Asian Journal of Information Technology 12 (1): 45-52, 2013
ISSN: 1682-3915

An Improved Information Retrieval System Using Fuzzy Set Similarity Measure

R.M. Periakaruppan and R. Nadarajan
Department of Applied Mathematics and Computational Sciences,
PSG College of Technology, 641 004 Coimbatore, Tamil Nadu, India

Abstract: Keyword based Information Retrieval (IR) Systems fails when there is no exact matching. Hence, IR Systems more focused on user relevant information retrieval. In this study, researchers proposed a technique to improve the searching based on Fuzzy set similarity measure, domain knowledge representation and semantics. In the proposed method, researchers extract candidate keywords from each document which reflects the topic of the document and map them with a suitable domain knowledge classification system. The mapping process associate a semantic weight to each keyword based on Latent Semantic Indexing (LSI) which reflects the significance of each keyword in the document with respect to a domain. As a result of mapping process the keywords along with their semantic weights are represented as XML document and they are clustered based on similarity measure. The experiments shows that the proposed similarity measure yields better results when compared with conventional similarity measure techniques.

Key words: Information retrieval, knowledge representation, fuzzy set, XML, ACM CR classification

INTRODUCTION

The main objective of Information Retrieval (IR) System is to provide user relevant documents for a query. Early IR techniques are keyword based and they do not consider semantic relationship between keywords, hence users often find difficult to express right keywords and due to this there is low precision and recall. To improve the searching process, domain or concept based IR Systems are widely used (Haav and Lubi, 2001; Kwasnik, 1999; Pan *et al*., 2011; Solvberg *et al*., 1992; Yoon and Dankell, 2005). Domain knowledge can be represented using Ontology or Taxonomy and they have significant role in information retrieval process (Dogac *et al*., 2002; Nagypl, 2005; Jan and Kostial, 2003). The term ontology is used for complete domain knowledge information including object relationship and property relationship. The term taxonomy is associated with hierarchical class relationship. In this study, researchers have used knowledge representation based on Taxonomy namely ACM CR classification system because the input documents considered deals with computer science subject classification. Examples for such documents are research articles, table of contents of a book, etc.

In information retrieval system, one of the popular methods of representing a document is Vector Space Model (VSM) (Salton *et al*., 1975). But VSM doesn't find semantic relation between the terms. Hence, LSI based document representation is used which discovers the semantic relation. The challenging task in LSI is to reduce the dimensionality of the matrix which can be done by extracting only important keywords from the documents. These keywords are used to identify the topic of the document which can be given manually, for example, researcher of research articles will mention the keywords or by automatic extraction based on text analysis. In case of books, keywords can be extracted from table of contents, since they will describe the topic of the book in a more precised manner. These set of extracted keywords are arranged in a hierarchical relationship by mapping with the domain knowledge classification system. In the mapping process the degree of similarity between the keyword and the concept in the taxonomy is computed using LSI. This measure is called semantic weight and it is used to represent each input document d_j as XML document where each tag $<t_i, w_i>$ corresponds to ith term and semantic weight of the document.

XML document similarity can be measured in terms of structure and semantic which is addressed by many researchers (Leung *et al*., 2003; Nierman and Jagadish, 2002; Tekli *et al*., 2009; Woosaeng, 2008). Jeong *et al*. (2008) addresses the semantic similarity of XML documents based on supervised classifier using neural networks for limited number of samples. Both semantic and structural similarity was discussed by Ghosh and

Corresponding Author: R.M. Periakaruppan, Department of Applied Mathematics and Computational Sciences, PSG College of Technology, 641 004 Coimbatore, Tamil Nadu, India

