http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (12): 2306-2314, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.2306.2314

Reinforcement Learning Approach for Adaptive E-learning Systems using Learning Styles

Balasubramanian Velusamy, S. Margret Anouneia and George Abraham
School of Computer Science and Engineering, VIT University, Vellore-632014, India

Abstract: This study aims to provide a mathematical modelling for an adaptive e-Learning environment using SARSA algorithm, by relating it to the concept of Reinforcement Learning. An adaptive e-learning system based on learning styles is very much alike an intelligent agent. The system needs to assess the various interactions from the user and provide them with the best possible content, so as to enhance the learning experience of the user. Successive interactions with the system by the same user should result in he/she being provided with the best content in the same manner as the previous time, if not for a better option. This is possible as the system will have learned from its previous interaction.

Key words: Semantic, e-learning, SARSA, ontology, learning styles

INTRODUCTION

The boom in the Internet sector and its percolation into all domains of human life as changed the dimensions of the education system with the introduction of virtual learning or e-learning. There has and will always be a steady rise in the demand of e-learning systems catering to different needs in the various fields and levels of education (Brusilovsky and Peylo, 2003; Nath *et al.*, 2012). An e-learning system with the facility of personalization or adapting to the users' needs is the ideal system that is required, as people with different skill sets use the system. Some people may be fast learners while some may be slow, some may need to practice more problems while others may need just example. These preferences are in general called the learning styles of an individual. Creation of a learner model enables in the capturing of all the preferences and needs of an individual. This learner model can be extracted from the personality factors of an individual like their learning styles, their behavioural factors like their browsing history or general patterns in browsing and knowledge factors like the user's prior knowledge (Ozpolat and Akbar, 2009). Out of all the three factors mentioned above, creation of a learner model from the learning styles of an individual yields the bests results if proper personalization can be provided based on it (Dag and Gecer, 2009).

The main challenge is the detection of the learning styles. Researchers have described various learning styles models like Myers and Myers (1980), Kolb (1984), Honey and Mumford (1992), Dunn and Dunn (1978) and Felder and Silverman (1988). Research has proved that the Felder-Silverman Learning Style Model (FSLSM) is the most suited for the engineering students' environment as it also considers the psychological aspects of a person (Felder and Spurlin, 2005). The Index of Learning Styles (Felder and Soloman, 2012) is a questionnaire-based approach for detection of learning styles based on the FSLSM. The problem with questionnaire-based approach is that it suffers from the "inaccurate self-conceptions of students" (Dung and Florea, 2012; Graf *et al.*, 2008) at a specific time. Moreover these questionnaires are incapable of tracking the changes in a learner's learning style, i.e., the dynamicity of the learning style.

As a result of these problems, various researches have been conducted to come out with alternate automated solutions for learning style detection. These works can be broadly classified into two groups: data-driven approach and literature-based approach. Some of the noticeable works in the data-driven approach are by using Bayesian Networks (Schiaffino *et al.*, 2008; Garcia *et al.*, 2007), NBTree classifiers (Ozpolat and Akbar, 2009) and Genetic Algorithms (Chang *et al.*, 2009). Literature-based approach is a relatively new method with some of the noticeable works being done by Graf *et al.* (2008), Dung and Florea (2012) and Simsek *et al.* (2010).

Personalization of the e-Learning system means to provide a system that adapts according to the learners' learning process.

Corresponding Authors: Balasubramanian Velusamy, Parallel and Distributed Computing Division,
School of Computing Science and Engineering, VIT University, Vellore-632014, India Tel: 0416-2202811

ADAPTIVE WEB-BASED EDUCATION

The concept of an adaptive system was initially stressed by Brusilovsky and Peylo (2003). They talk a about improving the system of web-based education by providing an Adaptive and Intelligent Web-Based Educational System (AIWBES) as an alternative to the traditional systems. AIWBES adapts to the learners' needs, knowledge and behaviour like a human teacher would do. An adaptive system modifies its solutions to a problem based on various factors, for instance the learners' previous experience with the system whereas an intelligent system provides the same solution irrespective of the different needs of the learners. AIWBES is a mixture of adaptive hypermedia technologies and intelligent tutoring technologies. It also contains adaptive information filtering, intelligent monitoring and intelligent collaborative learning. Adaptive hypermedia mainly consists of adaptive presentation and adaptive navigation support while intelligent tutoring mainly consists of curriculum sequencing, problem solving support and intelligent solution analysis.

AUTOMATIC LEARNING STYLE RECOGNITION

Due to the various disadvantages of questionnaire-based learning style detection, the process has to be automated so that it can incorporate various aspects of the learner while modelling the learner. The process of automatic detection of learning styles consists of two phases: Identifying the relevant behaviour for each learning style and Inferring the learning style from the behaviour (Graf, 2007), as shown in Fig. 1.

