CHAPTER 2

LITERATURE SURVEY

2.1 RELATED WORKS ON LEARNING STYLES

Learning styles are different kinds of methods and characteristics used in learning. The major objective of identifying the learning style is to enhance the performance level of the learners and aid them to find their best position to fit in the outside environments. Especially, in an e-learning environment, the impact of the learning style causes a greater effect on the performance of the learners and in the design of the e-learning systems. Several learning style assessment models and instruments are available online to effectively assess the learning style (Hawk and Shah 2007, Coffield et al 2004). The generic categorization of the learning styles fall under four categories namely Synthesis Analysis, Methodical Study, Fact Retention and Elaborate Processing. Among these, the Synthesis Analysis deals with the processing of information and organizing them into taxonomy. Methodical Study deals with careful study and completion of academic assignments. Fact Retention is the analysis of the correct output instead of understanding the logic behind it. The application of new ideas to existing knowledge is known as Elaborate Processing. The categorization of learning styles is helpful to group the learners and to provide relevant assistance in learning.
2.1.1 Kolb Learning Style Indicator

The learning style proposed by David Kolb (Boyatzis and Kolb 1997) is an indicator based on “Experiential Learning Theory” which considers experience of a learner as an important factor in learning. Therefore, it discussed two kinds of experiences namely grasping and transforming experiences. Grasping consists of two sub categories namely concrete experience and abstract conceptualization. Similarly, the transforming experience has two sub categories termed reflective observation and active experimentation. These experiences have been defined as followed by Cornwell and Manfredo (1994).

1. **Concrete Experience**—This experience deals with qualities of the world that rely on our senses and immersing ourselves in concrete reality of the knowledge.

2. **Abstract Conceptualization**—Thinking about, analyzing, or systematically planning, rather than using sensation as a guide is abstract conceptualization.

3. **Reflective Observation**—Transforming or processing experience some of us tend to carefully watch others who are involved in this experience and it reflects on what happens.

4. **Active Experimentation**—This experience focuses on transforming or processing experience some of us tend to choose and to jump right into applying the experience for doing things without carefully watching.

From these categories, it can be observed that the first three experiences shown in this model namely sensing, planning and watching others must be strengthened and the fourth one must be improved. Moreover,
David Kolb developed a Learning Style Inventory (LSI) in the year 1971, to assess the individual Learning Styles based on Experiential Learning Theory. The individuals were tested using LSI and four types of learners were identified. They are

1. **Diverging**—The Diverging style’s dominant learning abilities are Concrete Experience (CE) and Reflective Observation (RO). People with a Diverging learning style have broad cultural interests and like to gather information.

2. **Assimilating**—The Assimilating style’s dominant learning abilities are Abstract Conceptualization (AC) and Reflective Observation (RO). Individuals with an Assimilating style are less focused on people and more interested in ideas and abstract concepts. The Assimilating learning style is important for effectiveness in information and science careers.

3. **Converging**—The Converging style’s dominant learning abilities are Abstract Conceptualization (AC) and Active Experimentation (AE). Individuals with a Converging learning style prefer to deal with technical tasks and problems rather than with social issues and interpersonal issues.

4. **Accommodating**—The Accommodating style’s dominant learning abilities are Concrete Experience (CE) and Active Experimentation (AE). They enjoy carrying out plans and involving themselves in new and challenging experiences. Their tendency may be to act on “gut” feelings rather than on logical analysis. This learning style is important for effectiveness in action-oriented careers such as marketing or sales.
This categorization helps to develop learning materials to the learners based on their experience and learning styles.

### 2.1.2 Honey and Mumford’s Learning Styles Questionnaire

This model deals about the general behavioral tendencies. The Learning Style questionnaire of this model is derived from David Kolb’s LSI. This model probes the learners to indicate their general behavior tendencies rather than directly asking their behavior through questionnaires (Honey and Mumford 2000). Their reasoning is most people have never consciously considered how they really learn. According to this model, learners are classified into four types namely reflectors, theorists, pragmatist and activist who are defined as follows.

1. **Reflectors**—Prefer to learn from activities that allow them to watch, think, and review (time to think things over) what has happened.

2. **Theorists**—Prefer to think problems through in a step-by-step manner. Likes lectures, analogies, systems, case studies, models and readings.

3. **Pragmatist**—Prefer to apply new learning to actual practice to see if they work.

4. **Activist**—Prefer the challenges of new experiences, involvement with others, assimilation and role-playing.

The main difference between the basic Kolb model and this model is that the learners general behaviors are analyzed in this model whereas they are obtained from the questionnaires in the Kolb’s model. The trust of this
model has been proved to be better due to the fact that the analysis was based on the learners’ behavior.

2.1.3 **Gregorc Style Delineator**

This model is based on cognitive thinking aspects of an individual rather than experiences considered in Kolb model and general behavior tendencies discussed in Honey and Mumford model. In this cognitive model, the existence of perceptions leads to the notions of different kinds of learning styles (Gregorc and Ward 1977). Therefore, this model describes two kinds of perceptual qualities namely Concrete and Abstract and two kinds of ordering abilities namely Random and Sequential. In addition, the individuals also possess different kinds of perceptual and ordering abilities like

1. Concrete Sequential—Learning with hands-on experiences.
3. Abstract Sequential—use logic to grasp situations.
4. Concrete Random—prefers trial and error approach.

