The International Journal of Engineering and Information Technology

journal homepage:www.ijeit.misuratau.edu.ly

Embedding Structure into HTML for More Precise Retrieval of Information, A Novel XML Schema

Anwar Alhenshiri alhenshiri@it.misuratau.edu.ly

Zainab Afat z.m.afat2014@gmail.com

Hoda Badesh h.badesh@it.misuratau.edu.ly

Abstract— **This paper presents the core of a universal schema to transform each HTML document into XML format. The objective is to embed a sense of structure into textual documents prior to retrieving information. The structure is obtained from the HTML document based on the schema and applied in the form of an XML document. The resulting structure helps with identifying levels of significance in the HTML page. More relevant results can be obtained by including the hidden structure of the text document in the computation of relevancy during retrieval. The preliminary study indicates potential success with larger studies.**

Index Terms: **information retrieval, databases, HTML, XML, schema, structured retrieval, query, search, relevancy.**

I. INTRODUCTION

Web pages are written in HTML (Hyper Text Markup Language) format. Little structure is included in the format of HTML. In addition, most search engines ignore the structure and rely on text retrieval (term matching). The relevance of a document is computed first based on how many terms of the query can be found in the document. Other factors are then considered such as PageRank (Bring and Page, 1998). Even though some studies have highlighted the benefit of word location in the document (Kim et. al., 2006, Jiang, 2020), the structure of the text in the HTML document is seldom utilized in retrieval.

HTML documents can be written almost in any way and by almost anyone. The W3C allows variations in writing HTML with few restrictions at least until HTML5. This freedom resulted in having no general appearance (W3C, 2020) in HTML documents. Therefore, web pages are usually crawled and the pure human-readable text is extracted for further matching. Even with the structure given to organizing the text in special tags, that organization conveys little significance to each part of the document. That significance is obtained from the way a human being looks at the text.

To benefit from the structure of HTML documents, every element is considered at a hidden level of priority. This priority cannot be seen using the current vector space model used in representing text documents (Salton et. al, 1975). Therefore, the need is for a more structured format of text including the degree of priority, i.e. XML format. The schema proposed in this project is intended to move the current representation (mostly textual) of HTML documents into a structured form of XML.

In this study, a schema, i.e. a general structure is obtained from HTML and modeled into XML format. The schema demonstrates how the different parts of HTML can be prioritized and given different levels of significance. An HTML document is represented into a structure in which the main items are given the highest level of significance followed by the subitems and so on. When comparing the terms of a query to those of a document, not only does the word match matters but also the location of the term in the XML form of the document.

The main goal of this idea is to improve the user experience while using search engines to find relevant results (improving relevancy). That is by considering the position of the terms in the document in the conveyed structure in the XML format in addition to the known factors currently used in search and matching approaches.

The project is at a starting stage and more improvement is anticipated in the future. The model is created with a schema covering the main text items in the HTML format. More elements will be obtained and further enhancements to the schema will be applies. The research will also continue with conducting a large-scale study in which different topics within larger datasets will be used in the experimentation.

The remainder of this paper is organized as follows. Section 2 presents research work related to the subject of this article. Section 3 illustrates the schema design. Section 4 has the preliminary testing and evaluation part of the project. The results followed by a discussion of the

Received 3 Oct ,2020; revised 10 Dec, 2020; accepted 20 Dec, 2020.

Available online 23 Dec, 2020.

main findings are shown in Section 5. The paper is concluded in Section 6.

II. RELATED WORK

There have been many research attempts to tackle the problem of retrieving relevant results to users of Web information. For example, Lennon and Malkovich (2006) focused on one type of search on the Internet, and they discussed a method to extract information from text fragments found within a search engine, because search engines retrieve web pages, not the information itself. Then, the user has to search within the search results in order to acquire the information. They presented an algorithm that extracts, structures and combines information obtained from an internet search engine. The method of the algorithm is based on hand-crafted patterns which are tailor-made for the classes and relations considered and queried on them to Google. The results are scanned for new instances. Instances found can be used within these patterns as well, so the algorithm can populate an ontology based on a few instances in a given partial ontology. They found the results of the experiment to be encouraging.