The first step of identifying the relevant behaviour for each learning style consists of the following phases: Selecting the relevant features and patterns of behaviour, classifying the occurrence of the behaviour and defining the patterns for each dimension of the learning style (Graf, 2007), as shown in Fig. 2.

All this is performed by studying the various literatures of the respective learning model and other supportive research works that have already been done. The second step of inferring the learning style from the respective behaviour is where the approaches differ. But the initial step is of preparing the input data which is common. This input data is prepared from the extracted information and is formulated in the form of matrices that corresponds to each learning style. Then the calculation methodology can be data-driven or literature-based approach (Graf, 2007), as shown in Fig. 3.

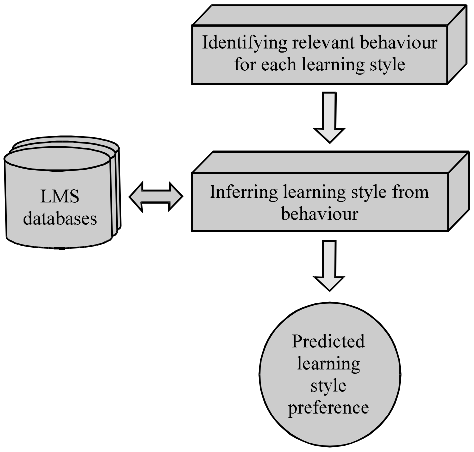
Identifying relevant behaviour for each learning style
LMS databases
Inferring learning style from behaviour
Predicted learning style preference

Fig. 1: Idea of automatic detection of learning style preference

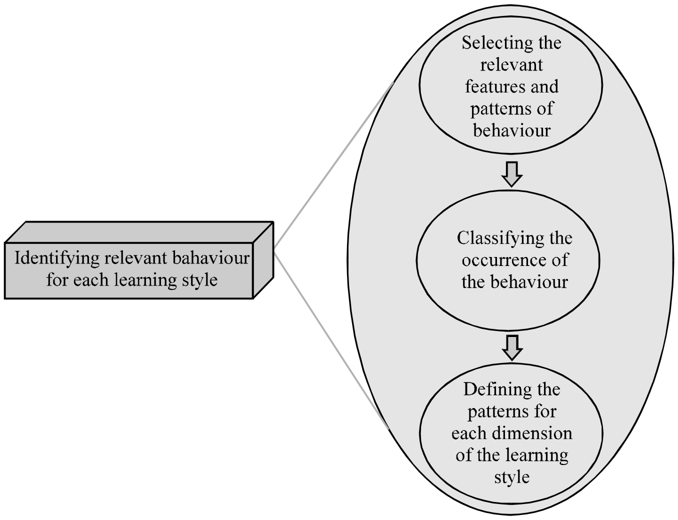
Identifying relevant bahaviour for each learning style
Selecting the relevant features and patterns of behaviour
Classifying the occurrence of the behaviour
Defining the patterns for each dimension of the learning style

Fig. 2: Identifying the relevant behaviour for each learning style

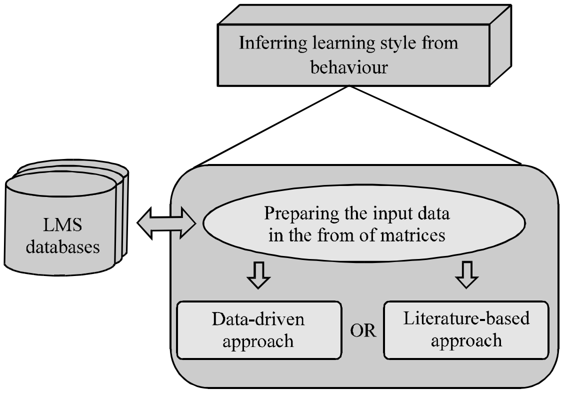
Inferring learning style from behaviour
LMS databases
Preparing the input data in the from of matrices
Data-driven approach
OR
Literature-based approach

Fig. 3: Inferring learning styles from their respective behaviour

The approach mentioned in this study focuses on the use the Felder-Silverman Learning Style Model and follows a mix of both the data-driven and literature-based approach by creating an ontological framework that can be then reasoned upon by a simple rule engine to detect the learning style. A lot of research work has been undertaken in the field of automatic detection of learning styles and modelling of student behaviour for providing an adaptive personalized e-learning environment. Various techniques have been proposed and researched upon to automate the learning style detection process.

