RESEARCH ARTICLE OPEN ACCESS

An Ontology Based Crawler for Retrieving Information Distributed on the Web

Wael A. Gab–Allah*,Ben Bella S. Tawfik**, Hamed M. Nassar***
**(Department Of Information Systems, Suez Canal University, Egypt)*
*** (Department Of Information Systems, Suez Canal University, Egypt)*
**** (Department Of Computer Science, Suez Canal University, Egypt)*

ABSTRACT
One of the principal motivations for the creation of the Web was to retrieve information in a fast and easy way. So, building systems for retrieving distributed information is crucially essential. This paper introduces an ontology based focused crawling system that exhibits high recall and high precision. The reason behind the power of the system is two–fold. First, it is focused, thanks to the underlying ontology–based retrieval subsystem. Second, operates in two phases, one to increase recall and the other to increase precision. We have implemented the proposed system using the Python language and the WordNet taxonomy. The results obtained by the system are given at the end of the paper and show clearly that it outperforms general purpose crawling systems built on approaches such as breadth first search.

Keywords: Focused crawler, Information retrieval, Ontology, Web search

I. INTRODUCTION

Distributed Information Retrieval (DIR) field lies in the intersection between Information Retrieval (IR) and Distributed Systems (DSs) as illustrated in Fig. 1. Distributed system is a collection of autonomous computers connected together and cooperate to achieve a common goal(s) [1].

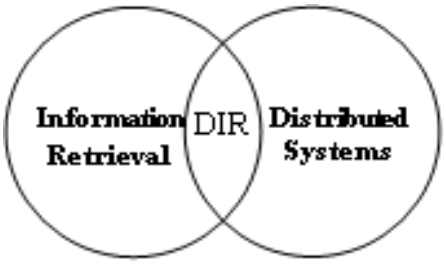

Fig. 1: Distributed Information Retrieval (DIR)

IR systems have been used for decades based on keywords matching. That is, a document may be retrieved if and only if it contains the query keywords. As a result, those systems cannot extract semantics in users' queries and hence information with similar semantics will be omitted. Unfortunately, the situation becomes worse with the exponential increase in the Web size. The problem of gathering information in scattered sources has become more critical, considering the huge number of databases on LANs and the Web [2, 3]. The issue of how to make use of the resources on the Web reasonably and efficiently has become an important concern of researchers. The goal of DIR is to provide a tool that searches the available databases and merges results into a single list back to the user.

A search engine was the tool designed to retrieve information on World Wide Web (WWW), or simply the Web. Nowadays, the researches of IR move closer to the semantic web, which utilizes different semantic technologies to analyze the semantics of documents and aims to extract significant information from them [4].

Normally, IR systems use crawlers for exploring the Web to find information. A crawler is one type of software agent. It is an agent which can automatically search and download Web pages [5]. There are two classes of crawlers: general purpose and focused [6]. The general purpose crawler searches the Web to construct its index. As a result, it confronts the hard job of creating, refreshing and maintaining a database of huge dimensions.

The focused crawler was introduced in 1999 [7] as a software agent that can traverse the Web and retrieve related information for specific topics, using semantic web technologies. The use of conceptual search, i.e. searching by meanings, has become a vital concern for researchers in the IR field to solve the limitations of keyword–based models. The goal of the focused crawler is to precisely and efficiently retrieve and download relevant information by understanding the semantics fundamentals of the predefined topics [8, 9].

The key feature of focused crawlers to potentially fulfill its task is the proper definition of the domain of interest. Therefore, ontology plays a vital role in the operation of focused crawlers and should be employed to enhance their performance by accurately defining the crawling boundary. In philosophy, an ontology is a theory about the nature of existence, or of what types of things exist. Form an

information science point of view, an ontology defines the relations among terms. Ontology can be defined as the specifications of conceptualization [10, 11]. It provides a terminology that can be used to identify a domain of interest, i.e. concepts and their relations. The most typical kind of ontology is a taxonomy.