Baeza-Yates et al. (2007) tried to achieve high quality answers, fast response time, high query throughput, and scalability. They assumed that they have 20 billion Web pages, which suggests at least 100 terabytes of text or an index of around 25 terabytes. For efficiency purposes, a large portion of this index must fit into the computer RAM (Random Access Memory). Using computers with several gigabytes of main memory, they needed approximately 3000 of them in each cluster to hold the index. They found that designing a distributed system is difficult because it depends on several factors that are seldom independent. Moreover, they found that designing such system depends on so many considerations that one poor design choice can affect performance adversely or increase cost. For example, changes to the data structures that hold the index may impact response time.

Sanderson et al. (2010) found that the evaluation of information retrieval systems relies on relevance judgments—human assessments of whether a document is relevant to a specified search request or not. This analysis shows that changes are not due to random error, but instead, reflects a relevance shift, whereby the assessor's conception of what constitutes a relevant document changes over time. Studying types of relevant judgment, they found that the shift in judgments is greatest between highly and partially relevant documents. They also examined the impact of this inconsistency on how retrieval runs and found that there appears to be a noticeable effect on such rankings.

Olsen (2011) studied information retrieval with respect to the search results. They found out that many of the search results share the same keywords. The result sets become too large to be efficiently presented to the user. He invented a method for computing summary information from documents containing hierarchies named scopes comprising a plurality of associations between a scope and a value or between a scope and a value-weight pair. Olsen's method helped to improve the quality of search results and reliability of the relation between facts returned in response to query. In addition, it helped with avoiding information overloading.

Bone et al. (2016) studied the volume of publically available geospatial data on the web, which is rapidly increasing due to advances in server-based technologies and the ease at which data can now be created. They found challenges with connecting individuals searching for geospatial data with servers and websites where such data exists. They presented a publicallyس available GSE (Geospatial Search Engine) that utilizes a web crawler built on top of the Google search engine in order to search the web for geospatial data. They found that the crawler seeding mechanism combines search terms entered by users with predefined keywords. They applied the GSE to search for all available geospatial services under these formats and provide search results including the spatial distribution of all obtained services.

The work of Liu (2017) investigated how to make controlled vocabulary more useful for searching. They presented an eye- tracking user study designed to help build natural user search interfaces to be used in formulating complex queries using MeSH (MEdical Subject Headings) terms. He found from several IR (Information Retrieval) user experiments, how the searchers, controlled vocabulary, IR system and the skill of indexers affect search outcomes. More recent eye- tracking study results further revealed that search experience, cognitive style, and perceived search task difficulty affect how users interact with search system user interfaces (Kim, 2006).

The aforementioned research and many other experiments have considered the direction of enhancing relevancy using terms (document building blocks). The focus has been on systems based on the Vector Space Model in IR. The structure of the document the effect it may have on the significance of the retrieved results has been seldom used or considered. There are other research directions such as XML-based retrieval which can benefit the current state in IR.

XML is a versatile markup language, capable of labeling the information content of diverse data sources including structured and semi-structured documents, relational databases, and object repositories. A query language that uses the structure of XML can intelligently express queries across all these kinds of data, whether physically stored in XML or viewed as XML via middleware. This specification describes a query language called XQuery, which is designed to be broadly applicable across many types of XML data sources (Robie et. al, 2009). XQuery can also be used on the back-end of a Web server, or to generate enterprise-wide executive reports (Marchiori and Quinn, 2017).

Liu & Cher (2008) focused on retrieving XML data in web and scientific applications. They found that it is hard to directly evaluate the relevance of query results due to the inherent ambiguity of search semantics. They studied keyword search strategies from a formal perspective, and then, they investigated an axiomatic framework that include two intuitive and non-trivial properties that an XML keyword search technique should ideally satisfy: monotonicity and consistency, with respect to data and query. They proposed a novel semantics for identifying relevant matches. An efficient algorithm is designed for realizing this semantics. They found that extensive experimental studies have verified the intuition of the properties and showed the effectiveness of the proposed algorithm.

