JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 30, 835-851 (2014)

DOI:10.1688/JISE.2014.30.3.17

Semantic Web Technologies for Pre-School Cognitive Skills Tutoring System*

MUHAMMAD AZEEM ABBAS, WAN FATIMAH WAN AHMAD
AND KHAIRUL SHAFEE KALID
Computer and Information Sciences Department
Universiti Teknologi Petronas
Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia
E-mail: azeem.abbas@uaar.edu.pk; {fatimhd; khairulshafee_kalid}@petronas.com.my

Current teaching strategies mainly follow instructional or constructive pedagogical models. In instructional model learning, contents are authored for teaching and evaluation of a concept. Whilst in constructive model, a student is exposed to the learning environment that allows him to learn new concepts, and his actions provide an evaluation of his understanding of the concept. Manual authoring of such learning contents not only requires specialized skills, thinking and dedication, but they also take large amount of time. Similarly within instructional environment, authoring any new content that supports the current teaching activity is very difficult. This paper presents an intelligent tutoring system (ITS) that facilitates the authoring of learning contents for both instructional and constructive teaching strategies. Proposed ITS uses semantic web formalism, namely ontology, for knowledge representation. This work shows that ontologies correctly and consistently models the domain knowledge that supports pedagogical activities, software development activities, ITS user friendly interface and provides robust reasoning facility for creation of dynamic contents. The proposed ITS provides authoring facility explicitly for teaching cognitive skills such as classification, relatedness, comparison (sequencing, discrimination, size *etc.*) and reasoning. With the use of ontologies, cognitive skills are translated into concepts/classes and properties/roles with restrictions. Reasoning and inference provided by ontologies are used for dynamic authoring of learning contents. Finally evaluation of proposed ITS is conducted for its correctness and consistency of domain knowledge, usability and efficiency & effectiveness of contents authoring.

Keywords: elementary education, intelligent tutoring systems, semantic web, cognitive skills, architectures for educational technology system, authoring tools and methods

1. INTRODUCTION

The need of a powerful learning environment is highlighted by Smeets [1], where the environment stimulates the required knowledge that a student is expected to master. There should be an open-ended learning environment rather than an environment where only concepts and facts are delivered. In this open-ended [2] learning environment students are engaged in active knowledge construction. Information and communication technologies (ICT) can contribute in creating such open-ended learning environment in several ways. One of its contributions is an educational software or intelligent tutoring system (ITS). Intelligent tutoring system (ITS) is a type of expert systems that has a very effective role in modern education. It facilitates both teachers and students in teaching

Received September 21, 2012; revised April 29, 2013; accepted May 24, 2013.
Communicated by Jan-Jan Wu.
* This work is a part of research project under Exploratory Research Grant Scheme (ERGS) # 15-8200-328. Thanks to Management of Universiti Teknologi Petronas (UTP) for their support.

and learning of some concept of the domain. ITS can also serve as an essential component to homeschooling environment [3]. Homeschooling is a less expensive and increasingly important type of learning environment especially for developing countries or for students who are living in remote areas.

According to Niederhauser and Stoddart [4], there are two types of ITS: skill-based transmission and open-ended constructive. Skill-based ITS is used in instructionist strategies. In instructionist strategies, a teacher defines knowledge and activities to be presented to the students, while students' skills are assessed through learning contents. Instructionist strategies are instructor-centered which makes student's cognitive skills (such as comprehension, application *etc.*) difficult. Authoring such learning contents not only requires specialized skills, thinking and dedication, but they also take a large amount of time. Similarly within instructional environment, authoring any new content that supports the current ongoing teaching activity is very difficult. On the other hand open-ended constructive ITS works under constructivist strategies. Constructivist strategies are student-centered where student is exposed to the learning environment that allows him to construct new knowledge.

The selection and use of ITS has a significant impact on the learning environment. Previous research shows that usage of ITS in general is of skill-based transmission in traditional learning environments [1]. Most of the available systems have limited capabilities with static representation of learning contents. Classical ITS can become potential systems by incorporating dynamic capabilities such as curriculum sequencing (we call it navigation), interactive problem solving and intelligent analysis of student's solution [5]. The reason is, currently open-ended constructive ITSs are very general systems or they are ITS shells (Epilist, Epistmic games, WETAS) that are too generic and weak to be adopted in today's world. A strong limitation in open-ended ITS development is the modeling of endless properties/possibilities of an underlying domain. Student constructs his knowledge by experiencing a new property that is correct and realistic. This new property is either modeled in the system in advance or student came across it through reasoning the domain knowledge. Construction of internal knowledge by a student is the development of generic cognitive skills. This construction of generic cognitive skills can be achieved by dynamic content creation (exploration) or authoring learning contents (instructional methods) in the proposed ITS.

Despite the vast application of Semantic Technologies in different fields of the real world, the semantic aspect of Semantic Technologies has not been fully applied to intelligent tutoring systems, and particularly for developing and delivering reusable learning contents. The research into Semantic Technologies based intelligent tutoring systems suffers from several limitations including high complexity of consistent and complete modeling of domain knowledge, student behavior, profile and interaction, pedagogical activities such as instructional or constructive. Ontology has not also only been developed to provide meaningful structure to syntactic information rather it can be used for modeling of domain knowledge, student model, pedagogical model and reasoning to generate learning contents.

This paper presents the framework and its details for a pre-school cognitive skills tutoring system (ITS). The proposed ITS provides learning content authoring facility through the use of an expert system. Here, in this work, learning contents can be authored through either automated or semi-automated way. In automated way the expert system performs reasoning over domain knowledge and infers new learning contents. The Expert knowledge is represented through the use of Ontologies. Ontologies offer several benefits such as modeling a complete and consistent domain, reasoning robustness, and time and space efficiency. The present work provides a detailed mechanism for knowledge acquisition, update and its maintenance. It addresses authoring of learning contents for both constructivist and instructionist pedagogical strategies used for tutoring cognitive skills.

