Automated Evaluation of Learners with ODALA:
Application to Relational Databases E-learning

Farida Bouarab-Dahmani
Computer science Department
Tizi−Ouzou University,Bp 17, (15000), Algeria
fax: (213)(0) 26 21 01 51
dahmani_farida@yahoo.fr

Malik Si-Mohammed
Computer science Department
Tizi−Ouzou University, Bp 17, (15000), Algeria
Fax: (213) (0) 26 21 01 51
maliksim@hotmail.com

Catherine Comparot
IRIT-UTM, IC3 team
Mirail University, 5 Allées Antonio Machado,
31047, Toulouse, France
comparot@univ-tlse2.fr

Pierre−Jean Charrel
IRIT-UTM, IC3 team
Mirail University, 5 Allées Antonio Machado,
31047, Toulouse, France
charrel@univ-tlse2.fr

Abstract

This paper deals with an approach for an automated evaluation of the learners' state of knowledge when learning by doing. This approach is called ODALA for "Ontology-Driven Auto-evaluation for e-Learning Approach". It takes place in the context of Computer Based Human Learning Environment (CBHLE) in a self-learning by doing mode. ODALA is based on the teaching domain ontology and on errors classification and detection. The evaluation process is composed of four stages: (1) form analysis of learner's solutions, (2) semantic analysis, (3) marking, and (4) updating of the learner's model. We bring the approach into play in the context of relational databases teaching: we present the results of the relational databases self-learning system (RDB-E-LEARN) development, where the main stages of our evaluation approach are implemented.

Keywords: Self-learning, Error diagnosis, Learners' evaluation, Learners' marking, Learner's model, E-learning.

1. Introduction

The development of teaching after information technology introduction has been accompanied by an evolution in the way of evaluating the learner's progression. Thus, a failure or lack of results is no longer seen as the consequence of a lack of work, attention or cleverness. The current educational thought replaces the concept of control by the much broader and more complex concept of evaluation which consists to reach in a more objective and rational way all parameters that have a bearing on student's outcomes. However, this development introduces three major difficulties in two features essential to any Computer Based Human Learning Environment (CBHLE) running in self-learning mode: the error diagnosis and the automatic marking process. These difficulties are due to the need:

- To automatically correct the freely built learners' answers in order to avoid common errors known with novices, particularly when solving open questions.
- To rate in the best possible matching to the actual state of the learner's knowledge.
- To find the adequate means to use these evaluation results for learning adaptation.

As response to these issues, we propose an evaluation process based on a gradual approach. It consists first in checking the form of the solution proposed by the learner. The semantic errors, such as the inadequacy of the solution of the learner with the exercise statement for example, are then detected through characteristics definition to each exercise. In the same vein, the definition of a hierarchical marking method allows to retain a set of marks such as understanding indicators. These indicators are synthesized into matrixes integrated to the learner's model.

To get the evaluation process described above, ODALA gives an approach for a given teaching domain. Indeed the automated evaluation is driven by the semantic of the domain (represented by the domain ontology), which is used at each step. As case study, we discuss in this paper the approach implementation in a prototype dedicated to teaching relational databases.

The paper is organized as follows: section 2 presents a literature review about teaching domain modelling and learner's evaluation in CBHLE; section 3 is about the ODALA approach; section 4 gives an application of the proposed approach and a conclusion is given in section 5.

2. Teaching Domain Modeling and Learners' Evaluation in CBHLE

Teaching domains modeling for evaluation purpose is an issue that is rarely studied in CBHLE research. Some works deal with modeling teaching domains and others deal with learners' evaluation. We present an overview of these works in the two following sub sections.

2.1. *Teaching domain modeling*

The teaching domain model (TDM) or domain model (as it is called in Ref. 1) represents the knowledge about teaching domain of the CBHLE. Many types of domain models have been defined. The model to use is more or less complex according to the concepts and relationships of conceptual network. The most popular relationships between concepts are "prerequisite", "part of" and "is a". These relationships are used in some e-learning standards such as LOM[*]. In these standards, these relations are mainly used to manage links between courses (documents). Other specific links between the TDM concepts are necessary for some learning activities such as learner's evaluation; learner's modelling ... Ontologies may be used to represent the sense of these relationships and concepts.

Different works, such as Refs. 2, 3, 4, 5, propose to use ontologies for CBHLE. Nevertheless, there are attempts to use ontologies for learner's evaluation purpose[6, 7]. Ontologies, in these works, are generally the representation of the tasks used to solve problems and/or are a representation of the teaching domain structure in order to infer new knowledge using the existing one. In addition, ontologies can be used for quiz generation (e.g. Ref. 6).

This paper is focused on the TDM we propose to enable and facilitate the learner's evaluation at its different steps when there are open questions and polynomial marks calculus. The conceptual level of TDM is presented in sub section 3.2 and its implementation (in a CBHLE for e-learning) is discussed in Ref. 8.

---

[*] Learning Object Metadata
http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

2.2. *Knowledge evaluation*

"The evaluation in an education context aims to contribute to the success of education, i.e. the construction of knowledge and skills by students."[9]. Indeed, evaluation "allows learning systems to have an interaction with the learner"[10]. But despite the acknowledged importance, little works have directly addressed the question of evaluating the learners, particularly the errors diagnosis in solutions expressed freely and completely by the learner. Perceived knowledge and skills of the learner solving problems remains a fundamental aspect of any training system[11]. This latter, used primarily to improve learner's training, has to allow him, and human tutors if necessary, to perceive his strengths and weaknesses.

Works focused on the evaluation as an integral part of the learning process (like Refs. 9, 10, 12, 13, 14, 15, ...) introduce some categorization able to facilitate the adjustment of the evaluation objectives. The following discussion allows extracting a number of fundamental issues concerning: the evaluation process, the diagnostic techniques that can be implemented and different types of diagnosis which it is possible to reach as appropriate.