All these techniques can be broadly classified into data-driven techniques and literature-based approaches. Some of the various data-driven techniques are the SAVER system based on Bayesian networks by Schiaffino *et al.* (2008) (Garcia *et al.*, 2007), the NBTree classification with Binary Relevance Classifier-based model by Ozpolat and Akbar (2009), the iLessons system by Sanders and Bergasa-Suso (2010) Bergasa-Suso *et al.* (2005), the fuzzy rule approach by Deborah *et al.* (2012), the enhanced K-nearest Neighbour (k-NN) combined with Genetic Algorithms (GA) approach by Chang *et al.* (2009), the social bookmarking and Learning Vector Quantization approach by Darwesh *et al.* (2011) the Evolutionary Fuzzy Clustering (EFC) with Genetic Algorithm (GA) approach by Montazer and Ghorbani (Ghorbani and Montazer, 2011) (Saberi and Montazer, 2012) and the recommender system-based approach by Jyothi *et al.* (2012).

Literature-based approach is a new methodology that is being followed by researchers. This method is beneficial as it is LMS independent and also the data need not be present while modelling the students' behaviour. Some of the noticeable works are those done by Graf *et al.* (2008), Dung and Florea (2012) and Simsek *et al.* (2010). These works differ in terms of the behavioural patterns that are considered for calculating the matching hints.

Table 1 gives a summary of the complete literatures survey, mentioning the approach, technology, key points, assessment methods and the precision/accuracy obtained. Data-driven approach and literature-based approach both have their own benefits. Data-driven approach is more accurate as it is based on pre-collected data set, while literature-based approach has the freedom of LMS and other inherent systems.

REINFORCEMENT LEARNING

In computer science, reinforcement learning (Sutton and Barto, 1998) is an area of machine learning concerned with what actions an agent, i.e. an intelligent program, should take in an environment so as to maximize the cumulative reward. The agent learns by sensing various parameters from the environment, exploring various possibilities for a better reward and when performing the same task again, exploiting the already learned best path.

The e-learning system that is the topic of concern in this study is similar to an intelligent agent. It senses the various user interactions and has to decide on the best possible responses to the user so as to enhance the learning experience of the user. If the same user uses the system again, then the previously learned options that were best suited to the person should be given again, if not for a better option.

Reinforcement learning scenarios are described by states, actions and rewards. There exist two main reinforcement learning algorithms - Q-Learning Algorithm (Watkins, 1989) (Watkins and Dayan, 1992) and SARSA Algorithm (Rummery and Niranjan, 1994).

Q-learning algorithm: Q-Learning (Watkins, 1989) is a form of model-free reinforcement learning (Watkins and Dayan, 1992). The problem domain consists of an agent its various states S and a set of actions per state A. The agent can move from one state to another by performing some action a ? A. The transition, i.e. the next state gives a reward to the agent. The goal of the agent is to maximize the total reward. This is achieved by optimizing the actions for each state. Hence, there exists a function Q that calculates the quality of each state-action combination. Initially Q returns a fixed value that the designer has set. Then during each step when the agent is rewarded, new values are calculated and updated (Eden *et al.*, 2013).