The rest of the paper is organized as follows. Section 2 defines the scope of our work. Section 3 reviews the previous work. Section 4 presents the ITS framework and details about its major components. In section 5 evaluation of proposed ITS is conducted and results are provided. Finally, Section 6 concludes the paper.

2. INTENDED DOMAIN

The proposed work specifically addresses pre-school children. Pre-school children are divided into two groups; the first group consists of children aged 3 and 4 years old, and the second group consists of children from 5 to 7 years old. Children belonging to the first group, in general, still do not have the skills to read and write; they can only listen, speak and draw lines. Usage of keyboard and mouse is unacceptable at this age, so we reject that a child himself is an active user of the proposed ITS. Instead his teacher or his parents use the ITS for generation of learning contents for him. These contents are to be solved using pencil and paper by the child. Upon completing the solution of the learning contents, the teacher/parents enters the child's solution back into the ITS. The ITS then navigates to the respective learning content based on assessment of the child's solution.

A pilot study was conducted to investigate the curriculum and practices used in pre-school teaching. More specifically to answer the following research questions: (1) What learning contents are used to teach some concept of the domain (in our case pre-school)? (2) How these learning contents are created or adopted by teachers at school or parents at home? A curriculum and teaching practices survey over books and worksheets available in local market or taught at schools was done to investigate: (i) type of learning content *e.g.* conceptualization, recognition, word formation, counting, arithmetic *etc*., (ii) difficulty level, (iii) student age and (iv) presentational style. The study initially involved 10 schools in the Perak state of Malaysia, 5 homes with 1 pre-school student and 3 book publishers, two local publishers and one international publisher. All schools and homes have a personal computer and adequate knowledge of its usage. The study conducted concludes that in general, cognitive skills are problem solving and reasoning skills. Pre-school children undergo the process of developing multiple cognitive skills such as reading-writing, mathematics, memorization, comprehension and development of new knowledge from their current knowledge. Moreover, syllabus/curriculum of pre-school cognitive skills [6] can be generalized to conceptualization, classification and relatedness of the real world.

Conceptualization is formally defining concepts (abstract idea or mental symbols sometimes defined as "unit of knowledge"). For pre-school children all real world objects are concepts. And every concept is learned through its associated properties. Concepts are classified with respect to their specific or some general properties called classification. For example vegetables and fruits, pets and wild animals, size classification (Bus-car-motorcycle, Elephant-rat) *etc*. In relatedness, concepts are related or disjointed to each other with respect to some properties between them. For example Dog eats Bone, Bee produces Honey, Clouds pour Rain, Water extinguishes Fire *etc*.

The learning contents in pre-school education are mainly small scale games, pictorial representations and sounds. The motor activities at this level are drawing lines, circling objects and coloring. Exercise patterns are mainly color or circle an object, match the columns and mark tick or cross. The proposed work provides an authoring facility to teachers and mediators to easily create and conduct learning contents. In our proposed scenario, a teacher or mediator is a facilitator rather than a direct instructor. These learning contents are used as an instrument for practicing and evaluating child's cognitive skills [7].

3. RELATED WORK

The primary need of any ITS is domain models, that is, domain knowledge represented in a way that allows the system to generate individualized learning contents and diagnose student's solutions for further navigation.

Nussbaum *et al.* [8] explained the knowledge structures for the development of an intelligent tutoring system. A knowledge structure is a way to model domain and any activity that can be performed within that domain. Their work covers only basic mathematics problems such as quantity, comparison, geometry and sequencing. Every individual problem is modeled to a separate knowledge structure, which we can say an equivalent to a rule as in any rule based system. Teachers can use these knowledge structures to generate exercises. This exercise generation aspect is similar to our proposed work, where learning contents are also generated by teachers/mediator for the children to solve.

Epilist, an intelligent tutoring system proposed by Ming *et al.* [7, 9], models the domain using sets. A set containing items that are related through a property is called a semantic network. According to the authors, any semantic network is significant and suitable if it is complete and unique. Epilist is a list-making game. The child, as a user of Epilist, classifies the provided items as a list. The classification formed by the child is then mapped with stored expert model (*i.e.* the expert model are also sets created by domain expert) for completeness and uniqueness. We have achieved conceptualization of complete and consistent reality of the domain by using the ontologies.

Epilist focuses on classification, generalization and comparison among all cognitive skills. Classification, generalization and comparison of items are only done for a single previously defined property. It is unable to use implicit or chain properties such as food chain where one animal produces something and other animal eats it. Both Nassbaum *et al*. [8] and Epilist [7, 9] have domain expert within their software model. Domain expert measures the mistakes of a student and consistently generates or reformulates domain knowledge. This consistent reformulation of knowledge structures [8] and semantic networks [7, 9] hinder lowering their development cost. In our case, completely modeled domain knowledge and well-formed reasoner have eliminated domain expert from our framework.

