a brief overview of the AIS techniques in the field of e-learning and CSCL. Section four presents the architecture of the proposed system. In section five, we give the obtained results and the analysis of these results. Finally, section six presents a conclusion and the future works.

**Literature review**

Traces are defined in different ways according to the context of their application. Therefore, they have various definitions. Jermann and his team (Jermann, Soller, & Muehlenbrock, 2001) defined a trace as “an observation or a recording of the interaction of learners with a system for an analysis.” In 2005, another definition was proposed by Pernin (2005) where he defined the trace as “an index of actors’ activity, in an instrumented or not instrumented learning situation.” In a different way, Setoutti (2011) defined a trace as “an object collection, which is collected from an observation.”

According to the definitions given by previous authors, we propose the following definition that considers the trace as “a sequence that is defined by a series of actions done by the user when interacting with an environment.”

Traces are exploited to conceive: (1) analysis and assistance systems of learning situations which aimed at improving the task of actors monitoring (followed learners in their learning tasks and tutors in their different tasks, (e.g., Bousbia, 2011; Heraud, France, & Mille, 2004; Loghin, 2008), (2) engineering/reengineering of learning devices systems according to the information collected during the learning scenario in order to improve their quality (e.g., Diagne, 2009; Hussaan, Sehaba, & Mille, 2011; Luengo, Vadcard, Dubois, & Mufti-Alchawafa, 2006), and (3) adaptation/personalization of learning environment systems which adapt the contents of the learning platforms to the users (e.g., Guettat, Chorfi, & Jemni, 2010; Settouti, 2011).

In the context of learning improvement based on learner’s traces, Heraud and his team (Heraud et al., 2004) proposed Pixed, a research project which uses the logs of learners’ interactions. These interactions are joined together as learning episodes to help the learners who were trying to find their learning paths. In another approach, France and her co-authors (France, Heraud, Marty, & Carron, 2007) introduce the traces’ visualization interface by elaborating the ClassroomVis system. The latter allows the tutor to observe and adapt learner group’ activities. Loghin (2008) regulates the learner’s activities by developing an observation station, which allows the collection of the student’s traces based on multi-agents systems. Diagne (2009) has revealed the re-use of the indicators that provide the tutor with information to control the learning activities on the cognitive, pedagogical, social and technical level. Compared with the previous approach, Bousbia (2011) has used also the indicators calculated from the learners’ navigation but to provide the teachers not the tutors with a perception of their learners’ behavior and to identify their learning styles for assisting them. In a different approach, the main aim of the work of Sehaba (2012) is to develop an adaptive helping system based on interaction traces. It consists in considering the traces left by the users as knowledge sources that the system can use to generate adapted help to the target user.

In the studies cited above, the process of assistance is made by the tutor and posteriori which is a major problem when the learners need a help and the tutor is not present online to help them. To solve these problems, we propose
an assistant system to improve the learning situation of learner in real time, by assisting the learners who follow a wrong scenario.

Other studies have the same aim of improving the user’s abilities in different fields, such as information retrieval (e.g., Fitchett, Cockburn, & Gutwin, 2013; Mele, 2013), Computer-supported cooperative work (e.g., Tausczik & Pennebaker, 2013), serious games (e.g., Bououd & Boughzala, 2013), pair programming (e.g., Radermacher, Walia, & Rummelt, 2012), etc. Table 1 gives a summary of the related works.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Studies sample</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance learning</td>
<td>Helping learners in their learning processes</td>
<td>(Bousbia, 2011; Diagne, 2009; France et al., 2007; Heraud et al., 2004; Loghin, 2008; Sehaba, 2012)</td>
</tr>
<tr>
<td>Pair programming</td>
<td>Increase student performance in programming</td>
<td>(Radermacher et al., 2012)</td>
</tr>
<tr>
<td>CSCW</td>
<td>Improving the collaborative work</td>
<td>(Tausczik &amp; Pennebaker, 2013)</td>
</tr>
<tr>
<td>Serious games</td>
<td>Enhance the collaboration skills for a perfect team’s management</td>
<td>(Bououd &amp; Boughzala, 2013)</td>
</tr>
<tr>
<td>Information retrieval</td>
<td>Improve the research results and minimizing the time spent on retrieving</td>
<td>(Fitchett et al., 2013; Mele, 2013)</td>
</tr>
</tbody>
</table>

