Domain-based Testbed for Peer-to-Peer Information Retrieval

Saloua Zammali
Dept. of Computer Science and Mathematics
Faculty of Sciences of Tunis, Tunisia
Email: zammalisalwa@gmail.com

Amira Ben Salem
Dept. of Computer Science and Mathematics
Faculty of Sciences of Tunis, Tunisia
Email: bnsalemamira@gmail.com

Khedija Arour
Dept. of Computer Science
National Institute of Applied Sciences and Technology of Tunisia, Tunis, Tunisia
Email: Khedija.arour@issatm.rnu.tn

***Abstract*—Information retrieval (IR) is a field that deals with storage and access to relevant information according to the user needs. The main goal of an Information Retrieval System (IRS) is to return to the user the most valuable documents in response to his queries. Classical models in IR are based on a general approach that meets the users invariably returning the same results for two users with the same issued query but having different information needs and different research preferences. Hence, the need to combine user domain interests with information retrieval becomes a challenge. The major issues raised by information retrieval, mainly, concerns domain interests modeling and domain exploitation in IR models. The major limitation of testbeds in distributed information retrieval (DIR) is mainly related to testbed that does not include domain interests as a source of evidence for evaluation of relevant documents. This problem becomes more insistent in Peer-to-Peer Information Retrieval (P2PIR) where there is not yet a standard testbeds for use. In this paper, we propose, *DBT*, a Domain-based Testbed for P2PIR. DBT is based on a new method for modeling peer and query domains. We represent these domains by using YAGO ontology.**

***Keywords-Testbed*; *Information retrieval*; *P2P systems*; *YAGO ontology*.**

I. INTRODUCTION

Information retrieval (IR) is a field that deals with storage and access to relevant information according to the user needs. The main goal of an information retrieval system (IRS) is to return to the user the most valuable documents in response to his queries. For a user query, an IR system allows to find a subset of potentially relevant documents, from a documents collection, responding to this query.

The growth of the web has delivered the IR face new challenges of access to information, namely to find relevant information in a diversified area and considerable size and that meets the need for specific user information. The major limitation of most classical information retrieval system is that they return, for a same query submitted by different users, the same results. However, users have different search background like interests, preferences, etc.

Studies, in [1], show that the problem of these systems lies partly in the fact that they are based on a general approach that considers the user information needs is completely represented by its query. To overcome this issue, the representation of user need must be extended in order to return the most useful information. As a result, the evaluation methodologies of these systems have been challenged by the consideration of extra external knowledge rather than the queries terms. That's why an appropriate testbed is needed, either in centralized IR or in distributed IR (particulary P2PIR) where queries and peers have limited descriptions. The testbed will be extended by semantic information provided from a semantic resource such as ontologies.

The main purpose of this extension is to make the testbed more enriched where user (*i.e.*, peer) need is not only represented by his queries but also through domains that describe the subject of the queries and peers. In this paper, we propose a domain-based testbed, suitable for the evaluation of P2PIR systems that takes in consideration the domain of queries and peers.

The paper is organized as follows. In Section II, we recall the key notions used throughout this paper. We review, in Section III, related work about building testbed for IR. In Section IV, we describe our approach of building distributed domain-based testbed. In Section V, we show our first experimental results. Finally, we present our conclusions regarding the current work and how this may relate to future trends P2PIR systems.

II. KEY NOTIONS

Before presenting our approach, we provide a simplified definition for some of the key concepts used throughout in this paper, namely, *testbed* and *ontology*.

A. Notion of ontology

An ontology represents knowledge as a set of concepts within a domain, and the relationships between those concepts. It can be used to reason about the entities within that domain and may be used to describe the domain [2]. An ontology can be constructed in two ways, domain dependent and generic. CYC [3], WordNet [4], and Sensus [5] are examples of generic ontologies.
One way of introducing external knowledge into IR is by

 ISBN: 978-1-61208-200-4

using ontology, for instance, by means of a list of keywords that reflect knowledge about the domain.

