SOON: A Scalable Self-Organized Overlay Network for Distributed Information Retrieval

Juan Li and Son Vuong

Computer Science Department, University of British Columbia
2366 Main Mall, Vancouver, B.C., Canada
{juanli, vuong}@cs.ubc.ca

Abstract. Locating desirable resources and information from a large-scale distributed system such as P2P system and grid is a very important issue. However, the distributed, heterogeneous, and unstructured nature of the system makes this issue very challenging. In this paper, we propose Self-Organized Overlay Network (SOON), an unstructured P2P overlay architecture, to facilitate sharing and searching semantically heterogeneous contents. In particular, we have proposed a semantics-aware topology construction method to group nodes sharing similar semantics together to create small-worlds. For this purpose, we have designed an algorithm to extract a node's ontology summary and use that summary to compute the semantic similarity between nodes. With this semantic similarity defined, nodes are grouped accordingly, forming semantic virtual domains and clusters. Resource information integration and searching can be efficiently performed on top of this topology.

Keywords: Overlay network, P2P system, Semantic Web, topology.

1   Introduction

A widely-held belief pertaining to social networks is that any two people in the world are connected via a chain of six acquaintances (*six-degrees of separation*) [1]. The quantitative study of the phenomenon started with Milgram's experiments [2]. Milgram's experiments illustrated that individuals with only local knowledge of the network (i.e., their immediate acquaintances) may successfully construct acquaintance chains of short length, leading to networks with "small-world" characteristics. In such a network, a query can be forwarded along acquaintance chains taking it closer to the destination. Randomized network constructions that model the small-world phenomenon have recently received considerable attention. To model the routing aspects of the small-world phenomenon, Kleinberg constructed a family of random graphs [3]. He considered a 2D $n \times n$ grid with n^2 nodes. Each node is equipped with a small set of "local" contacts and one "remote" contact drawn from a harmonic distribution. With greedy routing, the path-length between any pair of nodes is $O(log^2 n)$ hops, with high probability.

Small-world networks exhibit special properties, namely, a small average diameter and a high degree of clustering. A small diameter corresponds to a small

separation between peers, while high clustering signals tight communities. Small-world graphs contain inherent community structure, where similar nodes are grouped together in some meaningful way. Intuitively, a network satisfying the small-world properties would allow peers to reach each other via short paths while maximizing the efficiency of communication within the clustered communities.

We draw inspiration from small-world networks and organize nodes in our system to form a small-world topology, particularly from a semantic perspective. Our objective is to make the system's dynamic topology match the semantic clustering of peers, i.e., there is a high degree of semantic similarity between peers within the clustered community; this would allow queries to quickly propagate among relevant peers as soon as one of them is reached.

In our overlay network, SOON, peers use their ontology summary to represent their expertise. Unlike most existing systems, SOON does not assume a global ontology but heterogeneous ontologies. We have designed a novel algorithm to compute the semantic similarity between two nodes in the network; then we use the semantic similarity as the metric to organize the network topology. Nodes are loosely structured in this network. Each of them keeps track of a set of neighbors and organizes these neighbors into a multi-resolution neighborhood according to their semantic similarities. This way, the overlay network topology is reconfigured with respect to peers' semantic properties, and peers with similar ontologies are close to each other. Information can be integrated and discovered through nodes' current neighbors, rather than by contacting some central hubs or virtual central hubs, such as Distributed Hash Tables (DHTs). This architecture combines the efficiency and scalability of structured overlay network with the connection flexibility of unstructured networks. It achieves full distribution, high scalability, and robustness.

