International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 11, November-2014

Survey of Intelligent Tutoring Systems: A Review on the Development of Expert/Intelligent Tutoring Systems, various Teaching Strategies and Expert Tutoring System Design Suggestions

Shanky Sharma[#1], Snehal Ghorpade[*2], Anikit Sahni[#3,] Niti Saluja[#4]
[#]Department of Computer Engineering
Sinhgad Academy of Engineering,Savitribai Phule
Pune university
Pune, maharashtra,
India

***Abstract*— An ETS or an Expert Tutoring System is any program that enables its users to learn/understand a concept or topic and is able to customize its strategies by analysing the user's intelligence level. The paper explains a lot of tried-and-tested teaching mechanisms and aims at providing suggestions to engineer a platform that makes learning easier and more productive.**

Keywords*— Intelligent/Expert Tutoring Systems,Computer Aided Instruction (CAI), Affective Tutoring Systems (ATS),Interactive E-Learning, Applied Artificial Intelligence, Machine Learning, Fuzzy Sets, Expert Systems

I. Introduction

Since the 1970s, Artificial Intelligence and it's techniques have been amalgamated into the field of education and learning in attempts to produce educationally useful computer artifacts that can someday take the place of a human teacher or become a great tool for them in imparting knowledge with ease., these systems have taken various forms from Computer Aided Instruction media till finally Intelligent Tutoring Systems were born. Computer-based Intelligent Tutoring Systems (ITS) provide a promising option for helping students prepare for high stakes assessments. Recent developments on intelligent tutoring systems have clearly shown that users of tutoring software can make rapid progress and dramatically improve their performance in specific areas and skills. There has been significant improvement from the older day e-learning systems to today's ITS(s), forming systems that can support a wide range of multimedia to facillitate learning and a variety of analysis techniques provide students with the right motivation and assessment to push their limits to the farthest. Below are some popular definition of ITSs.

1. Tom Murray defines an ITS as "Intelligent Tutoring Systems are computer-based instructional systems with models of instructional content that specify what to teach, and teaching strategies that specify how to teach[1]."
2. Wikipedia defines an ITS as "Any computer system that provides direct customized instruction or feedback to learners, i.e. without the intervention of human beings, whilst performing a task[2].

II. Architecture of a typical Expert Tutoring System

A typical ITS, has the following four basic components:

- The Domain model
- The Student model
- The Tutoring model, and
- The User interface model.

The section below lists them with their functionality, individually and then by way of their integration.

A. *DOMAIN MODEL*:The domain model (a.k.athe expert knowledge model) consists of the concepts, rules, facts, problem-solving strategies, the pedagogical structure, heuristics, etc. of the domain in context. This portion analyses how well a student has understood a particular domain/concept. It serves the purpose of a source of expert knowledge, a standard for evaluation of the student's performance and diagnosis of errors.It is a cognitive model that analyseshow two or more processes interact (e.g., visual search and decision making), or to make behavioral predictions for a specific task or tool (e.g. how instituting a new software package will affect productivity).

B. *STUDENT MODEL*: It lays emphasis on cognitive and affective states of the student in relation to their evolution as the learning process advances, as it is an overlay on the domain model.. As the student works step-by-step through their problem solving process, the system starts to trace and record that process. Anytime there is any deviation from the predefined model, the system reports an error.

C. *TUTORING/EXPLANATION MODEL:*The tutor model (also called teaching strategy or pedagogic module) accepts information from the domain and student models and devices tutoring strategies with actions. This model regulates instructional interactions with student. It is closely linked to the student model, makes use of knowledge about the student and its own tutorial goal structure, to devise the pedagogic activity to be presented. It tracks the learner's progress, builds a profile of strengths and weaknesses relative to the production rules (termed as 'knowledge-tracing'). At any point in the problem-solving process the learner may request guidance on what to do next, relative to their current location in the model. In addition, the system recognizes when the learner has deviated from the production rules of the model and provides timely feedback for the learner, resulting in a shorter period of time to reach proficiency with the targeted skills. The tutor model contains several alternatives to put a concept into two states Learned and Unlearned. If a student succesfully solves a portion the system updates the probability of reaching the learned state and makes it higher. Systems continue this drilling process till the students acheve a satisfactory state where they are said to have learned or the probability is sufficiently high.