Artificial intelligence provides a major contribution towards knowledge representation or domain modeling. Several approaches such as semantic networks, fuzzy logic, neural networks, case-based reasoning and Bayesian networks have been proposed for knowledge-based systems. Hatzilygeroudis and Prentzas [10] utilized a hybrid knowledge representation approach *i.e.* combining symbolic rules and neurocomputing called *neurule* to model domain knowledge of a Web-based ITS. Student model is formulated by storing facts about user's interactions, exercises attempts, his pedagogical method, difficulty level *etc*. This work annotates learning contents with meta-description as a part of their domain modeling so that it can be provided to student for learning or exercise. Keeping such meta-description is similar to Teaching Material Design Center (TMDC) [11]. TMDC facilitates instructors in designing a new course from available learning contents shared over the internet. Currently these available learning contents are not indexed under concept hierarchy. So TMDC uses domain ontology of a course and associate available learning contents with its concepts. Ontology in TMDC [11] and Learning Objects Repository (LORs) [12] are used only for giving meaningful structure to the information of available learning objects. Another work that utilizes ontologies is EDUCA [13], a software implemented to provide authoring facility to instructors in course designing. The course designing through EDUCA starts with arranging the learning objects in a tree structure *i.e.* similar to TMDC and LORs. An instructor can insert different learning objects such as images, text and audio/video clips to the tree structure nodes. Student's learning style is selected by EDUCA through the use of self-organizing feature maps (SOM) or commonly known as Kohonen neural networks. In case of EDUCA, learning objects arranged in tree structure contains multi-dimensional aspect of a course design. Ontologies or structuring techniques in these works are only used to organize available learning objects whereas the creation of learning contents through reasoning over modeled domain using ontologies brings novelty to the present work.

An important aspect of intelligent tutoring system *i.e.* assessment of a student's solution is done through diagnosis rules [14]. Diagnosis rules take student's behavior (actions that are carried out by a student) and student's belief and goals as an input to diagnose student's maturity of understanding. In the work of Clemente *et al.* [14] student profile information, student behavioral traces, student knowledge state and student learning objectives are all modeled in ontologies. Their work can be considered as a precedent to a part of the work presented here.

Authoring systems are designed to reduce the time necessary for pedagogical activities, and may also make it possible for non-programmers and even teachers to develop ITSs for their courses. Some problems that may occur while designing a software for children are identified by Karuovic & Radosav [15]. These problems are limited usage of electronic devices and difficulties while interviewing the end users about their demand. Interface designer must understand the child's work tasks, his mental system when performing some task and tools that support his activities. In our work we used presentational templates for presenting learning contents to a child. These presentational templates are commonly used by teachers and curriculum designers.

An application named Self Learner Tutor (SLT) was proposed for preschool children [16]. SLT is based on Resource Description Framework (RDF) a component of semantic web technologies. Present work uses Ontology that is a component of semantic web but it is at a higher level in semantic web layered architecture than RDF. Ontology support complex and dense knowledge modeling as compared to RDF model. Constraint Acquisition System (CAS) [17] has used a similar approach like Epilist for domain modeling using ontologies. Epilist used sets for both expert model (the correct answer) and the student model (the solution). The student's model is then validated with the expert model. Similarly, in CAS author of a learning content provides both problem statement and a set of correct solutions. A solution is made from instances of the ontology. Different solutions are made by rearranging the author's provided solution. CAS experimented basic algebraic equations as their domain, but the ontology of algebraic equations is not provided.

Navigation among learning contents is an important component of our proposed ITS. This navigation is dependent on the assessment results of the child's solutions. Here in the proposed ITS the curriculum sequencing is controlled by reasoning engine and content authoring rules.

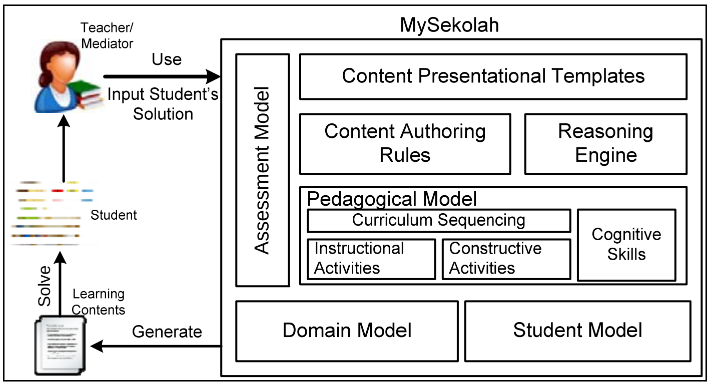


Fig. 1. Conceptual framework of proposed ITS.

4. MYSEKOLAH FRAMEWORK

This section presents the framework and its details for a pre-school cognitive skills tutoring system (ITS). The proposed ITS provides learning content authoring facility through the use of an expert system. Here, in this work, learning contents can be authored through either automated or semi-automated way. In automated way the expert system performs reasoning over domain knowledge and infers new learning contents. The Expert knowledge is represented through the use of Ontologies. Moreover, it presents authoring of the learning contents for both constructivist and instructionist pedagogical strategies. Fig. 1 shows the conceptual framework of the proposed ITS named MySekolah (Malaysian School) along with its potential users. Details and the methodological steps underlying major components of the conceptual framework are provided in the following subsections.

4.1 Users of the System

Teachers or mediators are the primary and active users of this system. They use this system to author learning contents through its user friendly interface. Also they input the solutions solved by the students against their profile. Teachers or mediators start by selecting a child's profile and cognitive skill from the list for which they want to generate the learning content.

As discussed in Section 3, MySekolah has eliminated domain expert from the system. With complete and consistent domain modeling through ontologies and reasoning through reasoners, the system provides dynamic construction and evaluation of learning contents without domain expert or expert model. In the next section, the learning contents generated from MySekolah shows the dynamic behavior of the system.

Pre-schoolers are the end user of this system but they are not active users. Learning contents generated from MySekolah are provided to the pre-school children by their teachers or parents. The children solve the learning contents on paper.

Table 1. WordNet senses and hypernyms.