B. Notion of testbed

Definition 1 (Centralized information retrieval testbed): **Testbed** = documents collection + queries collection + relevance judgments.
Indeed, a testbed must provide the documents and the queries to be raised on these documents. The answers to the queries are often data provided by experts, together with the relevance judgements [6].

Definition 2 (Distributed information retrieval testbed): **Testbed** = documents collection+ queries collection + documents and queries distribution method among peers + documents and queries replication method among peers + evaluation metrics+ queries responses [7].

Definition 3 (Domain-based DIR testbed): We define the main components which a domain-based distributed testbed must provide as follows:

1) Test collection: documents collection, queries collection and relevance judgments.
2) A definition of a documents and queries distribution methods among peers.
3) A definition of a documents and queries replication methods among peers.
4) A set of semantic resources, such as ontologies, which provide semantics information.

In the following section, we review various work on the building testbeds in a centralized and distributed systems.

III. Related work on testbed building for DIR

A. Testbeds for centralized systems

For centralized Information Retrieval, there exist a significant number of standard centralized testbeds, such as the yearly competitions conducted by Cranfield [8] TREC [9], DMOZ [10], etc.

- Cranfield: Cranfield is the first centralized testbed, was created under the direction of C. Cleverdon. It is started in 1957 [8]. Cranfield is composed of 1400 documents and 221 queries [8].
- TREC: Text REtrieval Conference(TREC) [9] is designed as a series of workshops in the field of information retrieval.

B. Testbeds for decentralized systems

Building testbeds for distributed information retrieval systems is a challenge, in particular in P2PIR systems. Indeed, there is not yet a standard testbeds for use. To overcome this lack, Peer-to-Peer Information Retrieval benchmarking (P2PIRB), a framework for building distributed testbeds, is proposed in our previous studies [6]. P2PIRB framework provides a certain nombre of testbeds (such as Uniform Testbed, Random Testbed and specialized Testbed) [6].

C. Summary

Testbeds for classical centralized/decentralized IR systems have several problems, among which we can mention:

1) Testbeds are based on queries which are the only resources reflect information needs key of the user. Indeed, information needs of the user is represented by a single key resource, including a query keywords often expressed in natural language.
2) The interests of users, having made these queries, does not form a part of testbed.
3) Absence of real users: traditional evaluation model does not include real users in research contexts and replaces them with experts responsible for creating relevance judgments for each topic.
4) Classical evaluation measures are not exhaustive in the sense that the document is considered relevant if it recovers query topic, independently of the context and the task of research.

In this paper, we focus on the two first limits. In the literature, few approaches have been proposed for integration of interest domains in centralized testbeds [11][12][13]. However, theses testbeds are not freely available and not standardized. To the best of our knowledge, building domain-based testbeds has not been widely addressed in distributed information retrieval.

To tackle this limitations of traditional testbeds, in recent years, there has been an increasing research interest in the problem of enrichment testbed with domains of interest. Addressing these issues, we propose a domain-based distributed testbed suitable for the evaluation of P2PIR systems.

IV. Domain-Based Testbed for P2PIR

A. Global architecture of creating a domain-based distributed testbed

The aim of our approach is to build a distributed testbed extended with metadata representing the domains of query and peer. The use of domain in evaluation approaches addresses the above limitations of the traditionnal evaluation. Therefore, the proposed approach consists of three parts: testbed building, domain modeling and domain integration. The architecture of the process of creating a domain-based distributed testbed is described in Figure 1.

B. Testbed building

To distribute documents and queries among the set of peers, we used the Benchmarking Framework for P2PIR [6]. This framework is configurable, which allows user to choose certain parameters (*i.e.*, number of peers, replication of queries, etc.) and provides XML files describing the nodes, the associated documents and the queries to be launched on the network.