Mitram (2008) however, the SVM (Support Vector Machine) has to be tuned with optimal weights. Another hybrid approach was discussed by Tekli *et al.* (2007) which computes similarity based on element/attribute labels not on element/attribute values. A fuzzy based similarity measure was discussed by Ceravolo *et al.* (2004) in which the semantic weight of the term is not considered. Structure and content similarity using LSI was discussed by Tran *et al.* (2008) in which all the terms in XML document are considered and hence the dimensionality of input matrix is higher. In this study, similarity between XML documents is computed based on Fuzzy set similarity measure. The semantic weight attribute present in XML document indicates the significance of a term with respect to a domain topic. Based on application this significance varies as high, medium or low. The threshold value α and β are fixed such that the semantic weight less than α is considered as low, greater than β as high and in between are medium. Hence, a fuzzy based approach ensures to retrieve user relevant documents. Now to cluster XML documents different methods were proposed by Dalamagas *et al.* (2006), Damiani *et al.* (2004), Gil-Garcia *et al.* (2006) and Guerrini *et al.* (2007) and researchers have used agglomerative hierarchical clustering algorithm since, it provides data at different levels. The experimental results based on the proposed technique yields higher precision and recall rate when compared with Tran *et al.* (2008)'s. The contributions in this study contain the following steps:

- Automatic identification of topic of the document in XML format
- Computing semantic similarity between XML documents using Fuzzy set
- Clustering XML documents based on similarity measure
- Searching the cluster of XML documents and retrieving results

DOCUMENT REPRESENTATION USING XML

Since, XML documents are hierarchical in nature, they are the best choice for representing a hierarchical classification system (De Vries, 2004; Hoelzer *et al.*, 2002; Zhu *et al.*, 2004). Consider an article written on the subject Data Structures. The extracted keywords from this document are Data Structures, Arrays, Lists, Graphs, Hash table and Sorting. These keywords are mapped with the ACM CR Classification System and a hierarchical relationship among these keywords is created with a classification number. The hierarchical tree and the corresponding XML document are shown in Fig. 1 and 2, respectively.

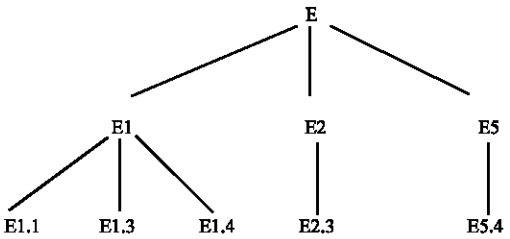

Fig. 1: The subject content of an article on Data Structures

```
<E>
    <E1>
        <E1.1/>
        <E.1.3/>
        <E1.4/>
    </E1>
    <E2>
        <E2.3/>
    </E2>
    <E5>
        <E5.4/>
    </E5>
</E>
```

Fig. 2: XML document representing the article on Data Structures using ACM CR classification

AUTOMATIC IDENTIFICATION OF SUBJECT OF THE DOCUMENT

A classification system is used to associate information with a set of predefined concepts. Examples of popular classification system are ACM Computing Classification System, Mathematics Subject Classification (MSC), Physics and Astronomy Classification Scheme, Journal of Economic Literature Classification System, etc. Automatic identification of topic of a document is a key issue in information retrieval. Once the information is organized, the retrieval process can be done effectively. The process of mapping topics of the document with the classification system is shown in Fig. 3.

The input text document can be any research article, books or any other electronic resources of a particular domain. Here , ACM CR classification is considered which deals with computer science domain.

The extraction phase involves identifying important keywords. In a research article the keywords given by the researchers is considered because they describe the topic of the article in a precised manner. Similarly, the table of contents in a book defines the topic of a book more elaborately than the title. In this phase, these keywords are extracted.