D. *USER INTERFACE MODEL:* This is the interacting front-end of the ITS. It integrates all types of information needed to interact with learner, through graphics, text, multi-media, key-board, mouse-driven menu[6], etc. There are three factors that should be considered for the design of a successful user interface; development factors, visability factors and acceptance factors.

- Development factors help by improving visual communication. These include: platform constraints, tool kits and component libraries, support for rapid prototyping, and customizability.
- Visability factors take into account human factors and express a strong visual identity. These include: human abilities, product identity, clear conceptual model, and multiple representations.
- Included as acceptance factors are an installed base,corporate politics, international markets, and documentation and training.
- Suitability for the task: the dialogue is suitable for a task when it supports the user in the effective and efficient completion of the task.
- Self-descriptiveness: the dialogue is self-descriptive when each dialogue step is immediately comprehensible through feedback from the system or is explained to the user on request.
- Controllability: the dialogue is controllable when the user is able to initiate and control the direction and pace of the interaction until the point at which the goal has been met.
- Conformity with user expectations: the dialogue conforms with user expectations when it is consistent and corresponds to the user characteristics, such as task knowledge, education, experience, and to commonly accepted conventions.
- Error tolerance: the dialogue is error tolerant if despite evident errors in input, the intended result may be achieved with either no or minimal action by the user.
- Suitability for individualization: the dialogue is capable of individualization when the interface software can be modified to suit the task needs, individual preferences, and skills of the user.
- Suitability for learning: the dialogue is suitable for learning when it supports and guides the user in learning to use the system.[ref wiki UI Design].

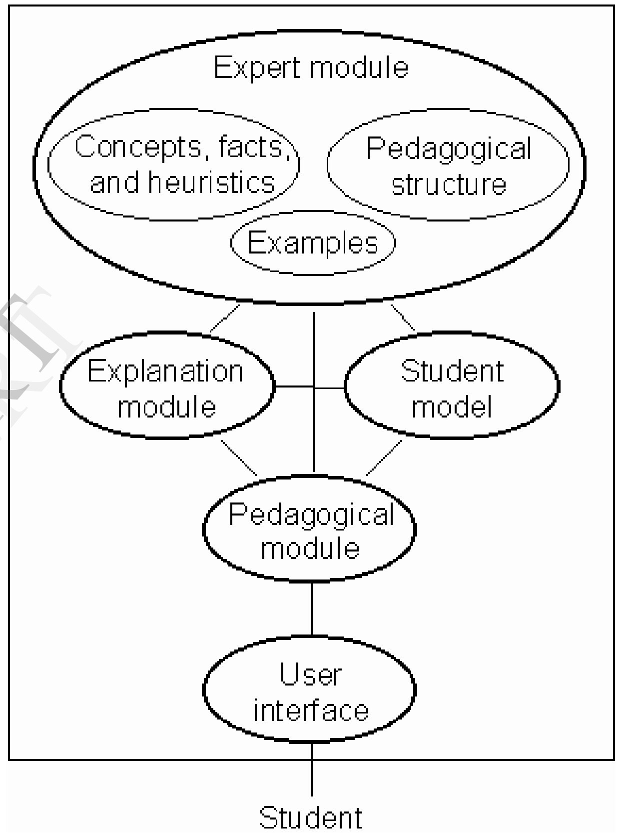


Fig. 1 Typical Architecture of an ITS[8]

III. A BRIEF CHRONOLOGY OF ITS'

This section of the paper presents ITS development across past (1970s-1999) and present (2000-2013). In the past decade, there has been tremendous growth in the field of expert systems and ITS with student modeling as a research area maturing sufficiently.

