Intelligent Algorithms for Reducing Query Propagation in Thematic P2P Search

Ana Lucía Nicolini, Carlos M. Lorenzetti,
Ana G. Maguitman, and Carlos Iván Chesñevar

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur, Av. Alem 1253, (8000) Bahía Blanca, Argentina
{aln,cml,agm,cic}@cs.uns.edu.ar

Abstract. Information retrieval is a relevant topic in our days, especially in distributed systems where thousands of participants store and share large amounts of information with other users. The analysis, development and testing of adaptive search algorithms is a key avenue to exploit the capacity of P2P systems to lead to the emergence of semantic communities that are the result of the interaction between participants. In particular, intelligent algorithms for neighbor selection should lead to the emergence of efficient communication patterns. This paper presents new algorithms which are specifically aimed at reducing query propagation overload through learning peers' interests. Promising results were obtained through different experiments designed to test the reduction of query propagation when performing thematic search in a distributed environment.

1 Introduction

The current information age has facilitated the generation, publication and access to geographically dispersed resources and heterogeneous content. As searching and sharing information directly from personal computers become more prevalent, new opportunities arise to preserve, foster and exploit the diversity of social communities in Internet. In this scenario we can identify several research challenges for developing mechanisms to manage and access distributed resources in a variety of formats. While research on peer-to-peer (P2P) systems has facilitated the implementation of robust distributed architectures, there are still several limitations faced by current search mechanisms. In particular, these mechanisms are unable to reflect a thematic context in a search request and to effectively take advantage of the peers' interests to improve the network communication patterns.

The main objective of this work is to provide P2P systems with mechanisms for context-based search and to propose algorithms that incrementally learn effective communication patterns in pure P2P networks, where each participant operates in an autonomous manner, without relying on a specific server for communications.

Current search services are rigid as they do not offer mechanisms to facilitate users access to information about potentially relevant topics with which they might not be familiar. Another limitation of the current search model is the lack of context sensitivity. Although some websites offer personalized search they do not offer proper mechanisms to facilitate contextualization and collaboration. These factors are crucial in thematic and distributed search environments.

In a distributed search model participants collaborate by sharing the information stored in their computers. Differently from the client-server model, P2P systems have the capability of increasing their performance as the number of users increases. To take advantage of this potential it is necessary to develop adaptive and collaborative mechanisms to exploit the semantics of users communities, the resources that they store and their search behavior. In order to address these issues, in this paper we present adaptive algorithms that learn to route queries to potentially useful nodes, reducing query propagation.

2 Background: Small World Topology and Semantic Communities

A good network logical topology is one that facilitates an effective performance and enables queries to reach the appropriate destiny in a few steps without overloading the system bandwidth [Tirado et al., 2010]. Moreover, it is desirable that the participants send their queries to other participants that are specialized in the query topic. Some results confirm this observation [Barbosa et al., 2004,Voulgaris et al., 2004]. This makes possible that a query be propagated quickly in the network through relevant nodes, and suggests that collaborative and distributed search can benefit from the context and the participants' community.

In order to evaluate the emergence of semantic communities in a P2P network we employ a methodology similar to the one applied in [Akavipat et al., 2006]. In particular, we adopt the concepts of "small world topology" and "clustering coefficient" [Watts and Strogatz, 1998] to study the structural properties of the emergent communication patterns.

2.1 Clustering Coefficient

The local clustering coefficient assesses the clustering in a single node's immediate network (i.e., the node and its neighbors) [Watts and Strogatz, 1998]. We consider undirected graphs $G = (V, E)$, in which V is the set of nodes and E is the set of edges. For a node v_i its neighborhood N_i is defined as the set of nodes v_j immediately connected to v_i, that is,

$$N_i =_{def} \{v_j \mid e_{ij} \in E, e_{ji} \in E\}$$

The local clustering coefficient is based on egos network density or local density. For each node v_i, this is measured as the fraction of the number of ties connecting v_i's neighbors over the total number of possible ties between v_i's neighbors.

Let k_i be the number of neighbors of a node v_i, that is, $|N_i|$. If a node has k_i neighbors then it could have at most $k_i(k_i - 1)/2$ edges (if the neighborhood is fully connected).

