A Popularity-Aware Semantic Overlay for Efficient Peer-to-Peer Search

Choonhwa LEE[1], Junwan CHOI[1], Eunsam KIM[2]
[1] *Division of Computer Science and Engineering, Hanyang University, Seoul 133-791, Rep. of Korea*
[2] *Department of Computer Engineering, Hongik University, Seoul 121-791, Rep. of Korea*
eskim@hongik.ac.kr

***Abstract*—This paper presents a novel semantic overlay scheme that connects peers based on the similarity of their contents. Semantic closeness among overlay peers can effectively be determined via the exchanges of their content summary using Bloom filters. The overlay link quality is further improved by carefully selecting semantic neighbors according to their potential to contribute to content-based searches. The basic idea behind the semantic neighbor selection is that highly replicated documents should not excessively dominate the overlay topology, overshadowing rare to modestly-replicated items whose query efficiency is often more critical for overall search performance. The efficacy of the proposed semantic overlay is validated through our simulation study which demonstrates superior overlay link quality and query routing performance.**

***Index Terms*—computer networks, distributed computing, distributed information systems, keyword search, peer to peer computing.**

I. Introduction

Over the past years, there has been a body of research dealing with the inefficiency problem of query flooding in unstructured peer-to-peer networks [1]. Typical approaches to the problem have been to investigate new overlay schemes to improve P2P search efficiency. A new overlay can be constructed by choosing proper neighbor peers according to new neighbor selection criteria. Alternatively, an additional routing index, e.g., routing shortcuts, may be built on top of a base overlay network to enable better search efficiency. The top-layer overlay is queried first before resorting to the base overlay that provides a fail-over search mechanism. Various information may be utilized to form a P2P overlay, including common interests, community membership, interactions, and user behavior patterns (such as tagging, bookmarked pages, comments and rating activities, and so on) of peers [2]. However, it is not surprising to find that a large portion of the approaches is on a basis of content similarities, if we consider the importance of P2P-based content search and sharing.

Semantic overlays are constructed based on the similarities between peers contents, i.e, semantic proximity. Their overlay links or routing tables are set to point to the peers that are semantically closer among others. Consequently, the overlay is expected to comprise peers with semantically related contents, which should yield better search performance. In this paper, we propose a new overlay scheme that discriminates against peers with popular contents; highly frequent documents are excluded from overlay membership decisions. Semantic neighbors for the overlay are chosen in favor of rare-to-intermediate keywords rather than overly common ones, because overly popular terms will less likely contribute to the results of keyword query.

The rest of the paper is organized as follows. In Section II, we first review prominent approaches to the semantic overlay problem. Then, the section continues to propose a novel document popularity-aware semantic overlay scheme. The effectiveness of the proposed scheme is evaluated and compared with existing protocols in Section III. Section IV discusses relevant research to clarify the differences of our approach and previous efforts. Finally, Section V summarizes and concludes the paper.

II. Content Popularity-Aware Semantic Overlay

A. Semantic Overlay Schemes

Semantic overlays are introduced to improve P2P search performance [3-4]. A peer chooses its neighbors among peers who store related contents to itself rather than among random peers. Being built according to the semantic closeness among peers documents, the semantic overlay is able to provide a fast-track path for query propagation. Semantic query routing forwards a query to the peers that likely hold a matching document, before resorting to the fail-over flooding mechanism. In other words, peer links through which to forward a query are chosen based on semantic proximity between query terms and the data that target peers keep [3],[5].

A semantic overlay node maintains a list of semantic neighbors of fixed small size l, which indicates a set of peer nodes that are semantically closer to itself. Content-based similarity can be measured by different metrics such as the term total frequency of term t in a file (referred to as the document frequency $df(t)$ used in IR) or a more sophisticated measure like CORI [6]. However, if we consider the volatility of P2P networking environments, a rather simple metric of content similarities like using file names might be more desirable. For example, content similarities can be measured in terms of the number of files that a peer has in common with its neighbor, and query keywords can be matched against the file names. Therefore, a semantic proximity function S between two peers P and Q can be defined as in (1), indicating how many files the two

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2014-H0301-14-1017) supervised by the NIPA(National IT Industry Promotion Agency) and by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(No. 2013R1A1A2009913).