2 Semantic Metadata

Metadata, the data about data, is a crucial element of a sharing and discovering infrastructure. An effective way of locating information of interest within large-scale information intensive environments is providing and managing metadata about the information. More important, metadata should be able to express the meaning of the information. An ontology, "a shared and common understanding of a domain that can be communicated between people and application systems", as considered in modern knowledge engineering [4], is precisely intended to convey that kind of shared understanding. An ontological representation defines concepts and relationships. It sets the vocabulary, properties, and relationships for concepts. The elements accumulate more meaning by the relationships they hold and the potential inferences that can be made by those relationships. This capability of formal ontologies to convey relationships and axioms make them ideal vehicles for describing the vocabulary for metadata statements, providing a rich formal semantic structure for their interpretation. Therefore, we use ontologies to represent information metadata semantics. To cope with the openness and extensibility requirements, we adopt two W3C recommendations, the Resource Description Framework (RDF) and the Web Ontology Language (OWL), as our ontology languages.

In our system the ontology knowledge is represented by OWL-DL and is separated into two parts: the terminological box (T-Box) and the assertion box (A-Box) as defined in the description logic terminology. The T-Box is a finite set of terminological axioms, which includes all axioms for concept definition and descriptions of domain structure, for example a set of classes and properties. The A-Box is a finite set of assertional axioms, which includes a set of axioms for the descriptions of concrete data and relations, for example, the instances of the classes defined in the T-Box. Generally speaking, there are many more A-Box instances than T-Box concepts. Separating the T-Box and A-Box enables different coarse-grained knowledge indexing, thus increasing the scalability of the system.

3   Semantic Similarity

To organize peers according to their semantic properties, we need a metric to measure peers' ontology similarity. There has been extensive research [6, 7, 8] focusing on measuring the semantic similarity between two objects in the field of information retrieval and information integration. However their methods are very comprehensive and computationally expensive. In this paper, we propose a simple method to compute the semantic similarity between two peers; this can easily be replaced with other advanced functions for a complex system.

3.1   Ontology Signature Set (OSS)

To measure the semantic similarity between peers, we need to extract each peer's semantic characteristics. The T-Box part of an ontology defines high-level concepts and their relationships like the schema of a database. It is a good abstraction of the ontology's semantics and structure. Therefore, we use keywords of a nodes' T-Box ontology as its ontology summary. For each node, we extract the class and property labels from its T-Box ontology, and put them into a set. This set is called this node's Ontology Signature Set (OSS). We can measure the similarity of two ontologies by comparing the elements of their OSSs. However, a semantic meaning may be represented by different labels in different ontologies, while it is also possible that the same literal label in different ontologies means totally different things. Ontology comparison based on primitive OSSs may not yield satisfying results. One improvement is to extend each concept with its semantic meanings, so that semantically related concepts would have overlaps. Based on this intuition, we use the lexical database, WorldNet [5], to extend the OSS to include words which are semantically related to the concepts from the original set.

WordNet is conceived as a machine-readable dictionary. It structures lexical information in terms of word meanings. WordNet maps word forms in word senses using the syntactic category as a parameter. Words of the same syntactic category that can be used to express the same meaning are grouped into a single synonym set, called *synset*. For example, the noun "computer" has a synset: {computer, data processor, electronic computer, information processing system}. An intuitive idea of extending an OSS is to extend each concept with its synset, i.e., its synonyms. In this

way, two semantically related ontologies would have common WordNet terms in their extended OSSs. Besides synonyms, WordNet also includes other lexical semantic relations, such as *is-a*, *kind-of*, *part-of*. Among these relations, *is-a* (represented by hyponym/hypernym in WordNet) is the most important relationship; it explains a concept by a more general concept. Therefore, we also extend OSS concepts with their hypernyms.