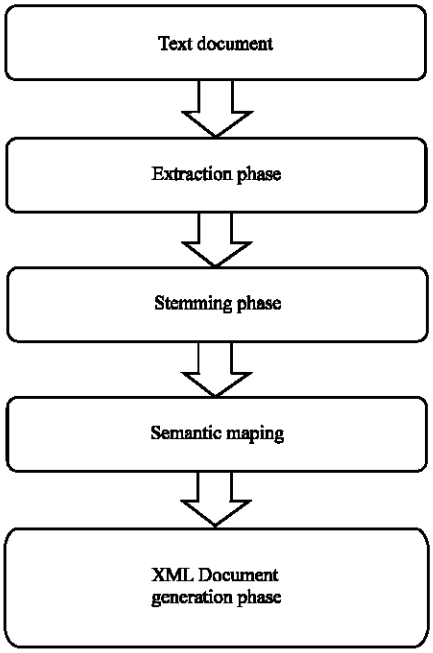

Fig. 3: Automatic topic identification

In stemming phase, key words are stemmed which we got in extraction phase using Porter Stemming algorithm. Each keyword can be combination of words and each word is stemmed. If the keyword is Data Structures and Algorithms, the stemmed output is: Data Structure Algorithm.

In semantic mapping phase, the stemmed keyword is considered as query and this is mapped with the taxonomy classification. In case of exact match researchers associate a semantic weight 1. In case of partial map we use LSI to associate the semantic weight which is in between 0-1. The algorithm for mapping process is shown in Algorithm 1.

Algorithm 1 (Semantic mapping phase):

```
Input: keyword K containing Stemmed words w1, w2…,wn.
       Hierarchical classification document D

Step 1: // Exact match //
        If (w1^w2…^wn) found in D then
          weight←1.0
          return (classification code, weight)
          exit function
        else
           goto step 2
        endif
Step 2: // Partial match. Possibility of more than
one match //
       q←minimum number of words to be present
       k←0
        while (not end of file D) do
              If (any combination of words w1,w2)
       (q<=i<n)
                found in D) then
             match[k]←classification hierarchy
      element
          k ← k +1;
          endif
      endwhile
        if (k = 0) //No partial match// return(false)
      endif
        r←0
        for each match[j] (1<= j < k)
              sim[r]←compute LSI between K
            and match[j]
              r←r+1;
        endfor
        weight←max(sim[r])
        classification code←match[j]
                    where j is max { sim[m] }, 0≤
      m≤r
          return (classification code, weight)
exit function
```

To explain the algorithm, consider the following example. Suppose we want to map Data Structures and Algorithms.

This is stemmed and we get the following keywords K = Data, Structures, Algorithms. Exact match searching is done for the Keyword K (i.e.) presence of all these words in the hierarchical classification Document D, namely ACM CR classification. Since, there is no exact match, searching for any of these words with a condition of presence of at least two words which results in the following partial matches:

- Data Structures
- Distributed Data Structures
- Representation Data Structures And Transforms
- Graphics Data Structures And Data Types

Now LSI similarity of the keyword K with each of the above partial strings is computed and the string with maximum similarity value is chosen (i.e.) the string Data Structures. Researchers associate this semantic weight with the corresponding classification code namely E1. The computation of LSI similarity value is shown.

Step 1: Stem the query string and partial matches are represented as (stop words are ignored):

- q: Data Structures Algorithms

All partial matches are considered as documents:

- d1: Data Structures
- d2: Distributed Data Structures
- d3: Representation Data Structures Transforms
- d4: Graphics Data Structures Data Types

Now the term document query matrix (T) is computed based on term weights and the same is shown in Table 1.

Table 1: Term document and query matrix T

Terms	d1	d2	d3	d4	Query q
Data	1	1	1	2	1
Structures	1	1	1	1	1
Algorithms	0	0	0	0	1
Distributed	0	1	0	0	0
Representation	0	0	1	0	0
Transforms	0	0	1	0	0
Graphics	0	0	0	1	0
Types	0	0	0	1	0

Step 2: The term document matrix T is decomposed as follows:

$$T = USV^T$$

Step 3: The coordinates of document vectors of each document d1, d2, d3 and d4 are found from the matrix V. Matrix V actually holds the eigenvector values.

Step 4: The new query vector coordinates is found using the equation:

$$q = q^T U_k S_{k-1}$$

where, U_k and S_{k-1} are reduced 2-dimensional space vectors.

