Semantic Web based Service Composition for E-learning

Swati Rai[1] , Dr. Ashish Chaurasiya[2] , Shubham Bind[3]
CSE Department (Kashi Institute of Technology Varanasi)

Abstract:- The amalgamation of web and Artificial Intelligence has produced miraculous results from the semantic web terms to be the most significant E-Learning application. This is because, E-learning performed efficiently only at the time of E-Content are customized, need based, real-time, on-demand, and searchable using keywords. The methodology created in the rule derived and customized preparation of e-content from the prevailing contents available on a specific topic, are based on certain keywords. This can be a holistic move to develop a concise digital collection of various topic/s. This kind of responsible work is to demonstrate the new approach by which an E-book Ontological search with the SQWRL and SWRL Rules are generated to obtain chapters and its requisite paragraphs (having keywords together with relevant factors) and thereafter, from the ontology developed, upon combining various paragraphs out of different eBooks, the desired contents can be made available.

Keywords:- *SWRL, SQWRL, OWL-DL, pellet as a reasoner, and JESS as a rule solver for both the rule languages SWRL and the SQWRL, RDF, RDF Schema.*

I. INTRODUCTION:

At present, the web concept appears to be extremely large, primarily static, information source. Obtaining the correct Web information seems to be a terrible thing. One can possibly go astray looking at the enormous and irrelevant subjects and can also overlook the relevant subjects, while searching specific information on the Web. Such searching process can prove to be imperfect to provide thousands of irrelevant pages. As per Fensel et al., (2005), the retrieved document search is also necessary to extract the requisite information. The Keyword-derived search-engines, become the prime methods, giving tools to work on prevailing Web, which usually puts a serious problem. This is because, its results are very sensitive and delicate to vocabulary. Due to this reason, in reality, identical queries provide strange results. If the requisite information is dispersed over several documents, one requires to commence many queries to obtain relevant contents and thereafter to physically obtain certain information to put them together. Hence, the prevailing search-engines fail to work as an 'information retriever' however, provide only as a 'location finder' (Antoniou & Harmelen, 2004). Hence, the prime reason for such problems is the Web content and its meaning, which is not automatically accessible.

Such situations puts a huge strain to access information, for extraction, and thereafter, its interpretation kept aside for further human act as web users. It indicates to provide machine understandable information and data to the web to get the perfect results, and this procedure has become highly important and necessary.

II. LITERATURE SURVEY

2.1 Service composition driven Model:

This paper (Orriens et al., 2003a) tries to introduce the Model Driven approach of Service Composition, based on the composition of dynamic service, because it should assist the development to the management of dynamic service composition. Unified Modeling Language (UML) is utilized to impart a high abstraction stage, and to help directly mapping to meet other standards.

In this case, Business rules are applied to structure and arrange the composition program of service, moreover, to describe the selection and bindings of service. Also, it identifies two major uses: the process of composition development service, which is sub-divided into four stages: Scheduling, Service Defining, Construction stage, and Execution process. To help illustrate every service composition, the study provides the model information, a meta-model abstract, to provide model components and the relationship between those components. Because of this reason, the elements of service composition and rules of service composition are explained and defined.

2.2 Semantic Web Services based on AI planned Composition:

The work performed by (Wu et al., 2003; Pistore et al., 2004; Lecue, & Leger, 2005; Wu et al, 2006) has informed the AI planning on the basis of SWS composition. The service composition based AI planning is primarily of plan generation for composition prior to performing the real composition.

As described by Wu et al, (2006), the Java based automatic SWS composition method does not require any human- involvement, is developed. It applies properly planned approach to managing the problem of service-based protocol-heterogeneity approach to manage data-heterogeneity problems. The system ontologies are managed applying Jena, using identical mapping technology to manage the complexity of multiple ontologies. The technique is highly adaptable to new situations, as it requires to adapt to task-specifications, preferences and discovery rules, while moving to fresh scenario.

2.3 Ontology-Derived Semantic Composition of Web Service:

Certain ontology-derived SWS composition approach are reported by Sell et al., (2004); Gomez-Perez et al., (2004); Arpinar et al., (2004); Charif & Sabouret, (2005). The service composition of Ontology involves composing work, based on their ontological explanations and the connection between them.