Table 1: Summary of the literature survey

Study	Approach	Technology	Key points	Assessment methods	Precision/accuracy
Bergasa-Suso *et al.* (2005)	Data-driven	Browser-based system with rules	Processing dimension	67 students-ILS (Training) 7 students-iLessons	71%-Processing
Garcia *et al.* (2007)	Data-driven	Bayesian networks	Detection only	27 systems engineering students-AI-SAVER	58%-Processing 7%-Perception 63%-Understanding
Schiaffino *et al.* (2008)	Data-driven	Bayesian networks	Detection+suggestions	42 Systems engineering students-AI-SAVER with eTeacher	83% feedback received was positive
Graf *et al.* (2008) Based on Graf (2007)	Literature-based	Simple rules on matching hints	LMS independent Better results that data-driven approach	127 students-Info. Sys. Comp.Sci.-Austria Univ.-Object oriented modeling-moodle LMS	77.33%-Input 79.33%-Processing 76.67%-Perception 73.33%-Understanding
Ozpolat and Akbar (2009)	Data-driven	NBTree	Detection+suggestion Uses only data objects selected by the user LMS independent	10 graduate student (Training) 30 graduate students (Testing)-PoSTech	53.3%-Input 70%-Processing 73.3%-Perception and understanding
Chang *et al.* (2009)	Data-driven	Enhanced k-NN clustering with GA	k-NN-pre-contrast and post-comparison reduced No. of behavioural features	IRIS dataset by UCI 117 students-SCORM-compatible Java-based LMS-windows XP	Increasing accuracy
Sanders and Bergasa-Suso (2010)	Data-driven	Browser-based system with rules for reasoning	More dimensions Improved rules Unknown category	67 students-ILS (Training) 7 students-same research task-iLessons	82%-Input 81%-Processing 69%-Understanding
Simsek *et al.* (2010)	Literature-based	Simple rules on matching hints	Processing dimension 6 features considered	27 students-Comp. Educ.-Derivatives-Moodle LMS	79.63%-Processing
Darwesh *et al.* (2011)	Data-driven	Analyzing web page content interactions using social bookmarking	No training methods used ILS filled using social bookmarking sit	Study conducted on 25 and 15 learners participating in a bookmarking site such as www.tagme1.com	For No. of learners = 25, Recognition = 72% For No. of learners = 15, Recognition = 86.66%
Darwesh *et al.* (2011)	Data-driven	Web pages tagging using social bookmarking and learning vector quantization	No special effort from student side for collecting data LMS independent	By varying the Learning rate No. of hidden neurons Values of epochs	For learning rate = 0.01, No. of hidden neurons = 40 and Epochs = 150, Recognitionrate=93.33% for 15 learners
Ghorbani and Montazer (2011)	Data-driven	Evolutionary Fuzzy Clustering (EFC) method using genetic algorithm	People with similar learning styles in a cluster High computational and memory usage costs, so use particle swarm optimization technique	98 undergraduate students-fundamentals of computer networks course	EFC has more accuracy than grouping based on behaviour in log files of LMS
Deborah *et al.* (2012)	Data-driven	Fuzzy logic	Bell-shaped membership function Better classification for “Unknown”	Comp. Sci. Engr.-Anna Univ.-C-language	-NA-
Dung and Florea (2012)	Literature-based	Simple rules on matching hints	LMS independent Parameters-No. of visits and time spent	44 UG students-Comp. Sci.-Politechnica Univ., Bucharest-AI course-Web-based LMS POLCA	70.15%-Input 72.73%-Processing 70.15%-Perception 65.91%-Understanding
Saberi and Montazer (2012)	Data-driven	ILS+LMS logs+ Bayesian networks	Improved accuracy Decreased uncertainty	40 M.Sc. students on 4 different courses, done in three phases	-NA-
Jyothi *et al.* (2012)	Data-driven	Recommender system based on ILS and clustering	Best for data sets less than 150 users Prior knowledge of learner needed LS captured by ILS	105 students-C-DAC Hyderabad R and D labs-R and D+courses like Embedded systems, system S/W and Adv. business computing	Good accuracy for student user profile data sets less than 150

The following represents the Q-learning algorithm (Watkins, 1989):

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t)+\alpha[R(S_t, a_t)+\gamma \max[Q(S_{t+1}, a_{t+1})]-Q(S_t, a_t)] \quad (1)$$

Where:

$\leftarrow$ = Updating the old value

t = Current interaction

$t+1$ = Next interaction

$Q(S_t, a_t)$ = The Q-values of the current interaction

$R(S_t, a_t)$ = Reward obtained for performing action a_t in S_t

α = Learning rate ($0 \le \alpha \le 1$)

γ = Discount factor that decides the importance of future rewards ($0 \le \alpha < 1$)

Q-Learning algorithm is also called as an “Off-Policy” of “Policy Independent” algorithm as it does not depend on any policy. Policy is the decision process used to select an action given a certain state. Q-Learning works on the greedy mechanism by selecting the maximum of the Q-values of all the state-action pairs that are possible from the current state. The agent learns through experience (unsupervised) by exploration. Each exploration is called an episode (Sutton and Barto, 1998). In an episode, an agent moves from initial to goal state after which the next episode starts.

In the Q-learning algorithm, if the discount factor (γ-value) is set to 0, i.e. $\gamma = 0$ then:

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) + \alpha[R(Q(S_t, a_t) - Q(S_t, a_t]) \quad (2)$$

This means that the updating can happen then and there itself without the consideration of a new state.

Now, if the learning rate (α value) is set to 0, i.e. $\alpha = 0$ then:

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) \quad (3)$$

This means that no learning takes place and the value remains as it is.

Else, if the learning rate (α value) is set to 1, i.e. $\alpha = 1$ then:

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) + R(S_t, a_t) - Q(S_t, a_t) \leftarrow R(S_t, a_t) \quad (4)$$

This means that the agent will consider only the most recent information, i.e., the reward only. This is too simplistic.

If $\alpha = 0.5$ (for example), the old and the new Q-values meet half way, given the reward.

Hence, when the discount factor γ is set to 0, the agent will consider only the current reward. It has only short term greedy goals.