Step 5: The similarity between each document and the query is computed by the equation:

$$\text{Sim}(q,d) = \frac{(q \cdot d)}{(|q| \, |d|)}$$

Similarity values for the above example computed using Eq. 1 are:

- Sim [1] = sim (q, d1) = 1
- Sim [2] = sim (q, d2) = 0.8166
- Sim [3] = sim (q, d3) = 0.3209
- Sim [4] = sim (q, d4) = 0.3742

The maximum similarity value is 1 which corresponds to the match Data Structures. The similarity value is taken as semantic weight. The candidate keyword is the pair <Data Structures, 1.0>.

In XML document generation phase the corresponding XML document is generated based on the semantic mapping. Every key word is mapped and the corresponding classification code is taken as XML element and the corresponding semantic weight is considered as the attribute. For the above example the classification code for data structures is E1 and the weight is 1.0. This is done for all extracted keywords.

For example if the keywords extracted from the document are Data Structures, Arrays and Trees then the corresponding XML document is shown in Fig. 4.

```
<root>
    <E1 weight = "1.0">
        <E1.1 weight = "1.0"/>
        <E1.7 weight = "1.0"/>
    </E1>
</root>
```

Fig. 4: XML document with semantic weight

COMPUTING SEMANTIC SIMILARITY BETWEEN XML DOCUMENTS

Once input documents are represented in XML form, the similarity between XML documents are computed based on fuzzy set. The computation process is explained with the following example. Document 1 with the following key words:

- Communication network
- Network architecture and design
- Distributed network

Document 2 with the following key words:

- Network analysis and design
- Network communication
- Distributed network management

The corresponding XML documents are generated shown in Fig. 5. Now each XML document is represented as fuzzy set with semantic weight as membership function.

The definition of fuzzy set representing XML document and the membership function is defined as follows.

Definition 1 (XML document fuzzy set): An XML document fuzzy set is defined as $A = \{t_1, t_2, ..., t_n\}$ that is represented by set of ordered pairs $\{(t_1, \mu_A(t_1)), (t_2, \mu_A(t_2)), ..., (t_n, \mu_A(t_n))\}$ where, μ_A is the membership function of the set A which assigns to each element x a real number μ (X) in the interval [0, 1] where the value of μ(X) represents the grade of membership of x in the fuzzy set and and $t_1, t_2, ..., t_n$ are the document terms which represent XML document.

Definition 2 (membership function μ): The membership function for matching keyword is defined as:

$$\mu(w;\alpha,\beta) = \begin{cases} 0 & \text{if } w \le \alpha \\ w & \alpha < w < \beta \\ 1 & w \ge \beta \end{cases}$$

```
<root>
  <c2 weight = 1.0>
    <c21 weight = 1.0>
      <c214 weight = 1.0/>
    </c21>
  </c2>
</root>
```

```
<root>
  <c2 weight = "1.0">
    <c21 weight = "0.80">
      <c214 weight = "1.0" />
    </c21>
  </c2>
</root>
```

Fig. 5: XML representation of documents 1 and 2

where, w is the semantic weight and α and β are threshold values which can be set based on applications. Based on (2) with $\alpha = 0.10$ and $\beta = 0.90$, document 1 and 2 shown in Fig. 5 is represented as:

$$A = \{(c2,1.0),(c21,1.0),(c214,1.0)\}$$
$$B = \{(c2,1.0),(c21,0.80),(c214,1.0)\}$$

The fuzzy similarity measure between A and B is given as:

$$Sim(A, B) = \frac{\min(\mu_A(t), \mu_B(t))}{\max(\mu_A(t), \mu_B(t))}$$

Using (3) the similarity between the fuzzy sets A and B is computed as:

$$Sim(A, B) = \frac{1+0.80+1}{1+1+1} = \frac{2.80}{3} = 0.93$$

Thus, a high similarity value between documents A and B is returned even though the keywords are not exactly same.

CLUSTERING XML DOCUMENTS BASED ON SIMILARITY MEASURE AND RETRIEVAL OF RESULTS

Based on the similarity measure explained above document similarity matrix is constructed and it is used for clustering. Agglomerative hierarchical clustering with complete linkage is used because it groups clusters at different levels.

