Human-centric Computing and Information Sciences
a SpringerOpen Journal

RESEARCH Open Access

Generating metadata from web documents: a systematic approach

Hsiang-Yuan Hsueh[1*], Chun-Nan Chen[2] and Kun-Fu Huang[3]

* Correspondence:
chyhsueh@itri.org.tw
[1]Computational Intelligence Technology Center, Industrial Technology Research Institute, Taiwan, R.O.C
Full list of author information is available at the end of the article

Abstract

In this paper, a mechanism generating RDF Semantic Web schema from Web document set as the semantic metadata is proposed. Analyzing both the structural and un-structural content of Web documents, semi-structured Web documents can be conceptualized as resource objects with inter-relationships in RDF diagram. Technically, hyperlinks, basic annotations, and keywords in web documents will be properly analyzed, and corresponding RDF schema will be generated following the mechanism and rules proposed in this paper. It is expected that with the semantic metadata of document sets on the Web being systematically translated instead of manually edited, the semantic operation on the Web, such as semantic query or semantic search, will be possible in the future.

Keywords: Semantic web, Resource description framework, Metadata

Introduction

With the popularity of Internet and World Wide Web (WWW, Web), the size of documents on the Web grows dramatically. It is indeed that content on the Web has become the dominant resource to users for problem solving purposes.

However, the utilizing and query of such information resource is a challenge. Owing to the semi-structured nature of documents on the Web, people could not get the contents or documents what they really need from the search and query processes on the Web. Typically, the semi-structured documents can only be “navigated” by user. It is almost impossible for a web document to be semantically understood by machine without preprocessing.

It is obvious that the main reason Web cannot be precisely queried by users is the lack of semantic metadata of web documents. One typical and well-known solution for users to utilize the web document is Internet Search Engines. It tried to acquire all available web documents on the Web, parse the documents, and generate semantic layer of documents with form of some factors or data structures, such as term frequency (TF), inverted document frequency (IDF), inverted index, or PageRank, etc. However, the semantic layers in the Search Engine are limited, since they are usually designed for full-text information retrieval process. Users may need more advanced retrieval functionalities, such as attribute-based or arithmetic-based query (e.g. Finding all documents describing Ubuntu operating system newer than Ubuntu 8.04), which cannot properly provided by Search Engines.

One of the solutions to enable the ability for web documents utilized like a database is the generating of "schema" of web documents. There must be a plenty of approaches to express the Web with more schema-like manners. The dominant approach is the utilization of Semantic Web standard [1]. The main goal of Semantic Web is to play the role of extension of Web so that information can be linked together at the semantic level and interpreted by machine [2]. In other words, the core of Semantic Web is to provide schema-model-like metadata of web documents, Resource Description Framework (RDF) [3], so that information implicitly embedded in web document set can be operated and queried semantically.

The limitation of Semantic Web standard is the popularity. Currently, the Semantic Web still cannot be widely adopted on the Web, since a large number of un-structure web documents available on the Internet contain texts in natural language that can only be read by human beings. To be properly handled by machine automatically, providing corresponding schema of web documents is the most straightforward way for content providers and developers so that data service providers such as Internet Search Engine can understand semantic of web documents with efficient way. However, for content providers and developers, it is almost impossible to generate such metadata manually. The schema of web document can only be declared and edited by publishers using < meta > tags or other modern annotation methods, such as Open graph [4], microformat [5], or microdata [6], etc. Even when RDF has been recognized as the future standard for schema-model of web documents, publishers must edit and publish the RDF manually. The case will be even worse because RDF must be created and edited following underlying eXtensible Markup Language (XML) syntax. For publishers of Web resource, it is indeed a time-consuming work. It is also impossible to ask publishers of all web documents currently available on Internet to provide the corresponding RDF schema. While some solutions such as [7,8] claims that generating useful annotations as metadata from unstructured web documents is possible, there is still no scalable and semi-automatic solutions to generate semantic metadata of web documents based on the semantic related to the topic implicitly embedded in the content and relationships of web documents.

In this paper, we propose a systematic mechanism generating RDF Semantic Web schema from web document set as the corresponding schema-model-like semantic metadata. By analyzing the structure and content of Web objects in the web document set, they can be conceptualized as resource objects with inter-relationships in RDF diagram. It is also expected that when the semantic metadata of document sets on the Web being systematically translated instead of manually edited, the semantic-ready web documents will be more popular on the web since the Semantic Web standard can be adopted by content providers and developers. The semantic operation on the whole Web, such as semantic query or semantic search, will be therefore possible in the near future. Both content developers and data service providers will be benefit from the web environment with rich semantic natures.

The remainder of this paper is organized as follows. In Section "Literature review", we briefly introduce the related works about the Semantic Web engineering. Section "Solution to generate metadata from Web documents" describes our proposed approach that generates semantic metadata of web documents based on the actual content and relationships of web documents. This approach will then be demonstrated and discussed with the illustrative example in Section "Feasibility study and discussion". Finally, Section "Conclusion" concludes this paper and discusses some future applications.

Literature review

Utilizing Web with semantic manners is always a big challenge. Some solutions and approaches had proposed in order to generate the semantic information of semi-structured web documents, such as:

A. Search Engine: All search engine vendors provide internal semantic layers in their own search engine architectures for full-text information retrieval purposes. For example, the solutions of Google extract information about links and the content of documents by means of keywords. Google's solution also emphasizes the "quality" of links using PageRank model [9]. However, as we discussed in Section "Introduction", such internal semantic layer can be applied for full-text search. Some advanced query mechanisms are not widely supported.

B. Annotation Standards: On the other hands, there are some standards, such as OpenGraph, microformats, and microdata, proposed as extension of Hyper Text Markup Language (HTML) so that the semantic information can be embedded in the web documents as form of HTML elements or attributes. However, such elements or attributes are typically applied to annotate the data pieces in the web document. It is not suitable for "modeling" the web document sets or other textbases.

In order to modeling the web document sets or other textbases with form of schema model, it is potential that Semantic Web Standards can be applied. The Semantic Web is the next-generation Web that can be understood and be processed directly by machine. The scenario of Semantic Web deployment is that the information sources on the Web bring the metadata as semantic in a well-defined format for machine to operate, so that it is possible to support the integrated and uniform access to information sources and service as well as intelligent applications for information processing on the Web [10].

Technically, the Resource Description Framework (RDF) specification, which has become the recommended standard from World Wide Web Consortium (W3C) at 1999, is the most dominant enabler of Semantic Web. RDF is actually the "semantic model" of Web. In the model, any assertions about propositions can be created with simple language [11]. By such simple and formal language, everything on the Web can be treated as individual "resource" with a set of "properties". Concepts about resources can be modeled as the "object-property-value" triple.

Modeling the semantic information embedded among resources on the Web, it is possible for operation of Web documents with more semantic manners. For example, users can perform some attribute-oriented or arithmetic-based query on the whole Web, such as "*ALL documents published by W3C*". The Web, which is currently the largest pool of information resource, can be utilized by users with more effective way by applying proper schema-model-like metadata layer on the Web.