There exist different types of evaluation[16]: prognostic evaluation, diagnostic evaluation, formative evaluation, summative evaluation, self evaluation[17], *etc*.

In this paper, we are interested both in summative and formative evaluation since we calculate marks and we do an "inventory." of the acquired skills.

An automatic error diagnosis can be defined as the process provided by an automated module for analyzing and/or detecting errors in the behavior and/or the answers of the learner (see Ref. 15), respectively studied by: behavioral diagnosis(e.g. see Ref. 12), and cognitive diagnosis (e.g. see Refs. 13, 14).

We are interested here by the cognitive diagnosis since we try to detect what the learner has acquired (or has not acquired) by using analysis and correction of its solutions.

The use of Multiple Choice Questions (MCQ) even if it introduces a convenience in an evaluation process by reducing answers to a dichotomy "true / false", cannot be appropriate for the evaluation needs of complex training systems. Indeed, in their traditional approach, the MCQ are part of an overall summative evaluation which does not inform about learners' cognitive processes. Therefore it cannot provide a precise diagnosis of their skills and knowledge. This evaluation is possibly adapted at end of course, but it is probably insufficient in the case of a follow-up training.

This is the main reason why we are interested in more sophisticated processes, such as those of correct answers freely drafted by the learner to open questions. The bibliography in this area indicates several diagnosis techniques which are used either separately or combined in a single system: definition of plans for each possible solution[18, 19], use of bases of rules[20, 21, 22], and construction of grammars[23, 24, 25].

There are also emerging approaches based on specification and/or classification of errors[13]. In this case the diagnosis consists to find errors in learner's solutions. Despite some shortcomings, for example related to the incremental nature of the errors inventory, these approaches are very interesting because they are independent of the domain. We chose this kind of technique as a basis of our approach. Indeed, this kind of learning by recognition of errors is applicable in various fields and always gives a significant result even if the errors database is incomplete. This can lead the learner to avoid, at least, the basic mistakes, which is not a negligible result in the case of learning by exercise.

That is why in our approach, we integrate the concept of taxonomy of errors and retain the assumption mentioned in Buggy[26] system and its derivatives. This assumption stipulates that an error found in a learner's solution is usually an indication of poor understanding of the teaching domain, thus establishing a priori a link between systematic errors and the domain's concepts, what is dealing with the following section.

3. The ODALA Approach

We define ODALA (Ontology Driven Auto-Evaluation Learning Approach) as a methodology to develop automatic evaluation using a TDM. ODALA assists the designer at two main levels:

- First, at the domain ontology construction.
- Second at the development of the evaluation's modules (form analyzer, semantic analyzer, learner's marking module and learner's module update) by browsing in the ontology and by suggesting techniques based on the ontology's concepts and links.

In the context of a CBHLE, the evaluation process, developed with ODALA, requires the collection, within a teaching expert module[27, 28], data relating to a session of learning by students. These data will be subsequently processed by a teacher module, in order to synthesize an overall evaluation integrating other data concerning the learner and its learning pathways. It should be noted that we are interested in an evaluation process intended primarily to self-learning by doing systems[28]. A session begins with an automatic exercise generation. The exercises are produced on teacher's request, according to the learner's profile. There exist three types of production[27]: (1) simple generation when the exercises are selected from a prerecorded bank; (2) semi-automatic generation by the use of configurable exercises; (3) automatic production when existing exercises are combined.

3.1. *The evaluation process*

The evaluation process (see Fig. 1.), in a learning session, begins when the learner's solution to the generated exercise is captured. The diagnosis module is then requested (by the learner or the teacher, according to the applied teaching strategy). A lexicon-syntactic analysis (form analysis) is launched to verify that lexicon and syntax used in the learner's solution (free text) follow the language taught in the domain (notations and algebraic operators in mathematics for example, or even keywords and syntax in the case of teaching a programming language ...). If this first step takes place without errors, a semantic analysis is performed to check the solution's "sense", and thus consider whether the learner has correctly answered to the problem and / or has used the adequate domain notions according to the exercise statement.

In the case of completely free learning, learner's rating begins at his demand. It is also possible, in the case of automatic guidance, to move to scoring after a number of attempts to correct. The marking module allows marks and understanding indicators calculation on the various notions in the taught domain. The final stage of the evaluation is the learner's model update. The learner's model is a sort of memory about his progress where the evaluation results are saved in appropriate structures.

3.2. *The ontological model*

The design and implementation of the proposed approach, as well as the development of each module of the evaluation process, lead us to a 3-parts structure of domain's knowledge manipulated by this process:

- The main knowledge required is primarily related to the exercises themselves, collected in the exercises' database.
- The teaching domain's structural components in particular those we call *knowledge items* (*knowledge items* are basic or granular notions).
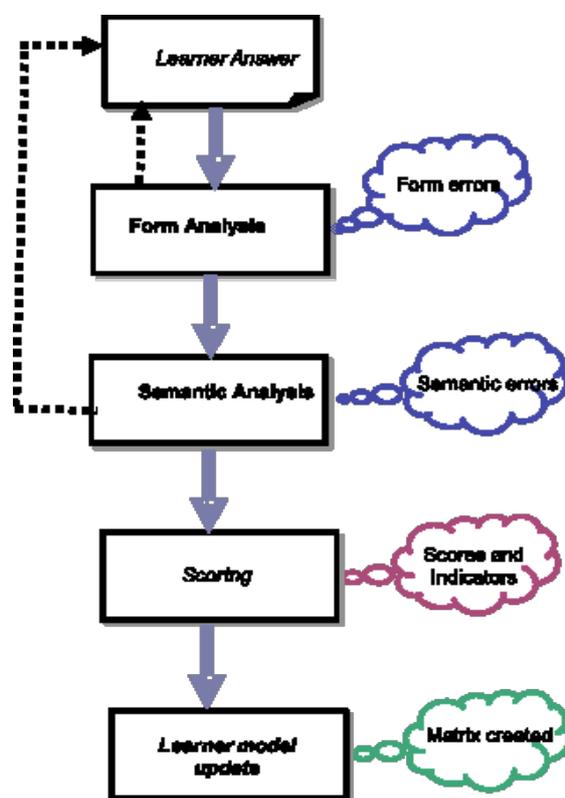- The taxonomy of errors which collect all the learners' potential errors.