A design framework applied to the SWS, semi-automatic and automatic composition applying ontologies were built by Gomez-Perez et al. (2004). The work process composition is done on the basis of ontology stack, which describes various SWS parts and have designed-rules, verified by the instances of ontology. The format of specific three models: translated model, checking model and instance model, whereas, the checking model checks the consistency of the instance model, while the translate model helps translate the instances of ontology to form semantic web language, like OWL-S.

2.3 Semantic Web Services, Agent based Composition:

In SWS, agent based composition, there are various agents included in the system to provide various individual services. An entire multi-agent functional system is known for SWS composition. This work was performed by Kungas & Matskin, (2006); Abela, (2003); Ermolayev et al., (2004); Burstein et al., (2005); Valee et al., (2005); Cao et al., (2005); Preist et al., (2002); Kungas et al., (2004); Kungas & Matskin, (2005) among several other works done on agent derived SWS compositions.

The Kungas & Matskin (2006) work proposed the multiple-agent composition system, while the SWS discovery was done on the basis of chord P2P network (Stoica et al., 2001). This system gives services of dynamic composition utilizing agents, which cooperate and apply symbolic distribution reasoning. The OWL-S, web ontology semantic language, is applied in the system to describe the services. For the web service composition, initially, the SWS available is considered as an OWL-S input service profile.

2.5 Another Web Service Semantic Composition Methods:

A model, based on the BPEL4WS (IBM et al., 2003), has also been presented providing SWS composition (Charif and Sabouret., 2005). This model explains web services in the form of business procedures. The model imparts a notation, for various web service interactions. The process model separate service-roles are treated as associates providing incorporation. The model operates on a bottom-up method (Mandell & McIlraith, 2003). Initially, it collects every OWL-S or DAML-S service profile to get into a repository. Thereafter, the required partner service properties are defined. The service semantics in the repository get exploited by means of partner's querying, depending on the defined partners' descriptions and properties.

III. SEMANTIC WEB SERVICES, LANGUAGES AND REASONERS

It is created as a new-generation Web, which tries to present information, so that, it can be utilized by any machines for display purposes, and also for integration, automation, and reuse in all applications (Boley et al., 2001). Known to be the best research and development (R&D) topic in the AI community recently, and also in the entire Internet community, where the Semantic Web carries very vital functions of the World Wide Web- www Consortium, called W3C (W3C SW Activity, 2005). This 'semantic web' is the recent method extending the web using semantic sequences (Berners-Lee, 2001). The semantic web aim is to create technologies to enable machines to incorporate more wisdom and sense on the web. The eventual goal is to transfer web as a useful application for humans to provide better kinds of services.

Semantic Web Services: The existing Web is a static data repository and provides interfaces to Web- reachable services for the manual user, starting from regularly generated dynamical pages for merely information terms to additional complicated services for booking trips, purchasing books, or for trading purpose with various internet-users globally for commercial or private applications. The further step after completing the data process on the possible Web machine, it facilitates the direct services, application, and interaction over the Web.

Semantic Web Languages: Together with certain famous web languages, which provided machine processable and formal knowledge representation are: SWRL, OWL/OWL-S, RDF/RDF-S, WSML, and DAML/DAML-S. Such languages vary in their modeling primitive setting, semantic representing strength, and the power of expression. As per Manola & Miller (2004), RDF is a strong triple-derived Universal Resource Identifier's (URIs) representation language, which assigns particular URIs to the individual resources and fields. It explains resources in the simple property terms and property values.

Semantic web languages and Reasoners: Ontologies provide common and shared domain models and they are the prime semantic web component. The reasoning services, which are Ontology- supported can be Operationalization Semantics to provide network-based service. Ontology helps machines and people to assess, process, and communicate information. Ontologies will be implemented using various languages ontology and presently, the most well known languages of ontology for the Web are:

• RDF; • Schema of RDF; and • OWL.