 ISBN: 978-1-61208-200-4

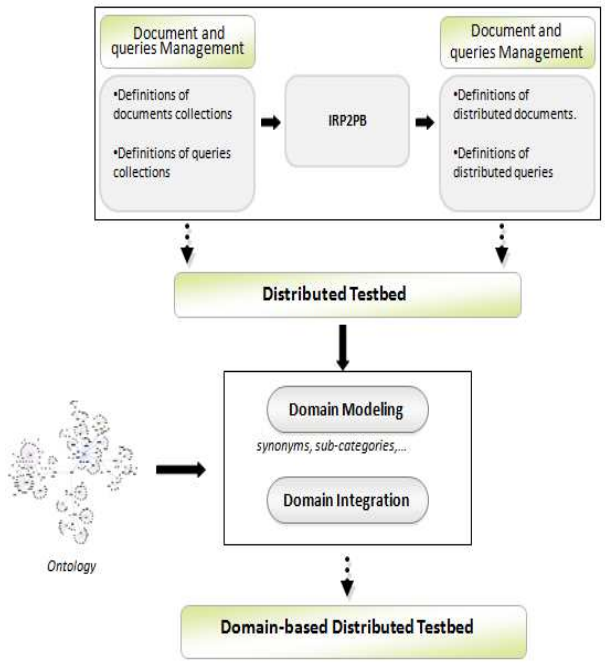

Figure 1. Global architecture of creating a domain-based distributed testbed

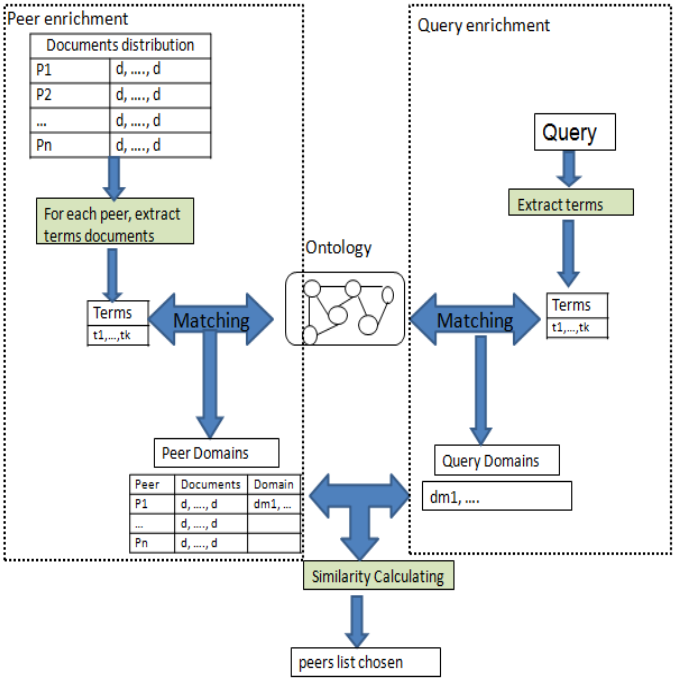

Figure 2. Domain modeling process

C. Ontology-based domain modeling

Modeling domain, in our work, is based on extracting knowledge from a given ontology. Knowledge extraction refers to the automatic extraction of structured information such as entities, relationships between entities, and attributes describing entities from unstructured sources such as text. Our domain modeling can be formulated as follows:
Let $\mathcal{T} = \{t_1, \ldots, t_n\}$: a set of terms and O: an ontology.
After doing the correspondence between the terms of $\mathcal{T}$ and the entities of $\mathcal{O}$, we obtain, from $\mathcal{O}$, a set of entities (*i.e.*, terms). For several terms, we will obtain a larger set of ontology entities. The entities with higher frequency are selected and it represents the most appropriate sense to the set of terms: $\mathcal{E} = \{e_1, \ldots, e_i\}$.
For each e_i, we extract from $\mathcal{O}$:

- a set of synonyms *Syn* $= \{s_1, \ldots, s_j\}$,
- a set of general terms $\mathcal{G} = \{t_{g1}, \ldots, t_{gk}\}$,
- a set of specific terms $\mathcal{S} = \{t_{s1}, \ldots, t_{sl}\}$.