Fig.1. The evaluation process with ODALA.

We opted for modelling with ontological concepts and their links. Indeed, the specific contribution of ontological engineering for CBHLE consists on the one hand in the formal representation of knowledge reporting, coupled with an inference mechanism used by these environments[4]. On the other hand, the ontology

construction is done through a consensus, and thus represents the shared understanding of a group or a community, which is perfectly adequate to the education area where different agents (pedagogical agent, learner's management agent, domain expert agent) can process domain's knowledge.

Finally, the construction of domain ontology facilitates the identification process of the language used to express the solutions to the proposed exercises. This is a very important element for the construction of lexicon-syntactic analysis grammar or another form analysis module used by the proposed evaluation.

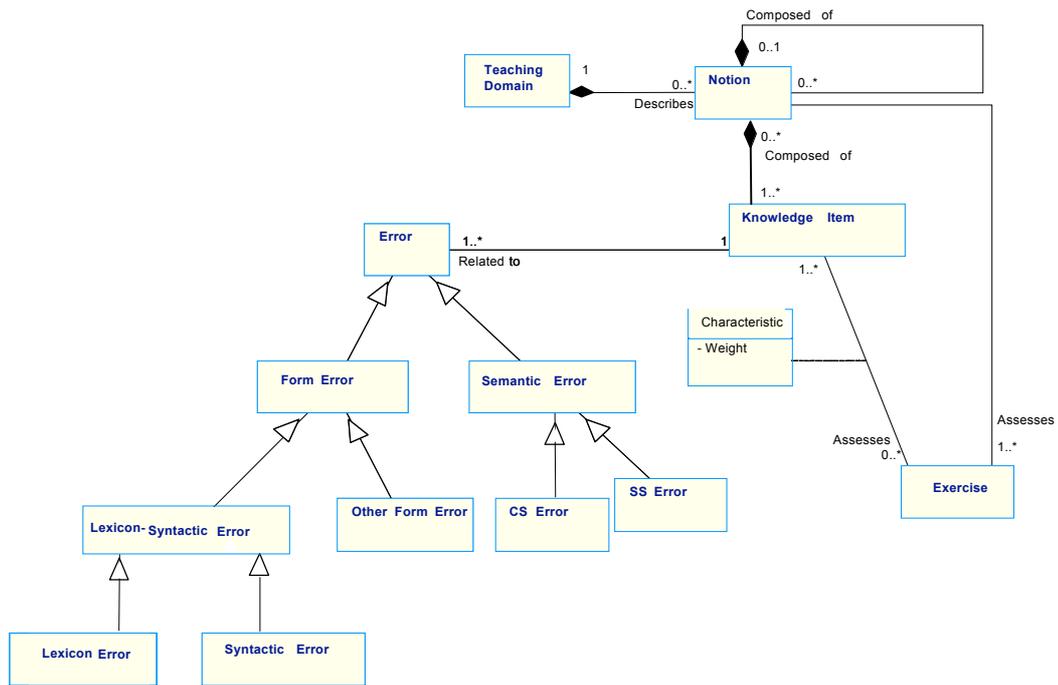The domain ontological modeling that we intend to use is represented as the UML class diagram of Fig. 2.

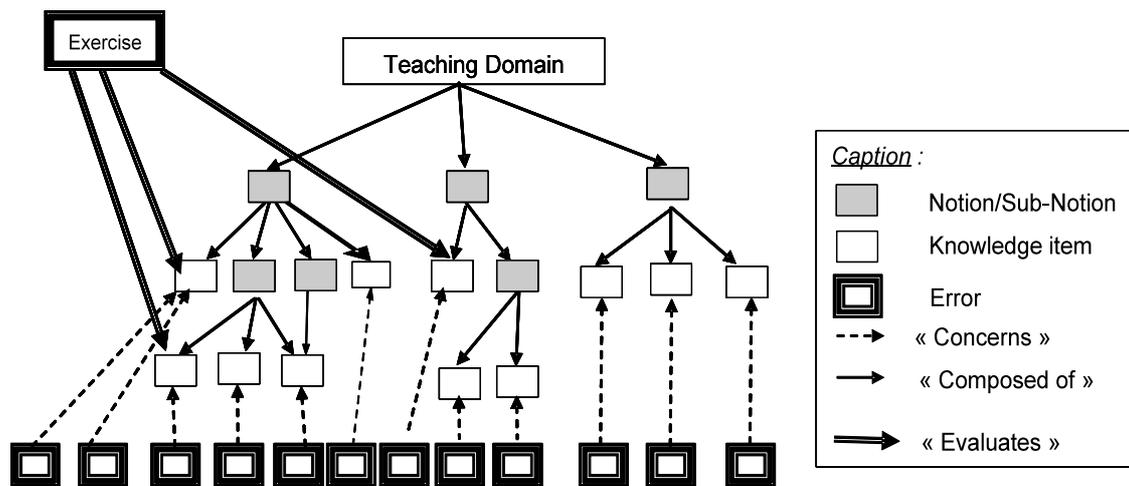Fig. 2. The UML class diagram of an ontological teaching domain model.