After extension, an OSS may get a large number of synonyms for each concept. However, not all of these synonyms should be included in the set, because each concept may have many senses (meanings), and not all of them are related to the ontology context. Having unrelated senses in the OSS will diminish the accuracy of measuring the semantic similarity; thus we have to prune the expanded OSS to exclude those unrelated terms. A problem causing the ambiguity of concepts in OSS is that the extension does not make use of any relations in the ontology. Relations between concepts are important clues to infer the semantic meanings of concepts, and they should be considered when creating the OSS. Therefore, we utilize relations between the concepts in an ontology to further refine the semantic meaning of a particular concept. Only words with the most appropriate senses are added to the OSS. Since the dominant semantic relation in an ontology is the *subsumption* relation, we use the *subsumption* relation and the sense disambiguation information provided by WordNet to refine OSSs. It is based on a principle that a concept's semantic meaning should be consistent with its super-class's meaning. We use this principle to remove those inconsistent meanings. The refined algorithm to extend the primitive OSS is illustrated with the pseudocode in Fig. 1.

```
/* This algorithm generates the refined ontology signature
set OSS for an ontology, O */

createOss(Ontology O)
{
   OSS={};
   for each c ∈ {concepts of ontology O} do
        pc is parent concept of c
        add c, pc to oss
        for each Sc ∈ {senses of c} do
            Hc={hypernyms of Sc}
            for each Spc ∈ {senses of pc) do
                if Hc ∩ Spc !=null
                    add Sc,Spc to OSS
}
```

Fig. 1. A refined algorithm to create the Ontology Signature Set of an ontology O.

The algorithm in Fig.1 creates the refined OSS of an ontology by adding the appropriate sense set of each ontology concept based on the *sub-class/super-class* relationships between the parent concepts and child concepts. For every concept in an ontology, we check each of its senses; if a sense's hypernym has overlap with this concept's parent's senses, then we add this sense and the overlapped parent's sense to the OSS set. In this way we can refine the OSS and reduce imprecision. Possible improvements could be obtained by using other relations in the ontology.

3.2　Peer Semantic Similarity

To compare two ontologies, we define an ontology similarity function based on the refined Ontology Signature Set. The definition is based on Tversky's "Ratio Model" [9] which is evaluated by set operations and is in agreement with an information-theoretic definition of similarity [10].

Definition 1: Assume A and B are two peers, and their extended Ontology Signature Sets are S(A) and S(B) respectively. The semantic similarity between peer A and peer B is defined as:

$$sim(A,B) = \frac{|S(A) \bigcap S(B)|}{|S(A) \bigcap S(B)| + \alpha |S(A) - S(B)| + \beta |S(B) - S(A)|}$$

In the above equations, "$\bigcap$" denotes set intersection, "$-$" is set difference, while "$\|$" represents set cardinality, "α" and "β" are parameters that provide for differences in focus on the different components. The similarity *sim*, between *A* and *B*, is defined in terms of the semantic concepts common to OSS of *A* and *B*: $S(A) \bigcap S(B)$, the concepts that are distinctive to *A*: $S(A)-S(B)$, and the features that are distinctive to *B*: $S(B) - S(A)$. With the similarity measure specified, we have the following definition:

Definition 2: Two nodes, node A and node B are said to be semantically equivalent if their semantic similarity measure, sim(A,B) equals to 1 (implying sime(B,A)=1 as well). Node A is said to be semantically related to node B, if sim(A,B) exceeds the user-defined similarity threshold t (0<t≤1). Node A is semantically unrelated to node B if sim(A,B)<t.

4　Self-Organized Semantic Small-World Overlay

We follow the idea of the Kleinberg experiment to construct the semantic small-world network. In Kleinberg's experiment each node keeps many short-range contacts, as well as a small number of long-range contacts. In our system, a node distinguishes three kinds of neighbors based on their semantic similarity. A peer A's neighbor, B, can be one of these three types: (1) zero-distance neighbor (or semantically equivalent neighbor), if $sim(A,B)=1$, (2) short-distance neighbor (or semantically related neighbor) if $sim(A,B) \geq t$ ($0<t<1$ is A's semantic threshold), (3) long-distance neighbor (or semantically unrelated neighbor) if $sim(A,B)<t$. A node always tries to find as many close neighbors as possible, but it also keeps some long distance neighbors to reach out to other ontological clusters.