Semantic Web Reasoners: The group called MindSwap has created Java, an open-source OWL-DL based reasoner known as Pellet.8 It is done on the basis of a tableau algorithm to support expressive logic description. It also supports Expressivity completely of OWL-DL along with nominal reasoning. It makes sure of completeness and soundness by incorporating the SHOIQ decision procedures.

ISSN No:-2456-2165

OWL-DL Reasoner Pellets

The Web Ontology OWL Test Categories of W3C Recommendations define two document checkers kinds OWL syntax and OWL consistency checkers. It further explains four conformance categories of consistency checkers, Lite; DL; Complete OWL consistency checker; and OWL-Lite entire consistency checker. For a consistency checker it should match with the specific semantics species. To complete them, a proper procedure regarding that semantic should be taken. OWL is not Fully decidable, hence, such thing as complete does not exist regarding Full OWL consistency checker. In other words, the Test Case Recommendation fails to explicitly define the full consistency of OWL-DL checker, even though it can be explained that, there does not exist any known procedure to take a decision regarding OWL-DL. OWL-DL complete consistency checker is provided by Pellet and it is an incomplete Full, OWL consistency checker, which is otherwise a syntax OWL checker. As we know, Pellet is the initial and entirely complete consistency OWL-DL checker.

Ontology

"Ontology is a set of knowledge terms, including the vocabulary, the semantic interconnections, and some simple rules of inference and logic for some particular topic." (Hendler, 2001)

An ontology happens to be a clear, open and explicit and remains as a conceptualization of formal specifications.

Otherwise, an ontology explains a discourse domain formally, specifically, the ontology is of a finite term list and the link connecting these terms. The terms also describe vital concepts and categories of objects of this domain. For instance, in the location of a university, students, staff members, lecture theaters, courses, and faculties form the vital concepts.

Expressive Power and Limitations of RDF Schema

RDF along with RDFS permit the illustration of certain ontological explanation and knowledge. The modeling main primitive of RDF/RDFS involves the vocabulary of organization in category hierarchy: sub category and sub property connection, range restrictions, domain is the category instance. However, there are several different features, which are missing, as mentioned below:

• Local properties and scope of RDFS: The range explains the property range, like eating, and it is in every category. Therefore, in the case of RDF Schema, nothing can be declared regarding range restrictions, which relate to some categories. For instance, it is not possible to inform that cows normally eat merely plants, when different animals may also eat meat.

• Disjointness of category is when at time, we intend to inform that categories are disjoint.

For instance, female and male are disjoint. But in the case of RDF Schema, only statement can be made is the sub category relationships, which means, the female remains a sub category of persons.

OWL Compatibility with RDF and RDFS

In an ideal case, OWL remains as RDF Schema extension, in the view that OWL uses the RDF category meaning and properties (rdfs:subCategoryOf, rdfs:Class, and so on) and this will include the language primitives that support the rich expressiveness needed.

Unluckily, the RDF Schema simple extension can work against getting an efficient reasoning and expressive power. RDF Schema carries certain strong modeling primitives, while constructions like RDFS:Category (the best category) and RDF:Property (the category of every property) remains highly expressive, that can lead to unsuitable computational property, if the logic gets extended using this expressive primitive.

OWL DL

To recapture computational efficiency, the Description Logic, which in short known as OWL DL has become the OWL Full sublanguage that constrains the ways OWL/RDF constructors can be implemented. Basically, OWL constructor's application to one another is not allowed; therefore, they ensure that this language relates to well-selected description logic.

Tools of Ontology

Simultaneously, such ontology languages were generated with tools and they emerged for editing, creating, and organizing written ontologies in several languages. Briefly described Protégé 2000 is to indicate the functionality kind on offer, along with certain famous tools as mentioned below. Protégé was created at Stanford University to work like editing of Open Source environment, wherein ontology can be developed by an interface graphical user. It was developed applying the plug-in architecture, wherein a fresh service could be included conveniently in the situation, and possibly the widely applied ontology tool. Further, it is used to manage OWL ontologies. RDF (S), XML, DAML+OIL, XML Schema,

Several well-managed tools of ontologies developed include:

OntoEdit was created by the Karlsruhe University and it contains ontology repository

OilEd was created at the Manchester University for simple OIL language development,

WebODE provides support many ontology lifecycle activities,

Ontolingua is called an editing ontology environment.

was created by the Karlsruhe University and it contains ontology repository.