WordNet is one of the most important resources for building an ontology [12]. The purpose of WordNet is to produce a combination of dictionary and thesaurus that is more intuitively usable, and to support text analysis. WordNet is used in many text classification methods as well as in IR because of its broad scale and free of charge license. WordNet is an ontology of lexical references whose design was inspired by the current theories of human linguistic memory [13]. Nouns, verbs, adjectives, and adverbs are gathered into sets of synonyms (known as synsets), each representing a distinct concept. Synsets are joined through conceptual–semantic and lexical relations as shown in the example in Fig. 2.

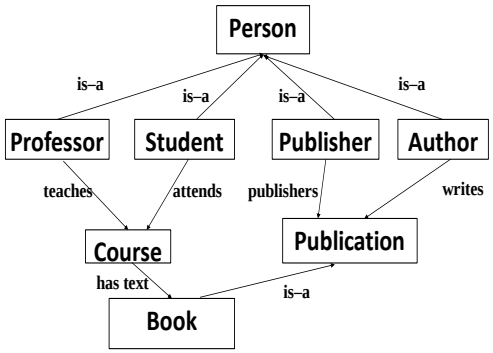

Fig. 2: Example of ontology structure

Semantic web technologies in general and ontology based approaches, in particular, are considered the foundation for the next generation of information services. This paper propose the use of ontology based focused crawlers models to provide a semantic level solution for IR so that it can provide fast, precise and stable query results. The proposed system aims to use a new algorithm to improve the accuracy of the distributed information retrieval process. The proposed system is named Ontology Based Distributed Information Retrieval (OBDIR).

The reminder of this paper is organized as follows: section 2 reviews related work. Section 3 introduces the proposed system architecture. Section 4 contains the results and discussions, and section 5 provides the conclusion.

II. RELATED WORK

A novel framework of open decision support system that is capable of gathering relevant information from an open–network environment was proposed in [14]. The authors exploited the context–based focused crawler architecture to discover local knowledge from interlinked systems. They elaborated on the knowledge alignment process to integrate the discovered local knowledge. However, they overlooked the intricate relationships between close terms.

In [15] the authors proposed a focused crawling framework supported by a statistical semantic association model with four kinds of semantic models: thesauruses, categories, ontologies, and folksonomies. They were able to boost the crawling performance for relevant prediction and ranking. However, the multitude of models they considered hindered the accuracy reaching convincing levels.

The study in [16] developed a new IR system integrating the semantic web with multi–agent. The system first analyzes and determines the semantic features of users' queries. Then it uses these features in extracting the most relevant information to the query in question. Also, it presents a new matching algorithm using semantics derived from content which can provide precise results meeting users' requirements. Furthermore, it collects information based on users' behavior. But the study lacked an ontology that could capture information with keywords that are subtly related to those supplied by the user.

In [17] the authors introduce ontology into query expansion and make efficient use of semantic relations of concepts in ontology to expand query and make the retrieval results more accurate and comprehensive. However, the time consumed by their system exceeds acceptable levels. Apparently, this has to do with the type and implementation of ontology they used.

In [18] the authors proposed a new semantic similarity based model (SSBM) and used it to cluster the topics in the domain. The model analyzed a document to get the semantic content. The SSBM assigns new weights to reflect the semantic similarities between terms. Higher weights are assigned to terms that are semantically close, whereas lower weights are assigned to those that are semantically further apart. However, the placement of the weights was not appropriate enough to return all relevant results.

III. PROPOSED ARCHITECTURE

The present study develops a novel distributed information retrieval system integrating an ontology extracted from WordNet with focused crawlers to handle the process of finding content semantics. Fig. 3 displays the block diagram of the proposed system. The major components of the proposed system can be summarized as follows:

- Ontology query expansion: Using WordNet to analyze and determine the semantic features of users' queries.
- Crawling technique: Based on ontology, which plays a vital role in this context to identify Web

pages for the domain of interest by using WordNet.

- Filtering strategy: The retrieved results will be scored semantically to determine their relevance to the user query. Pages with score lower than a certain threshold are filtered and only pages with high score will return back to user.