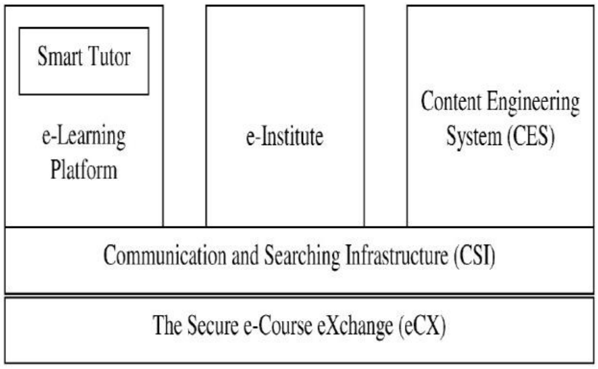
- The 70's saw an era of ITS development with knowledge representation, student modeling, Socratic Tutoring, skills and Strategic knowledge-based learning, buggy library, expert systems and genetic graph. "Bug Library" is a collection of mistakes. In genetic graph, "Genetic" related to the notion of knowledge being evolutionary, and graph denoted the relationships between parts of knowledge expressed as links in a network.

- Cognitive models [5] became a fashion of the 80s with other areas of research and development that gaining prominence such as Natural Language Processing (NLP), authoring shells, fault discovery and predictive modelling coming into play.SOPHIE was built on a powerful and original NLP technique developed by Richard Burton; called Semantic Grammar. It represented a powerful combination of carefully selected keywords with algorithms that searched the context for meaningful variables and objects. Authoring shells are kind of e-learning systems that feature authoring environments for system users, simplify the software development life cycle. Domain knowledge in such systems can be represented by using different knowledge representation specifications[7][11][12]..
- The 1990s saw a lot of multimedia and learner control coming into the sytems. Shift was taking place towards collaborative learning as opposed to individualized approaches adopted thus far
- On the turn of the millenia i.e the past 13 years have shown significant growth in the way the Expert Tutoring Systems were being modelled with new student learning models coming to urface, game-based approaches becoming popular and rather accessible, ETS(s) gained more power and control over their architecture, WIMP Interfaces, Fuzzy Linguistics, Machine Learning and Data Mining started sharpening the edge, and cutting edge technologies to aid the teacher and the taught were engineered.

IV. EXISTING ITS'

We studied various ITSs and their architectures for example Wayang Outpost[15], SMARTTUTOR, Autotutor Here we will discuss two of them.

*A. SMARTTUTOR:*SMARTTUTOR[9] is developed for tutoring of mathematics. Its a subsystem within the SOUL [SPACE Online Universal Learning] system. The SOUL platform is developed by the SOUL project team. It is an online e - learning platform. The system architecture is shown in Fig. 2, this architecture is called the PowerEdBuilder. The secure e-Course eXchange (eCX) is used to provide a secure layer for protecting the copyrighted materials. The Content Engineering System (CES) is mainly used by instructors to create or provide online course materials and launch them. The Communication and Searching Infrastructure (CSI) is used to provide an efficient communication mode among course-instructors, and students.The E-learning platform is a platform where students will interact with system for learning and downloading relevant learning aids provided.The system has multiple modes of usage either individual learning or for classroom with guidance by teacher. This paper claims that this feature of SMARTTUTOR made it different from others. While teaching a topic in the classroom, a human tutor does many steps;SMARTTUTOR is engineered to follow all the steps of Humanlike approaches to teaching such as Analysing the solution, probability of success or any other steps and accomplishing these tasks with utmost efficiency and accuracy



B. Fig.2 An overview of PowerBuilder source: (B. Cheung, March 2002) .

SMARTTUTOR is engineered to explain the core knowledge on the topic, tell which kind of knowledge can be applied to solve problems in the given area, provide examples of problem solving (by solving several problems and explaining each step of the solution), suggest appropriate test papers from previous examinations, and suggest the next most efficient activity for the student to do. SMARTTUTOR thus achieves the flexibility and generality of a tutor in ways that is required to hone individual students' needs and abilities and this became the keynote of SMARTTUTOR.