Therefore, the local clustering coefficient for a node v_i can be formalized as follows:

$$C_i = \frac{2|e_{jk} \in E : v_j, v_k \in N_i|}{k_i(k_i - 1)}.$$

In order to calculate the local clustering coefficient for the whole network, the individual fractions are averaged across all nodes [Watts and Strogatz, 1998]. Let n be the number of vertices in the network, that is $|E|$. Formally, the network average clustering coefficient can be defined as:

$$C_{average} = \frac{1}{n} \sum_{i=1}^{n} C_i.$$

A graph is considered *small-world* if its links are globally sparse (the network is far from being fully connected), its $C_{average}$ is higher than the average clustering coefficient associated with a random graph and the length of the path connecting two nodes is orders of magnitude smaller than the network size [Watts and Strogatz, 1998].

This metric represents the global knowledge of the network and was selected in this work in order to compare the ability of the proposed algorithms to understand the information associated with the nodes. When the amount of information about the nodes in the network is insufficient, $C_{average}$ is small. However, as this information grows, the value of $C_{average}$ will grow as well.

3 Algorithms

All the proposed algorithms share a common feature: each node has an internal table NT (Nodes' Topics) that contains the learned knowledge. Each entry maintains a topic and a set of nodes that are interested in this topic. The differences between the algorithms appear at the moment the table is updated and in the way a node is selected to send a query.

We designed eight context sensitive algorithms, adopting an incremental approach. Due to space limitations, in this paper we will only focus on the two algorithms that showed the best behavior. Despite showing small differences in the local behavior of each node, these two algorithms produced significant changes in the overall results. We will also present a brute-force algorithm as a baseline for comparative purposes.

3.1 Basic Algorithm

This algorithm does not have any intelligence, and therefore does not require the use of an NT table for each peer. The queries are routed in a brute-force search manner, as in Gnutella [Ripeanu, 2001]. Each time a node generates a query it sends this message to all of the adjacent nodes. If a node that receives a query message can reply, it sends a reply message, otherwise, it forwards the query to its adjacent nodes until exhausting the initially defined number of query hops.

3.2 Adaptive Algorithm

In this algorithm at the moment of generating a new query message, the query-issuing node looks into its NT table for nodes associated with the topic of the query and sends the query message to all of them. In the case that the query-issuing node does not have an entry for this topic in its NT table, it sends the query message to all of the adjacent nodes, in the same way as the basic algorithm. The learning phase occurs with the reply message. When a node can reply a query it sends a reply message that follows the same path as the issued query. Each intermediate node in this path updates its NT table with the topic of the query that is being answered and the node that answered it. There is another component in this phase: updating messages. When a node learns something, after updating its NT table, it sends an update message with the information learnt –in the format (topic,node)– to all of its adjacent nodes and to all the nodes which are "known" (through its NT table) to be interested in the topic of the reply message. There is another situation in which a node must send update messages: when a query message arrives by broadcast and the node is interested in the topic of the query but cannot reply, it will send an update message to the node that originated the query. This behavior avoids excluding nodes that do not have many resources.

3.3 Selective Adaptive Algorithm

The only difference between this algorithm and the Adaptive Algorithm is that this version skips update messages to adjacent nodes and only sends this kind of messages to those nodes that are interested in the topic of the reply message that arrived by broadcast.

4 Simulations and Results

These algorithms were implemented in Java, the physical network was simulated with the OmNet++ framework [Pongor, 1993] and the logical network was visualized with JUNG (Java Universal Graph). As an input we used more than 40,000 scientific articles that were distributed among the nodes such that each node contained articles related with its interests.

To find the best algorithm for query routing ten simulations were performed with each one, storing the results of the first, third, fifth, seventh and tenth execution. In order to complete our analysis of the simulations we considered:

- The average clustering coefficient of the logical network.
- The number of queries that have been satisfied.

- The number of messages sent by each node taking into account update messages to analyze whether these kind of messages were congesting the network.
- The maximum number of hops needed to find an answer.