In the Q-learning algorithm, if the discount factor (γ value) is set to 1, i.e., $\gamma = 1$ and also α=1 then:

$$\begin{aligned} Q(S_t, a_t) &\leftarrow Q(S_t, a_t) + R(S_t, a_t) + \max[Q(S_{t+1,} a_{t+1}) - Q(S_t, a_t) \quad (5) \\ &\leftarrow R(S_t, a_t) + \max[Q(S_{t+1}, a_{t+1})] \end{aligned}$$

This means that the updated Q-value for a state is equal to the reward plus the maximum of the possible Q-values from the state. As ε tends to 1 or more, the more it optimizes for long-term goals. It is ideally set in between 0 and 1 (Eden *et al*., 2013).

Also to be noted is that for values of α that are close to 1, the same Eq. 5 can be used.

Sarsa algorithm: SARSA (Rummery and Niranjan, 1994) algorithm is an improvement on the Q-learning algorithm where the name comes from the fact that the rule that updates the Q-value depends on the current state a_t+1 the action the agent chooses a_t, the reward r, the next state that the agent will be in after taking the action a_t+1 and the action that the agent will take in the new state a_t+1. The name SARSA stands for the same-state, action, reward, state (next state), action (next action).

The following represents the SARSA algorithm (Rummery and Niranjan, 1994; Eden *et al*., 2013; Van Hasselt, 2013):

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) + \alpha[R(S_t, a_t) + \gamma\, Q(S_{t+1}, a_{t+1}) - Q(S_t, a_t)] \quad (6)$$

SARSA algorithm is also called as an “On-Policy” of “Policy Dependent” algorithm as it depends on the decision process used to select an action given a certain state. It does not have a greedy approach like the Q-learning algorithm.

The policy selection process is not always selecting the action that results in the maximum Q-value as this will lead to a phenomenon of “local maxima”. Instead it is determined on a factor epsilon a which determines the extent to which the actions are randomized.

There are three types of action selection policies:

- **ε-greedy:** Action with the highest estimated reward is chosen independent of the Q-value estimates
- **ε-soft:** Best action is selected with a probability of 1 - ε-and rest of the time the actions are selected uniformly
- **Softmax:** The problem with both the above methods is the uniform selection that may result in the worst possible action being selected as second best

A solution to this is the softmax policy where a rank or a weight is assigned to each action according to its action-value estimate and actions are selected based on these weights, as a result of which the worst actions are unlikely chosen.

Assuming that the ε-soft policy is being followed Eq. 6 can be re-written as:

$$Q(S_t, a_t) \leftarrow Q(S_t, a_t) + \alpha[R(S_t, a_t) + k\gamma Q(S_{t+1}, a_{t+1}) - Q(S_t, a_t)] \quad (7)$$

where, is k-The probability of selecting the action.

GENERIC MODELLING SOLUTION

Consider the state-diagram, as shown in Fig. 4, with the action and their respective rewards mentioned on the

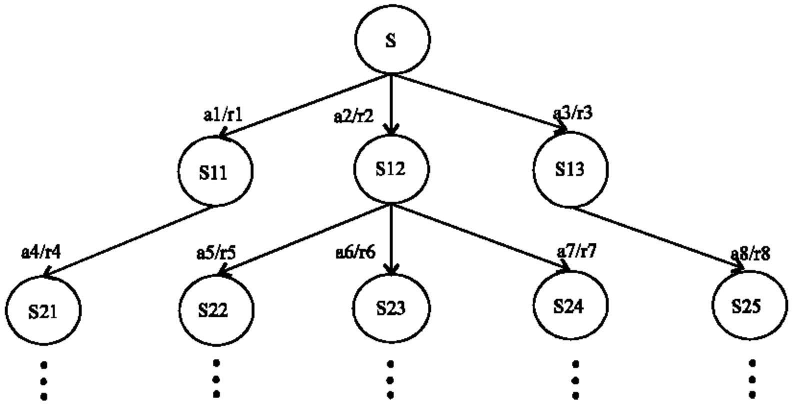

Fig. 4: Generic state-action-reward diagram

Table 2: Reward table of the generic state-action combination

	Action							
State	a1	a2	a3	a4	a5	a6	a7	a8
S	r1	r2	r3	-	-	-	-	-
S11	-	-	-	r4	-	-	-	-
S12	-	-	-	-	r5	r6	r7	-
S13	-	-	-	-	-	-	-	r8
S21	-	-	-	-	-	-	-	-
S22	-	-	-	-	-	-	-	-
S23	-	-	-	-	-	-	-	-
S24	-	-	-	-	-	-	-	-
S25	-	-	-	-	-	-	-	-