LinkFactory is known as a management system of formal ontology, designed to generate and manage very large and complex language-independent formal ontologies.

OWL and SWRL

SWRL and OWL are core Web Semantic languages. The development of OWL as a constructing ontology language provides Web content high-level description. The ontologies get developed by class building hierarchies by

ISSN No:-2456-2165

describing the domain concepts and relating the class to one another applying properties. OWL represents data as OWL class instances referred as individuals that provides the mechanism for reasoning using the data and also manipulating it. OWL further provides a strong axiom language, correctly defining and interpreting the concepts in the ontology.

OWL Queries Supporting with SQWRL

SWRL is known as rule language, not considered as a query language. But, several ontology-derived applications need the capability extract data from ontologies, further for reasoning providing the ontology information. For knowledge extraction supports, a query language was developed known as Semantic Query-Enhanced Web Rule Language- SQWRL7, which supports SWRL querying the ontology of OWL.

The implemented of SQWRL is used as a built-in library with the built-in mechanism of standard SWRL, which is compatible syntactically and semantically with standard SWRL. The built-in library of SQWRL contains built-ins SQL-influenced applied in a rule to retrieve information stored in OWL ontology.

Problem Description:

In case we are required to prepare some study material for a specific Tk topic, which is different from T1 to Tn topics, regarding a specific single subject Sp which remains different than other S1 to Sm subjects, This problem can be decomposed in two segments:

(a) By taking out chapters out of eBooks using key words, then

(b) By exploring paragraphs and taking them out from the chapters extracted on the basis of relevant keywords

For preparing required contents, certain steps must be followed:

(a). Firstly, to identify the requisite books and Ebook from Ebook1 to Ebookn concerning that specific topic, and subject(s).

(b). Based on particular keyword/s input the chapters will be extracted from the various identified ebooks in the previous step using Keyword based SWS Composition approach described in the previous chapter.

(c). Again from selected chapters, based on keyword/s and its relevance to the paragraph, the paragraph will be selected using again keyword based approach.

(d). Now, the collection of paragraph is our desired content for which this whole system has been evolved.

To solve above problem we do keyword-based Semantic Web Services Composition.

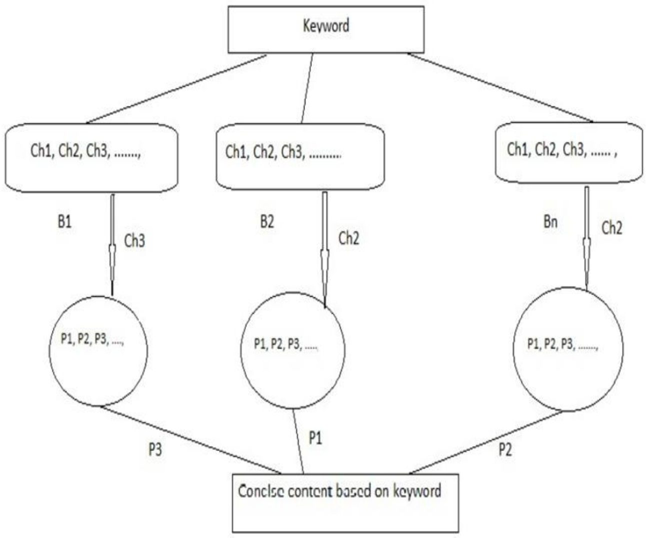

Figure 1: Illustration of content preparation using keywords

Suppose we want to create content for the keyword 'frame' (one of the topic of Artificial Intelligence subject).

Now among all available ebooks on the subject related to artificial intelligence, we have to filter out all the chapters that contain the keyword 'frame', now we have at least one chapter from each ebook. Suppose we get different chapters ch3, ch2, ch3… etc. from n number of Artificial Intelligence related ebooks.

As we know that each chapter contains n number of paragraphs, so based on keyword we have to extract one paragraph from each chapters, which has highest relevance factor i.e. corresponding to keyword 'frame' a particular paragraph has the highest relevance and that paragraph will be selected. Similarly, we have to repeat this selection process among all the available n number of chapters.