The main architecture of SMARTTUTOR consists of six components: Course manager, question bank, student model, content structure, expert model, and user interface (UI).

The Course Manager is the control center of Smart Tutor. It has the following responsibilities:

1. Invoke the Planner to create a tailored course content from Domain Knowledge for each student.
2. Invoke the Advisor to give individualized instructions and suggestions during each student's learning process.
3. Maintain student's personal information, learning status, testing results in the Student Model
4. Deliver the course via the User InterfaceCreates adaptive test or examination via the Question Bank.
5. Call other components to perform different tasks.

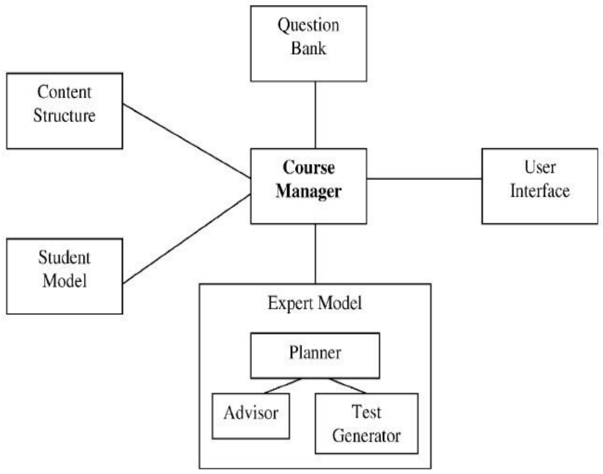


Fig.3 Architecture of SMARTTUTOR.

Content structure : In SMARTTUTOR,content structure consists of various learning materials.

C. *WAYANG OUTPOST:* Wayang Outpost is a web-based tutoring system that teaches students to solve geometry SAT problems[10]. Wayang Outpost uses multimedia to help students in their problem solving process, directing their attention, animating parts of the solution, and emphasizing concepts with sound. Multimedia was seen as a motivation for the students and aimed at making the coursework interesting, by adding a video-game style to the tutor. This tutoring system was an improvement over others as it used multimedia content like audio and video to provide the instruction. Wayang contains both short-term transfer problems as well as long-term transfer problems. Short-term transfer problems means variations of the operands of a previous high stakes exam problem and minor changes in figures to assist learning by providing the necessary scaffold to the students. Long-term transfer problems present real-world math problems following a story line . In Wayang Outpost, there is a centralized database. Every data-element likethe problem, its solution, hint(s), students' work, their performanceis stored in a centralized database. From this data, the system makes inferences on an ongoing basis to select problems at the appropriate level of challenge, and chooses hints that will be helpful for the student.

In Wayang Outpost architecture there are three part client side, server side and interface. Client side is basically a browser which is made of flash plug-in. It is the start point for users after login they directed to another side. Server side is completely database where all data are stored. It has some models like domain models, student models and support models. Domain model contains problem and its solution. It also contains the hints to solve the problem. Student model contains the information of student, its performance chart, its level of skills. Support model contains animation, sound effects, etc to make it effective.

There are two types of hints:

Wayang outpost created two kinds of hints, one based on an analytical approach and the second based on a visual estimations approach . It also provide other types of hint as required by the student. Students can adopt the approach according to their level of understanding. Hints are generally provided according to students' cognitive profile.Wayang Outpost doesn't trace each step of the student's solution, because this would be too expensive to implement for so many different problems. It uses the concept of data-centric approach with Bayesian Networks to categorize 3 types of students;. who already knows a skill,is learning a skill and is not learning skill. The tutor observes the hints requested by the student to reach the solution. If the student requests very less hints and gives correct solutions (not all) it means he/she already knows a skill. If the student makes correct answer after using hints it means he/she is learning a skill but the student is said to have not learnt the skill when the student has a low score for the skill at the beginning of the session and there is not a significant improvement in the number of correctly solved problems involving the skill from the beginning to the end of the session.