XQuery is used to retrieve information from within XML documents that have well-structured data. XQuery operates on the logical structure of an XML document, rather than its surface syntax. This logical structure, known as the data model (W3C, 2020). The use of current indices in Information retrieval is due to the textual non-structural nature of HTML/text documents. However, when transformed into XML, new opportunity open for the purpose of retrieving more concise information.

This research investigates the possible benefit of including the structure of web documents into the retrieval process. That is by obtaining the structure from the HTML page, prioritizing its elements, adding the hidden significance, matching the query terms, computing the rank of documents based on the significance of the matching terms, and providing the user need in the form of conventional search hits. The transformation of the non-structured HTML into the well-structured XML happens using a schema. The design and building of the schema are explained in the following section (as shown in Figure 1).

III. SCHEMA DESIGN

A set of rules were developed to reach the first objective of this research. The rules are described in the form of an XML schema. The schema will be used to transform every HTML document into XML format. The rules that will be developed in this research will allow the selection of certain elements to represent the HTML document into XML format. The suggested schema is built as follows:

1. Create a set of rules to transform any HTML document into an XML document. That is creating a general XML schema that can be applied to HTML documents.
2. The HTML document will be divided into categories according to the importance of tags in the search process.

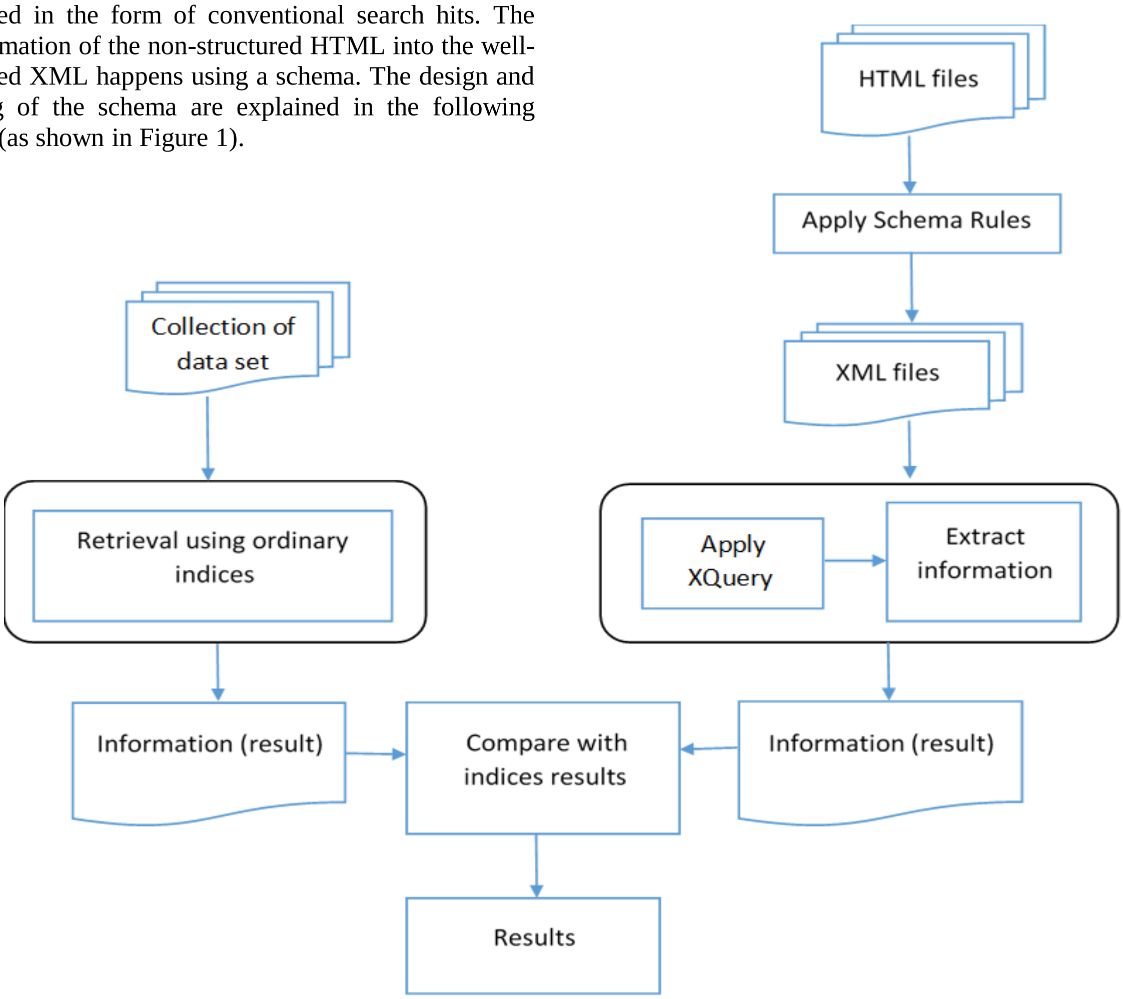


Figure 1. Steps of the Proposed Approach

i. Meta: with the highest search priority, because it consists of specific information about HTML documents.

ii. Titles: with the following search priority and it contains the HTML document's title.

iii. Header: all headers in the HTML document are grouped under this tag in XML.

iv. Footer: a footer typically contains information about the author of the section, copyright data or links to related documents.

v. Images: the image tag contains information on the image that should not be neglected.

vi. Words: all words with special format are considered with higher significance to the document than those unformatted.

vii. HREF (Links) and Anchors: those are also a different type of elements and are grouped together after titles and headers.

viii. Lists: used to store lists of items.

ix. Table: takes items in tables.

x. Text: this is used for all other text elements. This item will be further classified in a later stage of the research.

3. The developed schema excludes all tags which do not contain data such as: <br>, <hr> and the like.
4. All old HTML tags which are deprecated and not supported in HTML5 are also Excluded. They use other tags instead, such as: <acronym>, < applet >, < big >, <font>, etc. (W3C, 2020).
5. Each segment of an HTML document, which is an element of the XML resulted document, will have its own ranking degree based on its priority level.
