An approach for semantic-based searching in learning resources

Tran Thanh Dien
Can Tho University
Can Tho city, Vietnam
thanhdien@ctu.edu.vn

Le Van Trung
FPT Software Can Tho
Can Tho city, Vietnam
trunglv10@fsoft.com.vn

Nguyen Thai-Nghe
Can Tho University
Can Tho city, Vietnam
ntnghe@cit.ctu.edu.vn

***Abstract*- Currently, online learning has been widely applied in education and training. Especially, when it is difficult for lecturers and learners to get close to each other in the context of Covid-19 epidemic period, online learning shows its availability and necessary. Learning materials provided in the educational institutions are diverse; almost lectures are stored as files but have not been totally arranged in a standard database system. Therefore, searching information about curriculum and lectures still face difficulties. This paper proposes a solution for semantic-based searching in learning resources. Firstly, ontologies are built to represent information of lectures. When users enter a query, the system pre-processes it (e.g., word segmentation, removing stop words), and then provides it to classifier (e.g., SVM) to identify the corresponding domain (or topic), aiming to narrow the search space in the ontology. After classifying, the key phrases will be queried in the appropriate ontology to result in related lectures. Experiments on lectures in the domains of information technology show that the proposed model is feasible.**

Keywords- Learning resources, ontology, semantic web, topic classification.

I. INTRODUCTION

Along with the rapid development of information technology, more and more data in education domain is provided in educational systems, in which learning resources have been posted aiming to meet learners' needs. Learning resources could be defined as the instruments of presentation and transmission of the prescribed educational materials including images, maps, photographs, sketches, diagrams, films, written material such as newspaper clippings or articles from scientific and technical literature [1] provided on online learning systems, lecture and curriculum management systems, scientific publication system, etc.

Currently, most of universities' learning resources are stored as digitized files, but these files are not arranged in a certain standard database system. Meanwhile, search websites and other management support tools have not met the needs of semantic and quick search; doing a search still has many problems such as inaccessible learning resources, redundant search results, and only being searched through keywords without semantic support [2]. Therefore, semantic-based search models using ontology domain to better handle and retrieve documents are of interest today [3].

There are many studies related to semantic search. The authors [4] proposed an approach to compare the similarities in Vietnamese learning resources (e.g. books, theses, journals) for plagiarism check. Firstly, the data set is pre-processed, extracted, vectored and presented as TF-IDF. Then, the semantic similarity (cosine similarity) and word order similarity of the documents were calculated. Finally, the two similarities were combined to determine the semantic similarity between the documents.

Ontology-based information searching is becoming an interest of current studies on ontology and the semantic web [5]. An ontology-based system developed by [6] helps find knowledge for any field, overcoming the limitations of keyword-based approaches.

Another study of [7] proposed an ontology-based semantic search approach for educational management systems. Firstly, the authors presented a number of rules to build domain ontology from the learning resources of the educational management system, then use the semantic annotation for the built ontologies so that semantic information can be used in searching information resource. Finally, the ontology-based semantic search algorithm was used. Experimental results showed that this semantic search model on learning resource gave better results than the traditional search method for educational management systems.

In an effort to provide a solution for discovering resources for different user groups, the authors [8] presented an integrated model for personalized educational search. This model exploits technologies including ontology, metadata annotation schemas, and semantic web search engines to provide users with learning resources appropriate to their interests. This model also combines with an algorithm to prioritize returning relevant results, collect users' learning resource evaluation results as well as feedback to adjust subsequent results.

In fact, learning resources have many types of documents in different domains (topics), so it is necessary to build ontologies, in which each ontology describes documents in the same domain. Therefore, classification is needed to determine the domain of users' query, thereby conducting a search on the domain to have faster search results, satisfying users' needs.

This paper proposes a solution for semantic searching in learning resources based on ontologies representing information of lectures. When user enters a search keyword, the system will pre-process and classify to identify the corresponding domain to narrow the search space, then search in the appropriate ontology to return the results as related lectures.