In order to improve the efficiency of searching process an index file is associated with every cluster and this index file contains all parent element of classification code in the corresponding cluster. For example if the classification code is E112 then the parent element E with cluster id is stored.

The user query is mapped with the hierarchical classification document. The mapping process is same as explained above and the corresponding classification code is returned and the same code is searched in the index file and the corresponding documents within the clusters are returned.

EXPERIMENTAL RESULTS

To compare the XML document similarity algorithm with Tran *et al.* (2008) approach, researchers considered three real time data sets from InfoVis, DBLP and SIGMOD. InfoVis-2004 is a contest conducted in the field of information visualization (Fekete *et al.*, 2004) the dataset contains complete metadata for all the study of 8 years (1995 to 2002) of InfoVis Conference and their references. DBLP stands for Digital Bibliography Library Project and the DBLP server provides bibliographic information on major computer science journals and proceedings. A SIGMOD record is an index of articles from ACM SIGMOD. Table 2 presents the detailed information about the datasets.

In order to analyze the performance of the proposed method, experiment is conducted by varying the parameters of the membership function α and β with three set of values from $\{\{0.1, 0.9\}, \{0.2, 0.8\}, \{0.3, 0.7\}\}$. Also for each set of α and β the algorithm is executed about 10 runs. For analyzing the clustering performance, the terms True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) are used to calculate the Precision and Recall measures as defined:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Figure 6-8 depict the results received from each datasets. The approach yields high precision and recall when compared to Tien Tran approach.

From Fig. 9 for the datasets InfoVis and DBLP, the parameters α and β with the values {0.2, 0.8} yields better

precision and recall values where as for SIGMOD dataset the highest precision and recall values are obtained with {0.1, 0.9} for α and β. The experimental results for InfoVis dataset is shown in Table 3.

Library Books Search System: Researchers have developed an application to search library books based on the proposed method. Library books are normally searched by researchers name, book title and publisher.

But more details about the subject of the book are available in table of contents of the book. A research similar to the approach was done by Murty and Jain (1995) in which classification code is considered as string and it is very difficult to generate and manipulate. Also the above method manually created the table of contents document and the partial mapping is ignored.

To evaluate the proposed technique in Library Books Search System, table of contents from 38 books namely Data Structures (12), Operating systems (7), Computer Networks (9) and Software Engineering (10) are collected. To retrieve user relevant query average overall precision

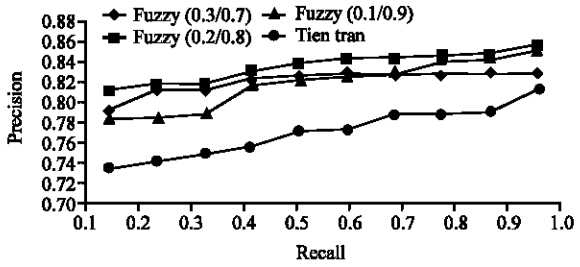

Fig. 6: Precision vs. recall for InfoVis dataset

Table 2: Dataset used for document similarity measures

Datasets	No. of articles	No. of key words/article extracted
InfoVis	4240	5-15
DBLP	3104	4-8
SIGMOD	6150	4-8