6. The schema is supposed to achieve a great improvement in terms of query execution time since not all relations will be scanned for each query.
7. The rest of the HTML text will be traversed sequentially as another element only when needed.

```
<?xml version="1.0" encoding="UTF-8"?>
      <xs:schemaxmlns=http://www.w3.org/2001/
      XMLSchema elementFormDefault="unqualified">
       <xs:element name="ElementSet">
            <xs:complexType mixed="true">
                 <xs:choice minOccurs="0"
      maxOccurs="unbounded">
                       <xs:element ref="Title"
      maxOccurs="1" />
                       <xs:element ref="Meta" maxOccurs="
      1" />
                       <xs:element ref="Header"
      maxOccurs=" unbounded " />
                       <xs:element ref="Anchor"
      maxOccurs=" unbounded " />
                       <xs:element ref="Link" maxOccurs="
      unbounded " />
                      <xs:element ref="Images" maxOccurs="
      unbounded " />
                       <xs:element ref="Text" maxOccurs=" 1
      " />
                       <xs:element ref="Words" maxOccurs="
      unbounded " />
                       <xs:element ref="List" maxOccurs="
      unbounded " />
                       <xs:element ref="Table" maxOccurs="
      unbounded " />
                       <xs:element ref="Footer"
      maxOccurs="1" />
                 </xs:choice>
            </xs:complexType>
       </xs:element>
      <xs:element name=" Title">
               <xs:simpleType>
                          <xs:restriction base="xs:string">
                          </xs:restriction>
               </xs:simpleType>
      </xs:element>

      <xs:element name="Meta">
               <xs:simpleType>
                          <xs:restriction base="xs:string">
                          </xs:restriction>
               </xs:simpleType>
      </xs:element>

      <xs:element name="Headder">
               <xs:simpleType>
                          <xs:restriction base="xs:string">
                          </xs:restriction>
               </xs:simpleType>
      </xs:element>

      <xs:element name="Anchor">
               <xs:simpleType>
                          <xs:restriction base="xs:string">
                          </xs:restriction>
               </xs:simpleType>
      </xs:element>

      <xs:element name="Link">
               <xs:simpleType>
                          <xs:restriction base="xs:string">
                          </xs:restriction>
               </xs:simpleType>
      </xs:element>

      <xs:element name="Images">
               <xs:simpleType>
                          <xs:restriction base="xs:string">
                          </xs:restriction>
               </xs:simpleType>
      </xs:element>

      <xs:element name="Text">
               <xs:simpleType>
```

```
                <xs:restriction base="xs:string">
                </xs:restriction>
        </xs:simpleType>
</xs:element>

<xs:element name="Words">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                </xs:restriction>
        </xs:simpleType>
</xs:element>

<xs:element name="List">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                </xs:restriction>
        </xs:simpleType>
</xs:element>
```

