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

MUHAMMAD AZEEM ABBAS, WAN FATIMAH WAN AHMAD, KHAIRUL SHAFAEE KALID

Computer & 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: architectures for educational technology system, authoring tools and methods; elementary education, intelligent tutoring systems

1. INTRODUCTION

The need of a powerful learning environment is highlighted by Smeets [28], 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 [11] 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 and learning of some concept of the domain. ITS can also serve as an essential component to homeschooling environment [2]. Homeschooling is a less expensive and in-
creasingly important type of learning environment especially for developing countries or for students who are living in remote areas.

According to Niederhauser and Stoddart [20], there are two types of ITS: skill-based transmission and open-ended constructive. Skill-based ITS is used in instructionist strategies [23]. 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) [14] 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 [28, 29]. 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 [31]. The reason is, currently open-ended constructive ITSs are very general systems or they are ITS shells (Epilist, Epistemic 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
The rest of the paper is organized as follows. Section 2 defines the scope of our work. Section 3 reviews the previous work in this field and critically analyzes it for domain modeling approaches, cognitive skills and its application to the real environment. Section 4 presents the ITS framework and subsequently provides 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 (Children World Publication Sdn Bhd and Honey Bee Publishers Sdn Bhd) and one international publisher (Oxford University Press). 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 [19] 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.

**Classification:** concepts are classified with respect to their specific or some general properties. For example vegetables and fruits, pets and wild animals, size classification (Bus-car-motorcycle, Elephant-rat) etc.

**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. Some survey results have shown that there is no distinction among subjects (science, mathematics etc) at pre-school [15].

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 understanding, including conceptualization, classification and relatedness along with generalization and comparison of Epilist [8].

### 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. [22] 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.

For any ITS, the conceptualization of a complete and consistent underlying domain is the first and difficult task for its development [22]. Epilist, an intelligent tutoring system proposed by Ming et al. [7,8], 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. Any semantic network is complete if it correctly classifies all items e.g. group of items according to male and female are complete. Similarly, animals grouped according to their habitat of land are complete but not unique as some animal possess multiple properties. Different properties are used to make such classification complete and unique; for example animals are grouped according to mammals, amphibians, birds and fishes are
complete and also unique since the defined property over amphibians is unique itself. 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.

Ontology development process is a tedious and sometimes cumbersome activity. For that reason, mostly researcher relies on domain experts to create these ontologies. A possible solution to overcome this limitation is to reuse the available ontologies. Because ontologies are well known for their reusability. Reusability of ontologies is a decision-making activity that require in depth knowledge of the previously created ontologies and the ontology to be created. The present work utilizes WordNet 3.0 [17] a large lexical database of English for development of ontologies required for domain modeling. OWL-VisMod [6] is a tool for developing ontological engineering based on visual conceptual modeling that support this decision-making activity.

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. [22] and Epilist [7,8] have domain expert within their software model. Domain expert measures the mistakes of a student and consistently generates or reformulates domain knowledge. According to Schneidewind [27] the non-reutilization design of typical artificial intelligence software makes them costly. The knowledge structures [22] and semantic networks [7,8] consistent reformulation hinders lowering their development cost. In our case, completely modeled domain knowledge and well-formed reasoner will eliminate 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. A neurule formed from an empirical data is compared to an equivalent fact to infer a student’s skill related to any given exercise. 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) [34]. 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 [34] and Learning Objects Repository (LORs) [25] are used only for giving meaningful structure to the information of available learning objects. Another work that utilizes ontologies is EDUCA [4], 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 [13]. Kohonen networks is a computational approach for analyzing and visualizing high-dimensional data. 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. modeling granularity of skills was highlighted by Fenget et al. [18]. They investigated that any problem at its finer granularity is composed of several different problems or scaffolding questions. Measurement of any student’s understanding is not correct for many cases if only solution to the original problem is measured; rather evaluating scaffolding questions individually give accurate results. The cognitive skills being considered in this work are atomic in nature that they cannot be decomposed into sub-skills. A statistical model mixed-effect logistic regression model is used for assessment of student’s solutions [18]. Similarly assessment of a student’s solution is done through diagnosis rules [5]. 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. [5] 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 [12]. 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.

A peer-tutoring system for mathematics named G-Math was proposed [16]. G-Math is a “game-like” environment that serves as activity manager for teachers and peer-learning environment for students. A teacher with G-Math can float basic arithmetic problems to be solved by students in a collaborative environment. Solution of a student can also serves as guidance to another student. An application named Self Learner Tutor (SLT) was proposed for preschool children [1]. 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) [33] 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. Any child exposed to new unseen content may not understand it internally, so according to Goswami [9] the content navigation should move in order to Zone of Proximal Development (ZPD) to achieve effective results. Keeping constructivist learning activities aligned with ZPD are quite difficult as it may lead pedagogical activities at a random. Here in the proposed ITS the curriculum sequencing is controlled by reasoning engine and content authoring rules.

![Conceptual framework of proposed ITS](image)

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. 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. Moreover, it presents authoring of the learning contents for both constructivist and instructionist pedagogical strategies. Figure 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.