Currently, there are some practical works addressed on the construction of metadata layer on the whole Web and create the user interface to users for querying the Semantic Web by indexing all available schemas on Internet. For example, the Swoogle [12] is a search engine for semi-structured knowledge information. The knowledge can be expressed by either RDF or Web Ontology Language (OWL) [13]. The search engine periodically acquires the knowledge files available on the Web. Users can perform search operations to query the knowledge repository managed by Swoogle. On the other hand, Sindice [14] is another project for Semantic search on the Web. It maintains the index to Semantic Web pages available on the Web. Users can perform semantic search based on either keywords or SPARQL [15]. However, the main problem of evolution of current semantic search engine is the insufficiency of Semantic Web resources on the Web. Nowadays, only a few portions of Web resources are created or maintained following the Semantic Web standard. It is not easy for users to acquire enough results that can be utilized for problem solving purposes.

As for the related works about enabling the Semantic Web and RDF as metadata of information resource, many studies have concentrated on enabling the ability of querying heterogeneous information resources using Semantic-Web-related approaches. For example, Jiang et al. [16] propose an architecture of exposing relational data source to the Semantic Web applications with SPARQL from the object-oriented perspective. Data source from relational database will be properly mapped to corresponding ontology from object-oriented perspective and make run-time translation efficiently. Then the Semantic Web applications can use SPARQL to query the ontologies and retrieve the knowledge back. On the other hand, Chen et al. [17] establish the database-to-ontology mapping functions. With these functions, it will be clarified whether SPARQL can support migration from relational database to semantic ontology, which is expressed by RDF. Once RDF as semantic layer is built, all applications can use SPARQL to search information in RDF data. Database application could be properly migrated as Semantic Web application by replacing SQLs with SPARQL queries.

Yet another category of studies have focused on looking for the translation mechanism to Semantic-Web-enabled information resource from traditional information sources. For example, Krishna [18] introduces a conversion of relational model databases into RDF formats. And the method from de Laborda [19] is to extract the semantic information of a relational database and transform it into Semantic Web metadata including RDF. On the other hand, D2RQ project [20] provides a mapping between relational database schema and Semantic Web concepts. D2RQ takes a relational database schema as input and presents the corresponding RDF interface of as output.

Furthermore, some studies provide solutions for generating RDF as annotations from information resources. For example, [7,8] provide mechanisms using either knowledgebase or natural language processing (NLP) technologies to annotate the content of Web page, and express the whole annotation map by RDF. However, the annotations in previous works are not enough for semantic search operations, since 1) the linguistic annotation, such as annotation of "part of speech" around the data pieces of the web document, cannot satisfy users, because users tend to get the answer about the questions which motivate users to search on the Web, and 2) semantic extraction of unstructured content using language grammars/parsers is not scalable.

There are also some studies focused on the conceptualization of Web documents. For example, Gu et al. [21] proposed a description method to express the structural content of Web pages using RDF. The structured parts in Web documents can then be conceptualized using RDF diagram. However, the conceptualization solutions totally based on structural information, such as < meta > tags, hyperlinks, or Resource Description Framework-in-attributes (RDFa) [22] information, still have some bottlenecks for semantic search operations. The most drawback is that the semantic and information that users want to query are often not available in the structural information of Web document sets. Analyzing the hyperlinks or other information cannot imply that users can query the Web document set with more semantic manners. For unstructured content, the human computing is the only way to be applied if users are interested in the semantic of such content.

In summary, RDF is indeed a useful data model to express semi-structured Web document sets. It has been widely adopted in many Semantic-Web-related literatures. However, there are still neglects for systematic or semi-automatic mechanism to generate the data model of Web information resources based on both the content and structure of Web document sets so that content publishers can maintain the semantic and users can utilize the information resource effectively.

Solution to generate metadata from Web documents

This paper proposed a mechanism for constructing Semantic Web with bi-directional approach: For content providers and developers, it is necessary to generate the schema-model-like metadata as semantic information of web resource/documents they maintained; Data service providers such as search engine vendors can acquire and maintain the semantic information on the whole Web so that it is possible for semantic search including attribute-oriented or arithmetic-based query operations. Table 1 discussed the design choice of proposed solution:

Figure 1 illustrates the proposed solution to generate the corresponding schema of Web site.

Table 1 Design choice of solution

Approach	Pros and cons
Top-down	Using traditional search engine index as semantic layer ●→**Pros:** No need for content providers and developers to generate semantic information. Search engine will discover the semantic information ●→**Cons:** Current search engines do not support advanced operations
Bottom-Up	Users provided annotations for data pieces in web documents ●→**Pros:** For users it is easier to annotate data pieces, there are specifications and standards for users to follow. ●→**Cons:** The annotations in the web documents may not relate to the topic or semantic of web documents because anything in the web document can be annotated. For data service providers it might not be useful for semantic engineering.
Bi-directional	●→**Pros:** It is easier for data service providers to provide semantic search schemes because schema of available web document can be collected easier due to the popularity ●→**Cons:** Systematic approach to generate schema is required to motivate the content providers and developers generating schema of web documents they maintained

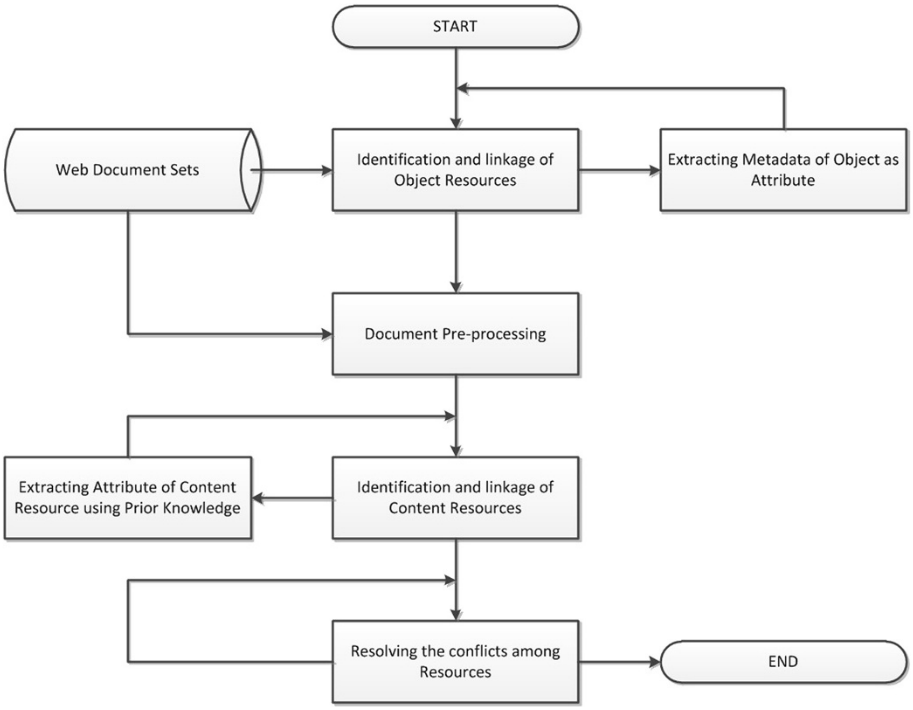

START
Web Document Sets
Identification and linkage of Object Resources
Extracting Metadata of Object as Attribute
Document Pre-processing
Extracting Attribute of Content Resource using Prior Knowledge
Identification and linkage of Content Resources
Resolving the conflicts among Resources
END