In recent years, many researchers have recognized the value of the use of AIS techniques in the field of Internet and web technologies. Artigues (Artigues, 2009) applies the AIS in the field of Virtual Reality Environments for Human Learning where he develops an intelligent tutoring system, which is based on the action recognition from users’ traces. In the field of web mining, Nasir and his colleagues (Nasir, A. Selamat, & H. Selamat, 2009) develop a system WMAIS (Web Mining using Artificial Immune System) that aims at recommending interesting and applicable news on the politics in Malaysia for the users. In another context, Romero and Nino (Romero & Nino, 2007) use the AIS for the extraction of the keywords from a document or a set of documents to enhance information retrieval. Also, AIS techniques are used in the field of Internet security (Chang, Venkatasubramanian, West, & Lee, 2013), spam filtering (Caruana & Li, 2012), pattern recognition (Hunt & Cooke, 1996) and anomaly detection (Gonzàles, 2003). According to the presence of these works, the current paper appears to be promising to develop and improve the models and algorithms using AIS techniques in e-learning.

**Overview of the AIS techniques**

**Basic concepts**

The field of Artificial Immune System has derived inspiration from many elements of the natural immune system to develop systems that operate in environments with constraints similar to those faced by the immune system (Hart & Davoudani, 2009). De Castro and Timmis (De Castro & Timmis, 2002) define the AIS as “the adaptive systems, inspired by the theories of the immunology, as well as the functions, the principles and the immune models, in order to be applied to the resolution of problems”.

The immunity is sub-divided into two distinct systems: *innate* immune system and *adaptive* immune system. The *adaptive* immune system has three principal processes (Timmis, Hone, Stibor, & Clark, 2008): negative selection, clonal selection and immune network. Whereas, Natural Dendritic Cells are the link between the *innate* and *adaptive* immune system.

**Negative selection**

The purpose of negative selection is to provide tolerance for self-cells (Aickelin & Dasgupta, 2005). The thymus is a gate against the non-self-antigens. The T cells presenting non-self-antigens are destroyed in this organ. All T cells retiring of the thymus and circulating in the body are said tolerantly towards the self.
Clonal selection

Clonal selection algorithm (De Castro & Von Zuben, 2000) is used by natural immune system to define the basic features of an immune response to an antigenic stimulus. It establishes the idea that only those cells that recognize the antigens are selected to proliferate. The selected cells are subject to an affinity maturation process, which improves their affinity to the selective antigens. The readers can read (De Castro & Von Zuben, 2002) for more details about the main stages of the clonal selection.

Immune network

Immune network theory is originally proposed by Jerne (Jerne, 1974). An artificial immune network is a bio-inspired informatics model that uses the ideas and the concepts of the immune network theory mainly the interaction between B cells and the cloning process. It receives an antigen as entry and sends back an immunized network compound of the B cells that are adjusting between them.

The immune network process is almost the same that the clonal selection, except that there exists a mechanism of deletion that destroys the cells having a certain threshold of affinity between them.

Dendritic Cells (DCs)

Dendritic Cells Algorithm (Greensmith, Aickelin, & Tedesco, 2010) is a second generation algorithm based on an abstract model of natural dendritic cells (DCs). It was introduced in 2005 (Greensmith, Aickelin, & Cayzer, 2005). Natural DCs are part of the innate immune system and are responsible for initial pathogen detection, acting as a vital link between the innate and the adaptive system.