Previous systems such as Nussbaum et al. and Epilist [7,8,22] have domain expert as an actor in their system. The domain expert evaluates and constructs new knowledge based on students’ solutions to make sure that the ITS produces new learning contents for the learners. 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. 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.

4.2 Domain Model

In recent years the development of ontologies-explicit formal specifications of the concepts in the domain and relations among them [30] has come from the realm of Artificial-Intelligence laboratories to the desktops of domain experts [21]. 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 [9] knowledge construction.

The Artificial-Intelligence literature contains many definitions of ontology; many of these contradict one another. For the purpose of this proposal, an ontology is a formal
explicit description of concepts in a domain of discourse (classes (sometimes called concepts)), properties of each concept describing various features and attributes of the concept (slots (sometimes called roles or properties)), and restrictions on slots (facets (sometimes called role restrictions)). Ontology together with a set of individual instances of classes constitutes a knowledge base. In reality, there is a fine line where the ontology ends and the knowledge base begins [21]. Ontology is a more enhanced and well formulated form of “conceptual graphs” [32]. It natively possesses all classification skills identified by Ming & Quek [8] i.e. matching, cross classification, serialization, equivalence and hierarchical tasks.

Ontologies natively provides solution to all requirements for modeling of our underlying domain i.e. conceptualization, classification and relatedness. Concepts are described by classes in ontologies [3]. For example, a class of birds represents all birds. Specific birds are instances of this class. A class can have subclasses that represent concepts that are more specific than the superclass. For example, we can divide meat concept into red and white meats. Properties of classes and their instances are described by slots. For example Hen is a bird and it produces eggs. In this example there are two slots: the slot isA with the value bird and the slot Produce with value egg. In this case, all the instances of Hen have slots isA and Produce.

All concepts are formally defined as classes in ontologies. Similarly the hierarchical structure (superclass-subclass) provides single as well as multiple classifications of concepts. Concepts/classes in ontologies are connected to each other with properties, which provide relatedness among concepts. In ontologies, concepts or classes at top level hierarchy are more generalized than the lower ones such as Animal. These generalized concepts are split into more specialized concepts at lower levels in hierarchy such as Vertebrates and Invertebrates. This splitting forms top-down knowledge construction.

Figure 3: Snapshot of food ontology
Similarly formulation of classes from instances and then creating their generalized concepts is a bottom-up knowledge construction approach.

We populated our ontology from WordNet 3.0 [17]. 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 col-lections of synsets for example car pool. 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. Root adjectives relation is used for adverbs in WordNet. 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. WordNet contains 117798 unique string nouns with 82115 synsets. Nouns have 78% percent of the total WordNet size. Classes are made according to the unique strings and their senses. For class hierarchy we used WordNet hypernyms relations. Figure 2 shows two senses of the word chicken and its hypernyms relations in the hierarchical structure. Sense 1 of word chicken shown in Figure 2 formulates our food ontology where sense 2 becomes the part of our animal ontology. Food ontology is depicted in Figure 3 using Protégé. 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. The populated ontologies retain these structural properties in addition to several other newly defined properties. 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 hypernym 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. Figure 4 shows a small example of properties among concepts of animal and food ontologies.

4.3 Pedagogical Model

Educationists and psychologists mainly follow two teaching strategies or models i.e. Instructional and Constructive. Prince and Felder have named them as inductive and deductive models respectively [24]. 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 inductive or instructional and deductive or constructive teaching strategies in term of content authoring.

Content authoring under instructional strategies are demonstrated in the next section where a teacher can create content easily with the help of our prototype application. Since we assume that the child is not a direct user of our proposed ITS, so 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” i.e. “What animals produce and what animals eats”. The content under classification cognitive skill “Pet and Wild Animals” is created dynamically by the reasoning engine 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 re-
spect 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. Some of those learning contents are Missing Shoes (containing individuals from professions and cloths ontologies), Shape of Feet (animal and body parts ontologies) and Random objects (such as Fish with aquarium, ball with bat, dog with bone).

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. [22] 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. Ordering of objects is done by cutting and pasting of object images. Discrimination (belonging and not belonging) is a type of comparison cognitive skills. In discrimination, one or two objects are placed with a group of other objects that they do not belong to. MySekolah creates such discrimination contents with the use of disjoint relationship, individuals from different ontologies, constraints and restrictions and properties grouping.