Therefore, the domain $\mathcal{D}$, of the set of terms in $\mathcal{T}$, is represented by a set of entities $\mathcal{E}$, called domains. Each domain e_i, is represented by the set of synonyms *Syn*, the set of general terms $\mathcal{G}$ and the set of specific terms $\mathcal{S}$: $\mathcal{D} =< e_i, Syn, \mathcal{G}, \mathcal{S} >$

D. Domain integration in testbed

A node, in a P2P network, contains a collection of homogeneous documents that represents its center of interest. In order to realize this, we use a dataset that reflects real scenarios. Figure 3 illustrates the structure of a peer content. Each peer is described by its documents and a set of domains:

$$p = (docs, \{dom_1, dom_2, \ldots, dom_k\})$$

where: $docs$ is the documents of peer p and k is the number of domains for considered peer and each domain is constructed as follows:

$$dom = (synos, sub\_catgs, sous\_catgs)$$

$synos$, sub_catgs, $sous\_catgs$ are respectively: synonyms, sub categories, sous categories of peer documents.
Figure 2 provides a visual representation of domain-based peer enrichment.

V. Evaluation Methodology

In order to get meaningful terms in the centralized collection, we decided to build a new test collection, where queries and documents terms are not generated randomly but in a way to ensure semantic between terms.
To build a test collection, you must specify:

- What are the criteria for the selection of documents.
- How to identify relevant documents for each query.

A. Document collection

We choose to use Delicious [14] tags and we consider the tags made by each user for a specified article as the terms of a document. For this purpose, we used the dataset published in Social-ODP-2k9 Dataset [15].

 ISBN: 978-1-61208-200-4

```
<peer peer-id="1">
  <documents>
    <document document-id="30" />
    <document document-id="31" />
    <document document-id="32" />
  </documents>
  <Domains nb="8">
     <Domain Domain- name = "music">
        <Properties>
          < synonyms >
            </word> concert </word>
            <word> tune </word>
          </ synonyms >
          <sub_categories>
            <sub_category> ballet </sub_category>
                 ....
            <sub_category> tango </sub_category>
          </sub_categories>
        </Properties>
  </Domains >
</peer>
```

Figure 3. Illustration of the peer's structure

Delicious is a website that save and share tagged web page and classify them according to the principle of folksonomy by tags. It was created in 2003 by Joshua Schachter in order to save their personal bookmarks. The site interface is based on HTML, which makes the site easy to use. Delicious content is organized via RSS (Rich Site Summary) and is based on *tags* notion. Tags are keywords describing the content of the document (*e.g.*, Sports, Cinema, Internet, etc.). We investigate the tag sets in delicious thanks to both its popularity and availability. Social-ODP-2k9 is a dataset created during December 2008 and January 2009 with data retrieved from Delicious and StumbleUpon social bookmarking sites, the Open Directory Project and the Web. It is available for research purposes and has XML format, as shown in Figure 4. The tags $<document>$ and $</document>$ mark

Algorithm 1: DOCUMENTS-BUILD

Algorithm: *Documents-Build*(AC, Pnb, n)
Input:
AC: Articles collection.
Pnb: Peers number in network.
n: Documents number per peers.
Output:
$\mathcal{DF} = \{D_{peer_1}\} \cup \{\ldots\} \cup \{D_{peer_{Pnb}}\}$: Documents collection.
begin
  for ($i = 1$; $i < |AC|$; i++) **do**
    $D_{peer_i} := \emptyset$;
    for ($j = 1$; $j < n$; j++) **do**
      d_j = ExtractTagsFromUsers(j, D_{Pnb});
      $D_{peer_i} = D_{peer_i} \cup \{d_j\}$;
  $\mathcal{DF} = \{D_{peer_1}\} \cup \{\ldots\} \cup \{D_{peer_{Pnb}}\}$;
  return ($\mathcal{DF}$)

```
<documents>
  .....
  <document>
     <tags>
       ....
       <tag>
          <name> Tag name </name>
       </tag>
       ....
     </tags>
       ....
     <detailedtags>
       ...
       <user>
       ....
       <tag>Tags assigned by a user </tag>
       ....
       </user>
        ...
       </detailedtags>
  </document>
       ...
</documents>
```

Figure 4. Illustration of the delious's structure

$<user>$ element) who have tagged (delimited by $<tag>$ element). The construction of test collection from delicious is described by algorithm 1. All articles (*i.e.*, documents) in AC collection is partitioned according to the number of documents per peers. To build the documents collection associated to peer i, we use $ExtractTagsFromUsers$ algorithm to extract the tags corresponding to the article i.