Fig. 3. Overview of Teaching Domain Ontology concepts.

3.2.1. *Notions and knowledge items*

These are components of the education field that the teacher wishes to convey to the learner. For didactic, educational or structural reasons, a notion can be broken down into several sub-notions. At this stage of our research, this decomposition of a domain in a hierarchy of notions is of human expert responsibility.

The ontological model that we propose is a structuring model of various notions components involved in the teaching domain considered in order to facilitate the automatic evaluation process. We therefore recommend a strictly hierarchical tree of notions (see Fig. 3.), except with regard to the leaves. Indeed, a knowledge item (KI) can fit into the composition of different notions. For example, in the algorithms case, the item "*form of instruction block*" can be linked, among others, to the notions "*alternative structure*" and "*iterative structure*".

3.2.2. *Exercises*

The resolution of an exercise requires the use of a range of items, which is expected to appear in the formulation of the solution by the learner, hence the intervention of notions located at different levels of the ontology. The domain's ontological representation allows a view about the notions required by the domain learning considered with the different exercises proposed by the system. According to this model, an exercise is described by a dynamic structure represented by links pointing to the domain's components required by the exercise, particularly the items (leaves) as shown by Fig. 3.

Table 1 represents the exercise 2 of the RDB-E-LEARN prototype for relational databases teaching introduced in the forth section of this paper. The representation focuses solely on giving the main notions (from two levels) and associated knowledge items (given with their respective numbers in the implemented database). In addition, each link between the exercise and a knowledge item may be characterized by a weight (as given in Table 1). Weights allow the teacher to define the importance of each item whose exercise aims to assess mastery. This aspect introduces an additional degree of flexibility because of the possibility to increase more or less the relative importance of the various items involved in the

resolution of the same exercise, according to educational objective pursued.

3.2.3. *Errors*

To increase the effectiveness of the error process detection, we propose to implement a hierarchical taxonomy of possible learner's errors, depending on the classification we define below. We consider here that two main types of errors can occur in the solution proposed by a learner:

- The form errors, which reflect non-conformity of the learner's solution to the conventions of the teaching domain's representations. In the case of textual solutions, we retain a particular interest to lexicon-syntactic errors. Thus, the ontological representation of the domain can limit the terms of the language used by learners to express their solutions to a finite set of constructions. Then lexicon-syntactic errors can be easily detected

- The semantic errors relate to a more abstract part of the learner's solution such as a logical expression and/or scheduling tasks inadequate in relation to the proposed exercise or to the domain's semantic rules. This obviously makes their detection more complex. To facilitate the detection and recognition process of these semantic errors, we divided them into two categories:

  (a) The *common semantic errors* (CSE) relate to errors that do not depend on the exercise, and which are simply the "generality" of the studied domain, like division by zero (arithmetic), infinite loop (programming) or same relations schemes (relational databases).

  (b) The *specific semantic errors* (SSE) express that the characteristics expected in a solution to the proposed exercise are not expressed by the learner. In fact, we name each reference in an exercise Exi, a knowledge item KIj of the domain ontology as a characteristic Cij of the exercise, to which at least one specific semantic error as potential error is associated. For the exercise 2 of Table 1, for example, they would be at least fifteen characteristics corresponding to the fifteen knowledge items. In the case of this example, errors like "*Union operator not expected for this exercise resolution*" or "*The attributes of the projection are incorrect*" would be SSE while errors as "*attribute identifier type different of the*

expression type allocated to" or "*unknown attribute identifier*" would be CSE.

Table 1. Example of bond Exercise-knowledge items-Notions in the case of Relational databases teaching

Exercise 2 Statement	First level Notions	Second level Notions	Knowledge items	Weight
'Going from the following relational diagram about an AIRBASE: -PILOTE(NumPil, NomPil, Adr, Sal) -PLANE(NumAv, NomAv, Capitale, Localis) -FLY(NumVol, NumPil, NumAv, Ville_Dep, Ville_Arr, H_Dep, H_Arr) Answer with algebraic operators this request: What are the planes (number and name) located at Toulouse and those that have already fly to Singapore"	- Introduction to the relational model -Relational algebra	-The relation concept -Algebraic operation -Binary operators -Monadic operators	06.-Definition of a relation 07 -Attribute of a relation 08 -Degree of a relation 09 -Key of a relation 10 -Diagram of a relation 19 -Principle of monadic operators 20 -The selection operator 22 -The projection operator 24 -Syntax of monadic algebraic operation 25 -Principle of binary operators 26 -The union operator 34 -Syntax of a binary algebraic operation 35 -Syntax of a condition in an algebraic operation 36 -Lexicon of the words 37 -Syntax of an algebraic operation	1 1 1 1 1 1 2 2 1 2 3 1 1 1 1

The semantic errors' classification shows that the fineness of the semantic analysis depends on controlling the exercises' characterization process (defining characteristics). This one, itself depends on the didactic wealth of the taught domain. Indeed, the richer teaching, gives the finer decomposition of the domain. The characterization is then accurate and can therefore define a greater number of potential SSE.

We chose to link all the possible errors (form or semantic ones) of the learner to knowledge items, knowing that these links can be propagated (as for the exercises) by bottom transitivity to all nodes of the tree representing the ontology. Thus, several errors may be linked to a given item. Each one provides an indication about the lack of control by the learner. However, we assume that every error may relate to only one item to avoid any ambiguity on the final diagnosis and the rating process. Far from being a constraint, this assumption imposes a structure as fine as possible in the domain, which could lead to define new items. Thus, the enrichment of the errors' base allows construction of more precise domain ontology.

