Semi-structured Semantic Overlay for Information Retrieval in Self-organizing Networks

Yulian Yang
Supervised by: Sylvie Calabretto and Lionel Brunie
Université de Lyon
LIRIS UMR 5205 - INSA Lyon
7, Avenue Jean Capelle
69621, Velleurbanne Cedex, France
yulian.yang@insa-lyon.fr

ABSTRACT

As scalability and flexibility have become the critical concerns in information management systems, self-organizing networks attract attentions from both research and industrial communities. This work proposes a semi-structured semantic overlay for information retrieval in large-scale self-organizing networks. With the autonomy to their own resources, the nodes are organized into a semantic overlay hosting topically discriminative communities. For information retrieval within a community, unstructured routing approach is employed for the sake of flexibility; While for joining new nodes and routing queries to a distant community, a structured mechanism is designed to save the traffic and time cost. Different from the semantic overlay in the literature, our proposal has three contributions: 1. we design topic-based indexing to form and maintain the semantic overlay, to guarantee both scalability and efficiency; 2. We introduce unstructured routing approach within the community, to allow flexible node joining and leaving; 3. We take advantage of the interaction among nodes to capture the overlay changes and make corresponding adaption in topic-based indexing.

Categories and Subject Descriptors

H.3.3 [**Information storage and retrieval**]: Information search and retrieval—*Selection process, Clustering*

Keywords

Information retrieval, Self-organizing networks, Topic-based indexing, Query routing, Semantic overlay

1. BACKGROUND

Self-organizing networks are distributed networks without central control. Resources are distributed in the nodes, and tasks are performed through the cooperation among the nodes (i.e. peer to peer network). It has a promising potential of scalability and flexibility, and attracts increasing research interests as an alternative to centralized information systems. One of the typical applications is Information Retrieval(IR) in self-organizing networks. In this task, each node manages a document repository, and works as both a resource provider and a resource searcher in the network. Since the documents aren't necessarily indexed by a central server, the single point of failure is avoided. Given a query, the relevant documents are often stored in remote nodes. Hence, IR in self-organizing networks mainly involves node selection and query routing.

In centralized IR systems, efficient IR mainly implies returning an effective result list from a pre-crawled document repository in a short time. In self-organizing networks, to perform efficient IR has more challenges. Appropriate nodes need to be located not only in a short time but also at a low traffic cost; the network changes dynamically. For example, the resources are constantly changing and the nodes are joining and leaving the network frequently. To meet these challenges in one system is not a trivial work. Traditionally, query routing is performed either randomly in unstructured network [8] or by inquiring indices stored in distributed hashing table(DHT) in structured network [7, 23]. The problems lie in: the former facilitates dynamic change in the network, but has the limitation in efficient query routing; the latter performs efficient query routing but spends a lot for dynamic network change(i.e. frequently index updating and network churn).

Semantic overlay networks [9] and small world model [13] are then introduced. They share a common idea: a logical overlay is built by connecting nodes using tight or loose links. Nodes with similar resources are connected by tight links. The node communities collected by tight links are connected by loose links. Construction of tight links is the major concern of research, while loose links are often built by randomly connecting two nodes in different communities. This makes it difficult for a new node to efficiently find its community in large-scale networks. In addition, the similarity is often measured by exact term matching [27, 6]. The semantics are missed in this way. In [18], a structured small world is proposed to perform efficient semantic search. But the scale of its structure would increase as the network size increases.

In this work, the self-organizing network is envisioned as a large-scale dynamic network, where IR problem can not be handled efficiently by only structured or unstructured approaches . we propose a semi-structured semantic overlay to integrate the advantages from both sides. We index the nodes in the network by the topics of their resources. A structured super-node layer is designed to maintain the topic-based indices. New node joins in the network by in-

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

quiring the indices and gets access to the nodes with topically similar resources. Once it finds similar nodes, flexible communication among them is allowed. In return, the flexible communication helps to capture the changes of the overlay structure. An adaptation mechanism is designed to make the indices adaptable to the changes. Specifically, our proposal and contributions are as follows:

- We extract topics from the network resources by measuring their semantic similarity in the subspace of a full semantic space. Refined semantic relationship is expected to be discovered in this way. This settles an effective basis for IR in self-organizing networks.
- We index the extracted topics, and store it on a structured super-node layer. By inquiring the topic indices, nodes with similar resources are connected via an efficient way and semantic overlay is built accordingly. For node joining and inter-community query routing, efficiency can also be guaranteed by the topic-based indexing.
- Each node is allowed to communicate freely with the neighborhood on the semantic overlay. This facilitate flexible searching within a community. In addition, changes(e.g. community shrinking and topic changing) in the network are expected to be captured by the flexible neighborhood communication. It motivates an adaption mechanism of the topic-based indexing.