Sense 1	Chicken, poulet (the flesh of a chicken used for food) => Poultry => Bird, Fowl => Meat => Food, Solid Food => Solid => Substance, Matter => Physical entity => Entity
Sense 2	Chicken, gallus (a domestic fowl bred for flesh or eggs) => Domestic fowl => Gallinaceous bird => Bird => Vertebrate => Chordate => Animal => Organism => Living Thing => Object => Physical Entity => Entity

4.2 Domain Model

In recent years the development of ontologies-explicit formal specifications of the concepts in the domain and relations among them [18] has come from the realm of Artificial-Intelligence laboratories to the desktops of domain experts. Ontologies possess all those characteristics that any ITS need to have such as domain modeling, reusability and reasoning ability. The primary and most important goal of ontologies is it includes human as well as machine-interpretable definitions of basic concepts of the domain and relations among them. Ontologies provides a way to model both top-down and bottom-up [18] knowledge construction. Ontology is a more enhanced and well formulated form of "conceptual graphs" [19]. It natively possesses all classification skills identified by Ming &Quek [7] *i.e.* matching, cross classification, serialization, equivalence and hierarchical tasks.

All concepts are formally defined as classes in ontologies [18]. Similarly the hierarchical structure (superclass-subclass) provides single as well as multiple classifications of concepts. For example, we can divide meat concept into red and white meats. Concepts/classes in ontologies are connected to each other with properties, which provide relatedness among concepts. Properties of classes and their instances are described by slots. For example Hen is a bird and it produces eggs.

We populated our ontology from WordNet 3.0 [20]. WordNet 3.0 is a large lexical database of English with over 155,287 words. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms called synsets, each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations that create a semantic network. The primary formation in WordNet is the collections of synsets. Synsets are connected to other synsets through semantic relations. Hypernyms, hyponyms, holonyms, meronyms are some relations between nouns synsets. Verbs synsets contain hypernyms, troponyms, entailment and coordinated terms relations. Adjectives have related nouns, similar to participle of verb relations. These semantic relations are applied to collections of synsets those share a common meaning or mainly synonyms, but WordNet also contain words connected through lexical relations such as antonyms.

Pre-school learning contents mainly contain nouns only. So for that reason we initially extracted and utilized only nouns from WordNet ontology. Classes are made according to the unique strings and their senses. For class hierarchy we used WordNet hypernyms relations. Table 1 shows two senses of the word chicken and its hypernyms relations in the hierarchical structure. Sense 1 of word chicken shown in Table 1 formulates our food ontology where sense 2 becomes the part of our animal ontology. Along with animal and food ontologies we have created several other ontologies such as vehicles, cloths, professions, weather, body parts, artifacts *etc.*

Properties in WordNet are only structural properties that are used to define structure of synsets. These properties do not provide semantic relationship/meanings among concepts. Verbs from WordNet are used to define properties among concepts. Symmetric, inverse and transitive property types are carefully inserted through the use of synonyms and hypernyem relations of verbs *e.g. eats* and *consume* are symmetric properties. Similarly individuals are inserted against every concept of populated ontologies. Since learning contents at pre-school level are mainly pictorial representations, small games and puzzles, so images that represent a concept are associated respectively with the concepts.

4.3 Pedagogical Model

Educationists and psychologists mainly follow two teaching strategies or models *i.e.* Instructional and Constructive. In instructional strategies learning items are created and delivered by the teacher. Similarly evaluation material against delivered contents is formulated by teacher too. In contrast, constructive strategies expose students to the learning environment where they learn through exploration. In this case evaluation is done through monitoring student's actions against a previously defined expert model that contains all possible solutions of the current problem under consideration. MySekolah addresses both instructional and constructive teaching strategies in term of content authoring.

In case of constructive pedagogical strategies, learning contents are dynamically created by ITS through the use of the reasoner applied over domain knowledge, patterns of previously created learning contents and user profile. A content created by the teacher is stored as a rule. All rules contain some patterns that can be used for creation of new contents by the reasoning engine. A content "*What animal eats*?" created by teacher contains individuals from animal and food ontologies associated with eats property. The reasoning engine has created content "*What animal produce*?" dynamically by inferring the stored rule of "*What animal eats*?" Similarly, a combination of both contents creates the content "*Food Chain*". The content "*Pet and Wild Animals*" is created dynamically through the use of disjoint constraint information between the concepts of pet and wild animal. Similarly, "*Ancestors and Descendants*" content is created through hierarchical classification information (superclass-subclass).

4.3.1 Cognitive skills

The cognitive skills implemented or focused on by MySekolah include conceptualization, classification, relatedness and comparison. Some brief details are given in section 3 about these cognitive skills. Here are few examples to show these cognitive skills within MySekolah.

(1) Classification: Through the use of ontology's hierarchical structure of super and subclass relationship, any specialized concepts can be classified by their super or generalized concept/class. For example the learning content of Fruits & Vegetables classifies the concepts of plant ontology under fruits and vegetables. Similarly fruits and vegetables are classified with respect to their taste *i.e.* sweet, sour, bitter *etc.* Also vegetables that grow underground and those that grow aboveground are classified with respect to their production method. Classification in MySekolah is also done through the use of properties defined among concepts.

(2) Relatedness: The most prompting results our ITS has shown are under Relatedness Skills. Learning contents for relatedness skills are created through the use of concept hierarchy, properties between concepts, semantic information and restrictions. Concepts are related to each other within an ontology, but concepts are also related to several other concepts in several other ontologies which makes our domain model heavily dense for relatedness skills learning contents. One of an inspiring content created through the use of reasoning is Umbrella with Rain and Snow Shoes with Winter. Rain and Winter are concept in weather ontology; Umbrella comes from the artifacts ontology and Snow Shoes is a part of cloths ontology. A large number of learning contents for relatedness cognitive skills are created manually with the use of MySekolah interface.

