Application of Semantic Web Concepts to Intelligent Tutoring Systems

Aparna Lalingkar, and Chandrashekar Ramnathan

International Institute of Information Technology, Bangalore, India E-mail: {aparna.l, rc}@iiitb.ac.in

Abstract. The application of concepts of Semantic Web to Intelligent Tutoring directs towards Adaptive Educational Hypermedia Systems (AEHS). AEHS contains a document space (i.e., domain ontology), a user model (i.e., a student model ontology), a set of observations (i.e., the logic through which students’ interaction with the system can be observed), and the adaptation component (i.e., suggested hyperlinks or even feedback to the student about what the student knows and still needs to learn). Reusability and interoperability offered through resource description framework (RDF) and ontology increase the scalability of the potential use of Intelligent Tutoring Systems (ITS). There are several research papers which discuss the benefits of the use of Semantic Web concepts for user modeling in Intelligent Tutoring Systems. There is a lot of research going on in the AI-ED community regarding enhancing intelligence in the ITS. There are some ITS and authoring systems in which Semantic Web concepts are used. In this paper the basic literature on ITS, AEHS and AI-ED is reviewed. Also, the basic concepts of Semantic Web are reviewed and its connection with research in all the above fields is examined. It is found that the combined use of research insights received from all the three fields, together with Semantic Web technology and AI techniques, would enhance intelligence for tutoring systems on the Web. This provides a basis for further research in the field of Intelligent Tutoring aimed at enhancing the intelligence of the systems.

Keywords: Intelligent Tutoring Systems, Adaptive Educational Hypermedia Systems, Ontology, Personalization, Semantic Web, AI-ED

1. Introduction

Intelligent Tutoring Systems (ITS) offer computer-based personalized instructions by imitating a human tutor, and provide a highly efficient learning environment [1]. Traditionally, ITS have four components [2, 3], namely: Tutoring Module (Pedagogical Module), Expert Module (Domain Knowledge), Student Module (Student Model) and Interface Module (Problem Solving Environment or User Interface), through which the student interacts with the system. While analyzing the state of the art of authoring ITS, [4] has mentioned Intelligent/Adaptive Hypermedia Systems as one of the authoring ITS. For more than fifteen years, many researchers [5-10] have studied Adaptive Hypermedia Systems with special applications to web-based education, which come from the combination of research in hypermedia systems and user-modeling. Adaptive Educational Hypermedia Systems (AEHS) contain a document space, a user model, a set of observations, and an adaptation component. Adaptation in AEHS, i.e., use of user models and adaptive component, added “dynamic” value in traditional “static” hypermedia systems by providing different educational content to different users based on their requirements.

Recent developments in Web Technologies and efforts to make them more intelligent led to the development of Semantic Web Technology. “The Semantic Web is a vision, the idea of having data on the Web defined and linked in such a way that it can be used by machines not just for display purposes, but for automation, integration and reuse of data across various applications” [1]. The Semantic Web is for im-

http://www.w3.org/2001/sw/Activity/
plementing reliable, large-scale interoperation of Web services and making such services computer-interpretable to create a Web of machine understandable and interoperable services that intelligent agents can discover, compose and execute automatically [11]. The Semantic Web offers scalability and expands the services of the existing web by enabling software agents to automate procedures currently performed manually and by introducing new applications that are infeasible today [12].

Application of Semantic Web concepts to Intelligent Tutoring directs towards AEHS because of similarity between the concept of student modeling and user modeling. Reusability and interoperability offered through Resource Description Framework (RDF) and ontology increase the scalability of the potential use of ITS [13]. There are several research papers [14-19] which discuss the use of Semantic Web concepts for user modeling in Intelligent Tutoring Systems. There are some AEHS, such as OWL-OLM [20], OntoAIMS [21] and ActiveMath [22], in which Semantic Web concepts are used.

In this paper, the basic literature of ITS and AEHS and the basic concepts of Semantic Web are first reviewed. This review is followed by discussion of the connection between AEHS and ITS when concepts of Semantic Web are used for Intelligent Tutoring. Finally, some AEHS, which use basic concepts of Semantic Web, are reviewed, discussed and classified to get an overview for further research.

2. Intelligent Tutoring Systems (ITS)

Intelligent tutoring systems are the application of Artificial Intelligence in the field of education. In the literature, tutoring generally means intelligent tutoring. According to [23, 24], ITS are computer-based instructional systems in which instructional content and teaching strategies are modeled in such a way as to specify what to teach and how to teach, respectively. Four components in the structure of an ideal ITS are described in Figure 1. In an ITS, the student is engaged with the interface module in order to get the actual tutoring. In the tutoring module, models of pedagogical strategies are included. The domain knowledge module is nothing but the expertise module which is used as a knowledge base for teaching. The student module (i.e. the student model) is information about the student’s interaction with the system: what the student has learned, which exercises the student has solved, etc. Basically, the student module is a diagnostic module which helps to diagnose students’ misconceptions and provide feedback [25]. The expert module and the student module are discussed in detail; this will help us to draw parallels between ITS and AEHS with respect to use of Semantic Web concepts.

![Fig. 1: Structure of an intelligent tutoring system [2, 3]](image)

2.1. Expert Module

For designing an intelligent system, one needs to have an abundance of domain knowledge. Domain knowledge can be encoded in three ways [26]. Firstly, there is the black box model, where only input-output information is available but no information is available regarding how the input information is processed to get the output. Secondly, there is the glass box model, which uses knowledge engineering techniques: i.e., it builds the knowledge model by consulting an expert. Finally, there are cognitive models, i.e. the way human beings use knowledge by coding different knowledge types. Cognitive modeling is used in Cognitive Tutors [27] (which are based on the Adaptive Control of thought Rational (ACT-R) theory developed by [28]), and is one of the examples of the third way of knowledge encoding. ACT-R theory consists of two parts: declarative knowledge and procedural knowledge. Declarative knowledge is nothing but factual knowledge stored in the form of chunks. Normally, chunking means dividing into small groups, but from a machine process ability perspective, chunks are group of prepositional structures connected with each other. Procedural knowledge is the knowledge displayed in our behavior and unconsciously implemented by production rules. Production rules are formalisms for representing reasoning in mathematics [29], and are the rules required for performing different cognitive tasks. One of the advantages of the rule-based approach is that it can be used for representing the student’s knowledge.
2.2. Student Module

Besides assessment of the student’s knowledge, the human tutor uses the student’s body language, facial expressions, and voice tone to recognize the student’s emotional state and then alter the teaching strategy as per the student’s need. This is the major idea behind adaptive intelligent tutoring, which is discussed more in detail in the section on AEHS. In the ITS, this information about the student’s competencies, learning achievements, and affective characteristics like emotional states, opinions, and learning is gathered from the student’s interaction with the system and stored in the student model [30].