II. SEMANTIC WEB AND ONTOLOGY

When using normal search engines such as Google, searching information will not take advantage of semantic web. A system for semantic search or a semantic-based knowledge network returns the complete structured information that the computer can "understand", thereby using or processing information becomes easier [9], [10]. Semantic search engines are built on the different techniques and technologies of certain platforms. To describe in detail the structure of a semantic search engine, first of all, it is need to have platforms for semantic search. Semantic web and ontology are the two main platforms for doing this work.

978-1-7281-4510-5/20/$31.00 ©2020 IEEE

A. Semantic web

Semantic web is built on the platform of the existing web system. The semantic web is considered an extension of the existing web that adds semantics to data on the web. The architecture of the semantic web consists of layers described in Fig. 1.

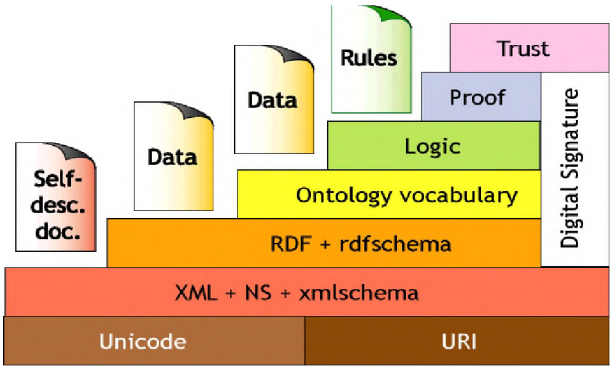

Fig. 1. The architecture of semantic web [11]

- Unicode and URI layers ensure the use of international character sets and provide means for identifying objects in the semantic web.

- XML, Namespace and XMLSchema layers ensure being able to integrate semantic web definitions with other XML-based resources.

- RDF and RDFSchema (RDFS) layers use metadata to describe web documents that computer can understand. This layer can be assigned types to resources and links, and is also the most important layer in the semantic web.

- Ontology layer provides common vocabularies for exchanging information between applications and web services.

- Digital Signature layer used for RDF, RDFS, Ontology, Logic, Proof, is used to identify the subject of the document, to ensure the reliability of the document.

- Logic layer allows writing rules, which is considered the basis of the semantic web rules.

- Proof layer is used to prove the inference of the system by linking the facts.

- Trust layer is a system being built on the basis of electronic signatures.

B. Ontology

One of the main ideas of semantic web is that semantic data can be shared among computers in the form of domain representation data model (or ontology) that allows creating global data [11]. According to [12], ontology is the expression of a set of concepts in a particular domain, and relationships between these concepts. Most ontologies describe individuals or instances, classes or concepts, attributes and relations. The most important ontology languages include XML/XML Namespace/XML Schema, RDF/RDFSchema and OWL [13].

RDFS (RDFSchema) is an extension of RDF that allows to describe the classification of classes and properties [14]. RDFS can also be considered a semantic extension of RDF to provide mechanisms that allow the description of related resource groups and their relationships. In RDFS, classes are a group of related resources; properties are the relationship between subjects and objects in RDF. OWL (Web Ontology Language) is an extension of RDF and RDFS, OWL's main purpose is to bring the ability of inference into semantic web. Basically, OWL and RDF have many similar characteristics, but OWL has a larger vocabulary (keyword) set and is a better computer-interpreted language than RDF. Currently, three types of OWL include OWL Lite, OWL DL and OWL Full; each of them has its own characteristics appropriate in the context of a specific application[1].

1) RDF (Resource Description Framework)

RDF as the platform of semantic web and metadata processing is defined by W3C organization. RDF is used to describe information about resources on web through the URI (Uniform Resource Identifier) and describe the semantics of that information in a way that computer can understand. The basic model of RDF consists of three parts: resources all described by RDF expression, properties or relationships describing the nature of resources, and statements. Each statement consists of three components as subject describing resource's address or location, predicate identifying the properties of the resource, and object being the content assigned to the predicate[2]. Each statement is called a triple as described in Fig. 2.

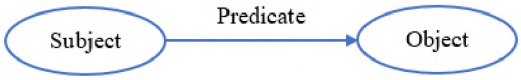

Fig. 2. The basic semantic triple model

For example, the following RDF statement "*http://www.example.org/index.html has a creation-date whose value is August 16, 1999*" is represented as a triple as follows:

ex:index.html exterms:creation-date "August 16, 1999" .