### Table 1. Axioms

<table>
<thead>
<tr>
<th>Animal ⊑ Thing</th>
<th>Dog ⊑ Carnivorous/∃livesIN.Farm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Place ⊑ Thing</td>
<td>Lion ⊑ Carnivorous/∃livesIN.Jungle</td>
</tr>
<tr>
<td>Farm ⊑ Place</td>
<td>Tiger ⊑ Carnivorous/∃livesIN.Jungle</td>
</tr>
<tr>
<td>Jungle ⊑ Place</td>
<td>Cow ⊑ Herbivorous/∃livesIN.Farm</td>
</tr>
<tr>
<td>Bird ⊑ Animal</td>
<td>Cow ⊑ Herbivorous/∃livesIN.Jungle</td>
</tr>
<tr>
<td>Carnivorous ⊑ (Animal∪Bird)/∀eats.Animal</td>
<td>Parrot ⊑ Bird/∃livesIN.Farm</td>
</tr>
<tr>
<td>Herbivorous ⊑ (Animal∪Bird)/∀eats.Plants</td>
<td>Pet ⊑ (Animal∪Bird)/∀livesIN.Farm</td>
</tr>
<tr>
<td>Cat ⊑ Carnivorous/∃livesIN.Farm</td>
<td>WildAnimal ⊑ (Animal∪Bird)/∀livesIN.Jungle</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

### 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.

Figure 5: Pet and wild animal inferred class hierarchy

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 1 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 etc as the result. The resulting asserted and inferred class hierarchy to get individuals of Pet and Wild animals is shown in Figure 5 using OntoGraph Protégé plugin.

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.

Figure 6 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 Figure 6. The time required for searching any concept from selected ontology is minimized by providing a selection view or selection control to the user. Hence, 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, category 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 figure 7) 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 contents 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.

The usage of MySekolah is described earlier in Section 4.4. Following the interactive pedagogical model the teachers/mediators just do selection of concepts and their associated properties to author learning contents. 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 2) by using their traditional way first and then later the same with the MySekolah.

<table>
<thead>
<tr>
<th>Classification</th>
<th>Relatedness</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td>What animal eats</td>
<td>Cloths and Weathers</td>
<td>Seating Capacity</td>
</tr>
<tr>
<td>Carnivores &amp; Herbivores</td>
<td>Profession and Vehicles</td>
<td>Speed Comparison</td>
</tr>
<tr>
<td>Fruit &amp; Vegetables</td>
<td>Shape of the body parts</td>
<td>Body Size</td>
</tr>
<tr>
<td>Soft and Hard</td>
<td>Things that Animal Produce</td>
<td>Containers Size</td>
</tr>
<tr>
<td>Ancestors</td>
<td>Random Objects</td>
<td>Longer and Shorter</td>
</tr>
</tbody>
</table>

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 bringing 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 3 shows the mean time spent to develop these same learning contents (difficulty
level 1) with use of the MySekolah and the traditional way.

Table 3. Time to create pre-defined contents (Mean Time mints)

<table>
<thead>
<tr>
<th>Classification</th>
<th>Relatedness</th>
<th>Comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional Way</td>
<td>13.94</td>
<td>17.56</td>
</tr>
<tr>
<td>MySekolah</td>
<td>1.33</td>
<td>1.46</td>
</tr>
</tbody>
</table>

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 3. The time difference is still pretty high (Table 3) 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 4. Dynamic content generated from MySekolah

<table>
<thead>
<tr>
<th>Classification</th>
<th>Relatedness</th>
<th>Comparison</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>18 (33.34%)</td>
<td>31 (57.40%)</td>
<td>5 (9.26%)</td>
<td>54</td>
</tr>
</tbody>
</table>

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 4 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-classof or associated with measurable concept in cognitive skills ontology.

Table 5. Dynamic content generated for relatedness cognitive skill

<table>
<thead>
<tr>
<th>Direct relationship</th>
<th>Indirect Relationship</th>
<th>Inferred</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>22 (70.96%)</td>
<td>6 (19.35%)</td>
<td>3 (9.68%)</td>
<td>31</td>
</tr>
</tbody>
</table>

Table 5 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 visualized from Figure 4). Finally an evaluation of dynamically created contents was performed by teachers to validate their correctness.

<table>
<thead>
<tr>
<th>Table 6. Dynamic contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correct</td>
</tr>
<tr>
<td>37 (68.51%)</td>
</tr>
</tbody>
</table>

The dynamic contents generated from MySekolah are separated in four categories shown in Table 6. 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 6 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 [17] that contains several terminologies that are difficult to comprehend by a child of age 4 to 6.

![Figure 8: Students Score (n=21)](image)

The remaining correct (68.51%) contents (Table 6) 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. Figure 8 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 each 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. & Chisnell D. [26] for usability testing. Figure 9 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.

![Figure 9: MySekolah evaluation](image)

5. 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


Muhammad Azeem Abbas received his master's degrees 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 PhD candidate in Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia. His interest researches are intelligent tutoring systems and semantic web applications.

Wan Fatimah Binti 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 also obtained Dip.Ed. from Universiti Sains Malaysia in 1992. She completed her PhD in Information System from Universiti Kebangsaan Malaysia in 2004. She is currently an Associate Professor in the Department of Computer and Information Sciences at University Technology Petronas (UTP). She was a lecturer at Universiti Sains Malaysia, Tronoh before joining UTP. Her main interests are in the areas of mathematics education, educational technology, e-learning, mobile learning and multimedia.

Khairul Shafee Bin 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 Department of Computer and Information Sciences at University Technology Petronas (UTP). His interest researches is knowledge management.