3.3. *The error diagnosis*

In a CBHLE, an accurate diagnosis must allow, during a session, the tutor to generate dynamically the succession of learner activities, and in particular, to select the content and difficulty level of exercises to be submitted depending on the history of previous answers[17]. This introduces relevant individual guiding rules in assisting the learner in its approach[29].

The diagnosis we propose here is therefore the correction of a solution given by a learner answering an exercise generated by the learning by doing system. The correction of the proposed solution may then take place during the writing thereof, the corrective module acting as soon as non-compliant input is detected. This is referred to as "automatic verification"[21]. We think that this "preventive" correction is likely to influence learning as its drafting of the answer, what is likely to introduce a bias that can affect the accuracy of the final evaluation. We therefore prefer to intervene only when the learner indicates to the corrector module that he/she has finished the input of his/her solution, which will result successively:

- Form analysis: The definition of grammar rules or another form analysis module specific to the

studied domain can detect any form errors like lexicon-syntactic errors relating to the terms used in the expression of the solution. The construction of this module is guided by the domain ontology.

- Semantic analysis: The semantic analysis fetches common semantic errors, and specific semantic errors defined by the "characteristics" of the modeled exercise[30]. Indeed, when a characteristic isn't recognized (by the semantic analyzer) in a learner's solution, its corresponding SSE is detected. The ontological links between the exercises and knowledge items and those between these last ones and errors are used at this step of the evaluation.

3.4. *Learner's marking*

The rating is the part of the evaluation, which expresses a quantitative evaluation of the solutions' correction proposed by a learner. It is based on the calculation of a parameter set which will enable the teacher to properly assess the learner's abilities compared to the educational objectives set.

Thus, leaving "granular" marks on the items, we calculate higher level marks relatively to an exercise or a notion of the highest abstraction level.

The formulas' development showed us the difficulty of replicating with a computer system the ability of a human corrector to rate the merits of a solution even when the form has errors. Indeed, this capability enables you to rate the merits without loop on correction of the form (what teachers do when they correct exam papers). In case of automatic rating, if there are form errors, it becomes particularly difficult to obtain a correct evaluation of semantics. A way to overcome this constraint consists in giving the learner the opportunity to correct its form errors and so the semantic analysis occurs after a determined number of attempts (limited by the teacher) to correct the form errors.

In the following formulas, let us consider that the number of detected errors will be represented by:

- The sum of the number of form errors with the number of semantic errors if the learner has been able to correct its form errors after the number of attempts allowed by the teacher,
- the number of form errors if the learner has failed to correct in the number of attempts allowed by the teacher, even if he/she did not commit semantic errors, which is the most restrictive case,

- The number of semantic errors if the learner has no form errors.

The main proposed formulas are described in what follows.

3.4.1. *Rating a knowledge item in an exercise*

Nij, which is the mark of the item KIi mentioned in the exercise Exj, depends on the number of the detected errors NDij on this item, and the total number of errors NTi linked in the errors database to KIi. We will have therefore the "Eq. (1)".

$$Nij = (1 - NDij / NTi) \tag{1}$$

With: $0 \leq Nij \leq 1$ and $NTi > 0$.

3.4.2. *Rating an exercise*

The mark *NExj* of the exercise Exj will be given by "Eq. (2)".

$$NExj = \sum_{i=1}^{i=NICj} Pij * Nij \tag{2}$$

With: *NICj*: number of items in Exj,

Pij: coefficient or weight, which indicates the importance of the item KIi depending on the educational objective assigned to Exj.

3.4.3. *Rating a session of N exercises*

NS is the mark obtained by the learner after treating N different exercises of a learning session. It represents the average of the N marks respectively calculated for N exercises weighted by the degree of difficulty of each exercise (DDExj) defined by the teacher. For example, this degree can be calculated by considering the number of used items and/or items' combinations needed to resolve the exercise. We will therefore get "Eq. (3)":

$$NS = \sum_{j=1}^{j=N} DDExj * NExj / \sum_{j=1}^{j=N} DDExj \tag{3}$$

The session mark is therefore necessarily evolutionary as it is updated by recalculating after each resolution of a new exercise by the learner.

3.4.4. *Knowledge items understanding Indicators calculation*

The understanding indicator IndCi of an item KIi is an incremental percentage, which assesses the overall understanding level of this item by the learner. It takes into account the marks obtained in relation to this item

in N treated exercises during the leaning session. We propose its calculation in "Eq. (4)".

$$IndCi = (\sum_{j=1}^{j=N}(DDExj * Nij) / \sum_{j=1}^{j=N} DDExj) * 100 \qquad (4)$$

3.5. *Enriching the learner's model*

The learner's model stores information about skills, knowledge and shortcomings of each learner. The aim of this model is therefore to guide the teacher to make best education decisions, suited to the learner[31]. The process of adaptation, used to get these decisions, is essentially a resources filter which is based on a set of parameters relating to the state of knowledge of the learner, its attitudes and preferences, and the objectives of learning sessions[32]. It is then a decision making with qualitative and quantitative information as it is studied in different works about decision making such as Refs. 33, 34 …

In this context, the evaluation process that we just present, provides information on the learner's state of knowledge. We therefore propose to pass this obtained information (understanding indicators, exercises' marks,) to the learner's model through the definition of matrixes. The stored values should enable the learning supervisor to adapt its guidelines to learner's knowledge. Among the possible matrixes, we propose:

- The *matrix of understanding* in which is allocated to each learner, his understanding rate regarding each item indicating the value calculated when the item is treated by at least one exercise, or the character "#" if it has not been mentioned in the proposed exercises.

- The *matrix of exercises* which indicates for each exercise and each learner the corresponding mark during the learning session or the special character "#" if the exercise has not been resolved. In each case the value is interpreted as the way to decide about the appropriate exercises to propose at the next sessions.