Delicious is based on tags technology. Tags are in the form of a word (*e.g.*, sports, movies, Internet, etc.) can quickly find relevant sites to the tag. Therefore, an ODP site (having url) can be tagged by multiple users with different terms. In our case, we considered:

- The URLs of ODP represent peers.
- The tags, for a given user and article, represents document terms.

The idea behind this choice is that each peer usually has a homogeneous collection of documents representing these interests. However, an ODP article is tagged by several users, but these tags, necessarily, have a certain correlation between them. To simulate this behavior and remain in a realistic environment, we have assigned the sets of tags (each set of tags represents a document), corresponding to a given URL, to a given peer.

B. Queries collection and relevance judgements

A query represents the user information need. Queries collection must adequately model human users behavior. Indeed, queries collection should represent the needs of non-expert users (for example, ambiguous query represented by a single term) and must also represent expert need users.

 ISBN: 978-1-61208-200-4

Studies have shown that the queries submitted by users are relatively short and are generally limited to less than three keywords [16]. For this, we have established three set of queries: the first contain one term, the second contain two terms and the third contain three terms.
Relevance judgements are obtained using the cosine function, given in equation 1.

$$S(d_j, q) = \cos(\overrightarrow{q}, \overrightarrow{d_j}) = \frac{d_j \times q}{|d_j| \times |q|} \quad (1)$$

The cosine function, given in equation 1, is often used to determine the similarty between a document d_j and a query q.

C. Queries Distribution

Queries distribution among peers is done in a completely random manner, but under the constraint that queries repartition is proportional to the documents one. We used the *IRP2PB* tool for queries distribution on peers [6].

D. Ontology

YAGO (Yet Another Great Ontology) is a huge semantic knowledge base. Figure 5 represent a fragment of YAGO knowledge representation. It contains 2 million entities (such as persons, organizations, cities, etc.). This ontology contains 20 million facts about these entities [17]. The main reasons for using this resource are:

- It is derived from Wikipedia and WordNet.
- It exists in many formats (XML, SQL, RDF, etc.).
- It covers a vast amount of individuals.

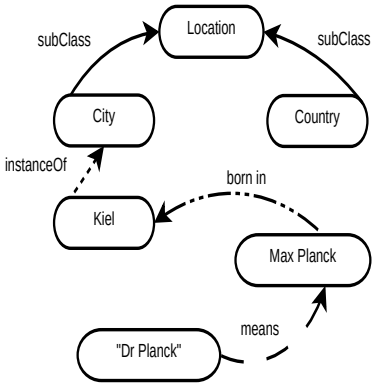

Figure 5. Fragment of YAGO ontology

E. PeerSim simulator:

To evaluate the approach proposed in this paper, we have chosen to use the PeerSim [18] simulator which is an open source tool written in Java. It has the advantage of being dedicated to the study of P2P systems [7]. It has an open and modular architecture allowing it to be adapted to specific needs. More precisely we use an extension of PeerSim developed by the RARE project [19]. This extension can be seen as a PeerSim specialization for information retrieval.

F. Routing Algorithms:

- *Gnutella*: a system that used a simple constrained flooding approach for search. A query was forwarded to a fixed number of neighbors until its time-to-live (TTL) in terms of forwarding steps was exhausted or a loop was detected [6].
- *DBR* (Domain-based routing): The pseudo-code for our routing algorithm is given by algorithm 2.
  The *DBR* peer selection algorithm uses the YAGO ontology for select suitable peers. This is due to the enrichment of peer structure by its interests (*i.e.*, extracted domains from YAGO). Indeed, initially, each peer has a set of documents representing their interests (*i.e.*, domains). In order to express, explicitly, we used YAGO ontology (as detailed previously in the section IV-C).
  For a query Q, the algorithm determines from ontology, a set of domains associated to the query (denoted by *QueryDoms*: getQueryDomains).
  Determine the set of domains, for each pair, denoted by $PeersDocsDoms$ (getPeersDocsDomains() function).
  For each peer domain, determine a set of domains similar to Q which are sorted according to the similarity value (getSimilarDomain() function of algorithm 2).
  The similarity between a domain $dom \in PeersDocsDoms$ and the domains *QueryDoms* of Q is determined by the formula as follows:

$$Sim(QueryDoms, dom) = \frac{|QueryDoms \cap dom|}{|QueryDoms \cup dom|} \quad (2)$$