Digital Object Identifier 10.4316/AECE.2014.04016

peers have in common [5],[7]. Notice that F_P and F_Q indicate the lists of files that peers P and Q own, respectively. A peer's semantic neighbor comprises a list of top-l closest peers according to the proximity function.

$$S(F_P, F_Q) = \left| F_P \bigcap F_Q \right| \quad (1)$$

Overlay peers periodically evaluate link quality to their semantic neighbors. For that purpose, each peer exchanges its neighbor view (i.e., the list of its semantic neighbors) along with additional information on what documents it has with its neighbors. Such content information can be a list of file names [7] or peer synopses that summarizes their content [8].

B. Popularity-Aware Semantic Overlay

Peer data can be summarized using Bloom filters [8-10],[12]. Bloom filter is a method to strike balance between storage and computation [13]. In other words, it is a summarization technique to compress information into a less amount at the cost of some possible information loss. It is actually a vector v of m bits initially set to 0. k independent hash functions h_1, h_2, ..., h_k are used to indicate bit positions to be turned on. More specifically, given an input w, the designated bit positions are set to 1, and others remain zero. Our simulation study uses MD5 for the hashing, since it is well-known and its implementation is readily available.

$$v[h_i(w)] = 1, \text{ where } i = 1, 2, .., k. \quad (2)$$

Note that, since a single bit array is used for all inputs, a bit position can be set repeatedly by more than one input. This can causes a *false hit* in which case the filter says "yes" misleadingly, even if a query word is actually not included in the filter.

Bloom filters provide an effective means to summarize peer documents. However, it is possible for the filter to be dominated by popular words appearing in highly-replicated file names. In other words, popular terms may outweigh intermediate-to-unpopular names, in computing semantic proximity, that would otherwise be useful for query routing. Excessively frequent document names may not be that useful from the query routing perspective, because such files can easily be found anyway. Instead, it is rather rare files that are desired to be encoded in a Bloom filter as an effective routing index. Therefore, we propose a new method of Bloom filter-based content aggregation, which takes into account the popularity of file names. By this popularity-aware semantic overlay scheme, rare to modestly-replicated documents are favored over highly-replicated ones. (It is often the case where a routing hint for popular terms is unnecessary, because popular files can easily be discovered in the neighborhood without any extra help.) As a result, our popularity-aware semantic overlays are built over the documents that have a better potential to benefit query resolution process later on.

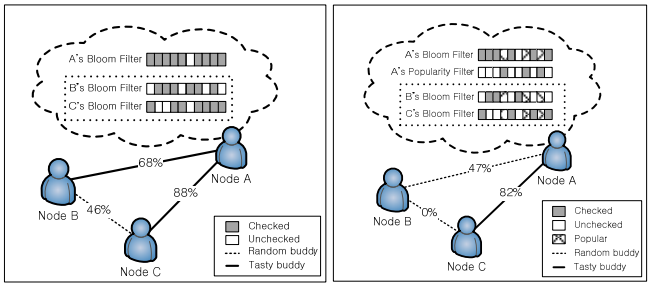
A peer first determines content popularity B_p using its neighbors Bloom filter BF_i. B_p indicates popular bit positions for which a certain portion of neighbors filters has 1, and S is the set of the neighbor peers. Based on that, our semantic similarity between two peers l and m is defined by (3). B_{ul} represents unpopular bits in peer l's Bloom filter BF_l.

$$B_p = \begin{cases} B_p[j] = 1, \text{ if } |\{i \mid BF_i[j] = 1, i \in S\}| > \text{threshold} \\ B_p[j] = 0, \text{ otherwise.} \end{cases}$$

$$Similarity\ (l, m) = \frac{\left| B_{ul} \bigcap B_{um} \right|}{\sqrt{\left| B_{ul} \right| \times \left| B_{um} \right|}} \quad (3)$$

where $B_{ul} = BF_l - B_p$ and $B_{um} = BF_m - B_p$.