Figure 1 **Process to systematically generate Semantic Web metadata.**

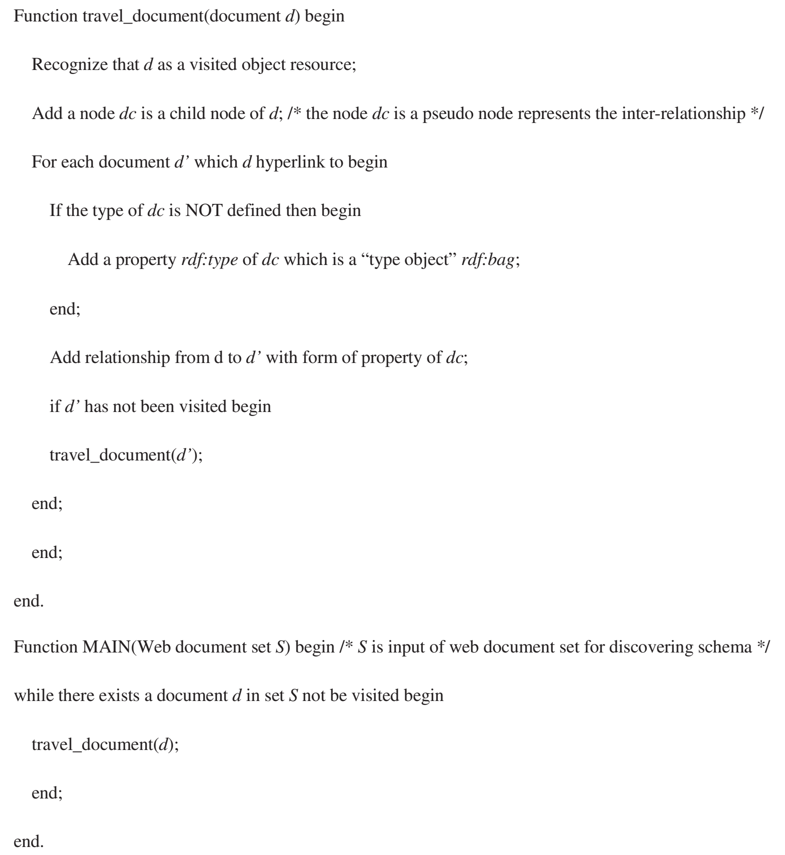
It is indeed that the bi-directional approach is a feasible choice for adoption of Semantic Web standards. The main goal of this work is to generate Semantic Web metadata, which is expressed by RDF, of a Web site or other Web document sets systematically. In this article, two types of resource in the Web can be obtained and expressed in RDF:

- Object Resource: The real object, such as files, can be identified as “resources” in RDF.
- Content Resource: On the other hand, some objects or concepts which are appeared in content of documents can also be identified as “resources” in RDF.

To achieve the goal, the following steps are necessary to extract semantic information from Web document set:

Step 1 Identification and linkage of Object Resources

The first step involves the identification of object resources based on file structure and hyperlinks. The object resources can be identified based on URL and hyperlink. For any Web document set, the set of Web resource with inter-relationship information can be obtained and expressed as a preliminary RDF diagram using a crawler-like algorithm, as shown in the following Figure 2:



```
Function travel_document(document d) begin
  Recognize that d as a visited object resource;
  Add a node dc is a child node of d; /* the node dc is a pseudo node represents the inter-relationship */
  For each document d' which d hyperlink to begin
    If the type of dc is NOT defined then begin
      Add a property rdf:type of dc which is a "type object" rdf:bag;
    end;
    Add relationship from d to d' with form of property of dc;
    if d' has not been visited begin
    travel_document(d');
  end;
  end;
end.
Function MAIN(Web document set S) begin /* S is input of web document set for discovering schema */
while there exists a document d in set S not be visited begin
  travel_document(d);
  end;
end.
```

Figure 2 Algorithm to generate relationships among object resources.

Basically, the structural relationships among object resources can be established by traversal of Web document set via hyperlink. It should be noted that:

1) There might be no semantic relationships among resources which have structural interrelationships.
2) For any web document set, one or more "entry points" might be available. That is, the documents can be considered as one or more tree structures conceptually. Documents (nodes) will be connected by hyperlinks (edges) in tree structures.

Step 2 Extracting metadata of Object Resource as Attributes of Object Resource

This step is responsible for extracting the metadata of object resource. Minimally, such metadata information includes the basic file information, such as file size and file authors, and the content information, such as MIME type or character encoding information which can be defined in < meta > fields in a Web document. It is indeed that the metadata must be translated as the properties of an object resource.

Step 3 Document Pre-processing

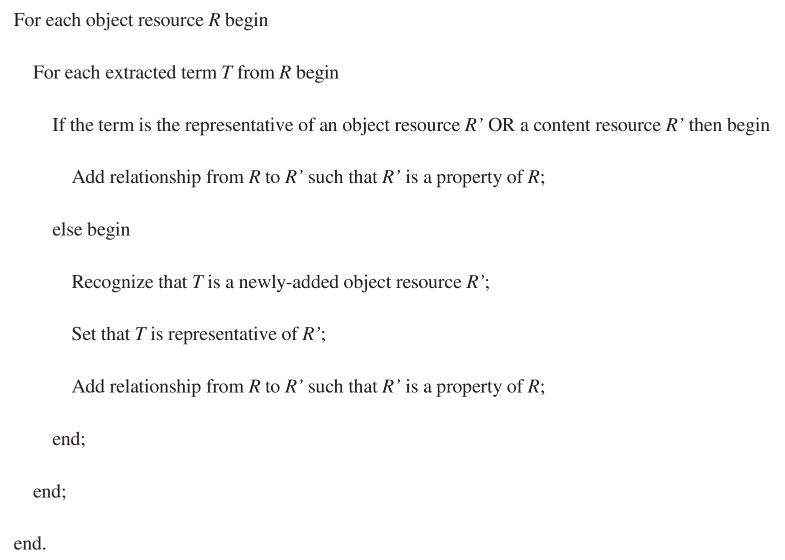
Typically, the content resources, which reflect some objects or concepts, are embedded in the content of object resource. In order to extract valuable information, the un-structured Web documents must be pre-processed so that the information embedded in documents can be handled automatically. Theoretically, all noun terms are potential content resources which can reflect some objects or concepts. In this article, however, for simplicity consideration, the information to be extracted only includes the keywords and the terms, which are already the representatives of extracted objects preliminary RDF diagram. For keyword extraction, there are many approaches to extract the keywords from Web documents. The most common way is the weighting approaches, such as TF-IDF factor, to determine the set of keywords of one document by calculating the "weight" of a term in a document. By this step, a set of terms are extracted as the potential representatives of content objects.

Step 4 Identification and linkage of Content Resources

In this step, content resources will be generated according to the relationships among extracted terms from previous step and preliminary RDF diagram. The following algorithm shown in the Figure 3 refines the preliminary RDF diagram using extracted term set:

Basically, the terms extracted as keywords will be considered as potential "content resource" in the step, since a keyword, such as "W3C" or "Linux", often reflects some physical objects or concepts. In this step, the inter-relationship among object resources and content resources will be preliminarily connected. It is then the semantic relationship among Web resources and concepts.