Nodes in the system randomly connect to each other through these three types of neighbors. They produce a semantically clustered small-world topology as shown in Fig.2. The clustered structure is not flat but multi-layered; nodes with similar ontological topics, i.e., short-distance neighbors, form a domain (a region formed by nodes with the same shape in the figure); inside the domain, nodes may create smaller clusters (sub-regions in a domain with same color) if they share the same ontology schema. For example, all peers in the medical domain are interested in medically related information. They may be interested in different aspects of the medical resources, and they may use different ontologies to describe their resources. They

connect with each other through short-distance links. Inside the medical domain, nodes further organize themselves to finer-grained clusters based on their ontologies. For example, node N_1, N_2, N_5, N_8 and N_{11} use the same ontology O_3 (e.g., a medical ontology, SNOMED-RT [26]). Clusters and domains do not have fixed boundaries; they are formed by randomly connecting relevant nodes.
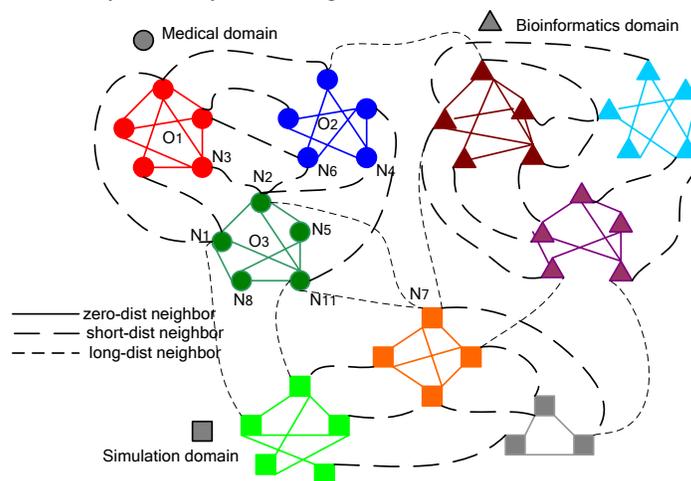

Fig. 2. The semantic small-world network topology.

The construction of an ontology-based topology is a process of finding semantically related neighbors. A node joins the network by connecting to one or more bootstrapping neighbors. Then the joining node issues a neighbor-discovery query, and forwards the query to the network through its bootstrapping neighbors. When a node N receives a neighbor-discovery query Q which tries to find neighbors for a new joining node X, N computes the semantic similarity between X and itself. If N is semantically related to X, N will send a reply to X. If the query's TTL does not expire, N computes the semantic similarity between X and each of its neighbors, and forwards the query to semantically related neighbors. If no semantically related neighbors are found, the query will be forwarded to N's long-distance neighbors.

With the semantic small-world topology constructed, information discovery can be efficiently performed. In most cases, a discovery query can be answered within the querying node's local domain, because queries reflect the querying node's ontology interest, and semantically related nodes are within the neighborhood of the querying node. When a node issues (or receives) a query, it first chooses its zero-distance neighbors to forward the query inside the local cluster. Since they use the same ontology, the zero-distance neighbors are the best candidates to forward the query to. Another important step in query processing is reformulating a peer's query over other peers on the available semantic paths. Starting from the querying peer, the query is reformulated based on the inter-ontology mapping over the querying peer's short-distance neighbors, then over their short-distance neighbors, and so on until the query TTL expires. Interested readers can refer our previous work [27] for the inter-ontology mapping schemes. Sometimes, users may want to locate resources in other

semantic domains. In this case, they would first locate the related domain using the long distance-neighbors.

6 Experiments

We have performed extensive simulation experiments to evaluate the performance of our overlay network structure.