Following are the major reasons described in [31] for why the student model is used in ITS. Firstly, for learning a structured curriculum through an ITS, if the student wants to move on to the next topic, s/he needs to master the topic which is the prerequisite of the next topic. Student modeling is used to determine which topics the student has mastered. Secondly, to give spontaneous advice at appropriate places, the system must know the student’s previous knowledge, which can be stored in the student model. Thirdly, based on the student’s capabilities known through the student model, some ITS generate problems suitable for that particular student. Finally, the student model is used for giving adaptive feedback based on what the student already understands, a record of which is stored in the student model. If students are provided customized feedback and are taught with instructions tailored to their individual learning styles, their interest in learning may be increased [30].

Classification of student models in a three-dimensional space has been made by [31], the three dimensions being: bandwidth, target knowledge type and differences between student and expert. The bandwidth dimension is related to the quality of input. This bandwidth (which is nothing but the time the student takes to solve a problem or exercise) can be used for the diagnosis of the state of a student. There are three types of states: mental states, intermediate states, and final states, which are mentioned in [31] under the bandwidth dimension. The other two dimensions are related to structural properties. Three target knowledge types are mentioned: flat procedural, hierarchical procedural, and declarative. The complexity of the interpretation of each type of knowledge affects the diagnosis, and hence the student modeling. According to [31], flat procedural knowledge makes student modeling easier, hierar-

<table>
<thead>
<tr>
<th>Knowledge Type</th>
<th>Knowledge Category</th>
<th>How Represented</th>
<th>Sample Tutors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topics</td>
<td>Concepts, facts, rules, skills, abilities, goals, plans, and tasks; declarative knowledge about objects and events</td>
<td>Overlay plans of facts and procedures, Bayesian belief networks; declarative knowledge</td>
<td>Student, Scholar, Wise, Winston, LISP tutor, Animal/Work, Cardiac Tutor, TAT</td>
</tr>
<tr>
<td>Misconceptions and Bugs</td>
<td>Well-understood states, “buggy” knowledge, missing knowledge</td>
<td>Bug library, bug parts library, and rules</td>
<td>BUGGY, Scholar, Why, GUIDON, Meep, PROGTEST, LISP Tutor, Geometry Tutor</td>
</tr>
<tr>
<td>Student Affect</td>
<td>Engagement, boredom, frustration, level of concentration</td>
<td>Reinforcement learning, Bayesian belief network</td>
<td>Auto Tutor, Animal watch, Learning companion</td>
</tr>
<tr>
<td>Student experience</td>
<td>Student history, attitude, interest, experience, goals, content of the exam</td>
<td>Receive all statements made by student, identify patterns of student actions</td>
<td>Ardenius and Gray, 2000</td>
</tr>
<tr>
<td>Stereotypes</td>
<td>General knowledge of student’s ability and characteristics; initial model of student</td>
<td>Build several default models for different students; store most likely values</td>
<td>Fitz et al., 1994, Roth, 1979, 1983</td>
</tr>
</tbody>
</table>
2.3. Techniques for building Student Models

There are Cognitive Science techniques and Artificial Intelligence techniques for building student models. Model-tracing and constraint-based are Cognitive Science techniques, whereas formal logic, expert systems, plan recognition or machine learning and Bayesian belief networks are AI techniques [30].

Model-tracing Technique: Cognitive Tutors [27] and the Andes Physics Tutoring System [33] are two of the systems which use the model-tracing approach to trace the student’s interaction with the tutors by using expert knowledge, thus determining whether students are heading towards an acceptable solution or not [34]. The Model-Tracing Technique uses procedural rules of the following form: \( IF \ c_1, c_2 \ are \ true, THEN \ take \ the \ action \ a \). Along with correct production rules, Cognitive Tutors have buggy production rules for tracking the student’s wrong approaches. One of the limitations of model-tracing is that it does not talk about the student’s misconceptions and only tells whether the student is going in the right direction or wrong direction; this is because of the predefined nature of bugs. Another limitation is that if a student makes a mistake which is not in the predefined model, then it cannot be captured by the Model-Tracing Technique.

Constraint-Based Technique: Limitations of model-tracing are due to the assumption that all mistakes by students can be captured and modeled for machine understanding. The constraint-based technique assumes exactly the opposite. The belief that students learn from their errors [35] is the basis for constraint-based techniques. SQL-tutor [36], WETAS [37], and KERMIT [38] are systems in which constraint-based techniques are used. According to [35], we make mistakes because of lack of internalization of declarative knowledge into our procedural knowledge, and by practicing the task we modify our procedural knowledge and learn [39]. In constraint-based modeling, the system is not interested in what the student is doing, but in the problem-state that the student is in. As long as the current state of the student is not identified as wrong, the system does not interrupt the process and the student can proceed freely. The domain model contains a set of rules in the following form: “\( \text{If } <\text{relevance condition}> \text{ is true, then } <\text{satisfaction condition}> \text{ had better also be true, otherwise something has gone wrong} \)” [39, p.315]. The relevance condition is a set of problem states for which the constraint is relevant, and the satisfaction condition is the subset of states in which the constraint is satisfied. In case of constraint violation, the system is expected to provide remedial action. The constraint-based technique is based on the assumption that the student’s learning cannot be completely recorded and only errors can be recognized by a system [30].

Machine Learning Techniques: Student modeling techniques from Artificial Intelligence are based on reasoning about the knowledge available about the student, and not on human learning. Machine learning techniques such as Bayesian belief networks, reinforcement learning, supervised learning, hidden Markov models, fuzzy logic, and decision theory are used for student modeling [30]. Most of the techniques above use probability theory to predict student behavior and make tutoring decisions. Bayesian belief networks facilitate the combining of prior knowledge or belief with new data and offer a guideline to keep new information in the proper position. In ITS, Bayesian belief networks are used for the classification of students’ knowledge, and for the prediction of their behavior and the steps where they will need help. There are two types of student models that can be generated by using Bayesian belief networks: expert-centric and data-centric [40]. In the expert-centric model, networks and conditional probabilities are specified either directly or indirectly by experts; whereas in the data-centric, structure and conditional probabilities are learned mainly from the data collected from real-world evaluations.

Expert Systems Technique: In the field of AI, expert systems (ES) are computer-based systems (production systems) that imitate the decision-making ability of a human expert [41]. They are designed for problem solving by using human-like knowledge expertise. Expert systems are composed of at least a knowledge base, an inference engine, and a user interface [42]. In ESs, production rules are used for diagnosing the problem and then suggesting solutions, which makes them a part of problem solving research. Expert systems have been researched and used extensively in medical diagnostic systems [43], intelligent patient monitoring systems, and management [44]. One of the major research links between expert systems, knowledge modeling and problem solving is the work done by [44-47]. The importance of the role of ontology (see section 3.4) for knowledge modeling is highlighted by [48, 49]. Opal, a knowledge acquisition system for oncologists and a tool for building Oncocin knowledge bases, was developed by [50]. Oncocin was an advice system for protocol-based cancer therapy developed by [51]. General development of Opal led to the development of Protégé-I, which succeeded in decreasing the difficulties of
knowledge acquisition for the purpose of medical advice [52]. Further, Protégé II was developed with the idea of reusable problem solving methods, and followed research by [47, 48] for the development of a tool for generating a hierarchy of problem solving methods and inference rules as separate components. Protégé has been developed into a tool named Protégé 4.0.2\(^2\), which is also used for editing and creating ontology with a class-subclass hierarchy and a hierarchy of properties and instances. AI techniques provided the basis for the use of concepts of Semantic Web technology in ITS. How this is connected with Adaptive Hypermedia Systems, with particular reference to Education, is discussed in section 4.