Table 3: Action table of generic state-next state combination

	Next								
State	S	S11	S12	S13	S21	S22	S23	S24	S25
S	-	a1	a2	a3	-	-	-	-	-
S11	-	-	-	-	a4	-	-	-	-
S12	-	-	-	-	-	a5	a6	a7	-
S13	-	-	-	-	-	-	-	-	a8
S21	-	-	-	-	-	-	-	-	-
S22	-	-	-	-	-	-	-	-	-
S23	-	-	-	-	-	-	-	-	-
S24	-	-	-	-	-	-	-	-	-
S25	-	-	-	-	-	-	-	-	-

arrows. This diagram represents the state S_i that can be attained on taking action a_i and also the reward r_i that is achievable on taking that action.

Table 2 represents the reward received on performing a particular action at each stage while Table 3 represents the action required to be performed to move to the next state.

Assume that we are starting from state S and moving towards some goal state.

Solving by Q-learning algorithm: Initially Q-values for all are set to 0.

Observations from Table 2 and 3 show that there are 3 possible actions that can be taken from S, i.e. a1, a2 and a3.

By randomly selecting an action, select a2:

$$\begin{gathered}Q(S,a2)\leftarrow Q(S,a2)+\alpha[R(S,a2)+\gamma\max[Q(S12), \text{all actions}] \\ -Q(S,a2)\leftarrow\alpha[R(S,a2)+\gamma\max[Q(S12,a5), \\ Q(S12,a6), Q(S12,a7)]-Q(S,a2)\end{gathered}$$

Solving by SARSA algorithm: Initially Q-values for all are set to 0.

Observations from Table 2 and 3 show that there are 3 possible actions that can be taken from S, i.e., a1, a2 and a3.

The first action a2 is selected randomly. On performing this action, the system will reach state S12. From here, the system can select any of the three actions- a5, a6 or a7. This selection is done with a probability $1- \varepsilon$ for the best action and uniform probability for the remaining actions. The best action is assumed to be the one that results in highest reward. This probability is substituted as the value of k in the equation.

Therefore:

$$Q(S, a2)\leftarrow Q(S\ a2)+\alpha[R(S, a2)+k\gamma[S12, \text{action})]-Q(S\ a2)]$$

In both the solutions, after the first episode is over, the system, i.e., the agent starts over again. In the consecutive runs, the agent is aware of the best path with the help of the Q-values. This helps the agent to work efficiently.

MODELLING THE E-LEARNING SYSTEM

For modelling the e-learning system, SARSA algorithm is best suited as it depends on the action that is performed rather than randomly selecting the action that gives the best result.

Assuming that the e-learning system is a Learning Management System (LMS) like Moodle (www.moodle.org). A course, suitable for e-learning environment, is uploaded to the system. The course contents can be theory relevant to the topic, examples, programming codes, practice exercises, diagrams and tests to evaluate the student. Users can browse through the system, select the course, read its relevant theory, go through the examples, practice exercise questions, solve tests and participate in forum discussions. The forum is assumed to be one where the student posts a question to which the teacher responds to. Peer discussions are not assumed to be present at the moment. The e-learning system keeps track of the students' log files, details from which we use for later analysis.

Assuming some of the possible states the system can be in, the possible actions that cause the transformation and also some of the possible rewards that the system can offer.

States: "Start", "ReadingTheory", "ReadingExtraMaterial", "SolvingEvercises", "GoingThroughQ and A", "WaitingForAnswer", "WaitingForResult", "UnderstandingAnswer", "Discussion", "UnderstandingExplanation", "GuidanceNeeded", "Idle", "End".

Actions: "read", "readMore", "solve", "submitting", "askDoubt", "discuss", "forMoreUnderstanding", "givingUp", "Q and A", "understanding", "exit".

Rewards:

- Completely understood = +100
- Well understood = +90
- Moderately understood = +50
- Slightly understood = +30
- Did not understand = -50
- No reward = 0

The rewards have been assumed based on the following assumptions:

- The reader will benefit more if he/she reads the theory first and then solve the exercises
- He/she further benefits if he reads some extra materials
- The learner stands to earn more reward if after understanding and discussion of the result; he/she goes to solve more exercises
- If he/she exists without understanding or discussing the explanation of the results, he/she earns a negative reward
- Exiting after understanding or discussion of the results earns the maximum rewards

The following diagram shows a part of the various state transitions and their respective rewards with respect to the above mentioned assumptions.

Table 4 gives a tabular representation Fig. 5, giving the rewards received on performing a particular action at each stage.

When we solve using the SARSA algorithm, the learning rate α is set to 0.8 (assumption) while the discount factor γ is assumed to be 1. The probability of selecting the best action is assumed to be 30% while the remaining actions are selected uniformly. This sets ε = 0.7.