The rest of this paper is structured as follows: Section 2 presents the state of the art; Our proposal is demonstrated in Section 3. Key issues and their solutions are described in Section 4 and 5.

2. STATE OF THE ART

To perform information retrieval in self-organizing networks, an efficient approach is to index the resources and store them in distributed nodes. A DHT-based infrastructure is implemented in this approach which provides efficient querying routing. But it costs too much for maintenance, especially when the network is a large-scale and dynamic one as envisioned in our work [15]. The approach based on unstructured network, on the other side, spends less to maintain the network infrastructure, but costs more to search relevant information stored in remote nodes [29]. Plenty of works have been done to improve the searching performance in unstructured networks, for example neighbor indexing [8], small world model [13] and semantic overlay network [9]. However, these works can only perform efficient IR when the relevant information to a query is stored in the neighborhood of the initiating node. In addition, it is costly to join a new node in the network, because random message dissemination is employed to find neighbors for the node. IR in hybrid system is proposed to combine the advantages of IR in both structured and unstructured networks, which is also the motivation of our work.

2.1 DHT based IR in self-organized network

DHT based IR in self-organizing networks stores resources as {key, data} pairs in DHT and provides inquiring service given a key [17]. In ALVIS [19],terms are indexed for each document. It achieves efficient search performance, but costs too much to update the indices. Moreover, the indices are not scalable to the amount of documents. In Minerva [2], terms are indexed for each node instead of document; In [23] and TSS [4], terms are replaced by highly discriminative keys(HDK) which improves the scalability. In PCIR [21] a cluster-based approach is proposed for publishing indices at low traffic cost. Besides, semantic searching is implemented in pSearch [7], where CAN, a distributed indexing strategy for multi-dimensional data, is employed to index document vector generated by latent semantic analysis(LSA). An extension of pSearch is proposed in [5].

2.2 Query routing in unstructured network

In unstructured network, nodes join and leave without causing too much traffic cost. IR doesn't rely on well-structured indices, and queries are forwarded mainly in two ways: blind informed ways. In blind query forwarding, no hint is utilized to guide the query's next hop. Typical works include random walk [11] and iterative deepening [28]. In informed approaches, query forwarding is supervised by hints, for instance, routing indices [8], small world links [13], social communities [3] and semantic overlay networks [9]. Small world is a phenomena in social network, where any two peers can be connected by mutual acquaintances [16]. They have two properties: high cluster coefficient and low average hop between any two randomly chosen peers. The idea is introduced in unstructured network to construct both short and long links for nodes [14, 24]. Short links are used to cluster nodes with similar resources, and long links to connect node clusters. The long links are often built based on those nodes whose resources belong to multiple clusters [12, 9, 1].

2.3 IR in hybrid network

In [10], a super-peer based similarity search strategy is proposed for high-dimensional data. The data is indexed by only few super-peers. The sub-peers with similar resources are connected to the same super-peer. Since the querying task is loaded on only few super-peers, this work might suffer from certain scalable problems when the network is in a very large scale. Other IR approaches involve small world model in CAN-based IR approach [20, 26], small world model in super-peer based network [6], peer clustering based on CAN infrastructure [18], and semantic overlay networks in structured network [27].