2.4. Tutoring/Pedagogical Module

In the field of education, the term tutoring can be understood as an individualized instruction (for one, two, or three students at a time) in which continuous interaction takes place between the tutor and student(s). For tutoring, the tutor needs to adapt the tutoring methods as per the requirement of the student and based on the interaction which the tutor has with the student. For this adaptation, the tutor needs to assess the student’s style of learning, diagnosing any errors in it, and then improve it by providing remedial feedback. Current research reveals that development in tutoring happens as a result of computing the student’s cognitive, meta-cognitive and affective knowledge [53].

In designing any educational application, the satisfaction of the four whys (what, when, where, who) and one why (how) is necessary. In the case of ITS, the when question does not arise because it is a computerized system, and so the student can learn at any time. Four questions that one must think about before designing a tutoring module are discussed by [53, 54]: Who is the learner? (relating to the learner’s cultural and academic background, etc.); What is going to be taught during the tutoring? (relating to subject matter, etc.); How will the tutoring be done? (relating to teaching and the assessment strategies to be used during the tutoring, etc.); and Where or in which environment is the tutoring going to take place? (relating to whether it will be in the home or the school or college classroom, whether it is mobile learning or web-based learning, etc.). For exploring the answers to the questions which are mentioned above, one needs to model the tutoring according to the knowledge gathered from the student’s model. This is where we find the major connection between ITS and AEHS.

In order to design the tutoring module in such a way that conceptual understanding and self-regulated learning are included, one needs to consider pedagogical philosophy, such as constructivist learning, collaborative learning, Bloom’s theory of mastery learning, and Anderson’s ACT-R theory. Decisions about pedagogical theories and their effective use should be made after consulting pedagogical experts (i.e. teachers). Some researchers [55, 56] in the AI-ED (Artificial Intelligence and Education) community have surveyed and analyzed the ITS. It was found that pedagogical experts were not involved in the authoring of ITS as the authoring tools were not user-friendly. A huge gap was found between pedagogical planning and domain knowledge. The domain-specific nature of existing ITS was not sufficiently flexible that it could be reused. Even though ITS are an application of AI for education, the research community of ITS remained unfamiliar with the AI research community; this is why ITS researchers had never used developments in AI for improving intelligence in the ITS. Several researchers [55-59] have recommended and used Semantic Web concepts for authoring ITS (this is discussed in detail in section 6).

2.5. Interface Module

The essence of a good interface is that it should provide instruction and feedback in a clear and explicit manner. The interface module must consider inputs from research on Human-Computer Interaction (HCI). Presentation of knowledge, reflection of instructional objectives, use of more graphics, effective use of natural language (such as dialog), appropriate choice of display size, use of advanced computer techniques such as touch screen and speech recognition, are some of the techniques discussed in [60, 53]. Presentation of knowledge through the interface plays an important role in the conceptual understanding of learners. While discussing the role of HCI in ITS, [60] proposed the breaking up of instruction as per the cognitive model, the use of dialog mode by the breaking up of instructions, the effective integration of graphics and natural language, and the presentation of semantic information in the domain for developing deeper understanding in learners. Discussing advanced applications for ITS, [61] described experiments conducted by the MIT media Lab aimed at determining the learner’s affective domain during

\(^2\) \url{http://protege.stanford.edu/}
interaction with the ITS; these experiments involved the use of various sensors, such as the mental state camera (for determining mental status), the skin conductance bracelet (for determining emotional status), the pressure-sensitive mouse, and the pressure-sensitive chair.

The basic concepts of Semantic Web technology are discussed in the following section.

3. Semantic Web Technology

Research into the application of AI concepts in Web development and efforts to develop the intelligent Web have resulted in the introduction of Semantic Web technology. The World Wide Web Consortium (W3C) is one of the leaders in Semantic Web Technology, particularly by the creation of standards. W3C has developed a Semantic Web layer cake (shown in Figure 2), which provides a basis for the development of Semantic Web technology.

![Semantic Web Layer Cake](http://www.w3.org/2005/Talks/1122-orf-sw/sw-stack-2005.png)

The Semantic Web can be built by exploiting the capacity of each block of the Semantic Web layer cake. Current research in the Semantic Web includes Linked Data [62], the practice of publishing and connecting structured data on the Web that can be used for developing applications of Semantic Web in various fields. In the Semantic Web layer cake, the semantics increase from the lower to the upper level. Some of the sub-sections (such as 3.6, 3.7 and 3.8) are largely taken from the paper [63] in which these concepts had already been discussed by the authors of this paper.

3.1. URI and Unicode

The base layer consists of URI, Unique Resource Identification, which uniquely identifies a resource found anywhere on the Web. "Unicode is a computing industry standard for the consistent encoding, representation and handling of text expressed in most of the world's writing systems". In the Semantic Web, the Unicode is used to represent and connect documents written in different languages.

3.2. XML

XML is Extensible Markup Language, which provides a structured way to present data which can be read by machine. Markup Language is a modern system of annotating the text in a way that is syntactically distinguishable from the actual text. Markup means the sequence of characters, letters or other symbols that a document creator inserts at certain places in a text or word processing file to indicate how the file should look when it is printed or displayed or to describe the document's logical structure. The markup indicators are often called "tags." For instance, in HTML (HyperText Markup Language, which is used for creating web-pages), to make a text for a page heading there are several tags, such as <h1> or <h2> or <h3>, which represent different levels of headings. XML Schema is the framework provided for the XML representation of any data. XML namespaces offer a way to globally use terms from more resources, as the Semantic Web needs to connect data from different resources. XML Query is a query language for extracting and manipulating the data in an XML document; it is also called XQuery. Signature is a digital signature for authentication of a document which is written in the XML format. Encryption means basically the same thing that it means in cryptography, referring to the process of hiding private or secret information from others and making it accessible only if some special key is used. The encryption here is in the XML format, and is especially developed as a standard for safety.

---

3.3. RDF

The next layers are RDF and RDF Schema. The Resource Description Framework (RDF) is a language for representing information about resources on the World Wide Web. RDF contains a triple of subject, predicate and object. The RDF model can be graphically represented as shown in Figure 3.

Fig. 3: The RDF model

In the English statement Swati ate Mangoes, Swati is the subject, ate is the predicate, and Mangoes is the object. This statement can be written in the form of an RDF statement as shown in Figure 4.