AIS techniques in CSCL systems

In CSCL context, group interactions allow to improve the learners’ competencies through collaborative production, and break the insulation process by gathering learners using various communication tools: email, forum, blog, etc. CSCL systems provide users with collaborative tools that allow synchronous and asynchronous exchange between different actors. So, we have chosen to apply our algorithm in this field due to the great quantities of traces that are provided. The immune system uses many strategies to protect the body against all foreign molecules (non-self or antigen). According to its different features: mechanisms, self and non-self-discrimination, memorization, destruction of antigen, and so on, AIS techniques has been applied in different fields: computer security, optimization, pattern recognition, etc. Hence, we have chosen to apply the AIS techniques in the field of collaborative learning.

Our proposed solution is to develop a subsystem for detecting learning faults and assisting learners based on AIS techniques. The subsystem uses the negative and clonal selections to detect the irrelevant traces in order to improve the cognitive and behavioral profiles of learners.

Proposed filtering method

According to the difficulties and the limitations presented in the introduction and on the basis concepts of the AIS techniques (negative and clonal selection), we describe in this section our system architecture and our novel algorithm of filtering.

System overview

The principal thematic of our work concerns the support required to the human actors to improve learning quality. The fundamental idea is detecting the irrelevant traces and helping learners by identifying wrong scenarios using AIS
techniques, especially negative and clonal selection. The analogy between the natural immune system principle and the problem proposed above (Table 2) prompted us to develop our system.

**Table 2. Analogy between the natural immune system principle and the developed system**

<table>
<thead>
<tr>
<th>Natural Immune System</th>
<th>Artificial Immune System applied in our context</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self (antibody)</td>
<td>Relevant trace</td>
</tr>
<tr>
<td>Non-self (antigen)</td>
<td>Irrelevant trace</td>
</tr>
<tr>
<td>Lymphocyte (B- and T- cell)</td>
<td>Detectors</td>
</tr>
<tr>
<td>Detect the antigen</td>
<td>Detect the irrelevant trace</td>
</tr>
<tr>
<td>Tolerization</td>
<td>Negative selection mechanism</td>
</tr>
<tr>
<td>Cloning and memorization</td>
<td>Cloning and update the self-data</td>
</tr>
</tbody>
</table>

**Overall system architecture**

As it is indicated by several authors, the treatment of traces passes by three main steps: collection, analysis and exploitation. Our context of work is situated in the second and the last step. Indeed, in the second step (i.e., traces analysis), the traces collected in the first step are filtered. In the last step, the aim is to recommend learners with the relevant traces and to provide the tutors with a small set of learner’s traces. Figure 1 illustrates the system architecture and its different functionalities.

*Figure 1. System architecture*

**Collection of traces**

The system collects all learners’ actions. Traces can be resulting from the interaction learner/learner and learner/system. Also, other traces can be resulting from a collector of traces. We propose a novel classification of the types of traces, where we have divided them into five classes (Figure 2):

- Identification information: first name, family name, age, birthplace and class.
- Activities: learning activities, assessment and search for learning objects.
- Interaction and collaboration: assistance, synchronous collaborative tools, asynchronous collaborative tools and virtual meeting.
- Use: navigation on platform and on computer.
- Formation time: session’s duration, presence, habitual time for connection and consultation type (superficial, medium and deep).
Analysis of traces

- **Reformulation**: from the data set generated in the first step, a new data set is created where the traces will be reformulated according to the proposed format. The general format of the trace is defined as follows (Figure 3):

\[
T = (A_1, A_2, ..., A_n).
\]

Each action \( A \) is characterized by the following 5-tuples:

- \( t \): Type of the trace.
- \( N = (N_W, N_M, N_Y) \): Number of traces by week/month/year.
- \( D \): Date of the trace.
- \( H_B \): Trace beginning Hour.
- \( H_E \): Trace end Hour.

- **Fusion**: in order to obtain all the traces left by learners during their use of the system, a fusion step is proposed. It consists in obtaining a hybrid traces having all the recorded traces according to their time (Figure 4).
• *Cleaning*: to eliminate the noise (false response from the server for example: page not found, error 404), an algorithm of cleaning traces is proposed (see Algorithm 1).