Step 5 Extracting Attributes of Content Resources



```
For each object resource R begin
  For each extracted term T from R begin
    If the term is the representative of an object resource R' OR a content resource R' then begin
      Add relationship from R to R' such that R' is a property of R;
    else begin
      Recognize that T is a newly-added object resource R';
      Set that T is representative of R';
      Add relationship from R to R' such that R' is a property of R;
    end;
  end;
end.
```

Figure 3 Algorithm to generate relationships among content resources.

In this step, the attributes of content resources must be extracted. The method to extract the attribute name about certain concept is out of the scope of this article. There are many approaches to extract attribute information of one concept from Web document automatically. Some common ways include the solution to extract attribute from structural part of Web document, or solutions using prior knowledge to determine the potential attribute with attributes of certain concept [23]. For any attribute of one concept to be extracted from Web document, it must be defined as the attribute of the corresponding content resource reflect the concept. After the step is done, all available resources, attributes, and inter-relationships will be extracted and expressed on the RDF graph.

Step 6 Resolving the conflicts among resources

Different from database resource, the Web document set cannot be normalized in order to keep the data consistency and storage minimization. There might be redundant or even conflict resource or attribute items in the Web document set. For example, some documents indicated that the newest kernel version of Linux is “3.2”, while some out-of-date documents still said that the newest kernel version is “2.6.18”. Strategically, to resolve the conflict among resource can be systematically done by the following procedure, as shown in Figure 4.

The first sub-step involves the resolution of linguistic conflicts in the RDF diagram, which might occur in resources identifications, attribute names, or even the values. Using thesauri or other prior knowledge, the conflict, such as synonym or homonym, must be eliminated first. For example, if two web documents represents identical attribute name with different value:

- *D(A): {Keyword = “Ubuntu”, CurrentVersion = “8.04”}*
- *D(B): {Keyword = “Ubuntu”, CurrentVersion = “12.04”}*

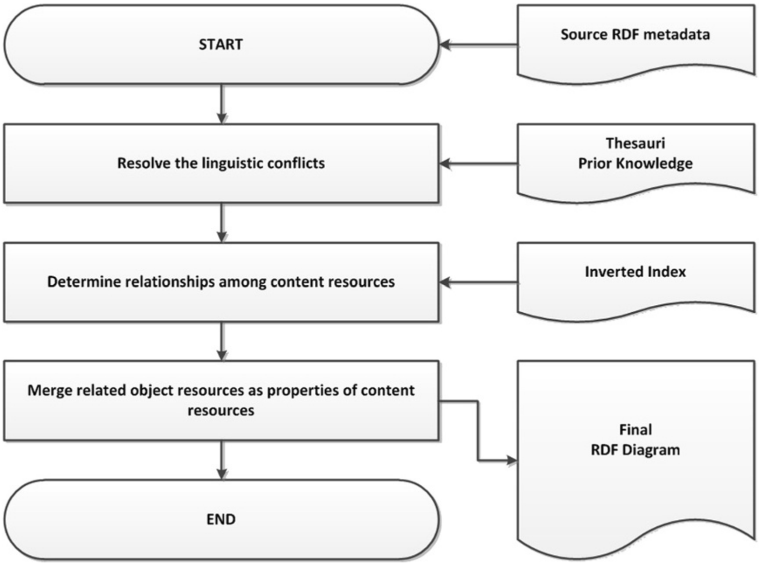

START
Source RDF metadata
Resolve the linguistic conflicts
Thesauri
Prior Knowledge
Determine relationships among content resources
Inverted Index
Merge related object resources as properties of content resources
Final
RDF Diagram
END

Figure 4 Refinement of RDF graph.

From the prior knowledge, the *CurrentVersion* attribute value in A must be ignored.

Next, the inter-relationships among content resources should be identified. In this article, the inter-relationship can be modeled by the "similarity" of two resources. Since the content resources come from the "keywords" extracted from documents, "keywords" should be representatives of content resources. The similarity of any two content resources *X* and *Y* can be calculated by the probability of co-occurrence of two representative and conflict-free keywords. The relationships among two content resources can be recognized if the similarity value exceeds some threshold:

$$Similarity(X,Y) = \frac{P(X \cap Y)}{P(X \cup Y)} \geq \varepsilon$$

The last sub-step is responsible to annotate the relationship from content resources to object resources. The semantic of such relationship is to identify "ALL real objects which are related to some concepts". Strategically, in this sub-step, all object resources related to a content object can be merged as a "Bag" property of the content object. Like an inverted index, it is therefore possible for users to query about resources relevant to some concepts. After the conflicts are resolved, the final RDF diagram which represents the metadata of Web document set can be obtained.

Feasibility study and discussion

In this section, a set of experiments are applied to demonstrate the feasibility of proposed mechanism to translate a Web document into corresponding RDF model. We also discuss some lesson learned in this section.

Considering the Web document *D* = http://www.w3.org/News/2011#entry-9116. From the web document, it can be obtained that *D* is actually an anchor section of Web document http://www.w3.org/News/2011 and contains a News paragraph about Cascading Style Sheets (CSS). The keyword set extracted from *D* is *{CSS, W3C}*. *D* has linked to Web document set *{D1, D2, D3, D4, D5, D6, D7}*. Partial RDF diagram of *D* can be visualized as shown in Figure 5:

Please note that in this case, only the semantic directly related to *D* is shown in the graph and only one content resource *{CSS}* is visible as illustrative demonstration in the graph. Semantic relationships about another content resource *{W3C}* are not shown here.

The translated XML document of the RDF is shown in Figure 6:

It should be noted that the all resources in Semantic Web will be identified using Uniform Resource Identifier (URI), which may be denoted as Universal Resource Locator (URL) or Universal Resource Name (URN). For example, the content resource "CSS" can be identified using the URN "*urn:object:CSS*".

Since the feasibility of proposed mechanism is basically certified, in order to demonstrate the advantage of proposed mechanism, we apply a set of illustrative experiments to compare the effectiveness of proposed mechanism with previous studies [7] and [21].

The Extractiv project [7] provided a knowledge-engineering-based mechanism to generate the annotation for "contents in web documents". The basic principle is similar to the microdata approach proposed in HTML 5: To annotate according to known ontology and prior-knowledge. The Figure 7 illustrates the semantic information of a web document *Z* = http://www.ubuntu.com generated from Extractiv project:

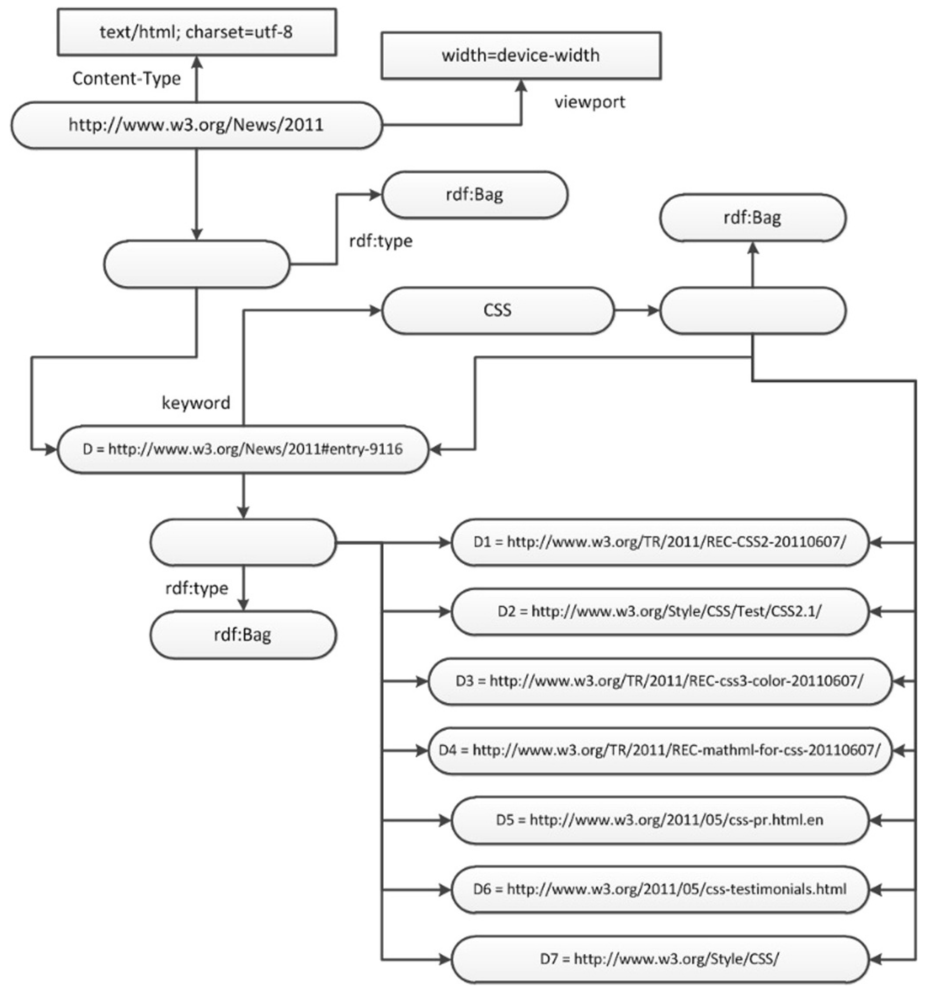
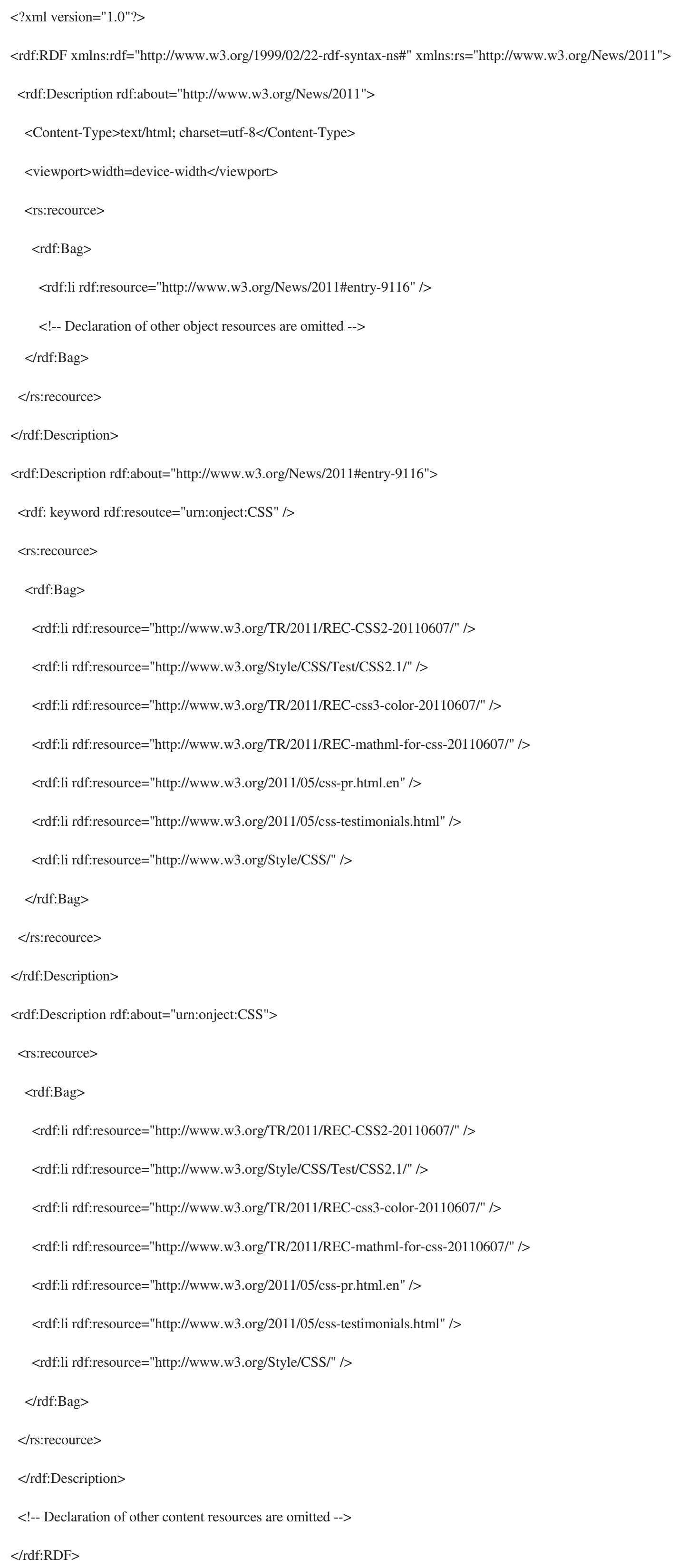

text/html; charset=utf-8
width=device-width
Content-Type
viewport
http://www.w3.org/News/2011
rdf:Bag
rdf:type
rdf:Bag
CSS
keyword
D = http://www.w3.org/News/2011#entry-9116
rdf:type
rdf:Bag
D1 = http://www.w3.org/TR/2011/REC-CSS2-20110607/
D2 = http://www.w3.org/Style/CSS/Test/CSS2.1/
D3 = http://www.w3.org/TR/2011/REC-css3-color-20110607/
D4 = http://www.w3.org/TR/2011/REC-mathml-for-css-20110607/
D5 = http://www.w3.org/2011/05/css-pr.html.en
D6 = http://www.w3.org/2011/05/css-testimonials.html
D7 = http://www.w3.org/Style/CSS/

Figure 5 **Partial result of generated RDF diagram.**

In the generated semantic information, users can easily obtain some types of "entities" in some location of the document. However, the main drawback of such solution is the requirement of inventing extremely rich prior knowledge. For example, the Extractiv system must understand that "Ubuntu is an Operating System". The generation process will be failed if no knowledgebase or ontology presented. On the other hand, the generated semantic information is not always useful for users. For instance, the fact that "2012 is an instance of DATE-TIME" might not be helpful for users with question answering purposes.