Fig. 4: Example of simple RDF statement

RDF Schema is a framework that provides a basic structure to RDF. The RDF schema structure uses a special vocabulary of terms like classes, properties, container classes and properties, and RDF collections. In RDFS framework, terms such as Resource, Class, Literal, Datatype, XML Literal, and Property are used for generalized categories named classes. Terms such as range, domain, type, subClassOf, subPropertyOf, label, and comment are used for defining predicate categories or categories of properties. Terms such as Container, Bag, Sequence, and Member are used as container classes and properties. Terms such as list, rest, and nil are used for defining the RDF collections. These terms are generally used as follows, rdfs: Resources; rdfs: Class; rdfs:Datatype; rdfs:domain; rdfs:type; rdfs:subClassOf; rdfs:Bag; rdfs:Container; rdfs:Seq; rdfs:list; rdfs:nil. This means that these terms are used with reference to their definition in the RDF schema. A collection of such RDF triples can be used to bring about interoperability between applications that exchange machine-readable information on the Web. RDF is used for annotation of resources on the Web, which is also called RDFa. The example shown in Figure 5 demonstrates the RDF/XML format as it is used with interconnections. There is a website http://www.hogward.org/index.html, which is created by a creator of value exstaff:65231. The same website has creation date of value April 1, 1860, and it is in the language with value en. This can be written in RDF/XML format as shown in Figure 5. In the above example, the terms xmnls and dc respectively stand for xml namespace (discussed in section 3.2) and Dublin core metadata (discussed in section 3.6).

Fig. 5: Example of RDF/XML format

3.4. Ontology

Generally the term Ontology means a philosophical study of the nature of being, existence or reality. In computer science or information science, ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. Ontology is a specification of a conceptualization [64]. Conceptualization means a structured interpretation of a part of the world in which people think and communicate. Ontology is explained by [49] in two ways. Firstly, ontology is a representation vocabulary specialized to some domain, in which the terms in the vocabulary are intended to capture the conceptualizations of the domain. Thus, translation of an ontology from one language to another, say from German to English, does not change it conceptually. Secondly, ontology also refers to a body of knowledge describing some domain using a representation vocabulary, where some facts about domain knowledge are represented.

For instance, in biology ontology, where facts in biology can be represented by using connections between the concepts, Thing can be a super class, Living Things and Non-living Things can be two sub-

9 http://www.w3.org/TR/2004/REC-rdf-primer-20040210/

10 http://en.wikipedia.org/wiki/RDFa


12 http://en.wikipedia.org/wiki/Ontology_%28information_science%29
classes of the super class Thing, and the class of Living Things can be further divided into two subclasses, Vertebrates and Non-vertebrates. In other words, ontologies are essentially content theories representing the content in the form of classes and sub-classes of objects and the relations among them. Ontology contains hierarchies and tree structures. The spectrum for increasing semantics of ontology mentioned in [65] and described by [66] is shown in Figure 6. An ontology is defined by [67] as a set of knowledge terms including the vocabulary, the semantic relations among the terms, and some simple rules of inference and logic for some particular topic (this is quite similar to what is explained by [49]).

Fig. 6: The Ontology Spectrum [66]

There has been much debate about the best definition of ontology. The definition given by [65] is widely accepted, but one criticism of it is the general nature of the term specification, as there are no agreed-upon borders [68, 69]. Gruber’s definition of ontology is modified by [69] in the following words: “ontology is a formal specification of a shared conceptualization”. The inclusion of the two words ‘shared’ and ‘formal’ in the modified definition emphasize, respectively, the agreed upon conceptualization of the ontology that can be specified by a group, and the machine-readability of the ontology. Usually, ontologies can also be represented in the form of a set of RDF triples. They are used for information sharing and collaboration.

3.5. Logic and Reasoning

Logical reasoning is used to check the consistency of the data and infer conclusions which are not explicit in the ontology, and proof actually traces the steps of logical reasoning [70]. Trust stands for provision of authentication and trustworthiness of the statements derived from the data. There are various languages used for ontology, such as Resource Description Framework Schema (RDFS), Web Ontology Language (OWL), Ontology Inference Layer (OIL), and DARPA Agent Markup Language (DAML). OWL is a W3C project for standardizing a more capable ontology framework than RDFS [70]. OWL comes in three sublanguages named OWL Lite, OWL DL and OWL Full. OWL DL is based on Description Logic and OWL Lite is OWL DL with more restrictions. OWL Full is a complete language with both OWL-Lite and OWL-DL. We can use Protégé to create ontology. After creation of the ontology, one can create rules by using the Semantic Web Rule Language (SWRL) [13], and by combining OWL and Rule Markup Language as defined by W3C standards. Logic is the third last building block of the Semantic Web layer (Fig 2). It is used for applying and evaluating the rules defined. It can help in inferring the facts that are not directly given in the ontology, and can interpret information which is not explicit. Logic can also detect contradictory claims and statements in the ontology [70]. The ontology, along with logical rules, forms a knowledge representation. Hence, it can be said that another role of logic is to represent and infer knowledge. There is a tool named RacerPro [14] (renamed ABox and Concept Expression Reasoner Professional), which can be plugged-in into the protégé; with the help of defined SWRL rules one can perform reasoning over the ontology. A-Box and T-Box are two notions that are frequently used in the world of ontology. T-Box stands for all the terms which are used in the ontology, which include classes-subclasses, predicates, and the logical statements formed with them. A-Box includes all the instances of the classes and subclasses and the rules associated with the instances. RacerPro uses logic for reasoning the ontology through A-Box and T-Box. The ontology can be reasoned out without tools like RacerPro. As mentioned above, ontologies can be represented in RDF triple format. There is a query language named SPARQL [15] (SPARQL Protocol And RDF Query Language) which can be used to query ontology in RDF format.

Apart from the basics of Semantic Web technology such as XML-RDF & Ontology, there are two other

---

13 http://www.w3.org/Submission/SWRL/
14 http://www.racer-systems.com/products/racerpro/index.phtml
15 http://www.w3.org/TR/rdf-sparql-query/
important notions, metadata and learning objects, which are connected with Semantic Web and are used in AEHSs extensively. For completeness of the discussion of the application of Semantic Web concepts in ITS, these two notions are discussed briefly in the next sub-sections.

3.6. Metadata

Metadata is data about data. In computer science, metadata is generally taken to mean information about a set of data in a particular representation, which typically means “information about objects” [71]. There are some metadata standards, such as Dublin Core [16] and IEEE Learning Object Metadata (IEELOM) [17], that are used all over the world. Dublin Core Metadata Element Set (usually referred to as just Dublin Core) is a metadata standard used by librarians to broaden the scope of book searches, and to facilitate search and information retrieval in cross-domain information resource description [72]. DC Metadata contains mainly bibliographic information, such as Title, Name of Creator, Subject, Description, Publisher, Contributor, Date, Type, Language, Source, etc. IEEE LOM standard is a principal metadata standard approved by IEEE, which contains seventy-six elements grouped in nine categories [73]. These have the nature of attribute and value. Sometimes metadata can also be seen as a fragment of ontology. Sharable Content Object Reference Model (SCORM) [18] is a collection of standards and specifications for web-based learning that facilitates sequential content presentation and is commonly supported by learning management systems.