V. PROBLEM WITH ITS' AND DESIGN SUGGESTIONS FOR NEW - AGE SYSTEMS

The one common problem with most of the tutors is, that they, are designed, only for one strategy, i.e. their architecture supported, only one teaching methodology. ITS' have been developed for Geography, Digital Circuits, Medical Diagnosis, Computer Programming, Mathematics, Physics, Genetics, and Chemistry, etc. As we can see all these ITSs are made to teach only one specific subject/concept.

The absence of a generic ITS framework which is not restricted to one strategy rather can incorporate multiple strategies at a time and also serves as a platform for teaching multiple courses.

Various Teaching strategies commonly implemented:

- Scaffolding
- Socratic Questioning and
- Game-based learning (much recent).

Each of these strategies have been explained below in a concise manner.

A. *SCAFFOLDING:*Scaffolding[15] also known as Instructional scaffolding is a learning process designed to promote a deeper learning. Scaffolding is the support given during the learning process which is tailored to the needs of the student with the intention of helping the student achieve his/her learning goals. The best and most effective use of instructional scaffolding helps the learner figure out the task at hand on their own. Once students build knowledge and develop skills on their own, elements of the support are removed.According to McKenzie (1999), the defining features of successful scaffolding include clear direction, purpose, and expectation. Results include on-task activity; better student direction; reduced uncertainty, surprise, and disappointment; increased efficiency; and palpable momentum.

B. *SOCRATIC QUESTIONING:* Socratic Questioning is an approach in which teaching-learning is performed in the form of question and answer. It is a kind of series of questioning in which an original question is split into more than one low level questions[17]. It is just like bottom up approach. In this strategy we start from the question which student or learner knows and goes to our target questions which we want to teach him. Socratic Questioning is basically a dialogue conversion between the teacher and student. First, instructor starts the question and student responses. In return, instructor reformulates a new question according to the response given by student. Questioning and answering is structured systematically to reach an ultimate goal.

C. *GAME BASED LEARNING:* Game based learning (GBL) is a branch of serious games that deals with applications that have defined learning outcomes. Generally they are designed in order to balance the subject matter with the gameplay and the ability of the player to retain and apply said subject matter to the real world. GBL uses competitive exercises, either pitting the students against each other or getting them to challenge themselves in order to motivate them to learn better. Games often have a fantasy element that engages players in a learning activity through a storyline.The basic idea is that students learn better through games.

This paper tries to shed some light on how an effective ETS can be developed if its made more adaptive and much more responsive, since yonder days educationists have been trying to develop that one perfect system that can teach effectively. The evaluation of ITS is an important though often neglected development stage. There are many evaluation methods available but literature does not provide clear guidelines for the selection of evaluation method(s) to be used in a particular context. Conventional computer programs are sometimes verified and validated through formal proofs of correctness. However, this technique is not suited for AI programs which deal with analytically intractable problems, represented as incompletely specified functions.Extensively validated research in cognition, perception, and learning as indicated by Jay, in 1983; Jonassen & Hannum, in 1987; Larsen, in 1985 suggests ways to design and improve educational programs, particularly the interface and user-related features.The way Humans interact with these "Expert" agents holds the key to unlocking the pandoras box of developments that can take place in the design of these systems, for this the subject of HumanComputer Interaction commonly HCI, has gained a lot of importance to channelize development in this field. An expert's knowledge/inspection (called evaluation) is used as an explicit standard for judging a program. Due to the complex and adaptive nature of these agents this phase is not easy to implement thus causing a lot of technical glitches in the process. Basic bottlenecks faced by ITS' are the scarcity of experts, their greatly varying degrees of expression and the one major boulder to overcome is the method of converting human knowledge into documented form. This is an area to further introspect.Although Machine learning has greatly improved the process of dynamic student modelling the complexities in the systems greatly increase is machine learning algorithms are to be incorporated. Moreover there has been next to negligible research done in bringing these systems to people with disabilities, which is an important area to look into, one can say that its where such expert systems are most required.