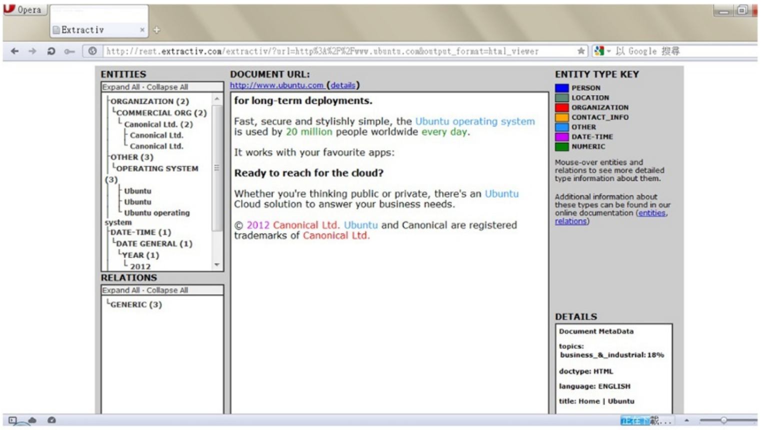
On the other hand, there are indeed some studies, such as [21], try to introduce approaches describing semantic of the web document based on the structural metadata. The Figure 8 shows partial RDF diagram of *D* using approach in [21]:

It is obvious that the generated RDF in [21] is based on 1) the structural relationships among object resources, and 2) the structural metadata in < meta > tags that can be easily acquired from the web document. However, the unstructured semantic information, which might be more valuable for users, cannot be extracted and expressed in the RDF diagram by such approach. The query on < meta > tag information such as



```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:rs="http://www.w3.org/News/2011">
  <rdf:Description rdf:about="http://www.w3.org/News/2011">
    <Content-Type>text/html; charset=utf-8</Content-Type>
    <viewport>width=device-width</viewport>
    <rs:recource>
      <rdf:Bag>
        <rdf:li rdf:resource="http://www.w3.org/News/2011#entry-9116" />
        <!-- Declaration of other object resources are omitted -->
    </rdf:Bag>
  </rs:recource>
</rdf:Description>
<rdf:Description rdf:about="http://www.w3.org/News/2011#entry-9116">
  <rdf: keyword rdf:resoutce="urn:onject:CSS" />
  <rs:recource>
    <rdf:Bag>
      <rdf:li rdf:resource="http://www.w3.org/TR/2011/REC-CSS2-20110607/" />
      <rdf:li rdf:resource="http://www.w3.org/Style/CSS/Test/CSS2.1/" />
      <rdf:li rdf:resource="http://www.w3.org/TR/2011/REC-css3-color-20110607/" />
      <rdf:li rdf:resource="http://www.w3.org/TR/2011/REC-mathml-for-css-20110607/" />
      <rdf:li rdf:resource="http://www.w3.org/2011/05/css-pr.html.en" />
      <rdf:li rdf:resource="http://www.w3.org/2011/05/css-testimonials.html" />
      <rdf:li rdf:resource="http://www.w3.org/Style/CSS/" />
    </rdf:Bag>
  </rs:recource>
</rdf:Description>
<rdf:Description rdf:about="urn:onject:CSS">
  <rs:recource>
    <rdf:Bag>
      <rdf:li rdf:resource="http://www.w3.org/TR/2011/REC-CSS2-20110607/" />
      <rdf:li rdf:resource="http://www.w3.org/Style/CSS/Test/CSS2.1/" />
      <rdf:li rdf:resource="http://www.w3.org/TR/2011/REC-css3-color-20110607/" />
      <rdf:li rdf:resource="http://www.w3.org/TR/2011/REC-mathml-for-css-20110607/" />
      <rdf:li rdf:resource="http://www.w3.org/2011/05/css-pr.html.en" />
      <rdf:li rdf:resource="http://www.w3.org/2011/05/css-testimonials.html" />
      <rdf:li rdf:resource="http://www.w3.org/Style/CSS/" />
    </rdf:Bag>
  </rs:recource>
  </rdf:Description>
  <!-- Declaration of other content resources are omitted -->
</rdf:RDF>
```

Figure 6 **The XML expression of generated RDF diagram.**


Opera
Extractiv
http://rest.extractiv.com/extractiv/?url=http%3A%2F%2Fwww.ubuntu.com&output_format=html_viewer
以 Google 搜尋
正在下載...

ENTITIES

Expand All · Collapse All

ORGANIZATION (2)
COMMERCIAL ORG (2)
Canonical Ltd. (2)
Canonical Ltd.
Canonical Ltd.
OTHER (3)
OPERATING SYSTEM (3)
Ubuntu
Ubuntu
Ubuntu operating system
DATE-TIME (1)
DATE GENERAL (1)
YEAR (1)
2012

RELATIONS

Expand All · Collapse All

GENERIC (3)

DOCUMENT URL:

http://www.ubuntu.com (details)

for long-term deployments.

Fast, secure and stylishly simple, the Ubuntu operating system is used by 20 million people worldwide every day.

It works with your favourite apps:

Ready to reach for the cloud?

Whether you're thinking public or private, there's an Ubuntu Cloud solution to answer your business needs.

© 2012 Canonical Ltd. Ubuntu and Canonical are registered trademarks of Canonical Ltd.

ENTITY TYPE KEY

🟦 PERSON
🟩 LOCATION
🟥 ORGANIZATION
🟧 CONTACT_INFO
🟦 OTHER
🟪 DATE-TIME
🟩 NUMERIC

Mouse-over entities and relations to see more detailed type information about them.

Additional information about these types can be found in our online documentation (entities, relations)

DETAILS

Document MetaData

topics:
business_&_industrial: 18%

doctype: HTML

language: ENGLISH

title: Home | Ubuntu

Figure 7 Semantic annotation discovered from Extractiv.

“Finding all documents with UTF-8 character encoding” might not be helpful for users with question answering purposes.

It is indeed that the proposed mechanism has some advantages compared to previous approaches. As for the application and adoption of proposed mechanism, it is potential to be the core technology of Semantic Web search engine based on Semantic Web or RDF [24,25]. The main characteristic of such search engine is to provide a semantic layer in the search engine so that users can perform semantic search operation based on semantic layer. With such automatically-generated RDF as metadata of Web document set, it is therefore possible for query operations on Web document sets with more semantic manner. For example, when users want to query about all documents about “CSS”, it is easy to acquire the result set *{D, D1, D2, D3, D4, D5, D6, D7}* from the semantic description of the RDF graph. Under such scenario, combining the current keyword-based information retrieval technology in finding potential semantic