This type of cognitive behavior analysis can be used to develop intelligent e-learning systems.

2.1.4 **Flemming VAK Model**

The most common and widely-used categorizations of the various types of learning styles is Fleming’s VAK model (Fleming 2001). The VAK learning styles model suggests that most people can be divided into one of three preferred styles of learning namely Visual, Auditory and Kinaesthetic. People with a Visual learning style have a preference for seen or observed things, including pictures, diagrams, demonstrations, displays, handouts, films
and flipchart. These people prefer to work from lists and written directions and instructions. People with an Auditory learning style have a preference for the transfer of information through listening: to the spoken word, of self or others, of sounds and noises. These are the people who are happy being given spoken instructions over the telephone, and can remember all the words to songs that they hear. People with a kinaesthetic learning style have a preference for physical experience touching, feeling, holding, doing, and practical hands-on experiences. These are the people who like to experiment, hands-on, and never look at the instructions first. This model is useful for categorizing the learners based on their interest in learning through visual, audio and hands-on experiences. These categories are helpful in developing efficient machine learning techniques for effective e-learning

### 2.1.5 Dunn and Dunn Productivity Environmental Preference Survey

This model defines learning style as the way in which individual learners begin to concentrate on, process, and retain new and difficult material (Dunn and Dunn 1989). It is a combination of many biological and experiential characteristics that work on their own or together as a unit to contribute to learning. This interaction with new information is unique for each individual. The model is discussed based on five variables that affect the learners’ ability to learn. The learners are generally affected by their immediate environment, own emotionality, sociological preferences, physiological characteristics, and processing inclinations. These metrics are helpful to understand the learners behavioral changes with respect to place and time. This model considers twenty-two factors representing environmental, sociological, emotional, and physical factors. Since the learning style factors identified by Dunn and Dunn are conditions external to the learner, they have a greater effect on external instructional conditions
rather than the learner’s internal learning strategies (Jonassen and Grabowski 1993). According to this model, the following learning style preferences need to be considered when instruction is created and implemented:

1. Environmental—Noise level, lighting, temperature, and furniture/seating design.

2. Emotionality—Motivation, responsibility, persistence, and need for structure.

3. Sociological—Learning groups, presence of authority figures, varied working patterns, and adult motivation (LSI only).

4. Physiological—Perceptual strengths, time-of-day energy levels, intake, and mobility.

5. Processing inclinations - Global/analytic, right/left, and impulsive/reflective (Dunn and Dunn 1989).

The main advantage of this model over other models is that it considers the environmental conditions. An intelligent agent can help the learner by sensing the environmental conditions effectively.

2.1.6 Chris Jackson Model

The hybrid model of learning in personality is a theory based model of personality which provides a way of understanding the processes which underlie functional learning that lead to successful work performance, and dysfunctional learning which leads to antisocial behavior in the workplace (Jackson 2002). The process model of the hybrid model of personality contrasts with the Big Five model of personality which is primarily based on exploratory factor analysis and which aims to provide a parsimonious social construction of personality. This model incorporates a new type of learners
called Deep Learning Achiever which takes inspiration from the experiential model of learning. The hybrid model considers four types of personalities (Boyatzis and Kolb 1984).

1. Sensation Seeker—The learners with high approach and low avoidance measuring exploration and curiosity.

2. Goal Oriented Achiever—This type of learner has mastery on long term and hard outcomes

3. Conscientious Achiever—Perseverance, responsibility and understanding about the complex social world are the major features of this type of learners.

4. Emotionally Intelligent Achiever—The learners who provide rational and logical

5. Deep Learning Achiever - This type of learners provides well thought out and well constructed outcomes.

This psychological modeling helps to understand the learners behavior which in turn can be used to develop a suitable intelligent agent for providing tutoring in e-learning (Robinet et al 2007).

2.1.7 Carl Jung and Myers Briggs Type Indicator

Myers–Briggs Type Indicator is the theory of psychological type as originally developed by Carl Jung. Myers–Briggs Type Indicator (MBTI) assessment is a psychometric questionnaire designed to measure psychological preferences in how people perceive the world and make decisions (Pittenger 2005). The learning style assessment in this kind of indicator is resolved using different aspects inclusive of psychological,
decision-making, information gathering and actions. The models of learning styles are as follows

1. Judging versus Perceiving—Attention towards the external world/things or internal world/things.

2. Thinking versus Feeling—Perceive world directly or perceive through impressions/imaging possibilities.

3. Sensing versus Intuition—Learners taking decisions through logic or through mere human values.

4. Extroversion versus Introversion—Learner viewing the world as a structured, planned environment or as a spontaneous environment.

The major application of this indicator was using among the companies in order to enhance the inter-personal relations among the employees by obtaining their psychological behavior.