G. Evaluation Metrics:

To compare the performance of the two routing algorithms, we used the metrics Recall (R) and Precision (P) defined as follow: given a query Q, consider RDR the number of relevant documents returned, RD is the number of relevant documents and DR the number of documents returned:

$$R(Q) = \frac{RDR}{RD} \quad (3)$$

$$P(Q) = \frac{RDR}{DR} \quad (4)$$

H. Initialize simulation parameters

The simulation, of both algorithms DBR and Gnutella, is based on the parameters:

- *TTL* (Time To Live): Maximum depth of research, initialized to 4.

 ISBN: 978-1-61208-200-4

- *Pmax*: Maximum number of peers which the query should be propagated to.
- *Overlay size*: Number of peers in the network, initialized to 500.

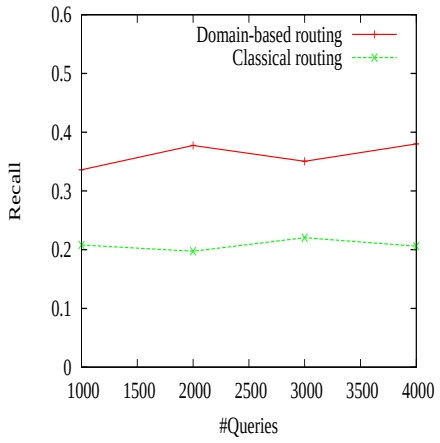

Figure 6. Relation between Recall and Nbr of Queries according to Gnutella and DBR algorithms

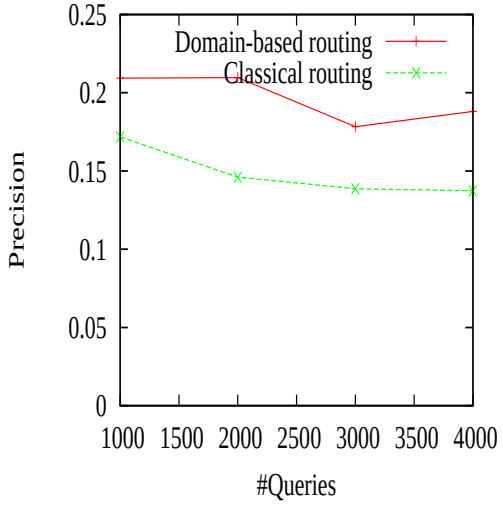

Figure 7. Relation between Precision and Nbr of Queries according to Gnutella and DBR algorithms

I. Experimental Results

In this experiment, we compared the performance of routing algorithms performed with and without considered interest domains of user.

To compare the performance of the domain-based algorithm (*i.e.*, DBR) and the classical routing (*i.e.*, Gnutulla), we calculate the average recall and precision per interval of 2000 queries sent by different peers in the system.

Figure 6 shows that the average recall of DBR algorithm is between 0.33 and 0.38 while the average recall for Gnutella is between 0.19 and 0.22.
Figure 7 shows that the average precision of DBR algorithm is between 0.18 and 0.20 while the average precision for Gnutella is between 0.13 and 0.17.

These results show that the *DBT* testbed significantly improves the effectiveness of the DBR routing algorithm.
In addition, when comparing classical routing to the domain-based routing, we see better recall and precision in the search results since domain-based query retrieve documents that would not be retrieved by using only the keyword-based query.

VI. Conclusion and Future Work

The field of information retrieval is very experimental in nature. We identify the need to create testbeds for information retrieval experimentation. We propose, *DBT*, a testbed for P2PIR, based on interests domains peers. In this paper, we demonstrated that ontology can be used to model peer interest domains and these domains can be used to improve distributed information retrieval.