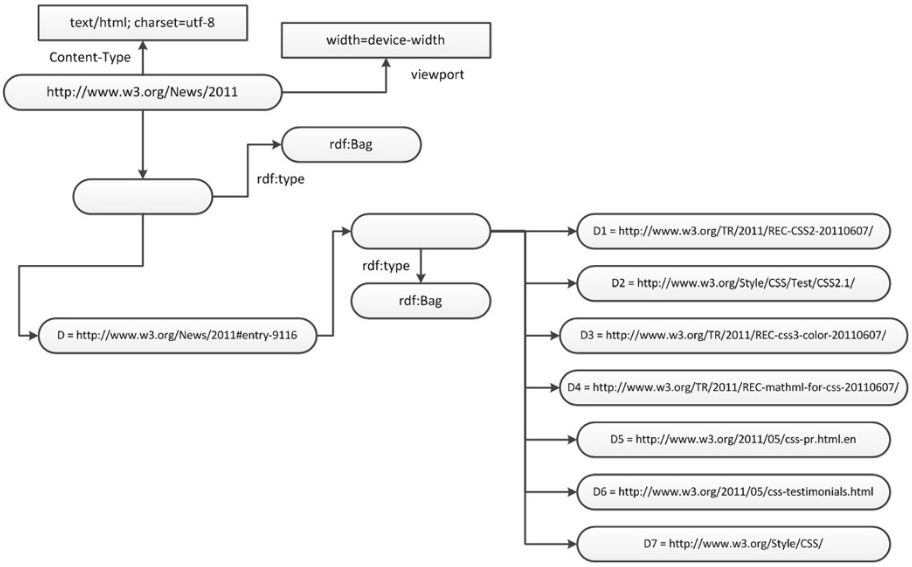

text/html; charset=utf-8
Content-Type
width=device-width
viewport
http://www.w3.org/News/2011
rdf:Bag
rdf:type
rdf:type
rdf:Bag
D = http://www.w3.org/News/2011#entry-9116
D1 = http://www.w3.org/TR/2011/REC-CSS2-20110607/
D2 = http://www.w3.org/Style/CSS/Test/CSS2.1/
D3 = http://www.w3.org/TR/2011/REC-css3-color-20110607/
D4 = http://www.w3.org/TR/2011/REC-mathml-for-css-20110607/
D5 = http://www.w3.org/2011/05/css-pr.html.en
D6 = http://www.w3.org/2011/05/css-testimonials.html
D7 = http://www.w3.org/Style/CSS/

Figure 8 Partial result of generated RDF diagram using structural approach.

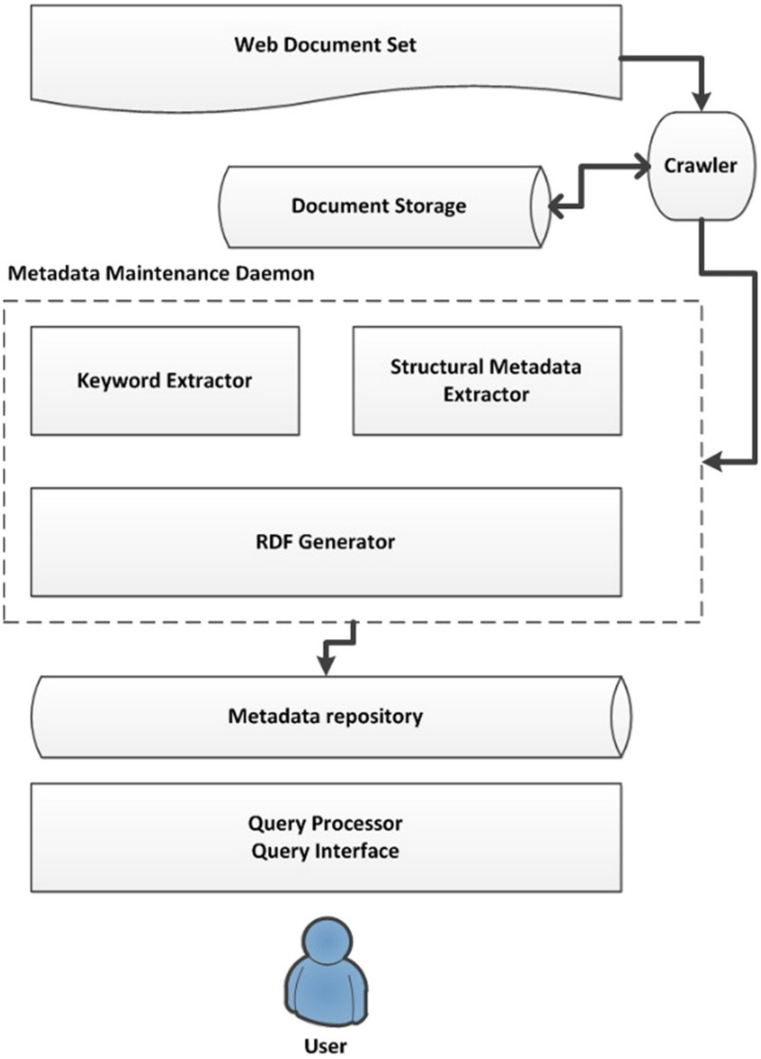


Figure 9 Architecture of semantic search engine.

information and formulating in RDF schema is a potential key technology for current search engine to seamlessly enable the semantic search functions.

As for the implementation of RDF generator in semantic search engine on Semantic Web, a referential architecture can be illustrated as shown in Figure 9. It should be noticed that in this article, the bi-directional approach is recommended that content providers and developers should be responsible for generating of RDF schema of web documents they maintained. However, implementation of RDF generator in semantic search engine is still required because the popularity of “schema-ready” web documents will highly depend on the willing of content providers and developers to generate the RDF schema with either manually or systematically approaches.

The main goal of semantic search engine illustrated is to provide a metadata repository, where the metadata with RDF form is automatically generated and maintained, as semantic layer for users to perform semantic query. In the referential architecture, the goal will be achieved by maintenance daemon. There are at least five modules included in the daemon:

- Crawler: For any Internet search engine, a crawler will be necessary to perform as backend service to acquire available Web resource.
- Structural Metadata Extractor: The module is utilized to extract structural information from Web document set created by crawler. Any structured and analyzable part of Web documents will be extracted and transformed into metadata, which described the structure information of the Web document set. In other words, the module is the extractor of object resources and performing the Step 1 and Step 2 of the proposed mechanism.
- Keyword Extractor: The module is responsible to extract semantic of content in Web documents. Based on keyword extraction methods, keywords reflecting certain concepts can be extracted from the content of Web documents. In other words, the module is the extractor of object resources and performing the Step 3 to Step 5 of the proposed mechanism.
- RDF Generator: The module is responsible for generating of RDF metadata, which will be physically expressed by extensible markup language (XML) or other formats, from extracted semantic and structural information. On the other hand, the module must reconcile the heterogeneity and conflicts come from metadata of different Web documents. In other words, the module is the extractor of object resources and performing the Step 6 of the proposed mechanism. In this module, external prior knowledge or thesauri might be necessary.
- Query Processor and Interface: The module is actually a human-machine interface. Users' intension must be properly expressed using some query formats such as SPARQL or XQuery [26] that can query the metadata repository directory. The processor will execute and return the query result back to users. In the module, a ranking mechanism based on the nature of Semantic Web, which is out of the scope of this article, might be necessary in order to determine the relevance of results to users' intension.

In summary, the Table 2 provides discussion and comparisons with other methods which are illustrated in Section "Literature review".

Table 2 Discussion and comparisons with other methods

Approach	Discussion and comparisons
Search Engine for Semantic Web [12,14]	The search space depends on the acquirable schema or other semantic information. No discussions for generating semantic information from semi-structured documents.
Middleware handling heterogeneity [16,17]	The middleware or translator can map or convert schema between structured data sources and Semantic Web schema. It is infeasible on the cases of semi-structured or even un-structured data resources.
Schema Translation from database [18-20]	
Schema generation from document based on knowledge engineering [7,8]	Current schemes can generate either linguistic or semantic annotation of data pieces in web documents using prior-knowledge or NLP technologies. It is not suitable for "modeling" the web document sets or other textbases. For problem-solving or topic search purposes, the solutions are not sufficient.
Schema generation from document based on structural part of document [21]	The structure-based approaches generate RDF only based on structural part of document. Such solution is simple to implement, while the generated RDF might not be helpful for users for question solving purposes.