VI. CONCLUSIONS

The paper chronologically presents the development of ITS..It presents the retrospect present and the prospect of ITS. Over the period, they have gradually moved closer to the individual student learning need. Adaptability and user-friendliness have been the key concepts. The paper explains how an ITS can be conceptualized by infusing different strategies can be infused into making one compltely interactive strategy that gives the multifold advantaghes. Further, the prospective areas for future Expert Tutoring System development have been outlined for recommended research work. As discussed Human Computer Interaction has emerged as an area offering definite potential in development of such Artificially Intelligent Agents and demanding intervention by research scientists.

ACKNOWLEDGMENT

The authors would like to sincerely thank Prof. B.B. Gite for his motivating support, guidance andvaluable suggestions during this project work. His guidance has helped the authors to understand theconcepts and he has been always motivating to them.The authors are extremely grateful to him for spendinghis valuable time to help them clarify their concepts..

REFERENCES

[1] Ivon Arroyo, Carole Beal, Tom Murray, Rena Walles, Beverly Woolf, Wayang Outpost: Itelligent Tutoring for High Stakes Achievement Tests , University of Massachusetts, Amherst.
[2] http://en.wikipedia.org/wiki/Intelligent_tutoring_system
[3] Hyacinth S.N, 1990, Intelligent tutoring Systems: An Overview, AI Review, pp 251-277.
[4] Brown, J. S., Burton, R. R., 1975. Multiple representations of knowledge for tutorial reasoning, Representation and understanding: Studies in Cognitive Science, pp.311-349.
[5] Shute, V. J., Psotka, J., 1994., Intelligent Tutoring Systems: Past, Present, and Future, Human resources directorate manpower and personnel research div., pp. 2-52.
[6] Padayachee I. 2002. Intelligent Tutoring Systems: Architecture and Characteristics.
[7] Brown, J. S., Burton, R. R., 1975. Multiple representations of knowledge for tutorial reasoning, Representation and understanding: Studies in Cognitive Science, pp.311-349.