After the initial episode, the system knows the best action to be considered for earning the maximum reward. In all further interactions of the user with the system, the

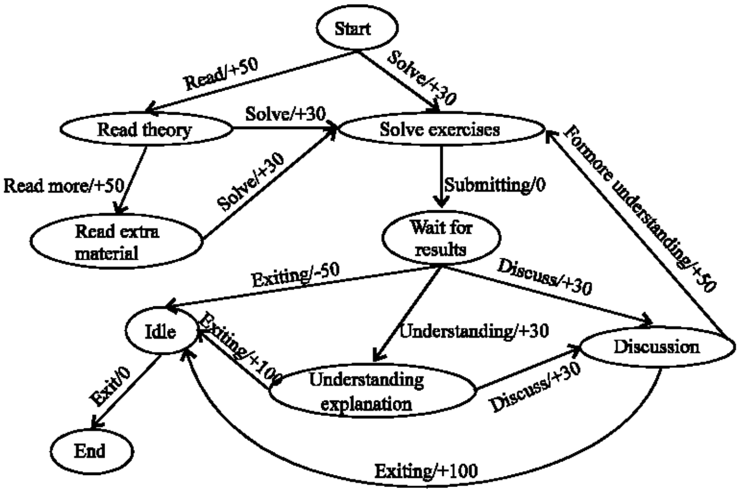

Fig. 5: State-Action-Reward diagram of assumed scenario

Table 4: Reward table of assumed state-action combination

	Action								
State	Read	Read more	Solve	Submit-ting	For more understanding	Exiting	Discuss	Understanding	Exit
Start	+50	-	+30	-	-	-	-	-	-
ReadTheory	-	+50	+30	-	-	-	-	-	-
ReadingExtra Material	-	-	+30	-	-	-	-	-	-
Solving Exercises	-	-	-	0	-	-	-	-	-
WaitFor Result	-	-	-	-	-	-50	+30	+30	-
Discussion	-	-	-	-	+50	+100	-	-	-
Understandingexplanation	-	-	-	-	-	+100	+30	-	-
Idle	-	-	-	-	-	-	-	-	0
End	-	-	-	-	-	-	-	-	-

system chooses the best path from its table and continues execution. However, it is also possible the user chooses to deviate from its previous path and chose some new action. In this case, the table is reworked completely to obtain the new values.

CONCLUSION AND FUTURE WORK

Adaptive learning in an e-learning system can be easily compared to an agent with reinforcement learning approach. SARSA algorithm suits the situation much more than Q-learning algorithm mainly because of its selection policy.

Every user has a unique characteristic called its learning style. It is the manner in which they learn or process information. The difference in selection of an action to shift from one state to another so as to earn the maximum is reward is due to their differences in learning styles. Learning in an e-learning system becomes more efficient if the system can be adaptive and personalized based on the user's learning styles. To provide more precision in the recognition of learning style, a new methodology that incorporates both the approaches by creating an ontological framework for modelling the learner and using fuzzy reasoning engine is proposed. Simple rule-based reasoning can be performed on the ontology to extract the required content. The recognized learning style can be stored within the system in the learner database and can be used for further interactions with the learner so as to provide the learner with his/her relevant content. This makes the learning process similar to reinforcement learning in machines.

REFERENCES

Bergasa-Suso, J., D.A. Sanders and G.E. Tewkesbury, 2005. Intelligent browser-based systems to assist internet users. IEEE Trans. Educ., 48: 580-585.

Brusilovsky, P. and C. Peylo, 2003. Adaptive and intelligent web-based educational systems. Int. J. Art Intell. Educ., 13: 156-172.

Chang, Y.C., W.Y. Kao, C.P. Chu and C.H. Chiu, 2009. A learning style classification mechanism for E-learning. Comput. Educ., 53: 273-285.

Dag, F. and A. Gecer, 2009. Relations between online learning and learning styles. Procedia-Social Behav. Sci., 1: 862-871.

Darwesh, M.G., M.Z. Rashad and A.K. Hamada, 2011. From learning style of web page content to learner's learning style. Int. J. Comp. Sci. Inf. Technol., 3: 195-205.

Deborah, L.J., R. Baskaran and A. Kannan, 2012. Learning styles assessment and theoretical origin in an E-learning scenario: A survey. Artif. Intell. Rev., (In Press).

Dung, P.Q. and A.M. Florea, 2012. An approach for detecting learning styles in learning management systems based on learners behaviors. Proceedings of the International Conference on Education and Management Innovation, February 26-28, 2012, Singapore, pp: 171-177.