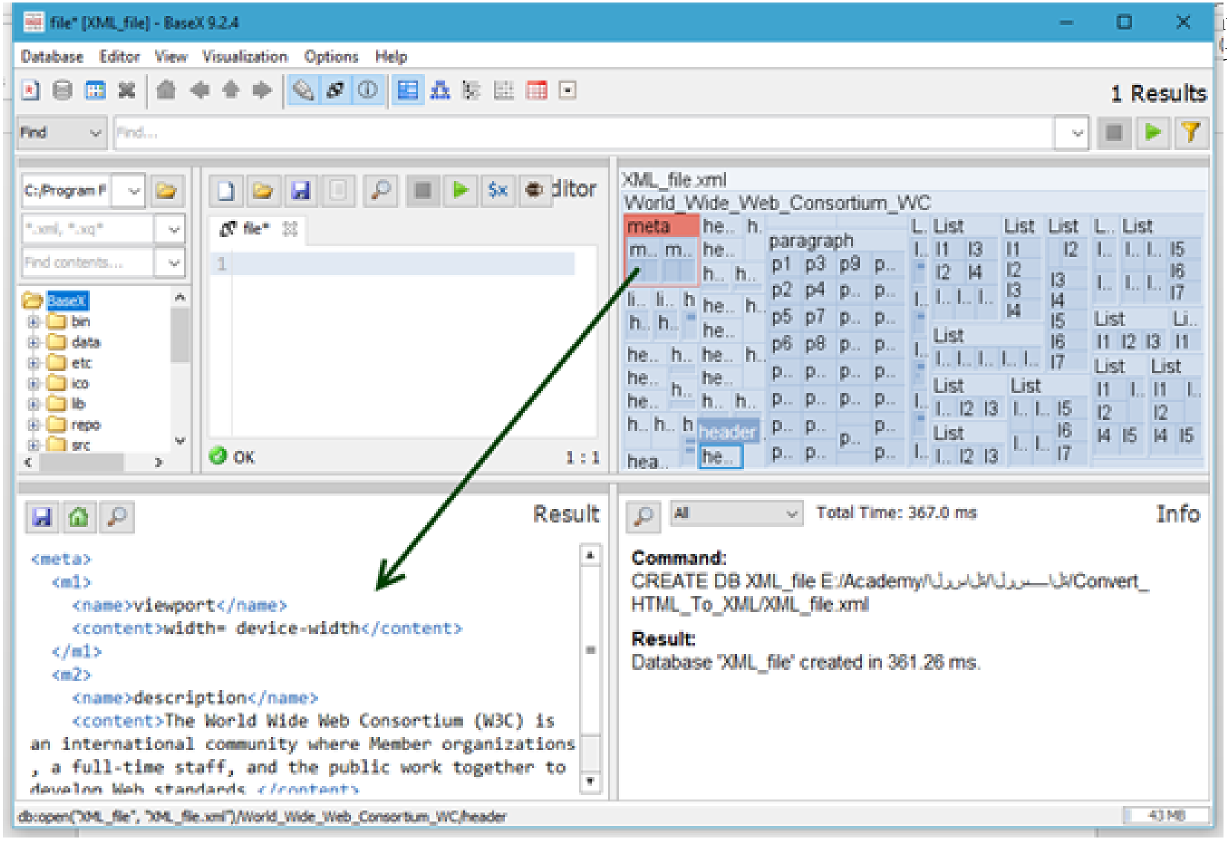
```
<xs:element name="Table">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                </xs:restriction>
        </xs:simpleType>
</xs:element>

<xs:element name="Footer">
        <xs:simpleType>
                <xs:restriction base="xs:string">
                </xs:restriction>
        </xs:simpleType>
</xs:element>
```