2.1.8 Howard Gardeners Multiple Intelligence

This is based on multiple intelligences and was developed in 1983 at Harvard University. The intelligence of an individual was tested using I.Q. testing, which is far limited. A broader range of human potential was found in eight different notions of intelligence (Gardner and Moran 2006). Gardener’s theory has emerged from recent cognitive research and hence a learner can learn, remember and perform well in different ways (Gardner 2004). Different kinds of intelligences were observed as different styles of learning which are as follows

1. Linguistic intelligence—This category of learners have highly developed auditory skills and often think in terms of words
and can be taught effectively by motivating them to read lots of articles, books and papers.

2. Logical–Mathematical intelligence—This characteristic involves high interest in reasoning and calculating and can be taught through logic games, investigations, unrevealing mysteries and problem solving skill.

3. Visual–Spatial intelligence—A person with this type of intelligence is well aware of the environments and can be taught through drawings, verbal and physical imagery.

4. Bodily-Kinesthetic intelligence—This type of intelligent learners are keen about body awareness and can be taught through physical activities, hands-on experiences and role playing.

5. Musical intelligence—This category of learners is highly sensitive to music and rhythm and can be taught by turning lessons into lyrics and speaking rhythmically.

6. Interpersonal intelligence—This deals with learners who have interaction with others and can be taught by conducting group discussions and collaborated learning environments.

7. Intrapersonal intelligence—The learners who shy away from others and do not interact with others fall under this group. They can be taught through independent study and introspection.

8. Naturalist intelligence—This type of learners relate their understanding to one’s natural surroundings and can be taught by asking them to apply their knowledge to environmental related applications like farming and gardening.
The theory of multiple intelligences provides a broader thinking approach to different styles of learning as compared to traditional levels of learning styles. Gardener challenges that everyone can learn the same material and show their efficiency when they are taught according to their interest as indicated above. This kind of teaching will effectively suit the broad spectrum of students with varying capabilities and interests.

2.1.9 Felder–Silverman Index of Learning Styles

This learning style model (Felder and Silverman 1988) often used in technology-enhanced learning and that is designed for traditional learning. Moreover, Felder–Silverman learning style model describes the learning style of a learner in more detail, distinguishing between preferences on four dimensions. It was used by many researchers since this model provides four dimensions of learning based on psychological aspects of the learners which is found to be important in an e-learning environment. The learning styles proposed by Felder–Silverman for categorizing the learners are:

1. Active–Reflective—Active learners learn best by working actively with the learning material, by applying the material, and by trying things out. Reflective learners prefer to think about and reflect on the material.

2. Sensing–Intuitive—Learners who prefer a sensing learning style like to learn facts and concrete learning material. Intuitive learners prefer to learn abstract learning material, such as theories and their underlying meanings.

3. Visual–Verbal—Learners who remember best and therefore prefer to learn from what they have seen (e.g., pictures, diagrams and flow-charts), and learners who get more out of
textual representations, regardless of whether they are written or spoken.

4. Sequential–Global—Sequential learners learn in small incremental steps and therefore have a linear learning progress. Global learners use a holistic thinking process and learn in large leaps. They tend to absorb learning material almost randomly without seeing connections but after they have learned enough material they suddenly get the whole picture.

Since most learners fall in the category of either active or reflective for the first dimension, this model is more suitable to evaluating the learners in an e-learning environment.

2.1.10 Browser-based Learning Styles Assessment

The iLessons system developed by Bergasa-Suso et al (2005) overcame the limitations of the previous e-learning systems by providing two additional features namely Drag and Drop facility for reuse and an intelligent recommendation system that recommends relevant web pages to the learners based on their learning styles. Therefore, iLessons is an important contribution for e-learning through the identification of learners as active or reflective. The work done by the authors promptly followed an algorithm in identifying the first dimension and classified the users into two kinds as either Active or Reflective. From his works it is understood that the learners belonged to either of the dimensions crisply. It is evident from their experiments that the classification accuracy for identifying such learners was 71% when considering the first dimension of Felder-Silverman learning style model.
2.1.11 Intelligent Browser-based Learning Style Assessment

Sanders and Bergasa-Suso (2010) proposed a new intelligent e-learning system with an effective user interface. The main advantage of this intelligent e-learning system is that it is capable of performing deductive inference to obtain the learning style of learners. They followed Felder-Silverman learning style model and developed an e-learning system that can infer the learning styles in real time and provided recommendations for selecting suitable e-learning contents. This is a significant achievement in the area of e-learning since, it uses Artificial Intelligent techniques for classifying the learners based on their learning style into three categories namely active or reflective or unknown. He categorized in such a way, since the users may not fall into one category virtually all the time and most learners tend towards a particular dimension in course of time, environmental factor, mood, need and psychological changes. The accuracy of classifying the learners Sander’s et al has increased to 81% compared to the earlier work done by Bergasa-Suso. One of the important contributions of Sanders and Begrasa-Suso is the inclusion of unknown sets in addition to the active and reflective categories of learners.