6.1 Setup

The test data is artificially generated. The T-Box ontologies are generated first, and then individuals are created by instantiating classes. We assume for simulation purposes that ontologies and queries are associated with a specific domain, and all ontologies in the same domain have ontology mappings defined in advance. The simulation is initialized by injecting nodes one by one into the network until a certain network size has been reached. After the initial topology is created, a mixture of joins, leaves, and queries are injected into the network based on certain ratios. The proportion of join to leave operations is kept the same to maintain the network at approximately the same size. Inserted nodes start functioning without any prior knowledge.

For comparisons, we simulate our SOON overlay in conjunction with the learning-based ShortCut overlay [11] and a random-walk based simple Gnutella overlay [25]. The ShortCut overlay, as will be described in the related work, is chosen as one comparison reference since it is simple yet effective, and many popular applications (e.g., [11], [12], [13], [14]) use this overlay as their basic routing overlay. Moreover, it is comparable to our network in the sense that it creates clusters on top of the unstructured network. Flooding-based Gnutella was chosen as another reference network for its simplicity and prevalence, which, in fact, made it a widely used baseline for many previous research efforts. The simulation parameters and their default values are listed in Table 1.

Table 1. Parameters used in the simulations.

Parameter	Range and default value
network size	$2^9 \sim 2^{15}$ default: 10,000
initial neighbors (node degree)	5
average node degree	14
TTL	1~20 default 9
resource-discovery query walkers	3 (propagate exponentially)
neighbor-discovery query walkers	2 (propagate linearly)
ontology domains	1~10 default: 8
ontology schemas per domain	1~10 default:8
resources per node	1~10
die/leave probability per time slice per node	0-21%, 3% default
query probability per time slice per node	5%
sample of nodes to compute diameter	5%

6.2   Results and discussion

Emergence of the small-world
As discussed, the topology of the peer network is a crucial factor determining the efficiency of the search system. We expect that the SOON semantic neighbor discovery scheme will transform the topology into a small-world network. To verify this transformation, we examine two network statistics, the *clustering coefficient* (*CC*) and the *average network path length (APL)*, as indicators of how closely the topology has approached a "small-world" topology. The *CC* is a measure of how well connected a node's neighbors are with each other. The *CC* of a node is the ratio of the number of existing edges and the maximum number of possible edges connecting its neighbors. The *APL* is defined as the average shortest path across all pairs of nodes. The *APL* corresponds to the degree of separation between peers. In our experiment, we use a random sample of certain percent of the graph nodes to compute *APL*.
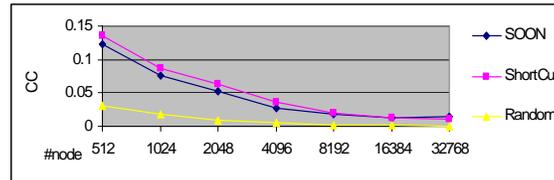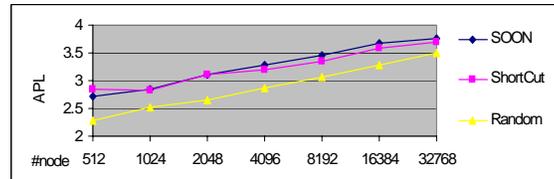

Fig. 3. Comparison of clustering coefficient.

Fig. 4. Comparison of average path length.

We performed experiments to measure SOON's *CC* and *APL*. An interest-based ShortCut topology and a random power-law topology with the same average node degree are used as reference topologies. The former has been proved to be a small-world network [15]. Fig.3 and Fig.4 show plots of the *CC* and the *APL* as a function of the number of nodes in the network. We observe that both the *CC* and the *APL* of SOON are very similar to those of ShortCut. The *CC* of SOON and ShortCut are much larger than that of the random power-law network, while the *APL* of SOON and ShortCut are almost the same as that of the random network. This indicates the emergence of a small-world network topology [16].