4. The RDB-E-LEARN Application

We already implemented ODALA through the development of an environment for self-distance learning called WebSiela[35] devoted to teach in French an algorithmic pseudo-language for beginners. In this section, we present a second teaching domain experimented with ODALA: the RDB-E-LEARN (Relational Data Bases E-LEARNing) system that uses

Onto-RDB, the relational databases domain ontology we designed. This system is implemented in PHP and it uses XML documents, XSL style sheets and a relational database for data implementation (managed with MySQL server).

4.1. *Relational databases domain modeling*

Onto-RDB, the relational databases domain (RDB) ontology is generated from the ODALA ontological model (described in section 3-2). With the current prototype, Onto-RDB proposes three levels for the domain decomposition corresponding to: notions, sub-notions and knowledge items. The implementation of this ontology reuses the knowledge acquisition interface (called author Space) already used with the WebSiela system to input the concepts (notions, knowledge items, exercises, errors) and the links. The current prototype includes three first level notions (notions), fourteen second level notions (sub-notions), ninety knowledge items, twenty exercises and an errors database referencing one hundred lexicon-syntactic errors and seventy semantic errors.

4.2. *Error diagnosis*

The error diagnosis development, according to the proposed approach, consists of:

- RDB solutions form analysis: since the RDB solutions forms can be different (textual, graphic with algebraic trees, request programs and algebraic expressions), we have chosen the algebraic expressions according to a standard algebraic relational language. The form analyzer implements the grammar (using Onto-RDB) of this algebraic language. The form analysis consists of a lexicon-syntactic verification. We are currently working about the other forms analysis (algebraic trees and SQL[†] queries).

- Semantic analysis: the common semantic errors are detected by functions integrated to the language grammar. Specific semantic errors are recognized by rules corresponding to the exercise's characteristics.

Figure 4 shows an example of the error diagnosis result given by the RDB-E-LEARN's semantic analyzer. It

---

[†] SQL : Structured Query Language

concerns a solution (displayed in the right window of the screen) given by a learner to exercise 2 (cf. Table 1). Among the semantic errors, the common semantic error numbered 110 indicates that the intersection operator used in the solution is applied to two relations (PILOTE and PLANE) that have not the same diagram.
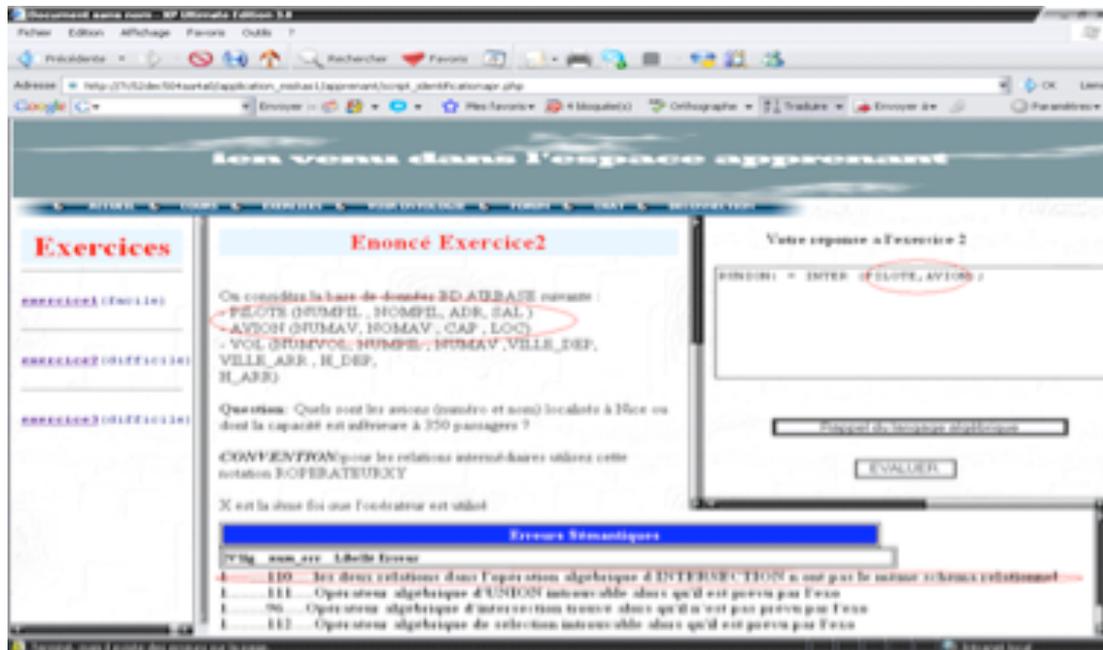

Fig. 4. This is the caption display of semantic errors with the RDB-E-LEARN system

4.3. *Rating process*

The marks and parameters are calculated using a reusable marking module which is an implementation of the ODALA'S approach and formulas presented in the sub-section 3.4.

4.4. *Tests and results*

Two experiments have been conducted with RDB-E-LEARN prototype.

The first experiment was conducted on thirty five (35) second-year students from computer science LMD group of our university. Three exercises, about the use of relational algebra, included in this prototype, were proposed to the students. The examination papers have been corrected by two human teachers and the evaluation module of RDB-E-LEARN. Each error detection rate displayed in Fig. 5. is based on the errors detected rate (errors detected by the corrector compared to the number of actual errors in the solution) for each examination paper among the thirty five.