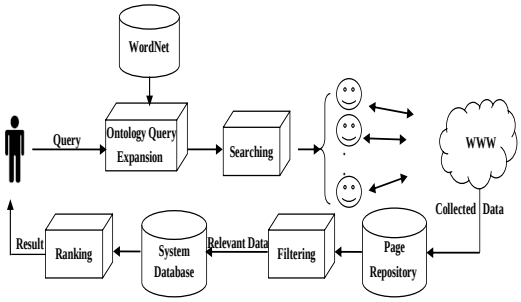

Fig. 3: The proposed OBDIR system architecture

```
Input the query Qin = {t1, t2,  ... ,tn} entered by the user, where the ti
                        are the terms in the query
Remove stop words i.e. words that do not convey a meaning such as 'the',
                                        'for', 'and', etc.
For i =1 to i <= n:
            Search WordNet for term ti
            If term ti is a leaf then:
                  Find Synonyms and Hypernyms
                  Add the found terms to Qex, where Qex is the query
                                        expanded set
            Else if term ti is a root then:
                  Find Synonyms and Hyponyms
                  Add the found terms to Qex
            Else
                  Get Synonyms, Hypernyms and Hyponyms
                  Add the found terms to Qex
            End if
End For
For i =1 to i <= n:
      For j =1 to i <= m:
            If SemSim (ti, tj) < Simth, where SemSim is semantic
                              similarity, Simth is semantic threshold,
then:
                  Remove tj from Qex
            End if
      End For
End For
Output Qout is the union of the initial terms in Qin and the list of
            accepted expanded terms in Qex i.e. Qout = Qin U Qex
```

Fig. 4: Algorithm for query expansion

3.1 Ontology Query Expansion

WordNet has been used ordinary for lexically expand the original query. A new algorithm for ontology query expansion of the input query is shown in Fig. 4. The proposed algorithm using WordNet to analysis the input query lexically and add semantic similarity terms to the input query.

3.2 Crawling Approach

The proposed crawler goes through the Web depending on the novel algorithm shown in Fig. 5. The algorithm is intended to control the operation of the crawler so as to be focused on the domain of interest. Specifically, it takes the concepts from the domain ontology and an initial list of Uniform Resource Locators (URLs) called seed URLs. Then it parses the hyperlinks in the page pointed to by the first URL, which necessitates other URLs to be crawled still. The crawler during discovering for relevant pages needs a mechanism to decide the relevance of these documents to the predefined domain of interest. Therefore, a specified domain relevance threshold, which is predefined by the user to extract only the relevant documents and discard irrelevant ones, will be used to determine if the extracted hyperlinks are relevant and have scores more than the domain relevance threshold. The crawler repeats these steps for all hyperlinks with scores exceeding the domain relevance threshold until all links are visited or the crawling limit threshold is reached.

```
lnput Seed URL (S_url), Domain Relevance Threshold (DR_th), Crawling Limit Threshold
(C_l)
Set PG_c = 0, where PG_c = number of pages downloaded,
While i <= S_url Do:
      Get URL
      Visit Web Page & Parse its content
      Set j=1, where j is the number of the hyperlink in the current page
      While j <= L_c, where L_c is number of links found in the page, Do:
            Calculate Relevance Score, RS_l, of the link
            If (RS_l > DR_th), where DR_th = Domain Relevance threshold, then:
                  Retrieve the document and dump it on the repository
                  PG_c = PG_c +1
            Else
                        Neglect the link
            End if
      End While
      If PG_c > C_l then:
            Break
      End if
End While
Output contents of the page repository, i.e. Relevant Web pages
```

Fig. 5: Algorithm for retrieving potentially relevant based on the domain relevance threshold

3.3 Filtering Strategy

Although results are retrieved through focused crawling, we still need a mechanism to verify the relevance of these documents to the specified domain of interest. All pages stored in the repository will be evaluated to calculate their relevance scores in order to remove irrelevant documents. To this end, a weight table containing weights for each term in the ontology, is used. In the weight table, we assign some weight to each term in our ontology. Terms which are common to many domains will take less weight, while terms which are specific to certain domain will take more weights. The task of weight assignment is made by knowledge experts.