The RDF/XML syntax can be represented below:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-
rdf-syntax-ns#"
xmlns:exterms="http://www.example.org/terms/">
<rdf:Description
rdf:about="http://www.example.org/index.html">
<exterms:creation-date>August 16,
1999</exterms:creation-date>
</rdf:Description>
</rdf:RDF>
```

2) RDFS (RDFSchema)

RDF provides the way to describe simple statements about resources, using predefined properties and values. However, users' needs require an ability to define the terms that they want to use in those statements. For example, the company *example.com* wants to describe classes such as *exterms:Tent* and use the properties *exterms:model*, *exterms:weighInKg* and *exterms:packedSize* to describe them or an application that wants to describe the classes such as *ex3:Person*, *ex3:Company* and properties such as *ex3:age*, *ex3:jobTitle*, *ex3:stockSymbol*, *ex3:numberOfEmployees*, etc.

[1] http://www.w3.org/TR/owlfeatures/2019

[2] https://www.w3.org/2001/sw/RDFCore/TR/WD-rdf-primer-20030117/

Similar to XML schema, RDFSchema is a set of keywords that allow users to define specific vocabulary set (resource, property) for RDF data (for example: *hasName*, *hasPrice*, *authorOf*, etc.) and define its relation to related objects[3]. For instance, the *hasName* is defined its relation to two objects: 'http://www.w3c.org/employee/id1321' and "Jim Lerners" as follows:

hasName('http://www.w3c.org/employee/id1321', "Jim Lerners")

3) Definition of class

Resources on web can be divided into groups called *class*; members of the group are considered instances of the *class*. *Classes* are also *resources* identified through URI identifiers and can be described using *RDF properties*. The *rdf:type* property is used to indicate a *resource* as an instance of a *class*.

ex:MotorVehicle rdf:type rdfs:Class .

For example, company *example.org* wants to use RDF to provide information about different motor vehicles. The company must first use a *class* to represent the vehicles. In RDFSchema, a class is any resource whose property is *rdf:type* and has a value of resource *rdfs:Class*. Therefore, the motor vehicle class is described by assigning the class a URIref *ex:MotorVehicle* (using *ex:* to replace URIref http://www.example.org/schemas/vehicles, used as a prefix for URIrefs from lexicon of example.org) and describe that resource with *rdf:type* property having value of resource *rdfs:Class*. Therefore, *example.org* is written as the RDF statement as follows:

ex:MotorVehicle rdf:type rdfs:Class .

4) Definition of property

RDFSchema also provides a vocabulary set to describe how properties and classes can be used together in RDF data. The most important properties used in this case are rdfs:range and *rdfs:domain*.

rdfs:range

The property *rdfs:range* refers to the value of an attribute that is an instance of a class. For example, if the *example.org* company wants to indicate that the *ex:author* property is an instance of the *ex:Person* class, then the RDF statement is represented as follows:

ex:Person rdf:type rdfs:Class .
ex:author rdf:type rdf:Property .
ex:author rdfs:range ex:Person .

This statement indicates that *ex:Person* is a class, *ex:author* is a property, and RDF statements that use the *ex:author* property have objects that are instance of the class *ex:Person*. However, property may have multiple *rdfs:range*, as in the following example:

ex:hasMother rdfs:range ex:Female .
ex:hasMother rdfs:range ex:Person .

rdfs:domain

The *rdfs:domain* property is used to indicate that a property is a given class's. For example, the company *example.org* wants the property *ex:author* to be the *ex:Book* class's, then the RDF statement is represented as follows:

ex:Book rdf:type rdfs:Class .
ex:author rdf:type rdf:Property .
ex:author rdfs:domain ex:Book .

III. PROPOSED APPROACH

For learning resource search systems interested in semantics, the first stage is to process the query to determine which domain it belongs to. Then, query classification plays an important role in limiting the search space, making the search process faster and more accurate [15], [16], [17]. For these systems, especially those with big data sources, searching across multiple ontology domains requires a query double-task classifier to identify the ontology domain of the query, called intra-domain classification, and to determine the query's domain (topic) in the ontology domain defined in the *intra-domain* classification.

The general architecture of the semantic-based search model is proposed in Fig. 3. The text classification process uses machine learning algorithms, specifically vector support machine (SVM) algorithm. This technique is more popular used by researchers [16].

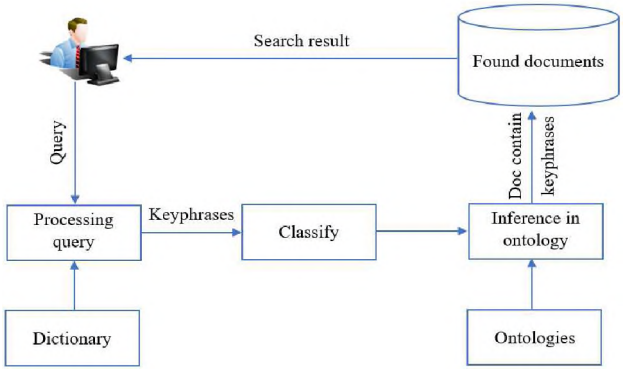

Fig. 3. The general architecture of semantic-based search system

The semantic search system diagram is described in detail as Fig. 4. Ontology building and data processing will be presented in the next sections.

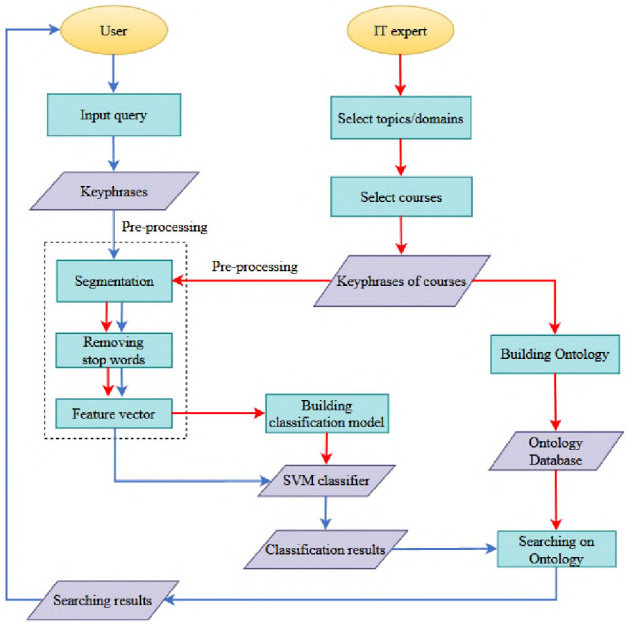

Fig. 4. Diagram of semantic search system

[3] https://www.w3.org/TR/rdf-schema/

A. *Ontology design*

In this paper, the semantic search system of learning resources was built in the information technology domain (including 4 majors: Information Systems, Computer Science, Software Engineering, Computer networks and Communications). However, it can be extended to apply to other fields. The information domains include lecture's name, lecture's content, lecture's topic. Classes and properties are described in Fig. 5.

Fig. 5. Classes and properties of ontology

Class **Thing** is superclass of the rest classes such as **Lecture_name**, **Topic**, **Lecture_content**, etc. Subclasses of **Thing** also have other subclasses. **Lecture**, **Topic**, **Lecture_content**, **Lecture_name** are sibling classes. The link property contains in the ontology. For example, **have** that represents the link of an instance of the **Topic** class, contains in the **Lecture_name** classes; **contain_in** that represents the relationship between **Lecture_content** classes is contained in the certain **Lecture_name**, etc. The **refer_to** property links between **Lecture_content** classes; **Lecture_content** links to other ones, and also has the links to other contents in the same lecture as Fig. 6.

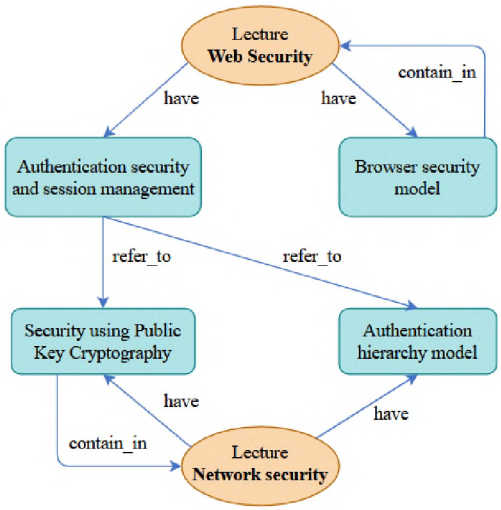

Fig. 6. Links between lecture contents

In Fig. 7, **Lecture_name** class contains the lectures within the defined information technology domain, a lecture can have many lecture contents. **Lecture_content** class is extracted from a lectures, many lecture contents belong to a lecture.

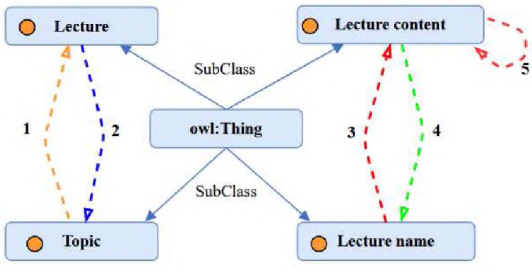

Fig. 7. Classification tree of ontology

In Fig. 7:

(1) **Topic** *have* **Lecture**

(2) **Lecture** *belong to* **Topic**

(3) **Lecture_name** *have* **Lecture_content**

(4) **Lecture_content** *contain in* **Lecture_name**

(5) **Lecture_content** *refer to* **Lecture_content**

After analysis, the ontology was designed using the Protégé tool to build the OWL database for querying data. The OWL file is used as a database in conjunction with the *Jena* open source library of the Java programming language to execute queries to return results for users.

Querying SPARQL data on ontology is done with the following SPARQL syntax:

```
PREFIX ow: <namespace ontology>
SELECT <select topics to be presented>
WHERE {
    ?topic ow:Object_properties ?topic
    [Filter(<filter search contents>)]
}
```

For example, SPARQL syntax to retrieve all lecture contents corresponding to lecture's name is done as follows:

```
PREFIX ow:
<http://www.semanticweb.org/username/ontologies/2019/9/research#>
SELECT distinct ?lecture_name ?lecture_content
WHERE {
    ?lecture_name ow:have ?lecture_content
    ?lecture_content ow:contain_in ?lecture_name
}
```

B. *Query classification*

As mentioned, in the semantic-based search system, the first step is to process the query to determine which domain the query belongs to in order to reduce the search space, increase speed and improve accuracy. In addition, classification is intended to determine the query's intra-domain and topic.

There are a number of studies on classifying query according to the regular expression approach based on hand-written grammar rules to determine the class of input query [18]. However, this approach still has certain limitations [19], [20] such as a small number of class, being not suitable for integrating into a large-scale semantic search system.

Therefore, another approach to solve the text classification problem is the probabilistic approach [16] including two main approaches of machine learning and language modeling, in which machine learning is of interest to many researchers. There are several existing text classification based on natural language processing and machine learning such as SVM, Naïve Bayes, kNN, etc. Many experimental results showed that SVM algorithm gives better classification performance than the remaining classifiers [21]. The SVM-based query classification model is described as Fig. 8.

Fig. 8: Diagram of query classification with SVM [22]

In Fig. 8, the pre-processing step performs the refinement function: segmenting words, removing stop words, then the query is fully extracted the features which were selected in advance. The input data of the SVM classifier will be a set of feature vectors.

C. *Data processing and model evaluation*

Data collection and pre-processing: From the identified topics, related lectures were collected, then, a dictionary for information technology domain was built and records were extracted from collected documents. Experimental collected data is shown in Table 1, including 1,312 records with 1,336 dimensions

TABLE 1: EXPERIMENTAL DATA BEFORE PROCESSING

No.	Label/ topic	No. of record	Total records	Dimension
1	Information Systems	168		
2	Computer Science	408		
3	Software Engineering	142	1,312	1,336
4	Computer networks and Communications	594		

In the data pre-processing stage, the tasks carried out including segmenting words, removing stop words; refining the records (deleting the duplicate records), etc. After pre-processing data, there were 1,114 records with constant number of vector dimensions (1,336) shown in Table 2.

TABLE 2: DATA AFTER PROCESSING

No.	Label/ topic	No. of record	Total records	Dimension
1	Information Systems	131		
2	Computer Science	325		
3	Software Engineering	114	1,114	1,336
4	Computer networks and Communications	544		

Vectorization of data: transforming string attributes into a set of numeric attributes that represents the text's appearance.

Training and evaluation of models: In order to classify topics, SVM model is used. The precision and F_1 are quite high, more than 95.42% (see Table 3).