3. PROPOSAL

We show our semi-structured semantic overlay in Figure 1. A topic-based indexing approach is designed on the structured super-node layer. The indices are designed as <topic,node> pairs. Only the relatively stable nodes for each topic are indexed in order to save maintenance cost. New nodes can join in the overlay by inquiring the indices with its topic and getting access to the nodes having similar resources. Nodes with similar resources are allowed to connect and communicate to each other randomly. The semantic overlay is formed accordingly. By the random communication within the neighborhood on the semantic overlay, community shift(community enlarging and shrinking, and community topic shift) could be captured. It could then be submitted to the structured super-node layer, as a feedback to update the topic indices. We call it dynamic adaption in the rest of the paper. For searching information within the neighborhood/a community, query routing is performed using the random connections among nodes. For searching

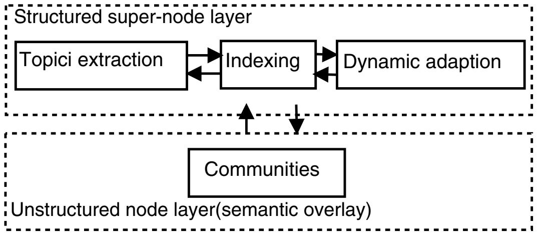
Structured super-node layer
Topici extraction
Indexing
Dynamic adaption
Communities
Unstructured node layer(semantic overlay)

Figure 1: Semi-structured semantic overlay

information in distant community, we inquiry the topic indices stored on the super-node layer, and then forward the query to corresponding community.

We will evaluate the performance of our proposal by considering its search efficiency, its cost in node joining and the scalability of its indices. The classic recall-precision metrics will be employed to measure the searching efficiency; Both the time and traffic cost will be evaluated for the node joining; The scalability of the indices will be measured by observing if the increase of the network scale would also cause the increase of the index scale. We will compare our proposal with both the structured and unstructured approaches in the literature.

4. TOPIC-BASED INDEXING

The basic idea is to extract topics from the network resources as the indexing granularity. It is motivated by pivot-based metric space indexing [22], where a small number of objects from the metric space are selected as pivots, and the index is built by storing the distances from each of them to the objects of the database. Searching space is sharply reduced by only comparing the distance between the query and the pivots. Since the topics as well as the super nodes are relatively stable in the network, this infrastructure is supposed to consume less maintenance cost and provides an efficient performance for node joining and inter-community query routing. In addition, only the representative nodes for each topic are indexed on the structured super-node layer. In other words, most of the nodes are not tightly controlled by the structured super-node overlay once they join the network. They can build their own connections to the other nodes in the neighborhood.

4.1 Topic extraction

Problem statement Let $\{N_1, N_2, N_3 ... N_n\}$ be n nodes in the self-organized network. The topic of each node is represented by a vector $V_i = \{v_{i,1}, v_{i,2}, v_{i,3} ... v_{i,D}\}$ with dimension D, which could be a semantic vector achieved by data mining techniques, the topic modeling approach Latent Dirichlet allocation for instance. Each element in the semantic vector refers to the contribution of corresponding dimension to the node's topic. The higher an element is, the more contribution the corresponding dimension makes. The topic vectors of all the nodes can be located as one point in this D-dimensional space, which we call semantic space. Their similarity is measured by the distance of their locations. Our aim is to extract the topics shared by the topic vectors and design appropriate data structure to maintain them. They are supposed to work as the pivots in [22].

Principally, a good set of pivots should be selected according to the data distribution(internal complexity) in metric space. Since similar topic vectors tend to aggregate in the semantic space, every bunch of aggregated vectors corresponds to certain topic. To our best expectation, if the pivots are scattered on the central of aggregated vector bunches, new vector can easily find its brunch by comparing its distances to the pivots. The challenge in our case is how to extract the shared topics in an incremental way, because the topic vectors arrives incrementally along with new nodes joining the network.

Solution According to the formation of topic vectors, two vectors are similar to each other if only they are similar in a subspace of the full dimension which has high contribution to the topic. Therefore, we propose a subspace clustering approach to extract topics and represent them in subspaces. We collect a set of vector samples from the semantic space. By mapping the vectors into their D dimensions, we get D 1-dimension value sets. L principle values $\{p_{d,1}, p_{d,2}, p_{d,3} ... p_{d,l}\}$ are selected from the value set in each dimension according to their intrinsic complexity. The selection of principle values in each dimension is performed as presented in [22]. The d refers to the dth dimension, and l refers to the lth principle value in this dimension.