Design using protégé

We use protégé 3.6.7 as a tool for generating knowledge base and rules.

Among the available various tools protégé is best suited for our requirements as protégé fully support the development of a knowledge base in the form of either classes or frames, along with the support for rule implementation over stored knowledge base; available separately for protégé as a plugin.

In our approach, we stored knowledge base in the form of classes, and the reason behind this approach is easy modeling of application domain the classes. In protégé every class requires some data property and/or object property.

E-booksearch Ontology:

E-book Class
i. Object Properties
   a. HasChapters
ii. Data Properties
   a. HasAuthors
   b. HasNames

Chapter Class
i. Object Properties
   a. HasParagraph
ii. Data Properties
   a. HasCKeys
Paragraph Class
i. Object Properties
   a. HasContents
   b. HasKeywords
   c. HasRelevance

Just the creation of classes is not sufficient and fulfills our requirement, until we impose certain conditions over it. For example the eBook must have exactly one name; the eBook must have at least one author and also contain at least one chapter. In ebooksearch ontology, we implemented these conditions using cardinality on each class.

Instances of chapter class
AUTOMATED_REASONING
KNOWLEDGE_REPRESENTATION
LOGIC_AND_COMPUTATION
HEURISTIC_SEARCH
Knowledge_and_reasoning
Planning
Learning
Problem-solving
Ch03_01
Ch03_02
Ch03_03
Ch03_04

Instances of Paragraph class
FRAMES
SEMANTIC_NETS
STRUCTURE_OF_AN_RBS
MODEL_BASED_REASONING
DEFAULT_LOGIC
RESOLUTION
CLASSICAL_CONCEPTS
FIRST-ORDER_LOGIC
COMPUTATIONAL_LOGIC
HILL_CLIMBING
INFORMED_SEARCH
WATER_JUG_PROBLEM
Logical_agents
Knowledge_Representation
First-Order_logic
Knowledge_in_learning
Planning_and_acting_in_the_real_world
Statistical_learning_methods
Planning
Learning_from_observation
Informed_search_and_exploration
Reinforcement_learning
Solving_problems_by_searching
Adversarial_search
P03_04_01
P03_04_02
P03_04_03
P03_03_01
P03_03_02
P03_03_03
P03_02_01
P03_02_02
P03_02_03
P03_01_01
P03_01_02
P03_01_03

Implementation SWRL:

SWRL, in this work, is utilized to generate rules while SQWRL is utilized to support OWL queries.

We have divided the implementation rule as shown below in four parts:
1. Rule Name
2. Rule Expression
3. Rule Description
4. Rule output

We have created five different SWRL/SQWRL rules to retrieve stored knowledge base.

Rule Name
Chapter_and_their_paragraphs

Expressing Rules
Ebook (?e) ∧ HasChapters(?name,?e) → Sqwrl:select(?name, ?e) ∧ Sqwrl:OrderBy(?e) ∧ Sqwrl:ColumnNames- "Chapter Name" and "Ebook Names"

Rule Description

This retrieving ebook rule and its related chapters of stored knowledge base, display results in the Ebook name columns and Chapter Names respectively.

Rule Output

Chapter Names	Paragraph Names
AUTOMATED_REASONING	
MODEL_BASED_REASONING	
AUTOMATED_REASONING	
RESOLUTION	
AUTOMATED_REASONING	
DEFAULT_LOGIC	
HEURISTIC_SEARCH	
INFORMED_SEARCH	
HEURISTIC_SEARCH	
WATER_JUG_PROBLEM	
HEURISTIC_SEARCH	
HILL_CLIMBING	