TABLE 3: DETAIL ACCURACY OF TOPICS

	SVM			
Label/ topic	**Precision**	**Recall**	**F_1**	**ROC Area**
Information Systems	0.937	0.908	0.922	0.965
Computer Science	0.977	0.920	0.948	0.965
Software Engineering	0.991	0.939	0.964	0.978
Computer networks and Communications	0.939	0.989	0.963	0.963
Average Precision	**0.955**	**0.954**	**0.954**	**0.965**

D. *System model*

The system diagram is described in Fig. 9. For building this system, many tools and software are used such as Java, Python 3.7, IDE Spring Tool Suite 3.9, Jena library, Spring MVC, Flask, and Bootstrap. In addition, the VnTokenizer was used for tokenizing Vietnamese texts.

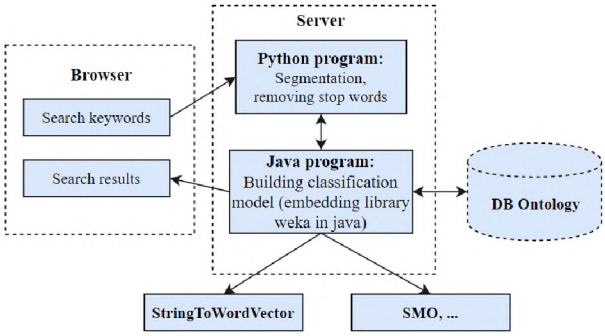

Fig. 9: System diagram

The interface of the system to perform semantic-based search is showed in Fig. 10.

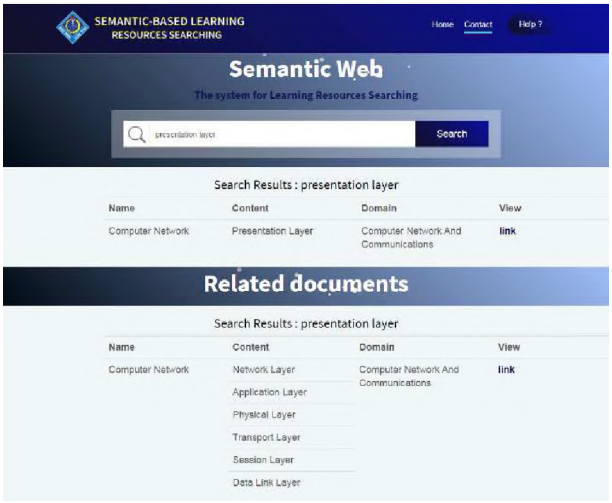

Fig. 10: The interface of semantic-based search

For example, when a user enters a query, such as a "presentation layer", the program runs in the command line mode to perform segmenting words, removing stop words, and returning processing results to the classifier to predict topics. The results returned after classifying will execute the query to the ontology. The results from the ontology are then processed and returned to the user.

IV. CONCLUSIONS

Online learning is an indispensable part in a learning society. This is a method that has been used in many higher educational institutions, especially in the context that lecturers and learners can not teach and learn as face-to-face by many different reasons, such as the Covid-19 pandemic taking place since end of 2019. The building a search engine is not new issue, but the semantic approach for online learning is of great interest to many researchers and educational managers.

This work presents searching in learning resources based on the semantic approach. When users enter a keyword, the system will pre-process and classify it to determine the appropriate topic of the keyword, aiming to narrow the search space in the ontologies. Then, it conducts the search in the relevant ontology before returning the results of the related lectures. An application is also developed to help lecturers and learners search their target lectures.

Further research should implement more experiments to compare machine learning methods with the state of the arts, especially deep learning to propose an appropriate

classification algorithm. Besides, comparasion of the proposed solution to other web-search approaches should also consider to suggest an effective searching approach. In addition, training models and ontology databases can be extended to better serve the needs of semantic web. Moreover, in the process of searching learning resources, in addition to the semantic issue, recommending appropriate learning resources related to learners should be further studied.

V. REFERENCES

[1] R. Bušljeta, "Effective Use of Teaching and Learning Resources", Czech-Polish Historical and Pedagogical Journal, vol. 5, no. 2, 2013. Available: 10.2478/cphpj-2013-0014.

[2] B. Wu, The Semantic Retrieval System for Learning Resources Based on Subject Knowledge Ontology. 2018.

[3] B. Yu, "Research on information retrieval model based on ontology," EURASIP Journal on Wireless Communications and Networking, vol. 2019, no. 1, p. 30, 2019/02/01 2019.