After selecting principle values in each dimension, we remap the vector samples to the closest principle values in each dimension. Each vector is represented by its mapping route $\{p_{1,l}, p_{2,l}, p_{3,l} ... p_{D,l}\}$. We extract the sub-routes that pass h highest principle values in any dimension, and set it as one of the bootstrap pivots if it hosts a lot of vectors(this can be achieved via ordering decreasingly the number of vectors in each extracted route and selecting the first N routes $\{P_1, P_2, P_3 ... P_N\}$). We reserve the dimensions as well as the values for each pivot which are taken account for pivot selection as $Pd_n = \{(d, p_{d,l})\}$. In this way, the vector collection in the semantic space is divided into $N + 1$ brunches $\{C_1, C_2, C_3 ... C_{N+1}\}$ by the following measurement:

$$V_i \in \begin{cases} C_n & if \ \ \forall d \in Pd_n, |v_{i,d} - p_{d,l}| \le M\alpha_d \\ C_{N+1} & if \ \ V_i \notin C_n \end{cases} \qquad (1)$$

Where $v_{i,d}$ is the dth element in V_i, $p_{d,l}$ is the dth element in pivot P_n, and $M\alpha_d$ is the distance from the principle values $p_{d,l}$ to its adjacent principle value in the same dimension.

This operation is repeatedly performed when the number of vectors in one brunch is above a prefixed threshold. Finally, a tree structure is generated as showed in Figure 2. The vectors are inserted in the tree by being associated with appropriate leaves via top-to-down lookup. The vectors in each leaf have similar topic and correspond to a bunch objects in the semantic space. Moreover, leaves whose vectors are in the adjacent neighborhood can also be connected by inquiring the tree. We assign unique identification for each node in the tree structure. Accordingly, we can generate a key for each topic vector by concatenating the identification of the nodes it passes. For example in Figure 2, topic vectors V_9 and V_{17} have the key OCC. The code for each node can also be calculated by other approaches.

4.2 Distributed indexing

Problem statement Since there is no central control in the network, indices have to be maintained in the distributed nodes and inquired distributively. In the literature, the only way to perform this task is the DHT based infrastructure.

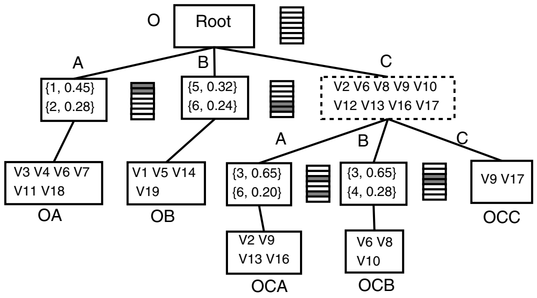

Figure 2: Tree hierarchy of topic vectors

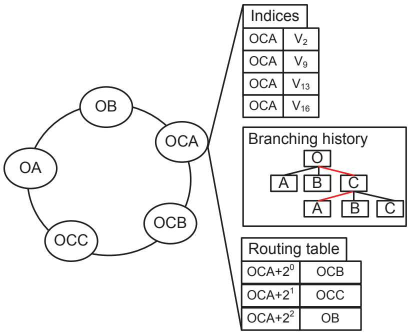

Figure 3: Chord based super-node layer

Specifically, it represents data objects as {key, object} pairs and keeps them in distributed nodes. Keys are distinguished for different data objects. In classic DHT based IR, the key is calculated by a predefined hashing function like SHA-1 [19] or a pre-trained model like LSA [5]. In this work, the keys for topic vectors can be calculated by checking the tree structure. It means each node in the network must keep the information about the tree in order to calculate the keys. However, it's not flexible to save the entire tree in every node.

Solution Suppose there is only one super-node identified as O in the network at the very beginning. It corresponds to the root in the tree-based index structure. As new nodes join in, their topic vectors are aggregated in this super-node. If the super-node becomes overloaded, for example if the number of vectors are above a threshold, a set of pivots are selected. The vectors are divided into several brunches in this way. This is corresponding to branching a leaf in tree-based index structure. We join several new super-nodes in the network, and make each super-node to manage one branch as well as its branching history. Super-nodes refer to those nodes with longer online time and better communication bandwidth. They could be servers or ordinary nodes. The ID of each super-node is assigned as the same as key of the data it stores. The super-nodes are then organized into a Chord ring in am increasing order by their IDs, as showed in Figure 3. Each super-node stores indices of the topic vectors whose keys are the same to its ID. It also stores the branching history to calculate the keys for new joining nodes. It's also possible for one super-node to manage the indices in more than one leaf if it has more power. In that case, the ID of the super-node is assigned as the shared substring of the keys in its indices. For maintaining the network and forwarding the message, each super-node keeps connections with its two adjacent neighbors in the ring. It also keeps a routing table which stores the information about the nodes 2^m hops away, where $m \leq 0$ and $2^m \leq n$.