The interpretation of the results (cf. Fig. 5.) shows that the automatic correction process implemented in RDB-E-LEARN for error detection in learners' solutions approximates human corrections, with the following exceptions:

- Detection of non existing lexicon-syntactic errors, which is a known problem in compilation. Indeed some lexicon-syntactic errors distort further analysis and give rise to other errors. The WebSiela system experimentations[35] have already raised this difficulty. We are currently working to improve analysis process adaptation to this type of anomaly.
- Semantic errors not detected because they are not already recorded into the system's errors' database. The solution is to enrich up to completeness, as far as possible, the database of the semantic errors. The experiment conducted has already been very instructive on this point.

Finally, corrections of RDB-E-LEARN are generally more comprehensive and uniform from a student to another while some human teachers corrections are

vague (errors are not precisely identified) and sometimes highly variable from an examination paper to another.

In addition to the e-learning by doing of relational algebra operators application, RDB-E-LEARN is a complete e-learning platform with an author interface for the ontology updating, a "learning by simple questions" module (like MCQ) where the learner is "directed" when answering, a course module for courses generation, …

The second experimentation was conducted on one hundred (100) novice students in relational databases from the second degree of our university and fifteen (15) senior databases teachers. After practicing RDB-E-LEARN we asked students: "Do you think that the system evaluation module will help you to learn
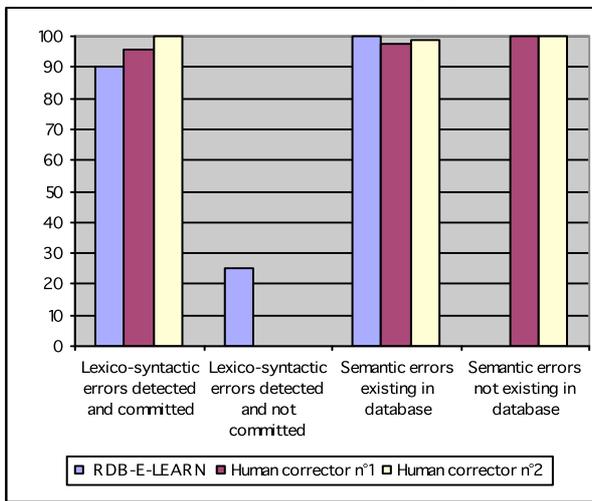
relational databases construction" and we asked teachers: "Do you think that the system evaluation module will help you to teach relational databases construction if your students use RDB-E-LEARN to self-practice" for the teachers. Fig. 6. synthesizes their responses by giving the percentage for each type of reply.

This evaluation results reflect, at least in its primary objectives, the validity of our approach and confirm its interest for teaching domains representations and learners state of knowledge evaluation in learning by doing systems. Making current prototype available to a greater number of learners will undoubtedly enable us to refine a little more the ODALA approach and perfecting the system RDB-E-LEARN to improve its performance.
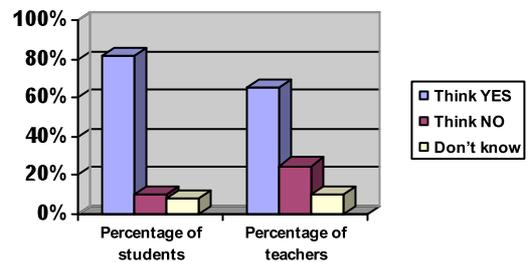

Fig.5. Histogram of the error's detection rates

Fig. 6. Histogram of the Second Test Results

5. Conclusion and future works

We have presented ODALA approach to facilitate the automated evaluation of learner's knowledge when learning by doing. ODALA is based on modeling an education field in the form of ontology. An ontology-based structure of the domain allows its representation and then, provides a sound transmission of the diagnosis results to each of its components aiming to assume an overall evaluation. We have also proposed the classification of learner potential errors depending on their type, thus building an errors database that can be enriched gradually during the sessions. The calculation

of various parameters with polynomial formulas is also an interesting aspect with learning by doing systems to get better results than the dichotomous evaluation (known or don't known) in closed questions (MCQ, checkboxes ...) very used nowadays in e-learning platforms.

We have presented RDB-E-LEARN, which is the second implementation of our approach in a prototype system for self e-learning of a relational algebra language with the relational databases domain. The results of initial tests are promising and guide us towards the need to improve the semantic errors in the field of relational algebra that we have chosen as a test.

These results complement those obtained with the WebSiela system we early developed for teaching algorithmic.

Both works on ODALA suggest us to extend this approach to other education areas in order to reuse automatically some difficult learning tasks, like error diagnosis of freely built solutions to open questions and learner's marking

References

1. Kazanidis, I., & Satratzemi, M. 2008. Adaptivity in a SCORM Compliant Adaptive Educational Hypermedia System. H. Leung et al. (Eds.): ICWL 2007, LNCS 4823, pp. 196 – 206. © Springer-Verlag Berlin Heidelberg

2. Knight, K., Gašević, D., Richards, G. 2006. An Ontology-Based Framework for Bridging Learning Design and Learning Content. Educational Technology & Society, 9 (1), 23-37.

3. Garcia, I., Benavides, C., Villar, J.R., Rodriguez, F., Alaiz, H. & Alonzo, A. 2006. An intelligent tutoring system based on domain ontology for the design of lead-lag controllers, seventh IFAC symposium on advances in control education, Spain, 21-23 June.

4. Psyché, V., Mendes, O., & Bourdeau, J. 2003. Apport de l'ingénierie ontologique aux environnements de formation à distance, STICEF Revient, vol. 10.

5. Snae, C., Brueckner, M. 2007. Ontology-driven e-learning system based on roles and activities for Thai learning environment., Interdisciplinary Journal of Knowledge and learning objects, volume 3, 2007.

6. Sampson, D., Karagiannidis, C. & Cardinali, F. 2002. An architecture for Web-based e-learning promoting reusable adaptive education, Educational Technology & Society 5 (4) 202, ISSN 1436-4522.