In our approach we examine the relevancy the pages found in the page repository and crawled based on the relevance of domain ontology. The examination is done according to the algorithm in Fig. 6.

```
lnput Weight Table (WT), containing the terms and their weights, Document
Score Threshold (DS_th)
While i <= D_n, where D_n is the number of documents in the repository Do:
      lnput document D_i
      Parse document i to extract its TC_i Terms counts
      Set RSD_i =0, where RSD_i is The Relevance Score of Document i
      Set j=1, where j is number of the current terms in document i
      While j <= TC_i Do:
            Get term T_j
            Look up the weight WT_j of term T_j in weight table
            RSD_i = RSD_i + WT_j
      End while
      If (RSD_i < DS_th) then:
            Discard the document
      Else
            Retain the document in the collection of relevant documents
      End If
End while
Output The Pages retained
```

Fig. 6: Algorithm for extracting relevant documents from retrieved ones (those dumped in the Repository)

IV. SYSTEM EVALUATION AND EXPERIMENTAL RESULTS

In this section we define the metrics to be used for the evaluation of the proposed system. Then the system is tested through some experiments to evaluate its performance. As far as the proposed system is concerned, we can divide logically it into two phases. The first phase includes the task of query expansion and the task of crawling and retrieving what the system identifies as relevant

based on domain ontology. The output of this phase may include some irrelevant documents. The second phase includes the task of filtering the output of the first phase based on the weights of ontology terms. That is, some of the documents that have been obtained by the first phase may be discarded here.

4.1 Evaluation Metrics

Based on the performance of the proposed system, we can speak of 4 disjoint sets of documents.

- *True positive* (TP): This set contains all *relevant* documents that have been identified by the system as relevant, thus retrieved.
- *False positive* (FP): This set contains all ir*relevant* documents that have been identified by the system as relevant, thus retrieved.
- *True negative* (TN): This set contains all *irrelevant* document that have been identified by the system as irrelevant, thus *not* retrieved.
- *False negative* (FN): This set contains all *relevant* documents that have been identified by the system as irrelevant, thus not retrieved.

Clearly, two only of these sets are desired: the set of true positive and the set of true negative, and for this reason two evaluation metrics are defined as follows.

The most common evaluation metrics for IR system performance are recall, R, and precision, P [19–21]. Recall is defined in (1) as the ratio of documents that the system labels (by the second phase) as relevant to all the relevant documents on the Web. Using |S| to denote the cardinality of the set S, i.e. the number of elements in the set, then:

$$R = \frac{|TP|}{|TP| + |FN|} \qquad (1)$$

The metric that reflects the efficiency of the first phase of our proposed system.

Precision is defined in (2) as the ratio of documents that the system labels (by the second phase) as relevant to the documents the system labels as relevant (by the first phase). That is:

$$P = \frac{|TP|}{|TP| + |FP|} \qquad (2)$$

Relating to the proposed system, this indicates that the second phase improves precision by getting rid of some of the false negatives.

It should be noted that there are many other evaluation metrics, such as accuracy, area under curve (AUC), and average gradient. However, we will leave these other metrics in future work.

4.2 Numerical Experiments

To evaluate the performance of the proposed system, we implemented it in Python and used WordNet for finding the ontology of the domain of interest. Python is chosen because it is Open Source and also because of its well–known power and rigidity in Web programming (actually all Google modules are implemented in Python.) WordNet is chosen because it is also Open Source and moreover because it is comprehensive and universally hailed.

A set of 2000 documents (related to "News") are collected manually by searching popular sites (namely, Yahoo and Google search engines). This set is used to test our system. Specifically, a number of experiments are carried out, in each a subset of the 2000 documents is used as a population. For each experiment, a comparison of the recall and precision is made between the proposed system and a system that uses the standard Breadth First (BF) search technique.