When a new node joins in the network by randomly connecting node N_r, its topic vector is sent to the super-node of N_r. The key for the vector is calculated in this super-node and then forwarded to the super-node which storing indices of this key. Since each super-node only keeps part of the tree structure (branching history of the leaves in it), it is unable to calculated a complete key for a topic vector out of its managing scope. In this case, we use the local tree structure to estimate a partial key for the new node, and forward it to a super-node whose ID is partial matched to the key. In the new super-node, the key is expected to be completed and forwarded to the target super-node. For inter-community searching, the same operation is employed.

5. DYNAMIC ADAPTION

As described in Section 3, we only store the information of those relatively stable nodes in indices. The nodes within a neighborhood are connected by random links. In this way, more flexible node joining and leaving is allowed. While as described in [25], dynamic node joining and leaving could effect the structure of the semantic overlay. We utilize the random links among nodes to detect these changes and submit them to the super-node layer as a feedback. On our super-node layer, the module of dynamic adaption is activated to make corresponding modification to the indices.

The structure changes we deal with include community enlarging, community shrinking and community topic shift. We formalize these changes as follows:

Definition 1. Suppose in a fixed region of the semantic space R, there are M nodes forming a community. If more and more new nodes join in this region, we call it community enlarging; If a large amount of nodes in this region are leaving the network, we call it community shrinking; If no node leave, only their topics change and their locations move outside the region, we call it community topic shift.

The community enlarging can be detected by the super node that stores the community's index. For the community shrinking, it can be detected simply by routing through the random connection between nodes on the semantic overlay. A specifically defined message is activated periodically in a indexed node of the community. This message is flooded through the random links, and the number of the nodes is collected finally. If the number decreases largely comparing to the previous result, an adaption action is activated accordingly.

To perform the adaption, each node in the community inquiries the indices to find other community in the neighborhood. The connections are made between this node and those from the closest community, while this node dose not need to change its identification. By doing this, the shrinking community could not be isolated in the semantic overlay. If the node detects that its topic is closer to the nodes from the neighborhood community, the index for that community is modified to include this node as its member. In other words, the region of that community need to be enlarged, which corresponds to modifying one of the tree branches of the topic structure. Once the branch is modified, all the super-nodes that keep this part of the tree need to be notified. This operation is performed by the infrastructure of the structured super-node layer.

6. CONCLUSION

In this proposal, we designed a topic-based indexing approach to facilitate the formation of semantic overlay in self-organizing networks. It would achieve scalable and efficient node joining and inter-community query routing. In addition, flexible communication is allowed among the nodes within the community, for the sake of network dynamics. he prototype of our proposal is being developed.

7. ACKNOWLEDGMENTS

This work is funded by the co-operation program between China and UT-INSA of France.

8. REFERENCES

[1] M. Bawa, G. S. Manku, and P. Raghavan. Sets: search enhanced by topic segmentation. In *SIGIR*, pages 306–313, 2003.

[2] M. Bender, S. Michel, J. X. Parreira, and T. Crecelius. P2p web search: Make it light, make it fly (demo). In *CIDR*, pages 164–168, 2007.

[3] M. Bertier, D. Frey, R. Guerraoui, A.-M. Kermarrec, and V. Leroy. The gossple anonymous social network. In *Middleware 2010*, volume 6452 of *Lecture Notes in Computer Science*, pages 191–211. 2010.

[4] H. Chen, J. Yan, H. Jin, Y. Liu, and L. Ni. Tss: Efficient term set search in large peer-to-peer textual collections. *Computers, IEEE Transactions on*, 59(7):969 –980, 2010.

[5] Y. Chen, Z. Xu, and C. Zhai. A scalable semantic indexing framework for peer to peer information retrieval. In *SIGIR-HDIR '05)*, volume 51, page 20, 2005.

[6] L. T. M. Choong Yong Liang. Small world bee: Reduce messages flooding and improve recall rate for unstructured p2p system. *IJCSSNS*, 11(5):88–99, 2011.