Scalability and efficiency
We examine the system performance in three different aspects, namely routing scalability, efficiency, and accuracy. The performance is measured using the metric of recall rate, which is defined as the number of results returned divided by the number of results actually available in the network. First, we vary the number of nodes from

2^9 to 2^{15} to test the scalability. The results are listed in Fig.5. As we expected, SOON gets higher recall in all these different sized networks. In addition, SOON's recall decreases less with the increase in network size. Fig.6 illustrates the system efficiency by showing the relationship between query recall rate and query TTL. With a small TTL, SOON gets a higher recall rate than the other two network. This means that SOON resolves queries faster than the others. In Fig.7 we show the effect of dispatching a different number of walkers to search the network. We can see that with the same TTL, SOON locates more results with fewer walkers.

SOON's small-world topology effectively reduces the search space, and its ontology summary guides the query in the right direction. This explains why SOON scales to large network size and why it achieves higher recall with shorter TTL and fewer walkers. Besides all these reasons, another factor contributing SOON's overall better recall rate is that SOON is able to locate semantically related results that cannot be located by the ShortCut and random-walk. Because of the semantic heterogeneity of our experimental setup, relevant resources may be represented with different ontologies. SOON may use its ontology signature set to find semantically related nodes and use the mapping defined to translate the query. Therefore, it can locate most of the relevant results. However, for ShortCut and random-walk, they have no way to find semantically related resources, but only resources represented in the same ontology as the ontology of the querying node.
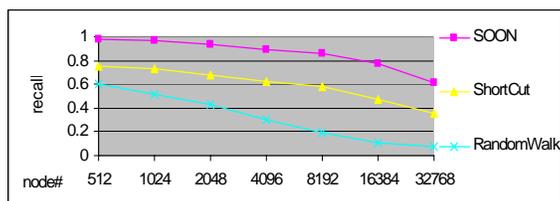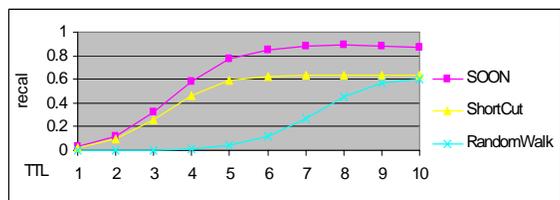

Fig. 5. Comparison of average path length

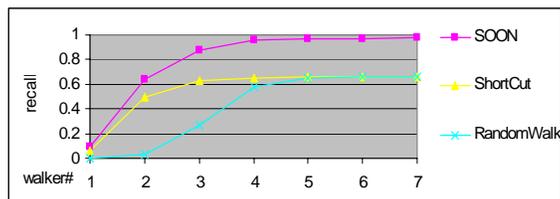
Fig. 6. Comparison of average path length

Fig. 7. Comparison of average path length

Overhead and adaptability to dynamics

The good recall performance of SOON does not come for free. Generally speaking, there is a tradeoff between query efficiency and maintenance overhead. Unlike ShortCut and random-walk approaches, which only create query propagating overhead, SOON also creates overhead for maintaining neighborhood relationship. We expect the extra overhead is reasonable and the saving from query cost exceeds the extra maintenance cost. To verify this, we examine the system's overhead in terms of accumulated bandwidth. System overhead has a close relation with the system dynamics, as a system must maintain consistent information about peers in the system in order to operate most effectively. Therefore, we measure the system dynamics together with the overhead. To evaluate the adaptability to different levels of dynamics, we measure the system overhead under different levels of peer "churn rate", referring to the rate of peers leaving/joining the system.