Table 1 presents the recall and precision of the proposed system and the BF system. These results are also plotted in Fig. 7 and Fig. 8 respectively. The results indicate clearly that the proposed system provides higher recall and precision, regardless of the population size. It is interesting to note that in BF the precision decreases as the population size increases. This is because the BF technique can be considered a general purpose crawler that acts on the population as a whole. This phenomenon does not exist in the proposed system because it is a focused crawler that operates on each element of the population in isolation.

Table 1: Comparison the proposed system and a BF–based system.

Number of Documents	Proposed System		Breadth First	
	Recall	Precision	Recall	Precision
100	0.53	0.64	0.31	0.51
250	0.55	0.73	.035	0.51
500	0.59	0.78	0.38	0.47
1000	0.62	0.81	.041	0.43
2000	0.66	0.82	0.43	0.38

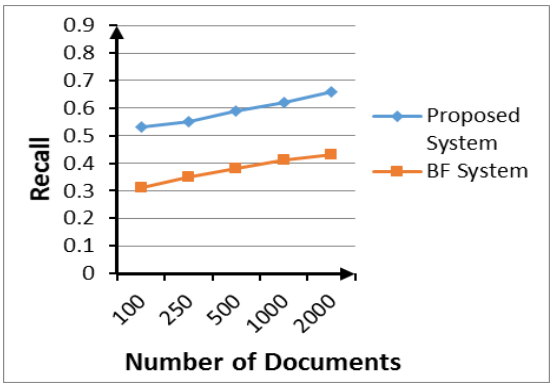

Fig. 7: Recall of the proposed system versus breadth first

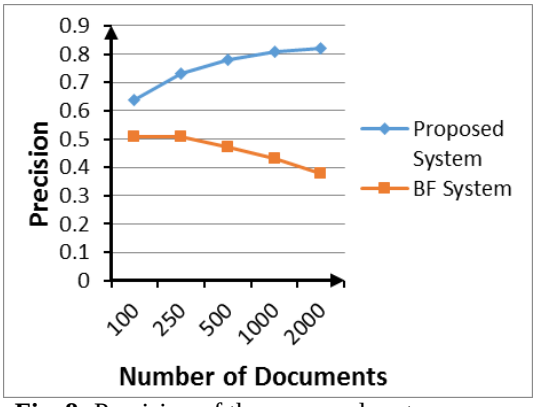

Fig. 8: Precision of the proposed system versus breadth first

V. CONCLUSION

Retrieving information distributed on the Web is a very complex task that integrates information finding, information filtering and information merging. A further complication added due to the large amount of available information. In this paper a framework for ontology–based DIR focused crawler system has been proposed to address the difficulties associated with information retrieval. The system is made up of two phases, one targeted to increase recall and the other targeted to increase precision. It is implemented completely in Python, because of its well–known power and rigidity in Web programming. The experimental results show that the proposed system greatly outperforms the general–purpose crawler, based on BF search.

For future work, the search technology is made personal and intelligent, to be close enough to users' needs and requirements. This necessitates further studies to design a semantic matching algorithm to calculate the matching degree between the retrieved results and users requests.

REFERENCES

[1]. De Krester, O., A. Moffat, T. Shimmin and J. Zobel, 1998. "Methodologies for Distributed Information Retrieval". Proceedings of the 18th International Conference on distributed computing systems, (pp. 26–29).

[2]. J. Callan, 2000. "Distributed information retrieval". In Advances in Information Retrieval, W. B. Croft, Ed. Kluwer Academic Publishers, (pp. 127–150).

[3]. A.M. Fard, M. Kahani, R. Ghaemi and H. Tabatabaee, 2007. "Multi–Agent Data Fusion Architecture for Intelligent Web Information Retrieval". World Academy of Science, Engineering and Technology, (pp. 73–77).

[4]. Dong, H., Hussain, F. K., and Chang, E., 2008b." State of the art in metadata abstraction crawlers". In Proceedings of the IEEE international conference on industrial technology, Chengdu, China, (pp. 1–6).