2.1.12 Proposed Model

Application of these existing learning models to the modern web based e-learning leads to a lot of new challenges. First, all these models assume that the teacher and learner meet frequently in the learning process so that the learning style of the learner will be known to the teacher easily. On the other hand, in e-learning the learner-instructor interactions are not frequent. Moreover, most applications in the real world have incomplete and vague information. Therefore, in this research work self learning is supported through learning style identification and provision of suitable learning materials. Therefore, this work uses fuzzy rules for making decisions since it
helps to make effective conclusions even from incomplete data understanding the learner characteristics. Finally, the existing models consider psychology as an important factor. In order to understand the learner’s psychology, this system identifies behaviours from web interface information. This helps to identify the learning styles easily and to provide suitable learning materials in the e-learning environment.

2.2 WORK ON AGENTS IN E-LEARNING

The traditional offline educational system is neither intelligent nor adaptive in nature (Beck et al 2004). In such systems, the instructional materials are delivered to the students through dialogues and interactions. The next era of educational system is intelligent tutoring systems. In these systems, the face-to-face interactions are replaced by e-mails, chats and online discussion forums. Several Artificial Intelligence techniques are used in such systems to provide personalized teaching and learning process (Chen et al 2004).

E-learning agents can be used to provide support to educate instructors and support learning object reuse by helping them locate existing e-learning content. In the dynamic world of the internet, the learning materials available on a particular topic are constantly changing, and hence there is a significant need to continuously monitor existing materials and search for new ones so that the most appropriate course/training materials can be selected. One of the benefits provided by an agent based e-learning system is it can continuously retrieve the most up-to-date educational materials available when creating customized lesson plans for learners. Another advantage of an agent based e-learning system is that it can assist instructors in monitoring learner progress and facilitate interactions between the instructor and learners that are struggling with a particular topic. They can also be used to optimally place learners in groups formed to solve specific problems.
Introducing agents into the e-learning environment will fundamentally change the way online education is conducted and the outcomes for both learners and instruction designers. As online learners and instructors increase their use of intelligent agents to automate information gathering, lesson planning, learning material customization and collaboration, the outcome for both learners and instructors will be improved. With sufficient information, agents should be able to select the most appropriate learning materials for individual learners based on both topics covered and learner characteristics, thus improving learning outcomes. Agents can also monitor learning effectiveness so the benefits of the e-learning program can be assessed by the organization. An agent-based learning system can be used in tutoring systems to provide assistance and to monitor the performance of those students (Beck et al. 2004).

Animated pedagogical agents interact with students in a manner that closely resembles face-to-face collaborative learning (Pearson and Graesser 2006). Successful collaborative learning approaches have five critical attributes namely a common task, small group learning, cooperative behavior, interdependence and individual accountability (Han et al. 2010). Prior researches have shown that pair programming when used as a collaborative learning positively affects students knowledge retention and knowledge transfer (Palmieri 2002). In such an environment, knowledge transfer is achieved by switching roles between a student and an agent. Interactive teaching and learning by a student and an agent achieves good performance in learning a programming course. Among the several merits of pair programming, there are some demerits listed in the study. One of the main drawbacks is that when students try to master a particular programming course from the scratch, the effect of self-efficacy of the students plays a major role. In such a scenario, it is very mandate to increase the self-efficacy of the students and this is achieved by making them to self learn using
suitable e-learning contents in e-learning servers based on their learning styles.

The INTELLITUTOR system attempts to teach novice Pascal and ‘C’ programmers as a human programming tutor (Ueno 2000). The system is capable of detecting logical errors and provides the types and location of bugs in the form of feedbacks. However, the system is not capable of understanding the students’ self-efficacy which helps in determining their success and failure rate.

RAPTIS is an intelligent programming tutor system that teaches Pascal looping concepts (Woods and Warren 1995). This system is adaptive in nature where a variety of teaching strategies are employed and the strategies could be changed according to the students’ history. However, this system does not aim to increase the self-efficacy of the students. Increasing the self-efficacy might be a permanent solution for an effective teaching-learning methodology.

Mahkameh Yaghmaie and Ardeshir Bahreininejad (2011) suggest a framework for building an adaptive Learning Management System (LMS). This framework is based upon multi-agent systems and uses both Sharable Content Object Reference Model (SCORM) and semantic Web ontology for learning content storage, sequencing and adaptation. This system has been implemented upon a well known open-source LMS and its functionalities are demonstrated through the simulation of a scenario mimicking the real life conditions. The result reveals the system effectiveness for which it appears that the proposed approach may be very promising.

Dawn Gregg and Steven Walczak (2007) described a set of interacting e-learning agents that have the capability of assisting instructors with online course design, course scheduling, and learning material location.
E-learning agents can also be used to personalize instruction based on learner’s prior knowledge (e.g. from knowledge surveys), learning style, and accessibility needs. These agents have the capacity to select and customize resources, problems, and hints. Finally, agents can be used to foster effective collaboration in the e-learning environment.

Pedagogical agents can be designed for personalization and adaptiveness. Pearson et al developed pedagogical agents that facilitate students learning by supporting collaborative, interactive and investigative learning in web-based learning technologies (Pearson and Graesser 2006). The agents in these systems merely guide a learner through a lesson using computer-assisted interactions. However, these agents have responsibilities in neither determining their self-efficacy nor increasing it.