All the simulations were executed in a server with these characteristics:

- 32 processors (4 x 8 cores) Opteron.
- 50 GB RAM.
- Debian GNU/Linux 6.0 64 bits.
- kernel 3.8.3.
- Oracle JRE 1.7.0_21.

	Basic Algorithm					Adaptive Algorithm					Selective Adaptive Algorithm				
	1	3	5	7	10	1	3	5	7	10	1	3	5	7	10
Answered queries	$\frac{92}{150}$	$\frac{80}{150}$	$\frac{63}{150}$	$\frac{120}{150}$	$\frac{75}{150}$	$\frac{135}{150}$	$\frac{97}{150}$	$\frac{94}{150}$	$\frac{83}{150}$	$\frac{91}{150}$	$\frac{128}{150}$	$\frac{94}{150}$	$\frac{89}{150}$	$\frac{95}{150}$	$\frac{98}{150}$
Hops	30	29	29	28	30	25	2	2	2	2	29	2	2	2	2
Clustering coefficient	0.013	0.013	0.013	0.013	0.013	0.696	0.706	0.708	0.709	0.713	0.686	0.701	0.705	0.709	0.709
Sent messages (Millions)	0.996	0.998	0.998	0.998	0.995	1.964	1.462	1.367	1.345	1.139	1.898	1.447	1.409	1.391	1.186
Update messages (Millions)	–	–	–	–	–	0.214	0.141	0.129	0.128	0.127	0.201	0.126	0.124	0.119	0.114

Table 1. Performance comparison between algorithms

Table 1 presents the results that are considered more important for the comparison of the different algorithms. From this table we can conclude the following:

- The number of answered queries is higher in the basic algorithm and in the first execution of the other algorithms. This is because in these cases the queries are propagated through the whole network.
- On the other hand the maximum number of hops to find an answer decreases as the overall knowledge of the network increases.
- Related with the previous item, the average clustering coefficient increases in the latest executions. This is because the knowledge of the whole network is higher, so that the nodes can send a query directly to potentially useful nodes.
- Concerning the number of messages sent, we can see that this number decreases as the number of executions increases. This is because the nodes find their queries in fewer hops, so they need to propagate fewer messages.
- The update messages are a part of the sent messages. We can see that the Selective Adaptive Algorithm sends less update messages than the Adaptive Algorithm and the global knowledge is not modified.

It is important to distinguish the physical network from the logical one. We have established a physical network of 1000 randomly connected nodes that remains static through all the executions. Each node in this network is associated with one or more themes of interest. On the other hand, the logical network is the result of the evolution of the network's global knowledge. Through its graphical representation we can see the semantic communities that emerged from this incremental knowledge.

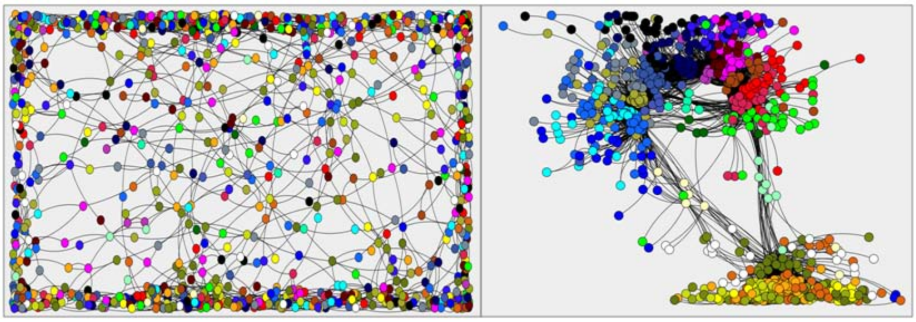

Fig. 1. Logical network obtained from a brute-force search algorithm (left). Logical network obtained from an intelligent algorithm (right).

On the left-hand side of figure 1 we can see a logical network obtained from the brute-force search algorithm. In this case the nodes only know their physical neighbors, disregarding the topics of interest associated with the rest of the pairs. This image shows us that this network does not reflect the existence of semantic communities. The logical network appearing on the right-hand side of figure 1 is the result of executing an intelligent algorithm. The colors are related with the topic in which each node is interested and allows us to appreciate the existence of semantic communities. It may be the case that in the physical network a pair of nodes is very far apart but the same pair is adjacent in the logical network. This is because the logical network reflects the semantic aspects of the nodes.