The experiment shown in Fig.8 gives an overview of how dynamics affect the system performance. We find that SOON performs similarly to the ShortCut algorithm which is proved to be resilient to churn [11]. When peers join or leave frequently, the performance of ShortCut and SOON deteriorate gracefully. Churn does not affect the two schemes dramatically because both algorithms do not depend on a strict structure to perform routing as DHTs do. Their unstructured random topologies provide multiple routes to a destination thus increasing the system resilience. In the worst case, they degrade to random-walk.
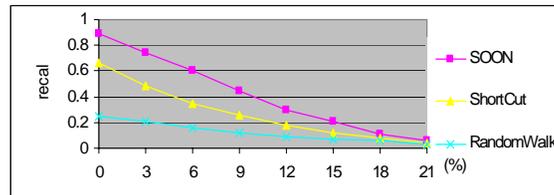

Fig. 8. Recall vs. churn rate

Fig.9 shows the accumulated bandwidth overhead of finding 10000 results under different churn rates. From the figure, we can see that in most situations SOON produces much less overhead than the other two methods. But when the system is very dynamic, such as when the churn rate is beyond 20%, SOON produces much more overhead. The high overhead problem of SOON in very dynamic environments can be solved by a simple solution: when the network is very dynamic, the system can give up the ontology-based topology construction but resort to basic Gnutella random-walk as the solution.
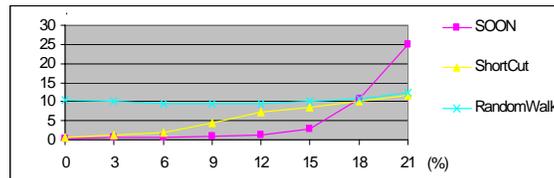

Fig. 9. System overhead (accumulated bandwidth) vs. churn rate

6   Related Work

Research has harnessed the power of semantic technologies to aid in information representation, retrieval and aggregation over large distributed systems. P2P technology has been used to improve the scalability and efficiency of the semantic searching. For example, systems such as Edutella [17] and InfoQuilt [19] use broadcast or flooding to search their semantic metadata, while many other projects, such as RDFPeer [20] and OntoGrid [21] attempt applying DHT techniques to the retrieval of the ontology encoded knowledge. pSearch [22] applies a dimension reduction technique, called rolling index, on top of CAN to realize a semantics-based search.

Recently, there has appeared the idea of grouping nodes with similar contents together to facilitate search. The latest super-peer-based Edutella [18] and Semantic Overlay Network (SON) [23] rely on centralized server or super-peers to cluster contents and nodes. Semantic Small Word (SSW) position peers and data objects in the semantic space, so that peers with similar data objects form into clusters. It then applies a dimension reduction technique on top of the DHT to realize a semantics-based search. In SSW, semantics of data objects is represented by a multi-attribute vector, but not Semantic Web-based data. Applications such as REMINDIN [11], Helios[13], and Bibster [24] add semantic short-cuts to group nodes. The short-cut approach relies on the presence of interest-based locality. Each peer builds a shortcut list of nodes that answered previous queries. To find content, a peer first queries the nodes on its shortcut list and only if unsuccessful, floods the query.

7   Conclusion

In this paper, we propose a self-organized semantic overlay network, SOON. It uses an ontology-based representation of the information metadata. It enables peers to automatically organize themselves according to their semantic properties to form a semantic small-world topology, so that information retrieval can be effectively performed within semantically related small-worlds. Our simulation results prove that SOON improves interoperability among network participants and aids efficient information discovery and access.

References

1. A. L. Barabási, 2003. Linked: How Everything is Connected to Everything Else and What It Means for Business, Science, and Everyday Life. New York: Plume. efficient information discovery and access.
2. S. Milgram, "The small world problem," Psychology Today, vol. 67, no. 1, 1967.
3. J. Kleinberg, "Navigation in a small world," Nature, no. 406, p. 845, 2000.
4. W. A. Gruver, and J. C. Boudreaux (1993) Intelligent Manufacturing: programming environments for CIM, Springer-Verlag, London.

5. G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. J. Miller, Introduction to WordNet: an on-line lexical database." In: International Journal of Lexicography, 1990.

6. J. Jiang and D. Conrath, "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy," Proc. Int'l Conf. Computational Linguistics (ROCLING X), 1997.