(3) Comparison: In MySekolah, learning contents related to comparison cognitive skills contain both structural and semantic comparisons among objects of the real world. The previous work proposed by Nussbaum *et al.* [8] contains structural comparison *i.e.* a single object of different sizes in random order is shown and a student is asked to find the smallest or largest among them. In our case, we consider semantic comparison among objects such as compare Elephant, Horse, Cheetah and Donkey. The property of comparison among them can be their Running Speed or their Living Place or Carrying Weight. Discrimination (belonging and not belonging) is a type of comparison cognitive skills. MySekolah creates such discrimination contents with the use of disjoint relationship, individuals from different ontologies, constraints and restrictions and properties grouping.

Table 2. Axioms.

$Animal \sqsubseteq Thing$	$Dog \sqsubseteq Carnivorous \sqcap \exists livesIN.Farm$	$Carnivorous \sqsubseteq (Animal) \sqcap \forall eats.Animal$
$Place \sqsubseteq Thing$	$Lion \sqsubseteq Carnivorous \sqcap \exists livesIN.Jungle$	$Herbivorous \sqsubseteq (Animal) \sqcap \forall eats.Plants$
$Farm \sqsubseteq Place$	$Tiger \sqsubseteq Carnivorous \sqcap \exists livesIN.Jungle$	$Elephant \sqsubseteq Herbivorous \sqcap \exists livesIN.Jungle$
$Jungle \sqsubseteq Place$	$Cow \sqsubseteq Herbivorous \sqcap \exists livesIN.Farm$	$WildAnimal \sqsubseteq (Animal) \sqcap \forall livesIN.Jungle$
$Bird \sqsubseteq Animal$	$Parrot \sqsubseteq Bird \sqcap \exists livesIN.Farm$	$Pet \sqsubseteq (Animal) \sqcap \forall livesIN.Farm$

4.4 Content Authoring Rules and Reasoning

One of the main reasons for building an ontology-based application is to use a reasoner to derive additional truths about the domain. Moreover reasoning can be used to ensure the quality of an ontology. It can be applied to different phases of ontology development. For example it can be used to evaluate any contradiction among concepts during ontology design phase. Reasoning can also derive implicit relations. This derivation of implicit relations provides support to construction of constructive pedagogical activities. For reasoning, a well-designed and well-defined ontology language (often a description language) is required. Description Logics (DLs) are a family of knowledge representation languages that can be used for defining domain semantics and performing powerful reasoning over modeled semantics. Important notions of the domain are described by concept descriptions. Statements or expressions of the underlying domain are expressed through the use of concepts, roles and individuals. DLs are dominant over their predecessors (semantic networks and frames) due to their formal, logic-based semantics and inference potential.

Axioms are the building blocks of the description logics. Axioms are logical statements that relate concepts and roles together. An axiom is shown in each line of the above example. An axiom is conjunction (∩) or disjunction (∪) of concepts and role constraints with existential restriction (∃) or the value restriction (∀). For example, Pet is a subclass of Animal but with a constraint that at least some of them live in a Farm. Similarly, Carnivores are Animal that only eats other animals. In general the knowledge in description logic is modeled into terminological (TBox) and assertional (ABox) parts. Axioms in Table 2 shows the terminological part (or simply terminology) describing the structure of the terms of the domain. While the assertional part (not shown here) contains individuals or instances of the domain concepts and their asserted properties, using definitions of its terminology. The Fact++ is used for reasoning to create learning contents. Fact++ is a powerful reasoner that is able to infer logical consequences from a set of asserted facts or axioms. Querying these axioms for PET give Cow, Dog, Cat, Parrot and Lion, Tiger, Elephant for Wild Animal.

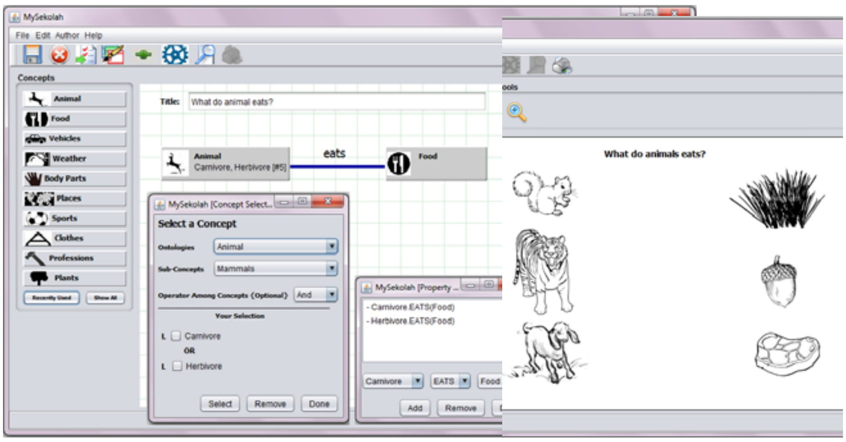


Fig. 2. (Left) Interface of MySekolah (Right) Learning content preview.

Jena framework is used for our application development. Jena is a Java framework for building Semantic Web applications. Jena provides a collection of tools and Java libraries to help develop semantic web and linked-data apps, tools and servers. Along with Resource Description Framework (RDF) Jena provides an API to manipulate OWL ontologies as well. A rule based inference engine and query engine compliant with SPARQL are part of Jena Framework too.

A prototype has been built for MySekolah. The user interface module provides the user interface of the content-authoring to the teacher or parents. Through this interface, the teacher selects the concepts for their content authoring and associates the selected concepts with each other through the use of properties. Properties are also selected from previously defined property hierarchy between any two selected concepts.

Fig. 2 shows the content authoring interface of MySekolah. The left side of the interface displays the domain ontologies (concepts) items that are used in the learning contents. Clicking on an ontology button would provide the concept selection in a hierarchical view as shown in Fig. 2. The teacher is able to select a concept from the concept hierarchy even though he/she is unfamiliar with the concept before.