Conclusion

In this paper, we propose a six-step systematic mechanism generating RDF Semantic Web schema from Web document set as the corresponding schema-model-like semantic metadata. In our approach, different from previous studies and solutions, both structural information and content information are analyzed using prior knowledge. Schema-model-like semantic information can therefore be generated systematically from our mechanism. By analyzing strategies for link and concept extraction, Web resources can be conceptualized as resource objects with inter-relationships in RDF schema diagram. We also demonstrate the feasibility of proposed mechanism using an illustrative case study. It is also expected that the proposed mechanism is general applicable. First, it is feasible to be the core technology of next-generation search engine. In this article, we discuss the architecture of semantic search engine on Semantic Web based on RDF and the proposed mechanism. With the semantic metadata of document sets on the Web being systematically translated instead of manually edited by either content providers or data service providers, the semantic operation on the whole Web, such as semantic query or semantic search based on certain semantic layer, will be possible in the near future. Furthermore, it is applicable as one of important module in web document development software. Many data service providers, such as Google rich snippet project [27], encourage content providers publish web document with rich semantic. The proposed mechanism is a feasible way for content providers develop semantic information systematically and semi-automatically. The reputation of such web document is expected to be basically certified and admired.

As for the future directions, the most important work is to enrich the schema of the web documents. It is indeed that in the schema, "keyword" might not be the only content resource to be extracted although in this article the scope is limited in keywords. It is recommended that the RDF schema will be more completed by integrating other categories of "semantic-like content", such as annotations, tags, or other acquirable properties.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

H-YH designed, conceptualized, and evaluated the solutions proposed in this article. C-NC and K-FH identified the innovation of proposed solution in this article by collecting and reviewing all available previous studies. All authors drafted the manuscript. H-YH refined the manuscript iteratively and did the final revision. All authors read and approved the final manuscript.

Author details
[1]Computational Intelligence Technology Center, Industrial Technology Research Institute, Taiwan, R.O.C. [2]Chunghwa Telecom Laboratories, Taiwan, R.O.C. [3]Information & Communication Research Lab, Industrial Technology Research Institute, Taiwan, R.O.C.

Received: 6 January 2012 Accepted: 16 April 2013
Published: 10 May 2013

References

1. (2010) Semantic Web, http://www.w3.org/standards/semanticWeb
2. Raimbault T (2010) Overviewing the RDF(S) Semantic Web. Proceedings of International Conference on Computational Intelligence and Software Engineering (CiSE 2010). IEEE Press, Wuhan, China, pp 1–4
3. (2004) Resource Description Framework (RDF), http://www.w3.org/RDF
4. (2012) Open Graph (Facebook Developers), https://developers.facebook.com/docs/opengraph
5. (2012) About Microformats, http://microformats.org/about
6. (2011) HTML5.1 Nightly: A vocabulary and associated APIs for HTML and XHTML, http://www.w3.org/html/wg/drafts/html/master/Overview.html
7. (2011) Extractiv Project, http://www.extractiv.com/

8. Mukhopadhyay D, Kumar R, Majumdar S, Sinha S (2007) A New Semantic Web Services to Translate HTML Pages to RDF. Proceedings of 10th International Conference on Information Technology (ICIT 2007). IEEE Press, Orissa, India, pp 292–294
9. Brin S, Page L (1998) The Anatomy of a Large-Scale Hypertextual Web Search Engine. Computer Networks and ISDN Systems 30(1–7):107–117
10. Decker S, Mitra P, Melnik S (2000) Framework for the Semantic Web: An RDF Tutorial. IEEE Internet Computing 4(6):68–73
11. Agarwal PR (2012) Semantic Web in Comparison to Web 2.0. Proceedings of 3rd International Conference on Intelligent Systems, Modelling and Simulation (ISMS). IEEE Press, Kota_Kinabalu, Malaysia, pp 558–563
12. Finin T, Ding L, Pan R, Joshi A, Kolari P, Java A, Peng Y (2005) Swoogle: Searching for knowledge on the Semantic Web. Proceedings of the 20th national conference on Artificial intelligence (AAAI 2005). AAAI Press, Pittsburgh, Pennsylvania, USA, pp 1682–1683
13. (2004) Web Ontology Language (OWL), http://www.w3.org/2004/OWL
14. Oren E et al (2008) Sindice.com: A Document-oriented Lookup Index for Open Linked Data. International Journal of Metadata, Semantics and Ontologies 3(1):37–52
15. (2008) SPARQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query
16. Jiang H, Ju L, Xu Z (2009) Upgrading the relational database to the Semantic Web with Hibernate. Proceedings of International Conference on Web Information Systems and Mining (WISM 2009). IEEE Press, Shanghai, China, pp 227–230
17. Chen Y, Yang X, Yin K, Ho A (2008) Migrating Traditional Database-based Systems onto Semantic Layer, Proceedings of International Conference on Computer Science and Software Engineering (CSSE 2008), 4. IEEE Press, Wuhan, Hubei, China, pp 672–676
18. Krishna M ( ) Retaining Semantics in Relational Databases by Mapping them to RDF. Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2006). IEEE Press, Hong Kong, China, pp 303–306
19. de Laborda C (2006) Bringing Relational Data into the Semantic Web using SPARQL and Relational OWL. Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDE 2006). IEEE Press, Atlanta, GA, USA, p 55
20. Bizer C (2004) D2RQ - Treating Non-RDF Databases as Virtual RDF Graphs. Proceedings of the 3rd International Semantic Web Conference (ISWC2004). Hiroshima, Japan
21. Gu Y, Dan L (2010) Web resources description model based on RDF. Proceedings of 2010 International Conference on Computer Application and System Modeling (ICCASM 2010), pp V9-222-V9-225
22. (2008) RDFa1.1 Primer: Rich Structured Data Markup for Web Documents, http://www.w3.org/TR/xhtml-rdfa-primer/
23. Nakane F, Otsubo M, Hijikata Y, Nishida S (2008) A basic study on attribute name extraction from the Web. Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC 2008). IEEE Press, Singapore, pp 2161–2166
24. Jin Y, Lin Z, Lin H (2008) The Research of Search Engine Based on Semantic Web. Proceedings of International Symposium on Intelligent Information Technology Application Workshops (IITAW 2008). IEEE Press, Shanghai, China, pp 360–363
25. Priebe T, Schlager C, Pernul G ( ) A Search Engine for RDF Metadata. Proceedings of 15th International Workshop on Database and Expert Systems Applications (DEXA 2004). IEEE Press, Zaragoza, Spain, pp 168–172
26. (2011) XQuery 1.0: An XML Query Language, Secondth edn, http://www.w3.org/TR/xquery
27. (2012) Rich snippets (microdata, microformats, and RDFa), http://support.google.com/webmasters/bin/answer.py?hl=en&answer=99170

doi:10.1186/2192-1962-3-7
Cite this article as: Hsueh *et al.*: Generating metadata from web documents: a systematic approach. *Human-centric Computing and Information Sciences* 2013 **3**:7.