Agents developed by Devedzic facilitate students’ motivation and provides adaptive feedbacks in e-learning environments (Devedzic 2004). They are designed in such a way to assist the learning process in different domains. But, these agents could not increase the students’ self-efficacy in terms of identifying their learning styles.

Three types of agents namely pedagogical, peer-learning and demonstrating agents helps to assist the human learners which was developed by Sklar and Richards (2006). Pedagogical agents interact with a learner and monitors the students performance indirectly which helps to understand the student and provides feedbacks. Peer-Learning agents acts like peers and are interactive partners but are less engineered than pedagogical agents. Demonstrating agents are agent-based simulations or educational robotics and act like interactive mediums for learning. However, this system could be enhanced by introducing new kinds of agents which could identify the students’
self-efficacy through the prediction of learning styles. Moreover, it could be increased by providing suitable e-learning contents as a pre-requisite for learning with peer-learning agents. The peer-learning agents developed by Han et al (2007) provide more positive impact on students’ achievements and their self-efficacy. The determined self-efficacy is not found to be increased by offering suitable e-learning contents for pre-tutoring to the students based on their learning styles.

In such a scenario, the psychological level of the learners is not well balanced since the learning styles of the learners vary from one individual to another. The success of the e-learning system may degrade if similar kind of e-contents is provided to all the students. The students can learn the basics of programming courses from the e-contents based on their choices namely documents, audio and video. This kind of e-contents provision can increase their self-efficacy and the students can now learn with the help of peer-learning agents to master the programming course.

From the discussions made on the intelligent tutoring systems, it is identified that increasing the self-efficacy can enhance the performance of the students in e-learning of programming courses. Therefore, self-efficacy of the learners are improved in this research work by allowing them to self learn before learning the ‘C’ programming language using agents. Moreover, identifying the learning styles of the learners can help in facilitating them with suitable recommendations on e-contents that are available in e-learning servers. For this purpose, Felder Silverman learning style model is used in this work for identifying the learning styles of the students (Felder and Silverman 1988). After identifying their learning styles, they are recommended with suitable e-contents based on their choice of learning. This helps them to self learn the basic concepts of the target course. This facility is provided in this
work since self learning the basic concepts of the target course helps in better collaborative learning with peer-learning agents which help in mastering the target course.

2.3 RELATED WORKS ON PAIR PROGRAMMING STRATEGY

White and Sivitanides (2002) consolidated prior research and accepted cognitive theory. It then presents a formulation of a theory that relates cognitive requirements of different computer programming languages and programmers’ cognitive characteristics. If the cognitive requirements for a programming language are beyond the cognitive characteristics of a programming student, the student may burn out. If the cognitive requirements are below the student’s cognitive characteristics the student may be bored. If they are similar to them, the student is able to meet the challenges. Motivation, interest, self-esteem and success may thus be optimized. Different programming languages are more suited for different cognitive characteristics. This theory extends prior research in cognitive theory and cognitive requirements of computer programming.

Van and Grissom (2001) explores an alternative pedagogical approach that emphasizes constructive and collaborative learning in virtual classrooms. After briefly discussing constructivism and providing examples of constructivist techniques in classrooms, empirical research results are provided. These results arise from a study that compares different classroom sections that utilized the techniques at varying frequencies. A positive correlation was found between frequency and mean final exam scores. However, no pair-wise differences between sections were statistically significant. These outcomes and others are discussed in addition to future research design implications.
Han et al (2010) developed a model for determining students' programming abilities. In addition, the roles of the tutor and tutee are like the roles of a navigator and driver in pair programming. The developed agent system is demonstrated to have positive effects on knowledge retention and transfer in a programming course, with a greater influence on transfer than on retention. This model combining peer-learning agents with a teaching and learning strategy is more effective in helping learners to acquire programming skills.

Ueno (2000) proposes the concepts and methodology of knowledge-based program understander which deals with plural programming languages, namely Pascal and ‘C’, by means of generalized syntax and knowledge representations based on it. A sub-system of intelligent programming environment INTELLITUTOR, which is available in the internet was developed. In addition to programming education, this subsystem demonstrates interesting features in acquisition and maintenance of knowledge base in a feasibility study.

Lui et al (2008) reported on two experiments that addressed the limitations of previous experiments, where program design was not disentangled from program coding. Because programmer skills, in particular computer languages and development environments, can vary, it is important to make sure that one is measuring the effects of pairs in problem solving and algorithm-related tasks and not their knowledge of computer languages. Aptitude tests can predict performance without relying on knowledge of specific language commands. Aptitude tests in problem solving and algorithm design were used in two experiments to test the effect of pairs in these tasks. In both experiments, the pairs outperformed individuals. In Experiment 1, on average, the pairs required fewer resubmissions and completed the tasks in less than half the time as compared to the individuals. In Experiment 2, where
subjects were required to complete a substantially more complex task, the pairs outperformed the individuals at an even higher level.