3.7. Learning Objects

Learning can take place anytime and anywhere, whether in traditional classrooms, outside the classroom, or while working through educational videos or online courses. Learning benefits significantly from the availability of what the IEEE Learning Technology Standards Committee defines as Learning Objects (LOs): that is, any entities, digital or non-digital, which can be used, reused, or referenced during technology-supported learning [73]. LOs stored anywhere, locally or globally, can be made available at any time and at any place to learners, provided that they have network connectivity. LOs can be used in any learning environment and across disciplines, and facilitate the creation and use of online educational content [74]. LOs are flexible, portable, and adaptable, and can be used in multiple learning environments and across disciplines. Learning objects are also reusable and interoperable [75, 76]. They may take a number of forms, such as a text document, a video/audio file, an image/animation, a simulation, or a virtual lab experiment. LOs can also be in the form of activities, games, or tests.

There are various Learning Object Repositories (LORs) available on the Web, such as Campus Alberta Repository of Learning Objects (CAREO) [19] and MERLOT [20]. Learners can use the content of these LORs for updating and enhancing their knowledge. LORs store both LOs and their metadata [77]. The metadata maintains links to the associated LO if it is stored elsewhere on the Web [78]. Ontologies and metadata are also sharable. The IEEE Learning Technology Standards Committee has defined Learning Objects Metadata (LOM) standards [73]. A number of LORs, such as ARIADNE [21], SMETE [22], Learning Matrix [23], MERLOT, CAREO, Learn-Alberta [24], and Lydia Inc. [25], contain IEEE LOM profiles.

3.8. Topic Maps

Topic Maps are covered by a number of papers [63, 79-85]. Generally, Topic Map technology is not accepted as a part of Semantic Web technology by the Semantic Web community because it lacks the powers of first order predicate logic for inference that are used in Semantic Web. However, Topic Maps do use XML-based schema, RDF for machine process ability, the notion of ontology-instances, and some logical inference by using Tolog query: these give it many of the capabilities of Semantic Web. Also, they are widely used by researchers [86, 87, 88] for web-based education. Hence, in this review, Topic Maps are also considered in the discussion of Semantic Web.

The basic concept model of Topic Map is TAO, where T stands for Topics, A for Associations, and O
for Occurrences [80]. An interesting facility provided by Topic Maps is the capacity to group Topics under Topic Types, Associations under Association Types and Occurrences under Occurrence Types. Also, there are Role Types which are used in the Topic Map to represent the roles played by different topics in the association. The use of a subject identifier for each topic in the form of a URI provides it with a namespace, i.e. a context. The facility of URI gives Topic Maps the power of merging and scalability. Various types (such as association types and role types) give Topic Maps the capacity to filter information. This is all explained in detail in [63, 79]. Topic Maps (TM) have general ontology and specific instances. Topic Maps were initially developed with a view to using them in the making of an encyclopedia. The advanced study and applications of TM in various fields revealed their usefulness for educational purposes. The standard used for creating Topic Maps is the XML-based Topic Map standard called XTM.

In the next section, basic concepts of AEHSs and some systems are discussed.

4. Adaptive Educational Hypermedia Systems (AEHSs)

Adaptive Hypermedia Systems are systems which provide user-driven services. “Adaptive Hyper Media Systems means all hypertext and hypermedia systems which reflect some features of the user in the user model and apply this model to adapt various visible aspects of the system to the user” [89]. AEHS have the capability of bringing about a paradigm shift of e-learning in two of its aspects: by the elimination of barriers of time, distance, and culture; and by the adaptation of the learner’s experiences [90]. Users with different goals, knowledge, background, experience and preferences get service according to their personal needs. Tailoring these services to the user’s need is called adaptation or personalization. The information which is used for making this adaptation or personalization is called the user model; alternatively, we could say that knowledge representation in the user model is used for adaptation. This information can be gathered by observing the user’s interaction with the system, by asking the user directly, and from the user’s profile. There are various adaptive hypermedia systems, such as content-based recommender systems, case-based recommender systems, personalized e-commerce applications, and adaptive news access systems. The systems which are used for educational purposes are called Adaptive Educational Hypermedia Systems. “User model” is an umbrella term under which the student model falls.

A good learning experience is one in which a learner can acquire new knowledge and skills, criticize any examined assumptions and beliefs, and engage in an energizing, collaborative search for knowledge and holistic personal development [91]. A User Model is: “a knowledge source, which contains explicit information on all aspects of the user that may be relevant to the dialog behaviors of the system” [92, p.6]. Personalization leads to more efficient learning [93–95]. Many researchers have discussed personalization in the systems described by them [96, 76, 97, 98]. Some researchers [99, 100] have described techniques of personalization. Access to Web-resources, especially learning objects, for use as supplementary material can be greatly facilitated by user models that describe an individual in terms of parameters such as name, age, grade of study, languages known, level of academic achievement, and preferences for types of learning material. Web-based tutoring systems like ActiveMath [22] (for teaching mathematics), ELM-ART[101] (for providing adaptive intelligent tutoring of material from an interactive online textbook), KBS Hyperbook [102] (an adaptive intelligent tutoring system for programming), Slide Tutor [103] (an intelligent tutoring system for visual classification problem solving), and SQL Tutor [36], are some examples of AEHSs.

As there was no formal way of representation for AEHSs in the beginning, [6] proposed some formal definitions: “An Adaptive Educational Hypermedia System (AEHS) is a quadruple (DOCS, UM, OBS, AC) with DOCS: Document Space: A finite set of first order logic (FOL) sentences with atoms for describing documents (and knowledge topics), and predicates for defining relations between these documents. UM: User Model: A finite set of FOL sentences with atoms for describing individual users (user groups), and user characteristics, as well as predicates and rules for expressing whether a characteristic applies to a user. OBS: Observations: A finite set of FOL sentences with atoms for describing observations and predicates for relating users, documents/topics, and observations. AC: Adaptive Component: A finite set of FOL sentences with rules for describing adaptive functionality.” For example, NetCoach [104] provides a framework for building adaptive hypermedia systems. It uses a knowledge base which consists of concepts, and the concepts are internal representations of pages that can be presented to the learner. This is the basis for adaptive support in NetCoach. By using this, authors can create content-specific
relations between concepts and the knowledge base.

The understanding for describing all the four parts has been developed from [6].

4.1. Document Space

The document space may include domain ontologies (with domain concepts as atoms and connections between those concepts as predicates), a set of documents and hierarchical connections between documents, a set of tests to be offered, and relations between test and document or test and concepts. For instance, in a general and very simple Adaptive Educational Hypermedia System, the document space can be combination of a set of documents \( D = \{D_1, D_2, \ldots, D_k\} \) and the relations among the documents are: pre-requisite relation, contained_in or part_of relation. Pre-requisite relation means: it is a set of all documents \( D_j \) that are needed to be studied before another document \( D_i \) and is written as: \( \text{pre_requisite}(D_j, D_i) \) for some \( D_i \neq D_j \).

4.2. Observations

Observations contain information related to tracking the user’s interaction with the system. For example, if there is a set of users say \( U = \{U_1, U_2, \ldots, U_k\} \) and an observation of set of users who has read a particular document \( D_m \) is written as: \( \text{observation}(D_m, U_i, \text{has_read}) \) for for some \( D_m \) and \( U_i \).