**Algorithm 1: Cleaning traces**

**Input:** $T =$ a set of traces  
$NT =$ \{error 404, erreur 404, page non trouvée, page not found, etc.\} //It depends on the browser language (English, French, etc.).  
**Output:** $CT =$ a set of cleaned traces  
**Begin**  
For all (trace $t \in T$) then  
For all (trace $nt \in NT$) then  
If ($t$ matches $nt$) then  
Discard $t$  
Else  
Place $t$ in $CT$  
End if  
End for  
End for  
End

• *Modem (modulator / demodulator)*: before the step of filtering, the modem is used as a modulator to convert the traces into scenarios (sequence of actions: $A \rightarrow B \rightarrow C$). Then, after the filtering step, the modem is used as a demodulator to transform the modeled trace into the original trace format according to the model proposed in the reformulation step (see Algorithm 2).

**Algorithm 2: Modem process**

**Input:** $CT =$ a set of traces (login to the system, learning, access to material, visualizing a resource, downloading resources, access to search engine, assessment, access to assessment, communication, send email, receive and answer email, access to the forum, a question on the forum, ask reply on the forum, request for assistance, receive a response to the request, response to requests for assistance, visualizing traces).

**Output:** $CTN =$ $CT$ in a novel format  
**Begin**  
For all (trace $ct \in CT$) then  
If ($ct =$ “Login to the system”) then  
$ct =$ “A”;

Else if ($ct =$ “Learning”) then  
$ct =$ “B”;

…

Else if ($ct =$ “Assessment”) then  
$ct =$ “G”;

…

Else if ($ct =$ “Access to the forum”) then  
$ct =$ “L”;

…

Else if ($ct =$ “Request for assistance”) then  
$ct =$ “O”;

…

End if  
End for  
End
Filtering: to detect automatically the wrong scenarios made by the learners during their learning sessions, we used an algorithm for filtering traces and keeping only the relevant ones. This algorithm is given in the next section (cf. Algorithm 3).

Exploitation of traces

The set of relevant traces is exploited by different actors:

- **Learner**: assistance and recommendation for learners when using the system (learning and collaboration).
- **Tutor**: minimize the tutors’ tasks by reducing the number of traces to visualize each time. We remember that the tutor has the possibility to visualize many types of traces left by learners every day, every week or during indicated dates. Furthermore, relevant traces are used to decrease the big number of the assistance requests sent by learners.

The generic algorithm

The pseudo code of the filtering algorithm is given in Algorithm 3 and its general structure is given in Figure 5. The algorithm has as input the self-data (a set of identified relevant traces) and gives us an output a set of irrelevant traces. In our context, the self-data are the relevant traces. The irrelevant traces are destroyed in order to let only the relevant ones that are used for recommending the learners to improve their learning scenarios. In fact, we use two mechanisms of AIS: negative and clonal selection. The negative selection is used to recognize the irrelevant sequences of actions. The clonal selection improves recognition and allows updating the initial database due to the memory cells. If a new trace arrives and it matches with the sequence of the initial, then the system made a clone, and it updates the initial base. Otherwise, it is an irrelevant trace; the system removes it and recommends learners with the relevant traces. For verifying if two traces match, we have used r-contiguous matching (Forrest, Perelson, Allen, & Cherukuri, 1994). In our case, r is equal to 3 because we have divided the sequence into equivalent sub-sequences that are composed of 5 actions.

Matching with r-contiguous:

We use the match between two strings which is proposed by Forrest and his colleagues (Forrest et al., 1994). The main aim of this matching is the following: when we have two strings x and y, match (x, y) is true if x and y agree (match) at least r contiguous locations.

Example of matching:

X: AMINAZEDADRA
Y: AMINEABDAOU1
Match (x,y) false in the case where r = 5 or greater.
Match (x,y) true in the case where r = 4 or less.