[5]. Kim, J.H, H.S. Shim, H.S. Kim, M.J. Jung, I.H. Chio and J.O. Kim, 1997. "A Cooperative Multi Agent System and its Real Time Application to Robot Soccer". Proceedings of the IEEE International Conference on Robotics and Automation, Albuqureque, New Mexico, (pp. 638–643).

[6]. S. Gunasekaran, M. Anisha and M. Anisha, 2015."A Comparative Study on Self Adaptive Semantic Focused crawler and Novel Focused Cell like Membrane Computing Crawler". International Journal of Computer Applications Vol. 112, No. 9, (pp. 12–18).

[7]. Chakrabarti S., van den Berg M. and Dom B., 1999. "Focused crawling: A new approach to topic–specific web resource discovery". In Proceedings of the eighth international conference on World Wide Web, Toronto, Canada, (pp. 545–562).

[8]. Hai Dong, Farookh Khadeer Hussain, and Elizabeth Chang, 2012. "Ontology–Learning–Based Focused Crawling for Online Service Advertising Information Discovery and Classification" in Springer–Verlag Berlin Heidelberg, (pp. 591–598).

[9]. DONG, Hai, Farookh Khadeer HUSSAIN, 2014." Self–adaptive semantic focused crawler for mining services information discovery". Industrial Informatics, IEEE Transactions Vol. 10, Issue 2,(pp. 1616–1626).

[10]. Dhingra, Vandana, and Komal Kumar Bhatia, 2015. "SemCrawl: Framework for crawling ontology annotated web documents for intelligent information retrieval". Intelligent Distributed Computing. Springer International Publishing, (pp. 213–223).

[11]. S.SASIREGA, A.Jeyachristy, 2014. "Ontology Based Web Crawler For Mining Services Informatiom Retrieval". International Journal of Computer Science and Mobile Computing, Vol. 3, Issue. 11, (pp.325–330).

[12]. Amine A., Elberrichi Z., and Simonet M., 2010. "Evaluation of Text Clustering Methods Using WordNet" The International Arab Journal of Information Technology, vol 7, no. 4, (pp. 349–357).

[13]. Kara, Soner, et al., 2012. "An ontology–based retrieval system using semantic indexing." Information Systems 37.4, (pp. 294–305).

[14]. Jung, J. J. 2009. "Towards open decision support systems based on semantic focused

crawling". Expert Systems with Applications, 36(2), (pp. 3914–3922).

[15]. Huang, R., Lin, F., & Shi, Z. Z, 2008. "Focused crawling with heterogeneous semantic information". In Proceedings of IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology, Sydney, Australia, (pp. 525–531).

[16]. J. LUO, and X. XUE, 2010. "Research on Information Retrieval System Based on Semantic Web and Multi–Agent. International Conference on Intelligent Computing and Cognitive Informatics, IEEE, (pp. 207–209).

[17]. W. Hongsheng, Q. Jiuying, and S. Hong, 2009. "Expansion Model of Semantic Query Based on Ontology". Web Mining and Web–based Application. WMWA '09. Second Pacific–Asia Conference on IEEE, (pp. 86–90).

[18]. G. Walaa and K. Mohamed, 2009. "Enhancing Text Clustering Performance Using Semantic Similarity". LNBIP 24, Springer–Verlag Berlin Heidelberg, (pp. 325–335).

[19]. Selamat, A. and M. H. Selamat, 2005. "Analysis on the Performance of Mobile Agents for Query Retrieval", Information Sciences, 172, no. 3, (pp: 281–307).

[20]. Jason J., Consensus–based evaluation framework for distributed information retrieval systems, Knowledge and Information Systems 18.2 , 2009, 199–211.

[21]. Amudaria, S., and S. Sasirekha, Improving the precision ratio using semantic based search, International IEEE Conf. on Signal Processing, Communication, Computing and Networking Technologies (ICSCCN), 2011, 465–470.