The center area of the interface is the workspace where the teacher creates the learning content. Any learning content can be saved by save option. All stored learning contents are objects having attributes such as concepts selection criteria, properties defined among them, title of the learning content, categories of cognitive skills, difficulty level, presentational style and student identification. With evaluate button, the teacher can run an evaluation over his created learning content for any inconsistency or contradiction. Preview button provides the preview (shown in Fig. 2 (right)) of the learning content that will be later delivered to the child.

5. EVALUATION AND RESULTS

We evaluated MySekolah from three perspectives *i.e.* correctness, efficiency and effectiveness. First the correctness and consistency of the domain model is evaluated. Evaluation of the domain model is done through the use of Protégé and Fact++ reasoner. Any inconsistencies or contradictions identified by the reasoner were eliminated from the domain ontologies. Secondly the evaluation of MySekolah usability was conducted. A likert scale is used for usability evaluation. Lastly the contents authored or generated from MySekolah were evaluated. All contents produced were manually evaluated by the teacher at the end of the operational session. The sample is fourteen kindergarten female teachers and twenty one students (age between 4 and 6) from five schools situated in rural and urban areas of Perak state of Malaysia. The demographic difference among these teachers was the primary reason for their selection. Among teachers most of them have degree or diploma in education and belong to urban areas. We visited these teachers at their schools. The experiment session continues for three hours starting with an introduction and basic training of MySekolah prototype for the first twenty minutes. MySekolah was installed on a laptop with an external mouse and keyboard for evaluation experiments. All of the teachers know the usage of personal computer and specially word processing software.

The efficiency of MySekolah is evaluated by comparing the time difference between the content creation activity done through traditional way and with the help of MySekolah. Therefore, the tradition way (teaching practices and curriculum used) for cognitive skills tutoring was investigated through the pilot study (discussed in Section 2). This pilot study leads us to following conclusions: that the traditional practices focus on mere delivery of pre-authored learning contents. These learning contents are mainly procured from text books and Internet sources. Similarly these contents are reproduced by creating copies for repetitive use or conducting the exam. A small number of teachers claimed that occasionally they create contents themselves. For creation of learning content teachers take resources such as images from the Internet. The theme or concept of the learning content is mainly adopted from books and Internet sources with a slightly altered form.

Table 3. Contents used for evaluation.

Classification	*Relatedness*	*Comparison*
What animal eats	Cloths and Weathers	Seating Capacity
Carnivores & Herbivores	Profession and Vehicles	Speed Comparison
Fruit & Vegetables	Shape of the body parts	Body Size
Soft and Hard	Things that Animal Produce	Containers Size
Ancestors	Random Objects	Longer and Shorter

The usage of MySekolah is described earlier in Section 4.4. The initial content of any sequence of contents is authored by the teachers/mediators, rest of them generated by MySekolah itself. Also teachers do intervention when any erroneous content is generated by MySekolah or giving appropriate labels to the created contents. While for constructive pedagogical model MySekolah generate new contents on a click of a button based on the user profile (containing his previous results), available contents (authoring rules) and reasoning over domain knowledge (Section 4.3). As discussed above these auto-generated contents are aligned with curriculum sequence by the use of contents authoring rules and reasoning engine. Later in this section we provide evaluation of these auto-generated contents manually performed by the teachers.

For the reason of efficiency comparison, a benchmark (set of fifteen learning contents) was established based on our pilot study. These benchmark contents were drawn from available books, curriculum guidelines and Internet sources (refer to survey in Section 2) that are common and widely used by the teachers. Details of all fifteen contents were provided to the participants and consent of agreement was taken before conducting the experiment. We started our experiment by requesting every teacher to prepare a set of fifteen contents (five against each cognitive skill, shown in Table 3) by using their traditional way first and then later the same with the MySekolah.

MySekolah required a substantially shorter time for creation of learning content than the traditional approach. In traditional approach teacher has to build a mind map of learning content before creating it. In case of classification content, first she has to separate elements herself in required classification similarly brining things related to each other through some relation is more difficult than classification. For comparison identifying comparison parameter and then inserting different objects require more time. Searching images from Internet adds up more time. Finally the creation of presentation template (match the column, circle an object, grid layout, randomly spread *etc.*) makes the job time consuming. Through our experiment we analyzed that creating a same learning content with different objects require more time. Specially creating a second level content (with increased difficulty) is much more time consuming then the previous. Table 4 shows the mean time spent to develop these same learning contents (difficulty level 1) with use of the MySekolah and the traditional way.

Table 4. Time to create pre-defined contents (Mean Time mints).

	Classification	*Relatedness*	*Comparison*
Traditional Way	13.94	17.56	22.33
MySekolah	1.33	1.46	1.89

In addition to defined contents, the participants are requested to create contents of their own choice. Participants created all those contents which they already experienced in the past. The time difference for these contents is more or less same as shown in Table 4. The time difference is still pretty high (Table 4) but a clear advantage of MySekolah was observed when we asked participants to create level 2 contents or a content that they had never seen before. For that, they spent large amount of time visualizing internally the content they want to author. But with the case of MySekolah, it generates contents automatically by bringing new concepts that are associated with the concepts of previous contents. For example MySekolah created "*Natural and Cooked Foods*" content based on *Food* concept selected in "*What animals eats?*" content and "*Mammal, Reptile, Fish*" content from *Animal* concept. Similarly "*Soft and Hard Fruits*", "*Identify Tastes of Fruits*" and "*Fruits that Grow on Trees*" from previously authored content "*Fruits and Vegetables*".

Table 5. Dynamic content generated from MySekolah.