Algorithm 3: Filtering traces

| Input: R= a set of identified relevant traces (R=CTN (output of algorithm 2)) |
| Output: IR= a set of detected irrelevant traces |
| Begin |
| Create an empty set D of detectors |
| Generate random traces TR |
| For all (trace tr ∈ TR) then |
| For all (relevant trace r ∈ R) then |
| If (tr matches r) then |
| Discard tr |
| Else |
| Place tr in D |
| End if |
End for
End for
While (there exist novel traces \( n \) to check) then
Retrieve novel trace \( n \)
For all (detector \( d \in D \)) then
   If \( n \) matches \( d \) then
      Place \( n \) in IR and output
   else
      Clone \( n \)
      Place \( n \) in \( R \)
   End if
End for
End while
End

Figure 5. General structure of the filtering algorithm

Presentation of some interfaces of the realized system

In order to validate our idea, we have implemented a subsystem called AIS4FT (Artificial Immune System for Filtering Traces). It aims at filtering the set of traces to let only the relevant ones. Also, the system assists the learners by sending recommendations’ messages to improve their learning profiles. Figure 6 presents the number of daily detected wrong scenarios (before and after application of the filtering algorithm), while Figure 7 shows an example of a message that was sent automatically to a learner after detecting that he follows a wrong scenario.

From Figure 6, the system filters the irrelevant traces (big number) to a small number of only relevant ones. Also, the figure shows that the communication and the collaboration between learners are low in the first few days, whereas they are high in the middle of the semester and they begin to decrease in the last few days. As it is shown in the figure, between March 16\(^{th}\) and March 31\(^{st}\), the interaction and the collaboration had reached more than 140, while there are only 25 relevant traces. During this period, the students were on holidays.

Figure 7 shows an example of recommendation message sent to a learner who following a wrong scenario. In fact, as it is shown in the figure, a learner had received a message having as subject: “recommendation”. It contains some
directions to follow in order to success his learning process. This recommendation message was sent automatically by the system after the application of our algorithm.

**Detection and filtering wrong scenarios**

Interaction and collaboration traces

![Graph showing detection and filtering of wrong scenarios before and after application of the filtering algorithm](image)

**Figure 6.** Number of bad scenarios before and after application of the filtering algorithm

**Figure 7.** Recommendation message sent to a learner

**Experiment: Results and discussion**

**Data set**

In order to valid our approach we have used a set of traces from LETline system. LETline was developed by Lafifi and his team (Lafifi, Azzouz, Faci, & Herkas, 2010). It is an online learning and tutoring environment, which is composed of two important parts. The first one is a learning management system, which offers the teachers all the means for preparing their courses. Furthermore, the students are brought to build their knowledge. The second one is a tutoring system. It offers mainly all the assistance and capture tools, analysis and traces visualization to allow the follow-up of the students by a tutor.

In order to validate our filtering algorithm, we have added a new module to LETline for supporting collaboration. In fact, LETline 2.0 is the new version of LETline by supporting collaborative learning. As a result, LETline 2.0 is a CSCL system. So, we have integrated two tools into the system: the first one provides the traces resulting from learner/learner and learner/system interaction while the second tool gives a report about the interactions of the learner on his computer. The proposed algorithm is applied on traces collected from LETline 2.0 (see Table3).

**Table 3.** Dataset of collected traces

<table>
<thead>
<tr>
<th>Primary traces</th>
<th>Interaction and collaboration traces</th>
<th>Learning traces</th>
</tr>
</thead>
<tbody>
<tr>
<td>5700</td>
<td>1939</td>
<td>4060</td>
</tr>
</tbody>
</table>
Participants

An experiment was conducted at Computer Science Department at Guelma University (Algeria). The participants were teachers, tutors and students of 2nd year (specialty: information systems). All these participants can use the platform at http://www.labstic.com/letline/ from any computer. The number of the actors participating in this experiment is shown in Table 4.

| Table 4. Number of human actors who used the LETline 2.0 system |
|-----------------|-----------------|-----------------|
| Learners        | Tutors          | Teachers        |
| 60              | 10              | 2               |

Methodology

Students and tutors can access to the LETline 2.0 platform after a successful enrollment. Each tutor has his own groups (from one to five groups based on their preferences). The assigning of tutors is performed by the administrator manually.