KNOWLEDGE_REPRESENTATION STRUCTURE_OF_AN_RBS
KNOWLEDGE_REPRESENTATION SEMANTIC_NETS
KNOWLEDGE_REPRESENTATION FRAMES
Knowledge_and_reasoning First-Order_Logic
Knowledge_and_reasoning Logical_Agents
Knowledge_and_reasoning Knowledge_Representation
LOGIC_AND_COMPUTATION CLASSICAL_CONCEPTS
LOGIC_AND_COMPUTATION FIRST-ORDER_LOGIC
LOGIC_AND_COMPUTATION COMPUTATIONAL_LOGIC
Learning Statistical_Learning_Methods
Learning Knowledge_in_Learning
Learning Learning_from_Observations
Learning Reinforcement_Learning
Planning Planning_
Planning Planning_and_Acting_in_the_Real_World
Problem-solving Adversarial_Search
Problem-solving Solving_Problems_by_Searching
Problem-solving Informed_Search_and_Exploration
ch03_01 p03_01_01
ch03_01 p03_01_03
ch03_01 p03_01_02
ch03_02 p03_02_01
ch03_02 p03_02_02
ch03_02 p03_02_03
ch03_03 p03_03_01
ch03_03 p03_03_03
ch03_03 p03_03_02

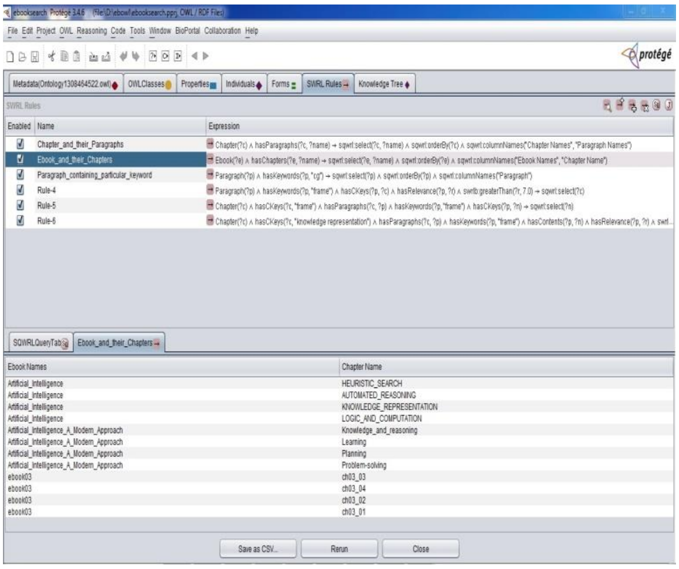

Figure 2: Various rules and their output using SWRL and SQWRL

IV. CONCLUSION

This study was recommended as a possible method to obtain keyword based service composition of semantic web. The OWL-DL derived ontology of Ebooksearch is formed applying protégé. In this case, we applied SQWRL and SWRL for gathering the ontology of Ebooksearch constructed applying several rules, and therefore, used rule engine JESS to resolve them. The ontology consistency check of was processed applying pellet reasoner. As shown in figure 4, it produces the E-content final created output. The ontology in this case is of enormous use to create content on the basis of specific keywords to generate personalized digital library.

REFERENCES

[1]. Berners-Lee, T., Hendler, J., Lassila, O, “The Semantic Web”, Scientific American, May 2001.
[2]. The Protégé project, http://protege.stanford.edu
[3]. SWRL-Submission, http://www.w3.org/submission/SWRL
[4]. SWRL Built-ins, http://www.daml.org/2004/04/Swrl/builtins.html
[5]. SWRLTAB Plug-in, http://protege.cm3.net/cgibin/wiki.pl?SWRLTAB
[6]. Pellet Reasoner, http://www.pellet.owldl.com
[7]. D.L. McGuinness, F. Van Harmelen, D.L. McGuinness, F. Van Harmelen, “OWL Web Ontology Language Overview”, in: W3C Recommendation, W3C, 2004
[8]. Web Ontology Working Group (2004). OWL Web Ontology Language Guide. Available at: http://www.w3.org/TR/owlguide/
[9]. SWRLJessTab http://protege.cim3.net/cgi-bin/wiki.pl?SWRLJessTab
[10]. R.B.Mishra, S. Kumar “Semantic web reasoners and languages”, Springer Science, Dec 2010
[11]. Christopher Brewstera, Kieron O’Harab, “Knowledge representation with ontologies:Present challenges - Future possibilities”, Int. J. Human-Computer Studies 65 (2007) 563–568
[12]. Li Ding et al., "Search on the Semantic Web", IEEE Computer, October 2005.