4.3. User Models

User models draw conclusions about characteristics of users and what users have learned after observing user interaction. An example of a simple user model is:

- A characteristic say ‘studied’ that is observed when user interacted with the system, and it can be written as: \( \forall U_i \forall D_l \exists D_m \exists Dt \text{pre_requisite}(Dt, D_l) \land \text{observation}(Dt, U_i, \text{has_read}) \Rightarrow \text{user_observation}(D_l, U_i, \text{Studied}) \)

4.4. Adaptation Component

Adaptation component is a set of rules which are used at run time to provide feedback to students according to the information gathered from their user model.

For the rule to state of the study of a document, an adaptive component used can be say suggested_for_reading and the traffic signal analogy for showing the adaptive feedback can be say red_flag and green_flag, can be written as: \( \forall U_i \forall D_l (\forall D_m \text{pre_requisite}(D_m, D_l) \land \text{observation}(D_m, U_i, \text{studied}) \Rightarrow \text{state_of_the_study}(D_l, U_i, \text{Suggested_for_reading}) \)

Similar notations with more sets of relations, observations, inference rules for user models, and adaptation components are used in NetCoach [104], ELM-ART [101], and KBS Hyperbook [102], which are described in detail in [6]. One of the major criticisms of [6]’s definition of AEHS was that it is difficult to re-use due to its usage of rules and logical framework at the higher level and its excessive domain dependence [105, 106].

5. Use of Semantic Web concepts in AEHSs

From the above review of four components in the definition of AEHS, it is clear that the key to adaptation is the knowledge model created or used at the back end, and the rules and reasoning used for manipulating that knowledge model. An AEHS is an interconnection of two hyperspaces: knowledge space (i.e. concept-based hyperspace) and content-based hyperspace [107]. The document space can easily be replaced by ontology.

Ontology has its own knowledge representation, navigational capability, and the ability to be used for machine learning, and hence can be considered as part of concept-based hyperspace. Ontology is not only about the hierarchical connection of concepts: it also includes the OWL, description logic, and the inference rules. Hence the same knowledge can be used for the observations, user models, and adaptation component. The combined use of domain ontology, goal ontology, user model ontology and logical reasoning to create an adaptive web is argued for in [108]; the same can be applied in the case of web-based intelligent tutoring systems by using various types of ontologies. In Figure 7, an excerpt of application domain ontology for Java e-lecture is shown, which is taken from [108, p. 708]. Metadata standards offer description of Learning Objects available on the Web, and ontology offers a way to describe domain knowledge and the learner. Due to RDF and the use of annotation properties, we can connect this ontological knowledge model to the educational content on the Web, which is termed as a content-based hyperspace. For creating online courses and sequencing the exercises in them, metadata of the course material, which is available in the form of LOs, can be used.
For distinguishing various kinds of information on a user/student (such as personal relations, security, preferences, performance, and portfolio information), a learner modeling standard is used, developed by IEEE, named Personal And Private Information (PAPI)\(^26\). A combination of RDF, RSFS, and standards like PAPI is used for applying personalization in web-based tutoring systems [109]. Ontological languages use terminology from different namespaces; this gives scalability and interoperability to ontologies. Many namespaces come from metadata standards such as Dublin Core (dc) and various other standards such as RDF Schema (rdfs) (as explained in section 3.3). These can be used for resource annotations in the ontology, and thus can be used to link the online resource with its metadata file. An RDF instance of performance security record for the Advanced Security Technologies for Networking II course, using PAPI standard, is shown in Figure 8, which is taken from [109].

```
<rdf:Description rdf:ID="BOB">
  <papi:performance>
    ...</rdf:Description>
```

Fig. 8: RDF instance of performance of security Record [106]

The use of RDF annotation can be applied for adaptive course generation on the Web. The use of such combinations of concepts in Semantic Web for creating AEHS has been demonstrated by [110, 55]. The use of RDF and metadata in AEHS for sequencing and reuse of course material has been demonstrated and discussed in [97, 9]. The use of ontologies in AEHS has been discussed in [10, 96, 111]. TM4L [86] is a Topic Map-based course authoring adaptive hypermedia system in which concepts like XML, Ontology, RDF, and metadata are used. Topic Maps are also used as a tool for distributed learning environments based on the Web [112]. OntoAIMS [21] is a course authoring system in which the author can create course ontology, subject-domain ontology, and the educational metadata for the resources that will be attached as the course material; the same can be used for collaborative course authoring. OWL-OLM [20] is a framework for capturing the user's conceptualizations based on dialogue generated by using ontology for implementation of effectual personalization in the Semantic Web. OWL-OLM is used by OntoAIMS for capturing the learner’s conceptualization by using the domain ontology. By OWL-OLM, the student’s interaction with the system (in the form of a graphical structure presented for each concept) is captured as a short-term knowledge model, and then the same is used for updating the long-term conceptual model of the student.

---

A framework named LAOS (Layered AHS Authoring-Model and Operators) (please see Figure 9), based on Semantic Web concepts, has been developed by [106] for Adaptive Hypermedia Systems (AHSs), and is discussed in [65]. LAOS model contains five layers, namely: Domain Model (DM), Goal and Constraints Model (GM), User Model (UM), Adaptation Model (AM) and Presentation Model (PM). Domain model contains linked resources; with respect to educational system, it will have linked concepts and learning material. Goal model contains goal-related information like instructional strategies or pedagogic information about the resources. User model contains user-related information such as information about student’s knowledge. Adaptation model contains the feedback specific to user/learner needs. Presentation model contains the display information (such as color scheme) used for presentation of the content. Presentation maps of all the five layers (i.e. domainMap, goalAndConstraintsMap, userMap, presentationMap, and adaptationStrategyMap) are defined formally [65]. MOT (My Online Teacher) [113] (discussed in section 7) is an adaptive educational hypermedia authoring system based on LAOS framework. The key point in LAOS is that it is formulated with a view to separating the content from the authoring tool so that it will offer flexibility, expressivity, reusability, interoperability, non-redundancy, cooperation and standardization, which are overlapping goals of Semantic Web technology [106]. For already existing AEHS, semantic integration of such systems is proposed and discussed in [114]. For effective implementation of personalization in integrated systems one needs to translate the user models. Multiple translations can be avoided by using integration of user information collected on the fly, which is also called evidence [114]. Several problems are inherent in integration, such as naming conflicts, difference of graph structure, difference of scope, difference of granularity, and difference of focus. Use of ontology solves these problems. Use of single-ontology integration and central-ontology integration are the two approaches discussed by [114]. Single-ontology integration [114] means the use of common domain ontology as domain models for two systems so that their user models, which rely on the domain models, can be integrated easily. The integrations of Problets27 and QuizJET28, and of SQL Tutor [36] and SQL Guide, are described in [114] to illustrate single-ontology integration and central-ontology integration respectively. For these integrations two types of architectures are used: ADAPT2 (Advanced Distributed Architecture for Personalized Teaching & Training) [115], which is a framework for integrating adaptive and non-adaptive tools into distributed learning environments; and CUMULATE [116], which is a centralized user modeling server. (More discussion of these two can be found in [115, 116, 117].) An OWL-based General User Model Ontology (GUMO) is also proposed and discussed by [118] for uniform interoperability among distributed user models.