Table 3: Performance measures for the infovis dataset

Alpha	Beta	TP	FP	TN	FN	PR	REC	ACC	FM
Fuzzy approach									
0.3	0.7	0.8064	0.1671	0.7509	0.1542	0.8284	0.8394	0.8290	0.8339
		0.8069	0.1695	0.7652	0.1560	0.8264	0.8380	0.8285	0.8322
		0.8092	0.1708	0.7669	0.1571	0.8257	0.8374	0.8278	0.8315
		0.8194	0.1726	0.7713	0.1597	0.8260	0.8369	0.8272	0.8314
		0.8238	0.1728	0.7747	0.1608	0.8266	0.8367	0.8274	0.8316
		0.8325	0.1730	0.7772	0.2022	0.8279	0.8046	0.8110	0.8161
		0.8374	0.1801	0.7790	0.2182	0.8230	0.7933	0.8023	0.8079
		0.8552	0.1971	0.7791	0.2222	0.8127	0.7937	0.7958	0.8031
		0.8554	0.1971	0.7864	0.2318	0.8127	0.7868	0.7929	0.7996
		0.8885	0.2344	0.8674	0.2318	0.7912	0.7931	0.7902	0.7922
0.2	0.8	0.9359	0.1909	0.8659	0.1633	0.8306	0.8514	0.8357	0.8409
		0.9390	0.2095	0.8744	0.1673	0.8176	0.8487	0.8279	0.8329
		0.9412	0.1762	0.9050	0.1891	0.8423	0.8327	0.8348	0.8375
		0.9419	0.2103	0.9100	0.2331	0.8175	0.8016	0.8068	0.8095
		0.9509	0.2211	0.9135	0.2303	0.8113	0.8050	0.8051	0.8082
		0.9531	0.1722	0.9206	0.1560	0.8470	0.8593	0.8509	0.8531
		0.9590	0.1617	0.9237	0.1899	0.8557	0.8347	0.8426	0.8451
		0.9647	0.1797	0.9239	0.2027	0.8430	0.8264	0.8316	0.8346
		0.9900	0.1819	0.9321	0.1917	0.8448	0.8378	0.8373	0.8413
		0.9919	0.1924	0.9424	0.2157	0.8375	0.8214	0.8258	0.8294
0.1	0.9	0.8634	0.1936	0.7868	0.1650	0.8169	0.8396	0.8215	0.8281
		0.8681	0.1811	0.7980	0.2160	0.8274	0.8008	0.8075	0.8139
		0.8763	0.2423	0.8020	0.2019	0.7834	0.8128	0.7907	0.7978
		0.8877	0.1930	0.8028	0.2473	0.8214	0.7821	0.7934	0.8013
		0.8891	0.1685	0.8202	0.2149	0.8407	0.8053	0.8168	0.8226
		0.8980	0.2405	0.8300	0.2300	0.7888	0.7961	0.7860	0.7924
		0.8997	0.2480	0.8304	0.1954	0.7839	0.8216	0.7960	0.8023
		0.9171	0.1939	0.8363	0.1932	0.8255	0.8260	0.8191	0.8257
		0.9293	0.1611	0.8526	0.2325	0.8522	0.7999	0.8191	0.8252
		0.9310	0.1758	0.8600	0.1583	0.8412	0.8546	0.8428	0.8478
Tien Tran approach		0.7629	0.2047	0.6024	0.1607	0.7885	0.8260	0.7889	0.8068
		0.6071	0.1796	0.6108	0.2154	0.7717	0.7382	0.7551	0.7545
		0.6195	0.2245	0.6152	0.1994	0.7340	0.7565	0.7444	0.7451
		0.6254	0.1689	0.6156	0.2279	0.7874	0.7329	0.7577	0.7592
		0.6284	0.2187	0.6168	0.2215	0.7418	0.7394	0.7388	0.7406
		0.6315	0.1684	0.6260	0.2404	0.7895	0.7243	0.7547	0.7555
		0.6342	0.1868	0.6305	0.2391	0.7724	0.7262	0.7481	0.7486
		0.6557	0.2126	0.6324	0.1834	0.7552	0.7814	0.7649	0.7681
		0.6784	0.2280	0.6331	0.2199	0.7484	0.7552	0.7454	0.7518
		0.6844	0.1581	0.6458	0.1698	0.8123	0.8012	0.8022	0.8067