The experiment was conducted in two stages: the first one was done without using the AIS4FT subsystem, while the second stage was done using the AIS4FT subsystem. During the experimental period, the students learn the concepts of “Languages Theory” subject. Two questionnaires were proposed by the teacher who is the responsible of the subject. These questionnaires contain a set of questions about two learning objects “languages” and “regular expressions”. Each questionnaire is composed of a set of Multiple Choice Questions (MCQ).

In the first stage of the experiment, a pretest (first questionnaire) was given to evaluate the cognitive profiles of the learners without an assistant system. However in the second stage, a post-test (second questionnaire) was given to measure the usefulness of the proposed subsystem on the cognitive profiles of the learners. The cognitive profiles were calculated basing on the results of the questionnaires (responses of the students on the proposed questions). We adopted the same formulas proposed by Lafifi and his team (Lafifi et al., 2010).

We have chosen ten students randomly to verify the usefulness of the recommendation proposed to the students when interacting with each other, and for verifying our experiment hypothesis.

During the system use, the students can visualize their cognitive and behavior profiles. Also, the tutors can visualize the profiles of their learners to assist them.

Results and discussion

The hypothesis of the research is:

- Null hypothesis H0: the use of the AIS techniques cannot help students to improve their cognitive profiles (learning process).

In order to verify our hypothesis, we calculated the difference of the cognitive profiles of the students. For doing this, each student must answer some assessment questions.

As we have the same sample of students who used the system in two phases (before and after) and the number of the learners is less than 30, we used student paired t-test. Before using the paired t-test, we must ensure that the data are normally distributed. For doing this, we have used Shapiro-Wilk test. The results obtained in both samples are: pretest data: \( W = 0.86, P = 0.07 \); post-test data: \( W = 0.95, P = 0.74 \). Since \( P_{value} \) in both samples is greater than the significance level of \( P = 0.05 \), our data sets were normally distributed.

To know if the difference is significant between the two means, a paired t-test is appropriate to compare how a group scores vary in two different test conditions (sample size is equal to ten). R software is used to calculate the statistical data. The following results are obtained with a confidence level of 95% (\( \alpha = 0.05 \)).
From the Table of t-test, \( t_{0.975} = \pm 2.26 \), so \( t_{score}>t_{0.975} \ (4.94 > 2.26) \). As results, the difference was statically significant; therefore the null hypothesis \( H_0 \) is rejected. So, the alternative hypothesis is proved and we can affirm that “the use of AIS techniques improves the cognitive profiles of learners”.

The main aim of this research was to investigate the effect of our AIS4FT subsystem on the students’ profiles. Firstly, there was a significant difference between the pretest mean of the students (Mean = 2.47; SD = 1.80) and the post-test mean of the students (Mean = 5.81; SD = 2.23). Secondly, Pearson’s correlation test was also performed to analyze the correlation between the students’ scores. In terms of the correlation coefficient, the size of the observed effect (\( r = -0.63 \)) indicates that the students’ scores are negatively correlated. From the obtained results, we observe that our system detects automatically the wrong scenarios in real time and improves them.

In brief, we can say that the use of AIS techniques to detect learning faults can enhance the cognitive profiles of learners. Concerning the collaboration and the dynamism between the learners, we have calculated their behavioral profiles. We found that the behavioral profiles of most students were improved. These profiles are calculated from the number of asynchronous tools used in the communication and collaboration processes.

The effects of the proposed algorithm on detecting the wrong scenarios

Figure 8 presents the variation of the scenarios of ten learners when they are connected to the LETline 2.0 platform for the first days.