[7] T. Chunqiang, X. Zhichen, and D. Sandhya. Peer-to-peer information retrieval using self-organizing semantic overlay networks. SIGCOMM '03, pages 175–186, 2003.

[8] A. Crespo and H. Garcia-Molina. Routing indices for peer-to-peer systems. In *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on*, pages 23 – 32, 2002.

[9] A. Crespo and H. Garcia-Molina. Semantic overlay networks for p2p systems. In *AP2PC*, pages 1–13, 2004.

[10] C. Doulkeridis, A. Vlachou, K. Nørvåg, Y. Kotidis, and M. Vazirgiannis. Efficient search based on content similarity over self-organizing p2p networks. *Peer-to-Peer Networking and Applications*, 3(1), 2010.

[11] C. Gkantsidis, M. Mihail, and A. Saberi. Random walks in peer-to-peer networks. In *INFOCOM*, volume 1, 2004.

[12] J.-C. Huang, X.-Q. Li, and J. Wu. A semantic searching scheme in heterogeneous unstructured p2p networks. *Journal of Computer Science and Technology*, 26(6):925–941, 2011.

[13] H. Jin, X. Ning, and H. Chen. Efficient search for peer-to-peer information retrieval using semantic small world. WWW'06, pages 1003–1004, 2006.

[14] W. Ke and J. Mostafa. Strong ties vs. weak ties: Studying the clustering paradox for decentralized search. In *LSDS-IR'08*, pages 49 – 56, 2008.

[15] W. Ke and J. Mostafa. Scalability of findability: effective and efficient ir operations in large information networks. In *SIGIR*, pages 74–81, 2010.

[16] J. Kleinberg. Small-world phenomena and the dynamics of information. In *NIPS*, page 2001. MIT Press, 2001.

[17] J. Li, J. Stribling, T. M. Gil, R. Morris, and M. F. Kaashoek. Comparing the performance of distributed hash tables under churn. In *IPTPS04*, 2004.

[18] M. Li, W.-C. Lee, A. Sivasubramaniam, and J. Zhao. Ssw: A small-world-based overlay for peer-to-peer search. *IEEE Trans. Parallel Distrib. Syst.*, 19(6):735–749, 2008.

[19] T. Luu, G. Skobeltsyn, F. Klemm, M. Puh, I. P. Žarko, M. Rajman, and K. Aberer. Alvisp2p: scalable peer-to-peer text retrieval in a structured p2p network. *VLDB*, 1:1424–1427, 2008.

[20] G. S. Manku, M. Bawa, and P. Raghavan. Symphony: Distributed hashing in a small world. In *USENIX Symposium on Internet Technologies and Systems*, 2003.

[21] O. Papapetrou, W. Siberski, and W. Nejdl. Pcir: Combining dhts and peer clusters for efficient full-text p2p indexing. *Computer Networks*, pages 2019–2040, 2010.

[22] O. Pedreira and N. R. Brisaboa. Spatial selection of sparse pivots for similarity search in metric spaces. SOFSEM '07, pages 434–445, 2007.

[23] I. Podnar, M. Rajman, T. Luu, F. Klemm, and K. Aberer. Scalable peer-to-peer web retrieval with highly discriminative keys. In *ICDE*, 2007.

[24] P. Raftopoulou and E. G. M. Petrakis. icluster: A self-organizing overlay network for p2p information retrieval. In *ECIR*, pages 65–76, 2008.

[25] P. Raftopoulou and E. G. M. Petrakis. Peer rewiring in semantic overlay networks under churn. OTM '10, pages 573–581, 2010.

[26] X. Sun. Scan: a small-world structured p2p overlay for multi-dimensional queries. In *WWW'07*, pages 1191–1192, 2007.

[27] J. M. Tirado, D. Higuero, F. Isaila, J. Carretero, and A. Iamnitchi. Affinity p2p: A self-organizing content-based locality-aware collaborative peer-to-peer network. *Computer Networks*, 54(12):2056–2070, 2010.

[28] B. Yang and H. Garcia-Molina. Improving search in peer-to-peer networks. In *Distributed Computing Systems*, pages 5 – 14, 2002.

[29] Y. Yang, R. Dunlap, M. Rexroad, and B. F. Cooper. Performance of full text search in structured and unstructured peer-to-peer systems. In *INFOCOM*, pages 1 –12, 2006.