Norsaremah Salleh et al (2011) described an SLR targeted at empirical studies of programming effectiveness and/or pair compatibility conducted in higher education settings. A total of 73 primary studies were used in our SLR, from which 14 compatibility factors potentially affecting programming effectiveness were identified. Of these, personality type, actual, and perceived skill level were the three factors investigated the most in PP studies. However, the effects of personality type toward pair compatibility and/or programming effectiveness were inconclusive. The studies that investigated actual and perceived skill levels achieved a consensus suggesting that students prefer to pair with someone of similar skills to themselves. However, in the second meta-analysis, the pooled results suggested that it was effective in helping students obtaining better scores in their assignments (effect size = 0.67). There were numerous methods employed when considering quality as a measure of programming effectiveness. They also discussed a number of implications of the SLR results for research and practice, including the need to replicate the studies in areas where findings were inconsistent, or to conduct studies in areas where there is scarcity of or no evidence regarding the effect of certain compatibility factors toward programming effectiveness as a pedagogical tool.

Jo Hannay et al (2010) reports on a study of the impact of the Big Five personality traits on the performance of pair programmers together with the impact of expertise and task complexity. The study involved 196 software professionals in three countries forming 98 pairs. The analysis consisted of a confirmatory part and an exploratory part. The results show that, their data do not confirm a meta-analysis-based model of the impact of certain personality traits on performance and personality traits, in general, have modest
predictive value on pair programming performance compared with expertise, task complexity, and country. They conclude that more effort should be spent on investigating other performance-related predictors such as expertise, and task complexity, as well as other promising predictors, such as programming skill and learning. They also conclude that effort should be spent on elaborating on the effects of personality on various measures of collaboration, which, in turn, may be used to predict and influence performance. Insights into such malleable, rather than static, factors may then be used to improve pair programming performance.

Thomas et al (2002) reported on the implication of different preferred learning styles on students’ performance in the introductory programming sequence and on work in progress on how to accommodate these different styles. Students were given a learning styles preference test and then their preferred learning styles were compared to their performance on the exam and the practical programming part of the introductory programming module. There were significant differences in performance between groups of students.

De Azevedo and Scalabrin (2005) introduced the design and implementation of a multiagent system based on a Collaborative Online Learning Environment (COLE). The purpose of developing such an environment is to improve social competences along with traditional content-related ones in lifelong learning. As educators would be unable to handle the huge amount of data concerning human interactions in such a learning environment, a multiagent system approach is adopted. The concept of human collaboration and the ways that Project-Based Learning (PBL) and portfolios can be used to improve social competences are discussed based on the Social Theory of Learning. The way that the System Analysis for Agent Systems (SAAS) method was used to identify services and agents is
presented. A general review of multiagent system architectures is presented to justify the choice of an open system. The basis and architecture of the COLE are explained. In order to facilitate the implementation of particular agents, a Generic Agent (GAg) and its functionalities are presented.

Crespo et al (2005) presented an effective matching algorithm in the context of peer reviewing applied to an educational setting. The problem is formulated as an optimization problem to search a solution that satisfies a set of given criteria modeled as “profiles”. These profiles represent regions of the solution space to be either favoured or avoided when searching for a solution. The proposed technique was deployed in a first semester computer engineering course and proved to be both effective and well received by the students.

In this work, pair programming using agents is proposed for providing effective collaborative learning.

2.4 RELATED WORKS ON ONTOLOGY CONSTRUCTION TECHNIQUES

Ontology construction is a knowledge representation technique that helps to organize knowledge effectively. It is useful for performing semantic analysis (Aldo Gangemi 2003).

2.4.1 Ontology Construction Techniques

Fok and Shing Ip (2004) addressed the issues and methodologies in the design and construction of education ontology and discuss the necessities of such an education ontology that can help retrieving, organizing, and recommending educational resources for personalized learning (Khribi et al 2008). They suggested the use of a systematic ontology construction
approach, the design and implementation of a Personalized Education Ontology (PEOnto) to demonstrate the flexibilities of ontology usages in performing different education tasks as well as enhancing system extensibilities and exchangeabilities.

The model provided a detailed examination of the opportunities and necessities of Personalized Education (PE) from the perspective of different learning pedagogies. To optimize the benefits of meaningful personalization technologies, they also propose a Personalized Education System (PES) Framework and introduce several PES features that can support personalized teaching and learning under pervasive computing.

Gilles et al (1999) introduced and demonstrated a feasible implementation of the Personalized Instruction Planner (PIP) to facilitate teachers to collaborate in using varied teaching instruction techniques to design lessons or thematic units. The beauty of using an ontology-driven approach in PIP is sets of theoretical vocabulary items can be expressively described with closely related educational goals and strategies. Static metadata of Learning Objects (LO) can semantically transmit to offer meaningful personalization services in education. PIP allows sharable, reusable, and extensible education resources, instructional presentation methods, and activities and evaluation projects to be searched and indexed for Personalized Education (PE). Collaboration can benefit both teachers and students. The construction of PEOnto (an education ontology for PE), that underpins the development of the Personalized Education System (PES) and its subsystem Personalized Instructional Planner (PIP), serves to help realize our education and research goal – personalized learning for whole-person development (Arpírez et al 1998).
According to Dahab et al (2006) most of existing ontologies construction tools support construction of ontological relations (e.g., taxonomy, equivalence, etc.) but they do not support construction of domain relations, non-taxonomic conceptual relationships (e.g., causes, caused by, treat, treated by, has-member, contain, material-of, operated-by, controls, etc.). Moreover, domain relations are found mainly in text sources. TextOntoEx constructs ontology from natural domain text using semantic pattern-based approach. TextOntoEx is a chain between linguistic analysis and ontology engineering. TextOntoEx analyses natural domain text to extract candidate relations and then maps them into meaning representation to facilitate constructing ontology. The paper explains this approach in more details and discusses some experiments on deriving ontology from natural text.