The next section discusses the application of Semantic Web concepts by the AI-ED research community.

6. Use of Semantic Web concepts in AI-ED research

AIED research includes research in AI, Instructional Design Science (IDS), and Learning Science (LS) [55]. Application of AI to Web-based ITS is very important because it offers many ways and techniques to implement intelligence via personalization, collaboration and adaptation in the ITS [56]. The issues found by the AI-ED research community were largely related to content. While analyzing the current state of the art of AI-ED research, [55] found several problems regarding authoring tools of intelligent tutoring systems. Some of the issues are: the huge conceptual gap between authors and authoring systems; limited use due to lack of reusability and interoperability; lack of user-friendly interface for authoring ITS; lack of awareness of research in Information Science and learning theories by ITS authoring systems. Considering the advantages of ontology, [55] proposed the use of ontological engineering for overcoming the problems stated above, and identified a need for extracting ontology from instructional theories and instructional design models available in the literature. The notion of ontology engineering is described by [119] as: ontology learning or automatic/semi-automatic extraction of ontology from textual/non-textual sources by a software system. It uses NLP-based techniques, statistical techniques, and machine learning techniques for extracting concepts, attributes, instances, and axioms from textual and non-textual sources. Domain ontology engineering tools and techniques from the literature are surveyed in [119], which also identifies the need for reusable services for ontology learning, editing, merging and evaluation. One of the major limitations of techniques

27 http://www.problets.org
28 http://adapt2.sis.pitt.edu/quizjet/help.htm
A blended learning approach, in which both humans and computers are involved in the ontology engineering, is proposed in [55, 119]. As per the modified definition of ontology, it is a collaborative effort. Hence, for using ontology in ITS one needs to integrate collaborative authoring of ontology into the authoring of ITS [120]. The use of XML and RDF is also proposed for the effective use of ontology and connecting the ITS with the Web. Some researchers [59] have offered support framework based on ontol- ogy for authoring intelligent tutoring systems. The knowledge is divided into two parts: static and dynamic. The static knowledge is about the curriculum instructions, and dynamic knowledge is generated from the adaptive component. The proposed framework includes the capacity to author a domain ontology, task ontology, and instructional design ontology. A description of the tasks as mentioned in the definition of ATO (authoring task ontology) by [121] is shown in table 2.

Table 2: Definition and Description of Tasks in ATO

<table>
<thead>
<tr>
<th>Definition of task in ATO</th>
<th>Description of the task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence of activities</td>
<td>Generic terms reflecting the roles of the objects in the authoring process</td>
</tr>
<tr>
<td>Goal</td>
<td>Generic roles representing authoring activities over the objects</td>
</tr>
<tr>
<td>Requirements</td>
<td>Generic adjectives representing the modifications of the objects</td>
</tr>
<tr>
<td>Constraints</td>
<td>Other authoring task specific concepts</td>
</tr>
</tbody>
</table>

To integrate learning design standards and connect those with theory, [58] described a formal process, which is closely parallel to the formal methods of creating ontology as described by [122-124]. The methodology is as follows: a) analysis of the domain; b) conceptualization, i.e. creating general classes and sub-classes and properties or relations among those classes; c) formalization; d) evaluation; and e) documentation. This reflects the true value and advantage of the use of ontology, because both the processes were discovered by researchers in different fields and at different times. It also shows that the creation of ontology is a collaborative venture which should be agreed upon, and reveals that the evolved definition of ontology is a formal specification of shared conceptualization. Creating a domain model or expert module of any ITS is a time-consuming task. To save time for the creation of a domain model in a constraint-based tutor, [14] used ontology to create a domain model automatically by extracting semantic and syntactic constraints from a domain ontology created by experts. Use of pedagogical agents in collaboration with educational servers and ontologies is one of the major contributions to AI-ED research achieved by the application of Semantic Web concepts to web-based ITS [56, 57]. The AI-ED research community is ready to accept the implementation of techniques of recommender systems as advancements in the ITS, which directs towards use of Semantic Web concepts and personalization. While discussing the development of an ontological web portal for educational ontologies, [125] presented a state of the art ontology of ontological technologies for education (as shown in Figure 10a & 10b), which gives us an overview of how to use ontology and the technologies involved in developing educational applications. This state of the art ontology is divided into two parts: using ontologies and building ontologies. Using ontologies (Figure 10a) include the application perspective of ontology and the technological perspective, i.e. technologies in which ontologies are used. Ontologies are applied as various types of knowledge and as a cognitive tool. Ontologies are used in various technologies like Knowledge Representation, Information Retrieval and Semantic Web technology. Building ontologies (Figure 10b) include theoretical issues in ontology engineering and development of ontologies. These are generated manually and/or by machine using automatic and or semi-automatic methods.

For automatic or semi-automatic generation of ontolo- gies, machine learning aspects, information retrieval aspects and some specific aspects of formal and semi-formal domains are used. For the development of ontologies there are various standards, languages and tools, which contain editors as well as ontology engineering environments. In the manual development of ontology, one faces subjective biases, pedagogical issues, and constraints; however, an ontology can be created collaboratively by considering concepts and
relationships among concepts, and writing the inference rules to extract knowledge from ontologies.

Fig. 10b: State of the art Ontology of ontological technologies for education [125]

For creating this ontological web portal, the authors used Topic Map technology as it has a simple formalization of ontology that can be easily used for knowledge structuring and representation. A similar use of Topic Map technology as an educational resource broker system for collaborative sharing of knowledge-centric content is discussed and demonstrated in [63].

The ITS research community has paid more attention to student modeling than to modeling the tutor and tutoring strategies. The AI-ED research community considers this to be the major gap in ITS research. Work done in the field of expert systems has been seen as an effort to model the expert; maintaining a knowledge base for this purpose has been found to be a challenging task. The use of expert system technology for developing ITS is criticized by AI-ED researchers [126] for ignoring pedagogical theory or learning science theory and for not justifying the rules used for knowledge modeling. This was the major motivation behind the development of OMNIBUS (Please see Figure 11), which is an ontology of learning theories and instructional theories that models the instructional activities openly or in a hidden manner.

Learning theories are concerned with conditions under which learning happens and the way in which it happens, whereas instructional theories are concerned with ways to facilitate learning and maximize its outcome, that is, with best learning. The OMNIBUS ontology is about target-world modeling, which contains an is-a hierarchy of not only the target concepts but also the states, actions, processes, events, and learning and instructional theories.

The change of state is defined as actions. Set of sequential and hierarchical actions is defined as processes, and set of processes is defined as events. Educational events are divided into two types: learning events and instructional events. In a learning event, the learner performs certain actions under certain conditions based on learning theories that have been designed to change her/his state of learning; whereas, in an instructional event, the instructor performs instructional actions to facilitate learning actions that are hoped to eventually bring about a planned change in the learner’s learning state. In OMNIBUS, this kind of nested structure of learning and instructional events is defined as Instructional Learning Event (I_Levent). Further I_Le vents are divided into three types of sub-events: simple event, reciprocal event (opposite of the simple event) and influential event. The influential event is further divided into an instructional event, an effective learning event, and a prepared learning event. The instructional event influences learning actions, thus causing an effective learning event (learning effect), which in turn prepares the base event (prepared learning event) for the
next learning action, so bringing about the required state of learning. From such a set of I_Levents an Instructional Learning scenario can be produced. By using OMNIBUS, an advanced authoring system named SMARTIES\textsuperscript{29} is implemented (discussed in [126]).