The first tests presented in this paper are very encouraging. One possible perspective to this work is to vary the number of documents and queries and use other routing algorithms in the aim of making *DBT* testbed more used. We plan to study a new dimensions such as peer location, time and integrate them in distributed testbeds to the aim of improving search effectiveness.

Algorithm 2: Domain-Based Routing Algorithm

Algorithm: *Domain-Based Routing Algorithm*(Q, O)
Input:
Q: Query.
O: Ontology.
Output:
$selectedPeers$: selected peers list.
begin
  $QueryDoms$:= getQueryDomains(Q, O) ;
  $PeersDocsDoms$:= getPeersDocsDomains(O);
  $SimQP$:= $\emptyset$;
  foreach $PDom \in PeersDocsDoms$ **do**
    $SimQP$:= $SimQP \cup$
    getSimilarDomain($PDom$, $QueryDoms$);
  $selectedPeers$:= getSelectedPeers($SimQP$) ;
  return *($selectedPeers$)*

References

[1] J. Budzik and K. Hammond, “User interactions with every applications as context for just-in-time information access,” in *Proceedings of the 5^{th} international conference on intelligent user interfaces*, Mars 2000, pp. 44–51.

 ISBN: 978-1-61208-200-4

[2] R. Gruber, “Toward principles for the design of ontologies used for knowledge sharing,” *International Journal of Human-Computer Studies*, vol. 43, pp. 907–928, December 1995.

[3] D. B. Lenat, “Cyc: a large-scale investment in knowledge infrastructure,” *Commun. ACM*, vol. 38, no. 11, pp. 33–38, 1995.

[4] G. A. Miller, “Wordnet: a lexical database for english,” *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[5] B. Swartout, R. Patil, K. Knight, and T. Ross, “Toward distributed use of large-scale ontologies,” in *Proceedings of the* 10^{th} *workshop on knowledge acquisition for knowledge-based systems*, Canada, 1996.

[6] S. Zammali and K. Arour, “P2PIRB: Benchmarking Framework for P2PIR,” in *Data Management in Grid and P2P Systems (Globe 2010)*, Spain, September 2010, pp. 100–111.

[7] ——, “An Evaluation of a Cluster-based Testbed,” in 6^{th} *International Conference on Internet and Web Applications and Services*, The Netherlands Antilles, March 2011, pp. 136–141.

[8] C. Cleverdon, “The cranfield test on index language devices,” in *Association of special libraries (Aslib)*, London, March 1967, pp. 173–194.

[9] “Trec web site,” January 2012, http://trec.nist.gov/.

[10] “DMOZ web site,” January 2012, http://www.dmoz.org/.

[11] L. Tamine-Lechani, M. Boughanem, and M. Daoud, “Evaluation of contextual information retrieval effectiveness: overview of issues and research,” *Knowl. Inf. Syst.*, vol. 24, pp. 1–34, July 2010.

[12] A. Sieg, B. Mobasher, and R. Burke, “Web search personalization with ontological user profiles,” in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, Portugal, 2007, pp. 525–534.

[13] A. Sieg, B. Mobasher, S. Lytinen, and R. Burke, “Using concept hierarchies to enhance user queries in web-based information retrieval,” in *Proceedings of the International Conference on Artificial Intelligence and Applications*, 2004.

[14] “Delicious web site,” January 2012, http://delicious.com/.

[15] “Social ODP web site,” January 2012, http://nlp.uned.es/social-tagging/socialodp2k9/.

[16] A. Spink, H. Ozmutlu, S. Ozmutlu, and B. Jansen, “U.s. versus european web searching trends,” *SIGIR Forum*, vol. 36, pp. 32–38, September 2002.

[17] F. Suchanek, G. Kasneci, and G. Weikum, “Yago: a core of semantic knowledge,” in *Proceedings of the* 16^{th} *international conference on World Wide Web*, Canada, 2007, pp. 697–706.

[18] M. Jelasity, A. Montresor, G. P. Jesi, and S. Voulgaris, “The peersim simulator,” `http://peersim.sf.net/`, January 2010.

[19] “RARE project,” January 2012, http ://www-inf.int-evry.fr/defude/RARE/.

 ISBN: 978-1-61208-200-4