5 Related Work

A P2P system uses the computational power and the bandwidth of the participants instead of relying on a small number of servers [Balakrishnan et al., 2003]. Mechanisms for distributed content search in these systems offer solutions to some of the scalability and coverage problems commonly recognized in centralized systems. These limitations are particularly evident when attempting to design thematic portals, where small search engines attempt to offer solutions to specialized communities of users [Menczer et al., 2004, Pant et al., 2004].

Many investigations were done on how to structure the network for routing queries. A proposed solution is to create a two layer architecture: the upper is the semantic layer that controls the super peers and the lower is the layer in charge of getting the relevant files [Athena Eftychiou, 2012]. Other approaches that use super peers were proposed in [Ismail et al., 2010], where decision trees are used in order to improve search performance for information retrieval in P2P network.

In the P2P scientific community there is an increasing interest in algorithms that dynamically modify the topology of the logical network, guided by mechanisms that allow the participants to learn about the thematic of the resources offered by other participants as well as their information needs [Wang, 2011, Yeferny and Arour, 2010]. This systems offers a way to relax the restrictions of centralized, planned and sequential control, resulting in decentralized and concurrent systems with collective behaviors [Watts and Strogatz, 1998].

There is a wide variety of search engines based on the P2P technology. For example the model proposed by the YouSearch [Bawa et al., 2003] project takes a centralized Napster-like design for query routing. At the same time each participant can find and index portions of the web. Other systems such as NeuroGrid [Joseph, 2002] attempt to send the query to potential nodes. Most of these systems use automatic learning techniques to adjust the metadata that describes the content of the nodes. Currently there exist some tools for decentralized search such as Faroo[1] and Yacy[2].

6 Conclusions and Future Work

The execution of the proposed algorithms in a simulation environment made it possible to obtain different statistics about their behavior such as response time and network congestion. With this statistical information we can conclude that the algorithms with better behavior are those that offer greater collaboration among peers (that is, when a node learns something, it should spread this knowledge across its community). Learning not only takes place to determine which node answers a query, but also when the node that generated the query is found to be semantically similar to the receptor node. In this case, learning occurs independently of whether the node replies or does not reply the query. These algorithms also showed that, after a number of executions, a logical network with a small-world topology and high average clustering coefficient emerges, reflecting the knowledge of the global network. The processing time that these algorithms require does not produce a significant overhead in the response time with the advantage that the processing time decreases as the available knowledge of the network increases.

Part of our future work will be focused on performing search based on semantic criteria, going beyond the currently existing syntactic search mechanisms. For example, if a query contains the term “house” an article that refers to a dwelling or an apartment could be also of interest for the user posing that query. This

[1] http://www.faroo.com.

[2] http://www.yacy.net.

kind of search by semantic similarity can reduce the precision but enables an increasing recall, reducing the number of ambiguities through context sensitivity.

A problem that arises in this scenario is what we could describe as "The Closed Communities Problem". In this setting, one or more nodes can be disconnected from their community or can form another community with the same topic without being related to each other. To solve this problem we plan to implement a curiosity mechanism that will prompt some participants to explore the network beyond their interest. Some results in this direction were already studied in [Maguitman et al., 2005,Lorenzetti and Maguitman, 2009]. Finally, we plan to run these algorithms in a real distributed environment where the participants could occasionally change their interests and generate queries dynamically. Research in this direction is currently underway.

7 Acknowledgements

We thank Maricel Luna and Mauro Ciarrocchi for their efforts in performing the initial analysis of the proposed algorithms. This research is funded by Projects PIP 112-200801-02798, PIP 112-200901-00863 (CONICET, Argentina), PGI 24/ZN10, PGI 24/N006 and PGI 24/N029 (SGCyT, UNS, Argentina).

References

Akavipat et al., 2006. Akavipat, R., Wu, L.-S., Menczer, F., and Maguitman, A. G. (2006). Emerging semantic communities in peer web search. In *Proceedings of the international workshop on Information retrieval in peer-to-peer networks*, P2PIR '06, pages 1–8, New York, NY, USA. ACM.