The schema includes the main elements found in an HTML document. Their types and restrictions in the XML document resulting from the transformation process are also described in the schema. Some elements such as

t.

ur



```
<meta>
  <m1>
    <name>viewport</name>
    <content>width= device-width</content>
  </m1>
  <m2>
    <name>description</name>
    <content>The World Wide Web Consortium (W3C) is
an international community where Member organizations
, a full-time staff, and the public work together to
develop Web standards.</content>
```

Total Time: 367.0 ms

Command:
CREATE DB XML_file E:/Academy/بحث/مؤتمر/بحث/Convert_HTML_To_XML/XML_file.xml

Result:
Database 'XML_file' created in 361.26 ms.

db:open("XML_file", "XML_file.xml")/World_Wide_Web_Consortium_WC/header

Figure 2. The Interface

The proposed model is intended to embed structure into a text document used for retrieval purposes. In order to do so, the HTML (text) document that has some implied significance to its elements cannot convey much of that importance when used as a bag of terms in the Vector Space Model currently used in retrieval systems. Hence, it has to, first, be transformed into XML using the above schema. The schema adds importance to each element since the nature of XML implies that the main element is the most important. The significance of each element decreases as one goes deeper into the XML structure (the inner sub-elements).

The main difficulty in this model is how to choose the proper significance level for each HTML elopement? Moreover, is that level of significance the same in each HTML document or is it different? Which factors determine such significance? At this level of research, significance is given based on the type and position of text element in the HTML document. For example, titles and headers are given high importance than plain text.

The Schema will continue to be modified to accommodate all possible types of elements in HTML documents. It may also differentiate among text elements in the same HTML item. That may involve using text formatting to convey further levels of importance to text elements. The schema will evolve to demonstrate complete and accurate description of the document in order to provide more precise results in the retrieval process.

IV. PRELIMINARY TESTING AND EVALUATION

In order to put the schema to possible use, the model had to be tested and evaluated. The evaluation has a main goal at the end. That is investigating the benefit of using XML along with XQuery and comparing that to the traditional Vector-Space based retrieval.

In order to reach the ultimate goal, a software converting HTML downloaded documents was developed (Figure 2). The software uses the schema described above to extract textual information in addition to structural semantics and create an XML variant from each web page for retrieval purposes. Then, an interface was built to query the XML files. The dataset used in the preliminary testing was basically a random collection of HTML pages crawled from different parts of the web and contain different kinds of information.

The interface was then tested using a number of pilots (three computer science professionals) to provide indications and guidance for possible larger experiment in the future.

V. RESULTS AND DISCUSSION

The purpose of the pilot study was to highlight the requirements of a large-scale experiment that will be conducted at a later stage. The participants in the pilot study indicated that they found retrieval using the designed interface very useful. They indicated that the results they located using simple term queries to an XML database were relevant. That was concluded through the debriefing that took place after completing the experiment.

The pilots were given the interface and the subjects covered in the data were explained to them. They knew the kind of information they were going to try to locate. They used the interface with conventional queries to satisfy their information needs. The engagement factor only was taken into consideration in the pilot study.

A larger scale study will be conducted in the future as follows. A large group of participants would be selected randomly. They would be given two interfaces that are similar in their design and usability but different in the underlying databases and retrieval algorithms. The first would use conventional indices used in current search engines which rely on matching terms to find related documents. The second interface would use XML databases.

The querying procedure would also be different. In the first interface, querying and matching would use term to term matching. The second interface would use XQuery search underneath. The user interface characteristics would be the same though. Furthermore, the results would also be similar in their appearance to the user. The study would compare the two interfaces in terms of efficiency, effectiveness and enjoyment. The study is shown in Figure 1 as a part of the entire project that will be conducted in the future.