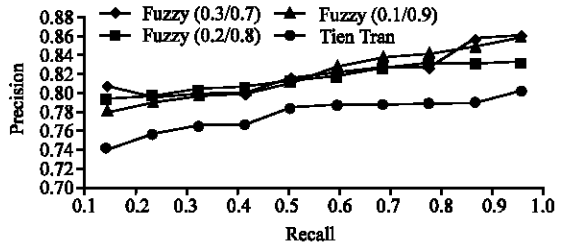

Fig. 7: Precision vs. recall for DBLP dataset

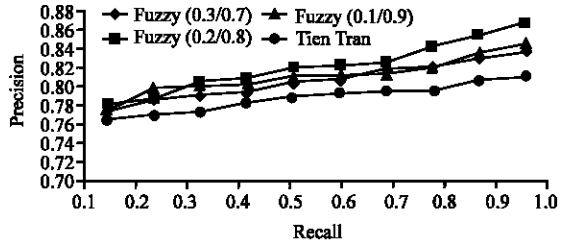

Fig. 8: Precision vs. recall for SIGMOD dataset

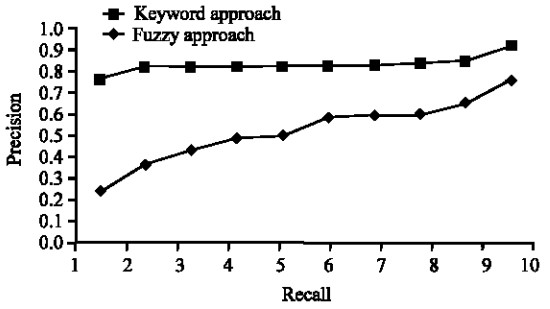

Fig. 9: Precision vs. recall analysis between fuzzy and keyword approach

value is calculated by supplying 94 queries. Comparison with traditional keyword based search is shown in Fig. 9.

CONCLUSION

In this study, researchers propose a method for retrieving user relevant documents based on domain knowledge, Semantics and Fuzzy set. The experimental result shows an improved information retrieval when compared with other exisiting approaches. In future extension, researchers are planning to test the system in large data set, namely, research corpus which are in Bibtex record format. Also, a hybrid method of representing knowledge in ontology and taxonomy will be considered, so that the accuracy of precision and recall will be further improved.

REFERENCES

Ceravolo, P., M.C. Nocerino and M. Viviani, 2004. Knowledge extraction from Semi-structured data based on fuzzy techniques. Knowled. Based Emerg. Technol. Relied Intell. Inf. Eng. Syst. Lect. Notes Comput. Sci., 3215: 328-334.

Dalamagas, T., T. Cheng, K.J. Winkel and T. Sellis, 2006. A methodology for clustering xml documents by structure. Inform. Syst., 31: 187-228.

Damiani, E., M.C. Nocerino and M. Viviani, 2004. Knowledge extraction from an XML data flow: Building a taxonomy based on clustering technique. Proceedings of the 8th EUROFUSE Meeting: EUROFUSE Workshop on Data and Knowledge Engineering, September 22-25, 2004, Warsaw, Poland, pp: 133-142.

De Vries, A., 2004. XML framework for concept description and knowledge representation. Artificial Intelligence. http://arxiv.org/pdf/cs/0404030.pdf.

Dogac, A., G. Laleci, Y. Kabak and I. Cingi, 2002. Exploiting web service semantics: Taxonomies vs. Ontologies. IEEE Data Eng. Bull., 25: 10-14.

Fekete, J.D., G. Grinstein and C. Plaisant, 2004. IEEE InfoVis 2004 contest, the history of InfoVis. http://www.cs.umd.edu/hcil/iv04contest/.

Ghosh, S. and P. Mitram, 2008. Combining content and structure similarity for XML document classification using composite SVM kernels. Proceedings of the 19th International Conference on Pattern Recognition, December 8-11, 2008, Tampa, FL, pp: 1-4.

Gil-Garcia, R., J.M. Badia-Contelles and A. Pons-Porrata, 2006. A general framework for agglomerative hierarchical clustering algorithms. Proceedings of the 18th International Conference on Pattern Recognition, Aug. 20-24, IEEE., pp: 569-572.

Guerrini, G., M. Mesiti and I. Sanz, 2007. An Overview of Similarity Measures for Clustering XML Documents. In: Web Data Management Practices: Emerging Techniques and Technologies, Vakali, A. and G. Pallis (Eds.). Idea Group Inc., USA., pp: 56-78.