In the next section the intelligent tutoring systems in which semantic web concepts are used are discussed in brief.

7. Discussion of the ITS that Apply Semantic Web Concepts

Although many researchers have separately used Semantic Web concepts for educational applications, there exist very few Intelligent Tutoring Systems which have applied concepts of Semantic Web. ActiveMath\textsuperscript{30} is a web-based intelligent tutoring system for teaching mathematics with a focus on high school and college level mathematics; it offers an interactive and exploratory learning environment to learners. ActiveMath\textsuperscript{30} uses ontology for knowledge representation as well as for annotation of LOs, which contributes to re-usability and interoperability of the content. It is compliant with Dublin Core metadata standard, and uses standards like OpenMath, MathML (Markup language for mathematical elements in the form of XML), and IEEELOM. It also provides various kinds of exercises and a user-friendly environment for practicing exercises. ActiveMath offers learners personalized feedback and the opportunity to create their own personalized hyper-book for learning various concepts. It is one of the intelligent tutoring systems that have adaptive hypermedia.

SlideTutor\textsuperscript{[103]} is an intelligent tutoring system for microscopic diagnosis of inflammatory diseases of the skin. It uses some architectural aspects of cognitive tutors, but also uses various types of ontologies such as domain ontology, domain task ontology, ontology of abstract problem solving methods, and pedagogical ontology. The abstract problem solving methods used in the expert model produce a dynamic solution graph with which the student interacts. Errors and hints are modeled on the instructional module (i.e. on pedagogical ontology) to provide personalized training. SlideTutor\textsuperscript{[31]} is a system which is used for visual classification problem solving, and is based on the information processing theory of cognition. Because of the use of ontologies, SlideTutor offers flexibility in the instructional layer, scalability, reusability, and easy maintenance. The dynamic solution graph offers backward and forward reasoning. For creating ontologies in SlideTutor, Protégé is used, and production rules that are used for manipulating the ontologies are written in Jess\textsuperscript{[12]}, a Java production rule system. SlideTutor is an intelligent tutoring system based on the expert system model.

MOT (My Online Tutor)\textsuperscript{[113]} is a web-based authoring AHS environment based on LAOS, a five-layer framework discussed in section 5. It uses concept map-based domain knowledge at the front end and XML-based format at the back end, which offers flexibility and reusability in the Web-based environment. Using MOT, users can create their own tutoring courses or modules. The use of LAG, the adaptation language, offers the greater level of semantics that is discussed in [127]. For converting MOT into an AEHS, common adaptation format (CAF) is used, whose semantic representation is similar to the LAOS XML schema. In MOT, Semantic Web notions such as XML, RDF, RDF Schema, LOs and metadata are used. The tutoring courses or modules authored by using MOT can offer intelligent tutoring because of personalization.

SMARTIES is an innovative intelligent tutor authoring system that is aware of learning and instructional theories based on OMNIBUS. Basically OMNIBUS is about “what to learn?” and “how to learn?” (as described at the end of section 6). OMNIBUS provides basic building blocks to SMARTIES for creating instructional learning scenarios in collaboration with the author, i.e. the instructional designer (ID). Decomposition of I_Levent is central to SMARTIES. The I_Levent can be considered an instructional strategy used by the author (ID). This decomposition can be done in two ways. The first way is to explain what to learn and how to learn. The second way is to display examples or content and let the learner to explore learning. The decomposition can be proposed to the author (ID) by the system based on OMNIBUS ontology, and then the author can select I_Levent components from the information available. For example, if a learner needs help, then way1 tells him/her what to learn by suggesting a hypothetical experiment; then, following the constructivist theory of learning, the system suggests an actual interactive experiment. Way2 would suggest

\textsuperscript{29}http://www.ei.sanken.osakau.ac.jp/pub/miz/AIE\textunderscore D07WS\_Invited.pps
\textsuperscript{30}http://www.activemath.org
\textsuperscript{31}http://sliderutor.upmc.edu
\textsuperscript{12}http://herzberg.ca.sandia.gov/jess/
providing related content and examples to display. The author (ID) could use his/her own instructional strategies to create an I_Levent and to produce a learning scenario. The instructional learning scenarios produced by SMARTIES are IMS-LD-compliant and can be connected to Learning Objects, such as virtual lab instruments available through GLOBE.

Table 3 shows a classification of the four systems as per the usage of Semantic Web concepts.

<table>
<thead>
<tr>
<th>ITS/Authoring Systems</th>
<th>Semantic Web Concepts</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActiveMath</td>
<td>RDF/XML Yes Ontology Yes LOs/Metadata Yes</td>
</tr>
<tr>
<td>SlideTutor</td>
<td>Yes</td>
</tr>
<tr>
<td>MOT</td>
<td>Yes</td>
</tr>
<tr>
<td>SMARTIES</td>
<td>Yes Yes Yes</td>
</tr>
</tbody>
</table>

It is clear from the above table that maximum application of Semantic Web concepts has been made by ActiveMath. The base of ontology editor Protégé is in expert systems. SlideTutor is a kind of expert system which provides personalized feedback. The system is very domain-specific, and confined to a small part of the domain, i.e. tutoring of medical diagnosis by using visual slides. Because of this limitation dictated by the needs of the tutoring domain, other Semantic Web concepts such as RDF/XML and LOs/Metadata are not used in that system. As per the intelligent tutor authoring tools, MOT comes from AEHS background and does not show any evidence of the use of ontology; however, there is a potential to use the ontology in MOT. SMARTIES come from AI-ED background, and apply all the concepts of Semantic Web. It is mentioned clearly in [126] that although SMARTIES looks like an expert system, it actually is not: it shows personalization in instructional strategies, but does not explore personalization of the learner by using information about the learner.

8. Conclusion

Reviewing the literature from ITS, AEHS and the AI-ED community, it becomes clear that all the three fields are moving in parallel, but have little interconnection and collaboration. Notions of Semantic Web like RDF/XML, LOs, and Metadata are building blocks that help to use ontology effectively. Student modeling techniques offered by ITS, personalization offered by AEHS, and the modeling of instructional strategies offered by the AI-ED community would add greater intelligence if used in combination with Semantic Web concepts. To combine the work in these three fields and enhance the intelligence of the tutoring systems, effective use of Semantic Web concepts in combination with some AI techniques such as ontology engineering would help. Also, for existing stand-alone AEHS, a way to increase scalability and re-usability would be to use a central ontological server in combination with annotation of LOs and metadata.

Acknowledgement

The research project for which this literature survey work has been done is funded by HP Labs, India and International Institute of Information Technology, Bangalore. We thank both the organizations for extending all their infrastructural support, including library resources.

References


33 http://www.imsglobal.org/learningdesign/lrdsip0.html
34 http://www.globe-info.org/