Dunn, R. and K. Dunn, 1978. Teaching Students Through their Individual Learning Styles: A Practical Approach. Prentice Hall, Reston, VA., ISBN: 10: 0879098082, Pages: 336.

Eden, T., A. Knittel and R. van Uffelen, 2013. Tutorial on reinforcement learning.http://www.cse.unsw.edu.au/~cs9417ml/RL1/index.html.

Felder, R.M. and L.K. Silverman, 1988. Learning and teaching styles in engineering education. Eng. Educ., 78: 674-681.

Felder, R.M. and J. Spurlin, 2005. Applications, reliability and validity of the index of learning styles. Int. J. Eng. Educ., 21: 103-112.

Felder, R.M. and B.A. Soloman, 2012. Index of learning styles questionnaire. http://www .engr.ncsu.edu/learningstyles/ilsweb.html.

Garcia, P., A. Amandi, S. Schiaffino and M. Campo, 2007. Evaluating Bayesian networks precision for detecting students learning styles. Comput. Educ., 49: 794-808.

Ghorbani, F. and G.A. Montazer, 2011. Learners grouping in E-learning environment using evolutionary fuzzy clustering approach. Int. J. Inf. Commun. Technol., 3: 9-19.

Graf, S., 2007. Adaptivity in learning management systems focussing on learning styles. Ph.D. Thesis, Vienna University of Technology, Vienna, Austria.

Graf, S., Kinshuk and T.C. Liu, 2008. Identifying learning styles in learning management systems by using indications from students behaviour. Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies, July 1-5, 2008, Santander, Cantabria, pp: 482-486.

Honey, P. and A. Mumford, 1992. The Manual of Learning Styles. 3rd Edn., Peter Honey Publications, Maidenhead, UK., ISBN-13: 978-0950844473, Pages: 94.

Jyothi, N., K. Bhan, U. Mothukuri, S. Jain and D. Jain, 2012. A recommender system assisting instructor in building learning path for personalized learning system. Proceedings of the IEEE 4th International Conference on Technology for Education, July 18-20, 2012, Hyderabad, India, pp: 228-230.

Kolb, D.A., 1984. Experiential Learning: Experience as the Source of Learning and Development. Prentice Hall, New Jersey. .

Myers, I.B. and P.B. Myers, 1980. Gifts Differing: Understanding Personality Type. Davies-Black Publishing, Mountain View, CA., USA., ISBN-13: 9780891060741, Pages: 228.

Nath, J., S. Ghosh, S. Agarwal and A. Nath, 2012. E-learning methodologies and its trends in modern information technology. J. Global Res. Comput. Sci., 3: 48-52.

Ozpolat, E. and G.B. Akbar, 2009. Automatic detection of learning styles for an e-learning system. Comput. Educ., 53: 355-367.

Rummery, G.A. and M. Niranjan, 1994. On-line q-learning using connectionist systems. Technical Report CUED/F-INFENG/TR166, Cambridge University. http://mi.eng.cam.ac.uk/reports/svr-ftp/auto-pdf/rummery_tr166.pdf

Saberi, N. and G.A. Montazer, 2012. A new approach for learners modeling in E-learning environment using LMS logs analysis. Proceedings of the 3rd International Conference of E-Learning and E-Teaching, February 14-15, 2012, Tehran, Iran, pp: 25-33.

Sanders, D.A. and J. Bergasa-Suso, 2010. Inferring learning style from the way students interact with a computer user interface and the WWW. IEEE Trans. Educ., 53: 613-620.

Schiaffino, S., P. Garcia and A. Amandi, 2008. eTeacher: Providing personalized assistance to e-learning students. Comput. Educ., 51: 1744-1754.

Simsek, O., N. Atman, M.M. Inceoglu and Y.D. Arikan, 2010. Diagnosis of learning styles based on active/reflective dimension of Felder and Silverman's learning style model in a learning management system. Proceedings of the International Conference on Computational Science and its Applications, Part II, March 23-26, 2010, Fukuoka, Japan, pp: 544-555.

Sutton, R.S. and A.G. Barto, 1998. Reinforcement Learning: An Introduction. 1st Edn., MIT Press, Cambridge, MA., ISBN: 0-262-19398-1.

Van Hasselt, H., 2013. A short introduction to some reinforcement learning algorithms. http://homepages.cwi.nl/~hasselt/rl_algs/Sarsa.html.

Watkins, C.J.C.H., 1989. Learning from delayed rewards. Ph.D. Thesis, Kings College, Cambridge, England.

Watkins, C.J.C.H. and P. Dayan, 1992. Technical note: Q-learning. Mach. Learn., 8: 279-292.