Athena Eftychiou, 2012. Athena Eftychiou, Bogdan Vrusias, N. A. (2012). A dynamically semantic platform for efficient information retrieval in P2P networks. *International Journal of Grid and Utility Computing*, Volume 3:271 – 283.

Balakrishnan et al., 2003. Balakrishnan, H., Kaashoek, M. F., Karger, D., Morris, R., and Stoica, I. (2003). Looking up data in P2P systems. *Commun. ACM*, 46:43–48.

Barbosa et al., 2004. Barbosa, M. W., Costa, M. M., Almeida, J. M., and Almeida, V. A. F. (2004). Using locality of reference to improve performance of peer-to-peer applications. *SIGSOFT Softw. Eng. Notes*, 29:216–227.

Bawa et al., 2003. Bawa, M., Bayardo, R. J., Jr., and Rajagopalan, S. (2003). Make it fresh, make it quick - searching a network of personal webservers. In *Proc. 12th International World Wide Web Conference*, pages 577–586.

Ismail et al., 2010. Ismail, A., Quafafou, M., Nachouki, G., and Hajjar, M. (2010). A global knowledge for information retrieval in P2P networks. In *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on*, pages 229–234.

Joseph, 2002. Joseph, S. (2002). Neurogrid: Semantically routing queries in peer-to-peer networks. In *Proc. Intl. Workshop on Peer-to-Peer Computing*, pages 202–214.

Lorenzetti and Maguitman, 2009. Lorenzetti, C. M. and Maguitman, A. G. (2009). A semi-supervised incremental algorithm to automatically formulate topical queries. *Information Sciences*, 179(12):1881–1892. Including Special Issue on Web Search.

Maguitman et al., 2005. Maguitman, A. G., Menczer, F., Roinestad, H., and Vespignani, A. (2005). Algorithmic detection of semantic similarity. In *Proceedings of the 14th international conference on World Wide Web*, WWW '05, pages 107–116, New York, NY, USA. ACM.

Menczer et al., 2004. Menczer, F., Pant, G., and Srinivasan, P. (2004). Topical web crawlers: Evaluating adaptive algorithms. *ACM Transactions on Internet Technology (TOIT)*, 4(4):378–419.

Pant et al., 2004. Pant, G., Srinivasan, P., and Menczer, F. (2004). Crawling the Web. In Levene, M. and Poulovassilis, A., editors, *Web Dynamics: Adapting to Change in Content, Size, Topology and Use*. Springer-Verlag.

Pongor, 1993. Pongor, G. (1993). Omnet: Objective modular network testbed. In *Proceedings of the International Workshop on Modeling, Analysis, and Simulation On Computer and Telecommunication Systems*, MASCOTS '93, pages 323–326, San Diego, CA, USA. Society for Computer Simulation International.

Ripeanu, 2001. Ripeanu, M. (2001). Peer-to-peer architecture case study: Gnutella network. In *Peer-to-Peer Computing, 2001. Proceedings. First International Conference on*, pages 99–100.

Tirado et al., 2010. Tirado, J. M., Higuero, D., Isaila, F., Carretero, J., and Iamnitchi, A. (2010). Affinity P2P: A self-organizing content-based locality-aware collaborative peer-to-peer network. *Computer Networks*, 54(12):2056–2070.

Voulgaris et al., 2004. Voulgaris, S., Kermarrec, A., Massouli, L., and van Oteen, M. (2004). Exploiting semantic proximity in peer-to-peer content searching. In *Proceedings of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems*, pages 238–243, Washington, DC, USA. IEEE Computer Society.

Wang, 2011. Wang, L. (2011). Sofa: An expert-driven, self-organization peer-to-peer semantic communities for network resource management. *Expert Syst. Appl.*, 38:94–105.

Watts and Strogatz, 1998. Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442.

Yeferny and Arour, 2010. Yeferny, T. and Arour, K. (2010). LearningPeerSelection: A query routing approach for information retrieval in P2P systems. In *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on*, pages 235–241.