Classification	*Relatedness*	*Comparison*	*Total*
18 (33.34%)	31 (57.40%)	5 (9.26%)	54

Lastly the effectiveness of MySekolah was validated by evaluating the auto-generated contents by MySekolah against a single user profile. A user profile was created containing initially all fifteen contents of the benchmark as a starting point. Solution of each content was entered randomly (correct, incorrect, partial correct) in MySekolah so that it progress further. MySekolah progress is based on results of student's solution. Since there is no sophisticated assessment model yet in MySekolah but the use of semantic knowledge modeled within ontologies is working. This activity of auto-generation of contents was stopped at a point when there are at least two contents for every child participant to solve. Table 5 shows total number of dynamically created contents. Classification contents mainly generated through classification defined among concepts of onotolgies. Relatedness contents are generated through either direct or indirect relationship present between concepts or inferred by the reasoner. Similarly comparison contents contain all those objects that sub-class of or associated with measureable concept in cognitive skills ontology.

Table 6. Dynamic content generated for relatedness cognitive skill.

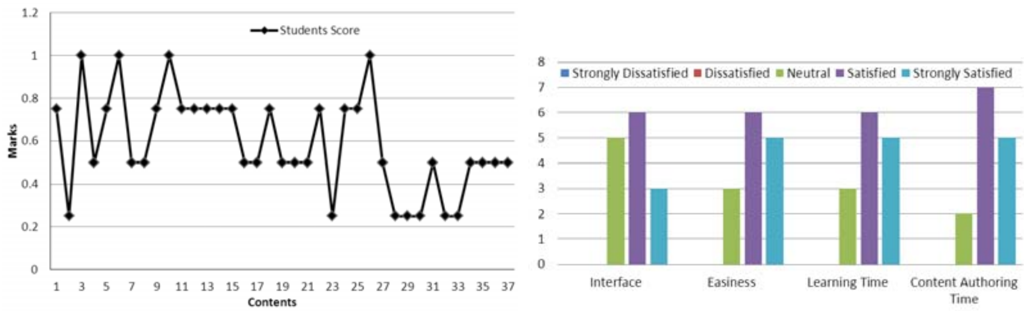
Direct relationship	Indirect Relationship	Inferred	Total
22 (70.96%)	6 (19.35%)	3 (9.68%)	31

Table 6 shows details of the relatedness contents. Relatedness contents are the highest in number among all dynamically created contents because generating contents using explicitly and implicitly defined relationship among concepts is the easiest job for MySekolah. Most of the relatedness contents (70.96%) were generated from the use of direct relationships (explicitly defined) among concepts. The contents generated through indirect relationships (symmetric, inverse, transitive *etc*.) and inferred by the reasoner engine provide a positive indication toward MySekolah potential benefits. "*Food Chain*" is an example of relatedness content generated by MySekolah using inference (can be in example Section 4.2). Finally an evaluation of dynamically created contents was performed by teachers to validate their correctness.

Table 7. Dynamic contents.

Correct	Erroneous	Repetitive	Difficulty Too High
37 (68.51 %)	2 (3.70%)	8 (14.82%)	7(12.96%)

The dynamic contents generated from MySekolah are separated in four categories shown in Table 7. A small portion of dynamic contents are generated repeatedly (14.82%) because MySekolah also uses previously stored or authored contents for generation of the new contents. Also repetition is sensitive to incorrect or partially correct solution of a student. Since in this experiment we entered solutions randomly but the similar repetition was observed in other in-house experiments where all solutions entered as correctly solved. The debate on these aspects of MySekolah is part of assessment model that is our future work. As shown in Table 7 some contents (3.70%) have erroneous objects e.g. in one content a *Jug* and a *Table* are shown under relatedness skill. Some contents generated are marked as high difficulty contents (12.96%) for children between 4 and 6 years age. The reason for these high difficulty contents is that the core ontologies were populated from WordNet 3.0 [20] that contains several terminologies that are difficult to comprehend by a child of age 4 to 6.



(a) (b)

Fig. 3. (a) Students Score (n = 21); (b) MySekolah usability evaluation.

The remaining correct (68.51%) contents (Table 7) are distributed among students to solve. Every student is given 3 or 4 contents so that a single content is at least solved by two students. A grading scale is used for quantitative analysis. A content graded 1 if correctly solved, 0 for incorrect answer and 0.5 for partially correct solution. Fig. 3 (a) shows cumulative grade score of 21 students against 37 contents. The final mean score (0.53) of student solution is not closer to maximum grade (*i.e.* 1) because students have not seen or taught about several contents before. Only four contents are correctly solved by every participant. Similarly a single content is at least partially solved by children that indicate the acceptability of these auto-generated contents and deficiency of cognitive skill teaching in current teaching strategies.

Finally the MySekolah is evaluated itself for its usability through post-test questionnaire. Frequency analysis was used to evaluate the level of agreement and disagreement of teachers on eight Likert type questions adopted from Rubin J. & D. Chisnell [21] for usability testing. Fig. 3 (b) shows the degree of acceptance of MySekolah. The graph depicts participants agreement is mostly toward “Satisfied” or “Strongly Satisfied” with respect to software interface, easiness to generate learning content manually and automatically, time required to learn MySekolah and time required to author learning contents.

6. CONCLUSION

In this paper, we have presented the framework and described the functionality of an intelligent tutoring system (MySekolah) for authoring learning contents to teach or evaluate pre-school cognitive skills. MySekolah uses Semantic Web formalism, namely ontology, for knowledge representation. Contents can be authored manually or generated dynamically by the expert system. The system’s pedagogical decisions are made by expert system using ontologies (OWL-DL) as domain knowledge and its corresponding inference mechanism. The use of Semantic Web technologies instead of knowledge structures or fuzzy sets approaches offers a number of advantages. Ontologies models the correct and consistent domain knowledge that supports pedagogical activities, software development activities, ITS user friendly non-programmer interface and provides robust reasoning facility for creation of dynamic contents. Additionally ontologies are highly supportive for incremental updates. Existing work used ontologies only for giving meaningful and common understandable structure to the information of available learning objects. In contrast, creation of learning contents through the use of ontologies brings novelty to our work. Our experiment results show that the authoring procedure is feasible and dynamic with the use of MySekolah that makes the cognitive skills tutoring more effective.