Guangzuo et al (2004) proposed a kind of flexible educational platform architecture for e-learning, which is called OntoEdu. It is composed of five parts: user adaptation, automatic composition, education ontology, service module and content module, among which educational ontology is the core. With OntoEdu architecture, it is promising to gain concept reusability, device and user adaptability, automatic composition, function and performance scalability. In the meantime, the grid-based system design of OntoEdu is also proposed. Experiments indicate that OntoEdu architecture is viable and flexible. At last, the author discusses some problems of OntoEdu

2.4.2 Works on Anaphora Resolution for Ontology Construction

Hobbs' algorithm (Markert Katja and Malvina Nissim 2005) relies on a simple tree search procedure formulated in terms of depth of embedding and left-right order. The tree procedure selects and replaces the pronouns by
selecting the first candidate encountered by a left right depth first search for
the tree. The algorithm chooses as the antecedent of a pronoun P the first NP_i
(Noun Phrase) in the tree obtained by left-to-right breadth-first traversal of the
branches to the left of the path T. If an antecedent satisfying this condition is
not found in the sentence containing P, the algorithm selects the first NP
obtained by a left-to-right breadth first search of the surface structures of
preceding sentences in the text. The algorithm is found to produce a success
rate close to 80% for intra-sentential anaphora resolution (Stuckardt and
Roland 1997).

Shalom Lappin and Herbert Leass (1994) report an algorithm for
identifying the noun phrase antecedents of third person pronouns and lexical
anaphors. The algorithm RAP (Resolution of Anaphora Procedure) applies to
the syntactic representations generated by RAP algorithm concentrates more
on resolving an intra-sentential syntactic filter for ruling out anaphoric
dependence of a pronoun on an NP on syntactic grounds. It employs an
anaphor binding algorithm for identifying the possible antecedent binder of a
lexical anaphor within the same sentence. The algorithm does not employ
semantic conditions or real-world knowledge in choosing among the
candidates. This algorithm is suited for intra-sentential anaphora resolution,
which will not be the case in most of the text corpus available in the WWW.
RAP is also not suited in identifying the exact antecedents and replaces of
such antecedents when the noun phrase is not a single but a compound noun
phrase. The major limitation of the algorithm is that the performance in terms
of resolving the entire set of anaphor is found to be very limited when the
input corpus consists of a number of compound noun phrases, even though
the algorithm employs a decision procedure for selecting the preferred
element of a list of antecedent candidates for a pronoun.
Aone et al (1995) described an approach to building an automatically trainable anaphora resolution system. The authors made use of a machine learning algorithm and used many training examples for anaphora resolution. This machine learning algorithm made use of a decision tree consisting of feature vectors for pairs of an anaphora and its possible antecedent. The feature vectors for the training samples include lexical, semantic, syntactic and positional features. The authors built six machine learning based anaphora resolvers and achieved about a precision close to 80%. However, the algorithm failed in cases when the machine learning algorithm has to resolve the anaphors between different sentences. The algorithm drastically showed lower performance when the inter-sentential anaphora resolution was performed.

Denis and Baldridge (2007) proposed a supervised maximum entropy ranking approach to pronoun resolution as an alternative to commonly used classification-based approaches. Classification approaches consider only one or two candidate antecedents for a pronoun at a time, whereas ranking allows all candidates to be evaluated together. In particular, the ranker obtains an error reduction of 9.7% over the best classification approach, the twin-candidate model. Furthermore, they show that the ranker offers some computational advantage over the twin candidate classifier, since it easily allows the inclusion of more candidate antecedents during training. This approach leads to a further error reduction of 5.4% (a total reduction of 14.6% over the twin candidate model).

Therefore, in this research work an enhanced ontology construction technique is proposed that focus on the existing anaphors. The proposed technique uses propositional logic based ontology construction that helps in building heavy weight ontologies for increased expressivity. Moreover, the ontology construction technique used in this research work also aims in
resolving inter-sentential anaphors that aids in increased accuracy in the construction of ontology.

2.5 RELATED WORKS ON ONTOLOGY ALIGNMENT TECHNIQUES

This section gives an overview of the existing work related to the notion of Semantic Analysis. Various kinds of techniques have been proposed in the past to solve the problem of Semantic analysis using ontology alignment techniques. Some of the well known systems for performing ontology alignment are Lily (Wang Minhong et al 2009), ASMOV (Jean and Kabuka 2007), RiMOM (Tang et al 2006), S-Match (Giunchiglia Fausto and Zaihrayeu Ilya 2009), Content-based techniques (Partyka et al 2008) and MUPRET (Assawamekin et al 2009). Most of these works used S-Match, Content-based techniques and MUPRET framework for evaluating their contributions because they are considered as the standard techniques. The work proposed in this paper is an extension to the work proposed by Assawamekin et al (2009) who developed MUPRET framework for ontology alignment and this proposed framework has been evaluated and tested for an e-learning system.