Haav, H.M. and T.L. Lubi, 2001. A survey of concept-based information retrieval tools on the web. Proc. East-Eur. Conf. ADBIS, 2: 29-41.

Hoelzer, S, R.K. Schweiger, R. Liu, D. Rudolf, J. Rieger and J. Dedeck, 2002. XML representation of hierarchical classification systems: From conceptual models to real applications. Proc. Amia Symp., 2002: 330-334.

Jan, J.P. and I. Kostial, 2003. Ontology-based information retrieval. Proceedings of 14th International Conference on Information and Intelligent systems (IIS), (ICIIS. 03), Varazdin, Croatia, pp: 23-28.

Jeong, B., D. Lee, H. Cho and J. Lee, 2008. A novel method for measuring semantic similarity for xml matching. Expert Syst. Applicat., 34: 1651-1658.

Kwasnik, B.H., 1999. The role of classification in knowledge representation and discovery. Library Trends, 48: 22-47.

Leung, H.P., F.L. Chung and S.C.F. Chan, 2003. A new sequential mining approach to XML document similarity computation. Proceedings of the 7th Pacific-Asia Conference on Knowledge Discovery and Data Mining, April 30-May 2, 2003, Springer-Verlag, Berlin, Heidelberg, pp: 356-362.

Murty, M.N. and A.K. Jain, 1995. Knowledge-based clustering scheme for collection management and retrieval of library books. Pattern Recogn., 28: 949-963.

Nagypl, G., 2005. Improving information retrieval effectiveness by using domain knowledge stored in ontologies. Proceedings of the 2005 OTM Confederated international conference on On the Move to Meaningful Internet Systems, October 3, November 4, 2005, Springer-Verlag Berlin, Heidelberg, pp: 780-789.

Nierman, A. and H.V. Jagadish, 2002. Evaluating structural similarity in XML documents. Proceedings of the International Workshop on the Web and Databases, June 6-7, 2002, Wisconsin, USA., pp: 61-66.

Pan, H., X. Tan, A. Han and G. Yin, 2011. A domain knowledge based approach for medical image retrieval. Int. J. Inf. Eng. Elect. Busin., 3: 16-22.

Salton, G., A. Wong and C.S. Yang, 1975. A vector space model for automatic indexing. Commun. ACM, 18: 613-620.

Solvberg, I., I. Nordbo and A. Aamodt, 1992. Knowl-edge-based information retrieval. Future Generat. Comput. Syst., 7: 379-390.

Tekli, J., R. Chbeir and K. Yetongnon, 2007. A hybrid approach for xml similarity. Proceedings of the 33rd conference on Current Trends in Theory and Practice of Computer Science, January 20-26, 2007, Springer-Verlag Berlin, Heidelberg, pp: 783-795.

Tekli, J., R. Chbeir and K. Yetongnon, 2009. An overview on XML similarity: Background, current trends and future directions. Comput. Sci. Rev., 3: 151-173.

Tran, T., R. Nayak and P.D. Bruza, 2008. Combining structure and content similarities for XML document clustering. In7th Australasian Data Mining Conference, November 27-28, 2008, Glenelg, South Australia -.

Woosaeng, K., 2008. XML document similarity measure in terms of the structure and contents. Proceedings of the 2nd WSEAS International Conference on Computer Engineering and Applications, January 25-27, 2008, Acapulco, Mexico, pp: 205-212.

Yoon, C. and D.D. Dankell, 2005. Domain-specific knowledge-based information retrieval model using knowledge reduction. Doctoral Dissertation, University of Florida Gainesville, FL, USA,

Zhu, H., J.H. Xu and X.F. Ji, 2004. An approach to XML-based knowledge representation and its application in dynamic E-business. Proceedings of the IEEE International conference on E-Commerce Technology for Dynamic E-Business, Seprember 15-15, 2004, IEEE Computer Society Washington, DC, USA, pp: 196-199.