REFERENCES

1. E. Smeets, “Does ICT contribute to powerful learning environments in primary education” *Computers and Education*, Vol. 44, 2005, pp. 343-355.
2. R. Hermans, J. Tondeur, J. van Braak, and M. Valcke, “The impact of primary school teachers’ educational beliefs on the classroom use of computers.” *Computers*

and Education, Vol. 51, 2008, pp. 1499-1509.
3. A. A. Aburas and R. Shreef, "On-line homeschooling learning management system using intelligent tutoring system," in *Proceedings of International Symposium on Information Technology*, Vol. 2, 2010, pp. 967-971.
4. D. S. Niederhauser and T. Stoddart, "Teachers' instructional perspectives and use of educational software," *Teaching and Teacher Education*, Vol. 17, 2001, pp. 15-31.
5. S. Stankov, M. Rosić, B. Žitko, and A. Grubišić, "TEx-Sys model for building intelligent tutoring systems," C*omputers and Education*, Vol. 51, 2008, pp. 1017-1036.
6. MOEM, "National pre-school curriculum ministry of education Malaysia," http://www.moe.gov.my/bpk/sp_hsp/pra/sp_prasekolah.pdf, 2001.
7. G. M. Goh and C. Quek, "EpiList: An intelligent tutoring system shell for implicit development of generic cognitive skills that support bottom-up knowledge construction," *IEEE Transactions on Systems*, *Man and Cybernetics*, *Part A: Systems and Humans*, Vol. 37, 2007, pp. 58-71.
8. M. Nussbaum, R. Rosas, I. Peirano, and F. Cárdenas, "Development of intelligent tutoring systems using knowledge structures." *Computers and Education*, Vol. 36, 2001, pp. 15-32.
9. G. Ming, Q. Chai, and D. L. Maskell, "EpiList II: Closing the loop in the development of generic cognitive skills," *IEEE Transactions on Systems*, *Man and Cybernetics, Part A: Systems and Humans*, Vol. 40, 2010, pp. 676-685.
10. I. Hatzilygeroudis and J. Prentzas, "Using a hybrid rule-based approach in developing an intelligent tutoring system with knowledge acquisition and update capabilities." *Expert Systems with Applications*, Vol. 26, 2004, pp. 477-492.
11. H.-C. Wang and C.-W. Hsu, "Teaching-material design center: An ontology-based system for customizing reusable e-materials," *Computers and Education*, Vol. 46, 2006, pp. 458-470.
12. P. Raju and V. Ahmed, "Enabling technologies for developing next-generation learning object repository for construction," *Automation in Construction*, Vol. 22, 2012, pp. 247-257.
13. R. Z. Cabada, M. L. B. Estrada, and C. A. R. García, "EDUCA: A web 2.0 authoring tool for developing adaptive and intelligent tutoring systems using a Kohonen network," *Expert Systems with Applications*, Vol. 38, 2011, pp. 9522-9529.
14. J. Clemente, J. Ramírez, and A. de Antonio, "A proposal for student modeling based on ontologies and diagnosis rules," *Expert Systems with Applications*, Vol. 38, 2011, pp. 8066-8078.
15. D. Karuovic and D. Radosav, "Educational software for preschool children," in *Proceedings of the 7th International Symposium on Intelligent Systems and Informatics*, 2009, pp. 291-295.
16. M. A. Abbas, W. F. W. Ahmad, and K. S. Kalid, "Resource description framework based intelligent tutoring system," in *Proceedings of International Conference on Computer and Information Science*, Vol. 1, 2012, pp. 324-328.
17. P. Suraweera, A. Mitrovic, and B. Martin, "A knowledge acquisition system for constraint-based intelligent tutoring systems," in *Proceedings of Conference on Artificial Intelligence in Education: Supporting Learning through Intelligent and Socially Informed Technology*, 2005, pp. 638-645.
18. S. Staab and R. Studer, *Handbook on Ontologies*, 2nd ed., Vol. XIX, 2009.

19. R. Stevens, C. A. Goble, and S. Bechhofer, "Ontology-based knowledge representation for bioinformatics," *Briefings in Bioinformatics*, Vol. 1, 2000, pp. 398-414.
20. G. A. Miller, C. Fellbaum, R. Tengi, P. Wakefield, H. Langone, and B. R. Haskell, "WordNet: An online lexical database," 2012.
21. J. Rubin and D. Chisnell, *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, 2nd ed., John Wiley & Sons, NY, 2008.

Muhammad Azeem Abbas received his MS degree in Computer Science from Mohammad Ali Jinnah University Islamabad, Pakistan in 2010. He is associated with PMAS-Arid Agriculture University Rawalpindi, Pakistan in Institute of Information Technology. Currently he is a Ph.D. candidate in Computer and Information Sciences (CIS) Department, Universiti Teknologi Petronas (UTP), Malaysia. His interest researches are intelligent tutoring systems and semantic web applications.

Wan Fatimah Wan Ahmad received her B.A. and M.A. degrees in Mathematics from California State University, Long Beach, California U.S.A. in 1985 and 1987. She completed her Ph.D. in Information System from Universiti Kebangsaan Malaysia in 2004. She is currently an Associate Professor in the Computer and Information Sciences Department at Universiti Teknologi Petronas (UTP). Her main interests are in the areas of mathematics education, educational technology, e-learning, mobile learning and multimedia.

Khairul Shafee Kalid received his Bachelor of Business Administration, Western Michigan University and Masters of Information Technology from Queensland University of Technology, Australia. He is currently a Lecturer in the Computer and Information Sciences at UTP. His interest research is knowledge management.