[8] Vladan Devedzic, John Debenham, Dušan Popovic (2000). Teaching Formal Languages by an Intelligent Tutoring System:ISSN 1436-4522.
[9] B. Cheung, L. Hui, J. Zhang, S.M. Yiu, SmartTutor: An intelligent tutoring system in web-based adult education, Journal of Systems and Software (2003), http://www.sciencedirect.com/science/article/pii/S0164121202001334
[10] Ivon Arroyo, Rena Walles, Carole R. Beal, Beverly P. Woolf, Tutoring for SAT-Math withWayang Outpost, University of Massachusetts, Amherst.
[11] Itko B., et al., 2009, Dynamic test generation over ontology-based knowledge representation in authoring shell, Expert Systems with Applications, Vol 36, No. 4, pp 8185-8196.
[12] Rosic, M, Vlado, G, Branko, Z., 2004, Intelligent authoring shell based on web services, Proc: INES -8th International conference on Intelligent Engineering Systems, pp 50-55.
[13] Clemente, J., Ramírez, J., de Antonio, A., 2011, A proposal for student modeling based on ontologies and diagnosis rules, Expert Systems with Applications, 38(7), pp 8066–8078.
[14] A. C. Graesser, S. Lu, G. T. Jackson, H. Mitchell, M. Ventura, A. Olney,and M. M. Louwerse, AutoTutor: A tutor with dialogue in natural lan-guage, Behav. Res. Methods, Instrum., Comput., vol. 36, pp. 180193, 2004.
[15] L.S. Vygotsky: Mind in Society: Development of Higher Psychological Processes, p. 86.
[16] Yogendra Pal's APS report:Intelligent Tutoring Systemto Teach Programming ToBilingual Students,IIT Bombay.
[17] Cotton, K. (2001). Classroom questioning. School Improvement Research Series(SIRS).http://www.nwrel.org/scpd/sirs/3/cu5.html(25.10.2005).
[18] M. Spectre, and M. Prensky (2001), Theoretical Underpinnings ofGames2Train.com's approach, Games2train, 2001.
[19] Garris, R., Ahlers, R., & Driskell, J. E., Games, motivation, and learning,Simulation & Gaming, An Interdisciplinary Journal of Theory, Practice and Research. Vol33, No.4 Dec. 2002, pp. 441-467.
[20] Clemente, J., Ramírez, J., de Antonio, A., 2011, A proposal for student modeling based on ontologies and diagnosis rules, Expert Systems with Applications, 38(7), pp 8066–8078.
[21] Brusilovsky, P., Peylo, C., 2003, Adaptive and intelligent web based educational systems, International Journal of AI in Education, 13, pp 156–169.
[22] R. W. Picard, Affective Computing, MIT Press, 1997.
[23] A. Sarrafzadeh,, 2008, How do you know that I don't understand? A look at the future of intelligent tutoring systems, Computers in Human Behavior, Vol 24, no 4, pp 1342-1363.
[24] He, Yulan et al., 2009, Automatic summary assessment for intelligent tutoring systems, Computers and Education, 53(3), pp. 890–899.
[25] O. Ozyurt,, 2013, Design and development of an innovative individualized adaptive and intelligent e-learning system for teaching–learning of probability unit: UZWEBMAT, Expert
[26] Systems with Applications, Vol 40 no 8, pp 29142940.
[27] Jay, T.B, 1983, The cognitive approach to computer courseware design and evaluation, Educational Technology, 23(1), pp 22-26.
[28] Jonassen, D.H.and Hannum, W.H, 1987, Research-based principles for designing computer software, Educational Technology, 27, pp 7-14.
[29] Larsen, R. E, 1985, What communication theories can teach the designer of computer-based training, Educational Tech., 25(7), pp 16-19.
[30] O. Mich, E, 2013, Interactive stories and exercises with dynamic feedback for improving reading comprehension skills in deaf children, Comp. & Education, Vol 65, pp 34-44.
[31] Sylvia Encheva, On Providing Automated Support to Students Based on Their Personal Preferences, IJCSI, Vol. 10, Issue 2, No 3, March, 2013.
[32] M. Goyal, E-learning: Current State of Art and Future Prospects, IJCSI, Vol. 9, Issue 3, No 2, May 2012.
[33] L. Anthonyn, J. Yang, K. R. Koedinger, 2012, A paradigm for handwriting-based intelligent tutors, IJHCS, Vol 70 no 11, pp 866887.
[34] [40]. C. Romero, S. Ventura, 2012, Data mining in education, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, Vol 3, no 1, pp 12-27.
[35] [41]. Y. Yang, H. Leung, L. Yue, L. Deng, 2013, Generating a two-phase lesson for guiding beginners to learn basic dance movements, Computers and Education, 61, pp 1-20.
[36] Ariel Arbiser, Practical Crossword Generation with Checkpoint Search, (Matrahaza, 1995), University of Buenos Aires Ciudad Universitaria, Argentina, 2001, pp. 1–6.
[37] Vassileva, J. (1990). An Architecture and Methodology for Creating a Domain-Independent, Plan-Based Intelligent Tutoring System. Educational and Training Technology International, 27 (4), 386-397.
[38] Issroff, K. & del Soldato, T. (1996). Incorporating Motivation into Computer-supported Collaborative Learning. In Proceedings of The European Conference on Artificial Intelligence in Education, Lisbon: Edicoes Colibri, 284-290.
[39] Anderson, J. R., Boyle, C. F., Corbett, A. T. and Lewis, M. W. (1990). Cognitive Modelling and Intelligent Tutoring. Artificial Intelligence, 42 (1), 7-49.
[40] Anderson, J. R., Boyle, C. F., Corbett, A. T. and Lewis, M. W. (1990). Cognitive Modelling and Intelligent Tutoring. Artificial Intelligence, 42 (1), 7-49.