7. Smrz, P. 2004. Integrating ontologies into learning management systems – A case of Czech., OTM workshop 2004, LNCS 3292, pp. 768-772, springer-verlag.

8. Bouarab-Dahmani, F.& Si-Mohammed, M., 2008. Un méta-modèle ontologique pour la représentation de domaines d'enseignement en E-learning, Proceedings of the international conference on Web and information technologies (ICWIT'2008), Sidi BelAbbes, Algeria, 29-30 June.

9. Hadji, C. 1997. L'évaluation démystifiée, ESF, Paris.

10. Michel, C., & Rouissi, S. 2003. E-learning : normes et spécifications, Etude des spécifications LOM et IMS-QTI caractérisant des documents inter-échangeables et réutilisables pour l'acquisition et l'évaluation des connaissances, « Document numérique » journal. 1.

11. Labat, J. M. 2002. EIAH : Quel retour d'information pour le tuteur, TICE Conference (France), 81-87.

12. Baker, R. S., Corbett, A. T., & Koedinger, K. R. 2004. Detecting student misuse of intelligent tutoring systems. ITS 2004, LNCS 3220, 531-540.

13. Horacek, H., & Wolska, M. 2006. Handling errors in mathematical formulas, ITS 2006, LNCS 4053, 339-348.

14. Jean, S. 2000. Pépite, un système d'assistance au diagnostic des compétences, doctorat Thesis, University of Maine (France).

15. Wenger, E. 1987. Artificial Intelligence and Tutoring Systems, Computational and Cognitive Approaches to the Communication of Knowledge, Morgan Kaufmann Publishers Inc., Los Altos.

16. Bisault, J., & Lavarde, A. 1995. Le mémoire en IUFM : Théorie et pratique, Beauvex cedex : CDDP Oise, ISBN 2903729700.

17. Ellis, H. J.C. 2007. An Assessment of a Self-Directed Learning Approach in a Graduate Web Application Design and Development Course, IEEE Tansactions on education, 50(1).

18. Johnson, W. L. 1986. Intention-Based Diagnosis of Novice Programming Errors, Morgan Kaufmann Publishers, Los Altos.

19. Py, D. 2001. Environnements Interactifs d'Apprentissage et Démonstration en Géométrie, Thesis, University of Rennes (France).

20. Anderson, J.R., & Reiser, B.J. 1985. The Lisp Tutor, Byte, 10(4), 1985, p. 159-175.

21. Nicaud, J.F., Chaachoua, H., & Bittar, M. 2006. Automatic calculation of students' conceptions in elementary algebra from aplusix log files, in ITS 2006, LNCS 4053, 433-442

22. Sleeman, D., & Smith, M. 1981. Modelling student's problem solving, Artificial Intelligence, (16), 171-187.

23. Goldstein, I., & Miller, M. 1976. Structured planning and debugging: a linguistic theory of design, Artificial intelligence Lab Memo 387, Massachusetts Institute of technology, Cambridge, Massachusetts.

24. Haire, S., & Vandeventer Faltin, A. 2003. Diagnostic d'Erreurs dans le Projet FreeText, Alsic journal, 6(2), 21-37.

25. Ruth, G. 1973. Analysis of algorithm implementations, Technical report MIT project MAC TR 130, Massachusetts Institute of technology.

26. Brown, J., Rubinstein, R., & Burton, R. 1975. Steps towards a theoretical foundation for complex knowledge-based CAI, BBN report 3135, ICAI report 2, Bolt Beranek and Newman Inc., Cambridge, Massachusetts.

27. Bouarab-Dahmani, F., & Ahmed-Ouamer, R. 1999. The Expert Component of an Intelligent Tutoring System for Teaching Algorithmic, Mediterranean international conference on computer technologies (MICCT'99), Tizi-Ouzou (Algeria).

28. Bouarab-Dahmani, F. Si-Mohammed, M., Comparot, C. & Charrel, P.J., 2008. Evaluating the Learner's State of Knowledge when he/she is learning by doing, Proceedings of the International Conference on Cognition and Exploratory Learning in Digital Age (IADIS- CELDA 2008), Freiburg, Germany, October 13-15.

29. Vin, L. 1986. Communication des connaissances et activité de l'élève, Presses Universitaires de Vincennes, Université Paris VIII (France).

30. Bouarab-Dahmani, F., & Si-Mohammed, M. 2005. Characteristic based diagnosis in an intelligent tutoring system for algorithmic, Proceedings of the international conference on machine intelligence (ACIDCA-ICMI'2005), Poseur, Tunisia.

31. Tchétagni, J.M.P., & Nkambou, R. 2002. Hierarchical representation and evaluation of the student in an intelligent tutoring system, ITS 2002, LNCS 2363, 708-717.

32. Bouzeghoub, A., Defude, B., Duitama, J.F., & Lecocq, C. 2005. Un modèle de description sémantique de ressources pédagogiques basée sur une ontologie de domaine. Sticefr journal, vol. 12.

33. Li, X., Ruan, D., Jun, L., Yang X. 2008. A Linguistic-Valued Weighted Aggregation Operator to Multiple Attribute Group Decision Making with Quantative and Qualitative Information, IJCIS journal, Vol. 1-3, August.

34. Cengiz Kahraman, C., Tolga, A. C., 2009. An Alternative Ranking Approach and Its Usage in Multi-Criteria Decision-Making, IJCIS journal, Vol. 2-3, October.

35. Bouarab-Dahmani, F., Si-Mohammed, M., Comparot, C., & Charrel, P.J. Learner's Error Diagnosis and Learner marking approaches for E-learning platforms. Proceedings of the Technology Enhanced Learning Conference 2008 (TELearn 2008), Hanoi, Vietnam, 2008, December 6.