Doan et al (2003) proposed a new system and is regarded as the first ontology alignment framework proposed in the literatures. It is a system based on taxonomical structure and uses instance-based methodology for ontology alignment. Moreover, this system exploits the concepts of statistical computation analysis and then uses the taxonomical structure of applications for decision making. The system that was developed was more suitable for applications like computer vision, natural language processing and hypertext classification which adopts relaxation labeling approach. Lily (Wang and Xu 2007) framework is a generic ontology alignment framework based on extraction of semantic sub graphs. The system
uses the concepts of both linguistic and structural information in semantic subgraphs to iterate initial alignments. Moreover, it combines all separate similarities iteratively using some weights assignments and hence it fails to consider ontology constraints directly. Later on, Tang et al (2006), proposed the Ontology Alignment technique called RiMOM which is based on Bayesian Networks decision theory. They used many kinds of taxonomical relationships like superClassOf, siblingClassOf, domain, etc are resolved using this system. All these systems are suitable only for situations where taxonomical classification is possible (Antoniol et al 2002).

ASMOV (Jean and Kabuka 2007) is an ontology alignment tool which calculates the similarity between concepts recursively by analyzing textual description. The tool effectively identifies the semantic inconsistencies like crisscross mappings and many-to-one mappings. However, ontology alignment must focus on both matching for equality and also detection of inconsistencies.

S-Match (Giunchiglia et al 2010) is an open source semantic matching framework that tackles the semantic interoperability problem by transforming several data structures including web services descriptions into lightweight ontologies and establishes the semantic correspondences between them. The major limitation of the framework is that S-Match could identify only three kinds of XML relationships namely “is-a”, “part-of” and “attribute-of”, whereas there are numerous relationships that are existing between the ontology for effective matching of them semantically. Therefore, it is necessary to extend the light-weight ontologies with axioms in order to provide heavy-weight ontologies.

Content-based techniques (Partyka et al 2008) proposed by Jeffrey Partyka et al (2008) for ontology matching deal with examining the associated instance data from the compared concepts and apply a content-matching
strategy to measure similarity based on value types. The major limitation of the content-based framework is that, it uses only syntactic structure of the two types of ontology for analysis. Therefore, the system could not identify the major relationships like partial overlap, intersection and subsumption. This can be enhanced by introducing semantic axioms for enhancing the efficiency.

MUPRET (Assawamekin et al 2009) framework is another ontology alignment technique used in web applications. This framework actually uses a semantic matching procedure based on dominant words matching. The main advantage of MUPRET framework over the existing systems is that it considers relationships such as equivalence, subsumption, overlapping and mismatch. However, it is necessary to find inconsistencies in order to apply the resolution procedure.

From the analysis made on the related work, it has been observed that most of the existing works and techniques used for Semantic Ontology Alignment were performed either by using light-weight ontologies or with axioms of propositional logic with limited expressivity. In such a scenario, the performance evaluation of the students in e-learning might not produce accurate results. On the other hand, accurate results can be obtained only when all kinds of relationships like equivalence, subsumption, overlap, partial overlap and inconsistencies are resolved during ontology alignment. Hence, effective techniques have been proposed to identify and resolve all these kinds of relationships. The work proposed in this thesis derives deontic relationships with increased expressivity through the introduction of axioms from deontic logic which helps in identifying all the above kinds of relationships and thus it increases the accuracy in evaluating the performance of the students in e-learning.
2.6 COMPARISON OF RELATED WORKS AND THE PROPOSED WORK

When compared with all the work present in the literature, the work proposed in this thesis is different and novel in many ways. First, all the existing works make decisions using complete knowledge. On the other hand, this work handles both complete and uncertain information about the learners for learning styles assessment by using fuzzy logic. Second, most of the systems were based on offline questionnaires only. This proposed system uses online web activities of users for learning style prediction in real time. Third, most of the hypermedia applications are based on Felder Silverman learning style model where the learners are categorized as active or reflective types only. However in the fuzzy logic based approach proposed in this thesis, learners are categorized into four categories namely active, medium active, medium reflective and reflective. Therefore, it provides an effective categorization of learners than the existing systems. Fourth, in the existing software agents systems for e-learning using pair programming strategies, self-efficacy of learners was not considered. However, it is an important criterion in e-learning for the success of the learners learning through web environments. In this research work, self-efficacy is considered in pair programming. Fifth, in ontology construction anaphora resolution was not focussed in the earlier systems. However, in this system anaphors are resolved effectively so that multiple sentences can be easily resolved during ontology construction. Finally, in ontology alignment only light weight ontologies and taxonomical structures were considered in the existing systems. On the other hand in the proposed work, heavy weight ontologies are constructed using deontic logic so that it is possible to identify both dominant and non-dominant words present in the text documents provided by the learners and the domain experts.