VI. CONCLUSION

In this paper, a new retrieval model was proposed. The model relies on the ability of transforming textual documents (HTML) into XML for a different form of retrieval that uses XML query language. The model consists of a general schema that contains rules developed based on the common content of HTML pages. The Schema is the standard used to transform every HTML document into XML for retrieval. The Query language used in the retrieval process is XQuery. The retrieval interface is similar to those used in conventional information seeking projects such as search engines for user comfortability reasons. Further research will involve larger experiments using larger collections of data for better understanding of the effectiveness, efficiency, and user engagement using retrieval systems based on this model.

REFERENCES

(Periodical style)

[1] Baeza-Yates, R., Castillo, C., Junqueira, F., Plachouras, V., & Silvestri, F. (2007, April). Challenges on distrfyur76iriylrer7tfugfkgur58tuftruyyrhgguuyyutyutibuted web retrieval. In Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on (pp. 6-20). IEEE.

[2] BaseX Location, Retrieved (23,3,2020) from BaseX: http://basex.org

[3] Bone, C., Ager, A., Bunzel, K., & Tierney, L. (2016). A geospatial search engine for discovering multi-format geospatial data across the web. International Journal of Digital Earth, 9(1), 47-62.

[4] Brin, S., & Page, L. (2012). Reprint of: The anatomy of a large-scale hypertextual web search engine. Computer networks, 56(18), 3825-3833.

[5] Extensible Markup Language (XML), Retrieved (10, 7, 2017). from W3C: https://www.w3.org/XML/

[6] XQuery: A Query Language for XML (W3C), Retrieved (20,08,2020).https://www.w3.org/TR/2001/WD-xquery-20010215/

[7] Jiang Y., Semantically-Enhanced Information Retrieval Using Multiple Knowledge Sources, Springer Link, Cluster Comput (2020).

[8] Kim, K.Y, Jin, D. S., Choi, Y.S., Kim, J. S., Suh, Y. K., and Seo, J., An Improvement of Information Retrieval Using World Location Information, International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria, 2006.

[9] Kim, J. Task Difficulty as a Predictor and Indicator of Web Searching Interaction. Proceedings of the 2006 Conference on Human Factors in Computing Systems, Montreal, Canada.

[10] Lennon, F. J., & Malkovich, B. J. (2006, March). Google-based Information Extraction. In DIR'06 Dutch-Belgian Information Retrieval Workshop (p. 39).

[11] Liu, Y. H. (2017). University Metadata and Retrieval: The Death of the Library Catalog?. Bulletin of the Association for Information Science and Technology, 43(4), 9-12.

[12] Liu, Z., & Cher, Y. (2008). Reasoning and identifying relevant matches for XML keyword search. Proceedings of the VLDB Endowment, 1(1), 921-932.

[13] Marchiori, M., and Quinn, L. (2017, April 22). W3C XML Query (XQuery). (W3C) Retrieved (6, 20, 2018). from W3C: https://www.w3.org/XML/Query/

[14] Olsen, Ø. H. (2011). U.S. Relevance-weighted navigation in information access, search and retrieval. Patent No. 7,966,305. Washington, DC: U.S. Patent and Trademark Office.

[15] Robie, J., Chamberlin, D., Dyck, M., & Snelson, J. (2009, December 15). XQuery 1.1: An XML Query Language. (W3C) Retrieved (22, 6, 2020), from W3C: https://www.w3.org/TR/xquery-11/#id-introduction

[16] Salton, G., Wong, A., Yang, C. S., A Vector Space Model for Automatic Indexing, Communications of the ACM, Vol. 18, Issue 11, pp. 1-12, 1975.

[17] Sanderson, M., Scholer, F., & Turpin, A., Relatively Relevant: Assessor shift in document judgements. In Proceedings of the Australasian Document Computing Symposium, pp. 60-67, 2010.

[18] What is XHTML, Retrieved (10, 7, 2020). from W3C: https://www.w3.org/TR/xhtml1/introduction.html#why

[19] W3C Retrieved (12, 6 ,2020). https://www.w3.org/standards/webdesign/htmlcss