7. J. Lee, M. Kim, and Y. Lee, "Information Retrieval Based on Conceptual Distance in IS-A Hierarchies," J. Documentation, vol. 49, pp. 188-207, 1993.

8. M. A. Rodriguez, M. J. Egenhofer, "Determining Semantic Similarity Among Entity Classes from Different Ontologies". IEEE Transactions on Knowledge and Data Engineering, VOL. 15, NO. 2, MARCH/APRIL 2003.

9. A. Tversky. Features of similarity. Psychological Review, 84(4):327–352, 1977.

10. D. Lin. An information-theoretic definition of similarity. In Proc. 15th International Conf. on Machine Learning, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998.

11. X. Tempich, S. Staab, A. Wranik, "REMINDIN': semantic query routing in peer-to-peer networks based on social metaphors" International World Wide Web Conference (WWW), New York, USA, 2004.

12. K. Sripanidkulchai, B. Maggs, and H. Zhang. Efficient content location using interest-based locality in peer-to-peer systems, In INFOCOM'03.

13. S. Castano, A. Ferrara, S. Montanelli, and D. Zucchelli. Helios: a general framework for ontology-based knowledge sharing and evolution in P2P systems. In IEEE Proc. of DEXA WEBS 2003 Workshop, Prague, Czech Republic, September 2003.

14. A. Castano, S. Ferrara, S. Montanelli, E. Pagani, G. Rossi: Ontology addressable contents in p2p networks. In: Proceedings of the WWW'03 Workshop on Semantics in Peer-to-Peer and Grid Computing. (2003).

15. A. Iamnitchi, M. Ripeanu, and I. Foster. Small-world filesharing communities. In Infocom, Hong Kong, China, 2004.

16. D. Watts and S. Strogatz. Collective dynamics of "small-world" networks. - Nature, 1998

17. W. Nejdl, B. Wolf, C. Qu, S. Decker, M.SIntek, A. Naeve, M. Nilsson, M. Palmer and T. Risch (2002). "Edutella: A P2P Networking Infrastructure Based on RDF." WWW2002, May 7-11, Honolulu, Hawaii, USA.

18. W. Nejdl, W. Siberski an M. Sintek (2003), " Design Issues and Challenges for RDF an schema-based peer-to-peer systems." ACM SIGMOD Record 32(3):41-46.

19. M. Arumugam, A. Sheth, and I. B. Arpinar. "Towards peer-to-peer semantic web: A distribuited environment for sharing semantic knowledge on the web." In Proc. of the International World Wide Web Conference 2002 (WWW2002), Honolulu, Hawaii, USA, 2002.

20. M. Cai, M. Frank, "RDFPeers: A scalable distributed RDF repository based on a structured peer-to-peer network", in proc of WWW conference, May 2004, NewYork, USA.

21. OntoGrid project: http://www.ontogrid.net/

22. C. Tang, Z. Xu, S. Dwarkadas, Peer-to-peer information retrieval using self-organizing semantic overlay networks, in: Proceedings of 2003 Conference on Applications, Technologies, Architectures and Protocols for Computer Communications, 2003.

23. A. Crespo and H. Garcia-Molina. Semantic overlay networks. Technical report, Stanford University, 2002.

24. Castano, A., Ferrara, S., Montanelli, S., Pagani, E., Rossi, G.: Ontology addressable contents in p2p networks. In: Proceedings of the WWW'03 Workshop on Semantics in Peer-to-Peer and Grid Computing. (2003)

25. Gnutella website. http://gnutella.wego.com/

26. College of American Pathologists. SNOMED RT - Systematized Nomenclature of Medicine Reference Terminology, VERSION 1.1, USER GUIDE, 2001.

27. Juan Li, Son Vuong, "An Ontological Framework for Large-Scale Grid Resource Discovery", in Proceedings of the IEEE Symposium on Computers and Communications (ISCC'07), Aveiro, Portugal, July 2007.