[4] T. T. Dien, H. N. Han, and N. Thai-Nghe, "An Approach for Plagiarism Detection in Learning Resources," in Future Data and Security Engineering, Cham, 2019, pp. 722-730: Springer International Publishing.

[5] R. Alfred et al., "Ontology-Based Query Expansion for Supporting Information Retrieval in Agriculture," 2014, pp. 299-311.

[6] S. Ma and L. Tian, "Ontology-based semantic retrieval for mechanical design knowledge," International Journal of Computer Integrated Manufacturing, vol. 28, no. 2, pp. 226-238, 2015/02/01 2015.

[7] L. Tang and X. Chen, "Ontology-Based Semantic Retrieval for Education Management Systems," Journal of Computing and Information Technology, vol. 23, p. 255, 01/01 2015.

[8] O. Okuboyejo, S. Misra, N. Omoregbe, R. Damasevicius, and R. Maskeliunas, "A Semantic Web-Based Framework for Information Retrieval in E-Learning Systems," 2018, pp. 96-106.

[9] S. Cohen, J. Mamou, Y. Kanza, and Y. Sagiv, "XSEarch: A semantic search engine for XML," in Proceedings of the 29th international conference on Very large data bases-Volume 29, 2003, pp. 45-56: VLDB Endowment.

[10] D. W. Gunter, "Semantic search," Bulletin of the American Society for Information Science and Technology, vol. 36, pp. 36-37, 2009.

[11] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," Scientific american, vol. 284, no. 5, pp. 28-37, 2001.

[12] C. Brewster and K. O'Hara, "Knowledge representation with ontologies: the present and future," IEEE Intelligent Systems, vol. 19, no. 1, pp. 72-81, 2004.

[13] G. Kaushal, "Role of Ontology in Semantic Web," DESIDOC Journal of Library & Information Technology, vol. 31, pp. 116-120, 2011.

[14] D. Stefan, "The semantic web: The roles of XML and RDF," IEEE Internet Computing, vol. 4, no. 5, p. 63, 2000.

[15] D. Zhang and W. S. Lee, "Question classification using support vector machines," presented at the Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, Toronto, Canada, 2003.

[16] J. Brown, "Entity-tagged language models for question classification in a qa system," Report on: http://www-2.cs.cmu.edu/jonbrown/IRLab/Brown-IRLab.pdf, 2004.

[17] M. Mishra, V. Mishra, and H. R. Sharma, "Question Classification using Semantic, Syntactic and Lexical features," International Journal of Web & Semantic Technology, vol. 4, 07/01 2013.

[18] B. Van Durme, Y. Huang, and E. Nyberg, "Towards light semantic processing for question answering," in Proceedings of the HLT-NAACL 2003 workshop on Text meaning, 2003.

[19] X. Li and D. Roth, "Learning question classifiers," in Proceedings of the 19th international conference on Computational linguistics-Volume 1, 2002, pp. 1-7: Association for Computational Linguistics.

[20] K. Hacioglu and W. Ward, "Question classification with support vector machines and error correcting codes," in Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume of the Proceedings of HLT-NAACL 2003--short papers-Volume 2, 2003, pp. 28-30: Association for Computational Linguistics.

[21] T. T. Dien, B. H. Loc, and N. Thai-Nghe, "Article Classification using Natural Language Processing and Machine Learning," in Proceedings - 2019 International Conference on Advanced Computing and Applications, ACOMP 2019, 2019, pp. 78-84.

[22] Nguyen Minh-Tuan, "Query classification head to search Vietnamese semantics in the health domain (in Vietnamese)," Hanoi University of Engineering and Technology, 2008.