![Figure 8. Variation of some learners’ scenarios](image-url)
As it is shown in Figure 8, the abscissas are the different traces and the ordinate are the order of traces in the scenario of each learner. Each trace has its position in the scenario from one to eighteen (e.g., all the learners login to the system which is the first trace in the scenario, after that each learner does his own activities). We observe that the variation of learner’s scenarios is not perfect, that are resulting from the wrong navigation of learners that let them waste time with wrong scenarios (For example, learner 8 logsins to the system first (1\textsuperscript{st} position), access to the learning space at the end of his scenario (18\textsuperscript{th} position), access to the forum (14\textsuperscript{th} position), assessment (6\textsuperscript{th} position), which gave a bad results with low profiles).

Figure 9 presents the variation of learner’s scenarios after applying our algorithm.

Figure 9. Variation of learner’s scenarios after applying the proposed algorithm

As it is presented in Figure 9, we observe that the learner’s scenarios are improved in a good way, where the subsystem of filtering detected the wrong traces and recommend learners with the right scenarios. In Figure 9, we observe that eight learners (learner 1, learner 2, learner 3, learner 4, learner 6, learner 7, learner 9, and learner 10) improve their scenarios using the recommendation provided by the subsystem, which includes the relevant traces to do (actions). In fact, the scenario of the learner 5 and the learner 8 is not improved because they didn’t take into consideration the recommendations proposed by the filtering subsystem.
Finally, we can confirm that the introduction of AIS techniques in CSCL systems for detecting learning faults gives good results.

**Conclusion and future work**

Immune system theory is a natural defense against the foreign molecules. It has different features such as detection of anomaly, learning, memorization, cloning and adaptation. For these reasons, the immune system can be used as a mechanism for inspiration in a variety of fields. According to the different characteristics of the AIS techniques, there are several implications for including AIS theory in CSCL systems. This inclusion aims at improving the learning quality and supporting students by providing several advices for them.

As one of the AIS techniques feature is the detection of anomaly, we have adapted this characteristic to our problematic, which is the filtering and the improvement of the wrong scenarios. A wrong scenario (set of irrelevant traces) is the sequence of actions that conducts to low levels of learners.

In this paper, we have adapted and applied AIS techniques in a CSCL system. The main aim of our work is to present to the learners their irrelevant traces (which returns to the problems of the bad navigation of learners where they are wasting a long time to understand the platform functionalities) and provide them with those relevant. The goal is to improve learners’ scenarios in real time and provide the tutors with a minimized set of traces to help them in their different tasks. The filtering process uses AIS techniques where the negative and clonal selections are combined. The negative selection is used to detect the irrelevant traces and the clonal selection is used to update the initial base. The developed subsystem detects all the irrelevant sequences of actions, and recommends learners to improve their learning process.

In order to verify and examine the proposed approach, some typical applied tests based on the LETline 2.0 platform were introduced and explained. From the experimentation performed by students from an Algerian University, we note that the majority of learners solves their problems without the intervention of the tutors. Tutors could be absent due to some unexpected reasons. So, it is legitimate to the tutors to don’t be online all the times when an assistant system is used to recommend the learners with the relevant traces in order to complete their learning sessions successfully.

With the increase of traces that are recorded on the database, the initial database is also increased. So, the execution of the algorithm will take a long time for filtering the traces, thus our future work is to optimize the complexity of the proposed algorithm. Whereas, the problem isn’t in the algorithm, but in the size of the initial base that is updated every time. To this end, we propose to classify the sub-sequences into several classes where the test will be made only on a subclass of the initial base and not on the whole database. The second future work is the interpretation of traces in CSCL systems. Furthermore, we propose to implement a search engine for finding the relevant traces.

**Acknowledgments**

The authors thank all those who have participated in the experiments in real situations (teachers, tutors, and students of Guelma University). Also, the authors thank the reviewers for their important suggestions for improving the quality of the paper.

**References**