A peer computes semantic similarities against every neighbor node of it, and semantic link qualities for the neighbors are ranked. It then promotes top k peers as its semantic neighbors, which means the formation of a popularity-aware semantic overlay network, while keeping the rest as its secondary random neighbors. This semantic proximity re-evaluation process is periodically performed to keep up with changes in the network.



(a) (b)

Figure 1. Popularity-aware semantic overlay. (a) Popularity-unaware overlay. (b) Popularity-aware overlay.

Fig. 1 illustrates how semantic overlay topology can be affected, if we consider content popularity when selecting semantic neighbors. Peers periodically exchange their Bloom filter summary with one another, based on which semantic proximity is measured for each neighbor. Neighbor peers with a higher similarity are promoted to become a semantic neighbor (indicated as *tasty buddy* in the figure.) This semantic overlay provides as a primary means for query routing before resorting to the fail-over search mechanism of flooding. Content similarities are calculated for popularity-aware case by (3), while Bloom filter BF_l and BF_m are used for the popularity-unaware overlay, instead of B_{ul} and B_{um}. In this particular example, neighbor nodes with link quality of over 50% are designated as a semantic neighbor. When taking content popularity into account, node A finds that its link quality to node B is lowered to 47%, which results in the node being demoted to a *random buddy* as illustrated in Fig. 1 (b).

Our Bloom filter-based semantic proximity function provides an effective means to approximately measure the content similarity between any two peers. The proximity function is further refined to take into account the popularity of files, so that rare to modestly-replicated documents can be favored over highly-replicated ones.

III. Performance Evaluation Study

In order to prove the efficacy of our popularity-aware overlay construction scheme, we performed a simulation study that compares its performance with that of its base

popularity-unaware version and Tribler [7]. We built our simulator on PeerSim P2P simulator (http://peersim.sourceforge.net) with the following setups.

A. Simulation Setup

Peer behaviors of our simulation study are modeled to follow Can-O-Sleep data set that consists of MP3 files shared among users on a campus network (https://kdl.cs.umass.edu/display/public/Can-o-sleep). The P2P file-sharing trace data contain file sharing activities around an OpenNap server including peer queries and subsequent downloads. File names in the dataset are used to capture semantic relationships among peers. Individual words of the file names are hashed into a Bloom filter for our simulation study, so that constituent words, instead of the entire file names, can be used as a search query. Also, it is noted that the proposed semantic overlay can be readily formed over file contents rather than file names, so that file content-based keyword search can be supported. Important simulation parameters of our simulation are summarized in Table I.

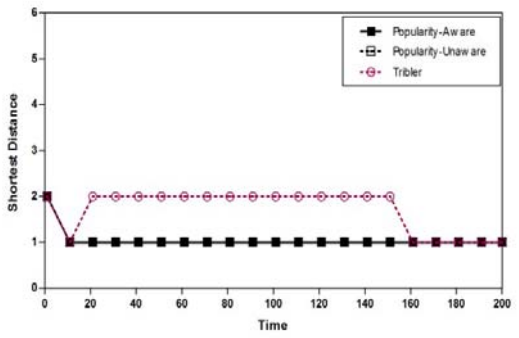
TABLE I. SIMULATION PARAMETERS FOR SEMANTIC OVERLAY

Parameter	Value
Number of peers	6,464
Number of files	291,925
Number of terms	64,112
Bloom filter size	4,096
Number of hash functions (k)	4
Popularity threshold	50%

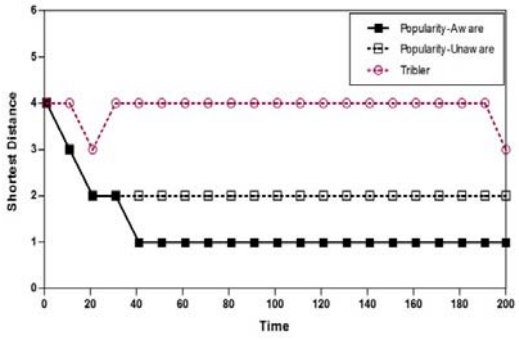
Popularity threshold determines whether a particular bit position in Bloom filters should be considered popular or not. When more than half neighbor filters have a certain bit set, then it is not included in semantic proximity calculation for our simulation. In other words, a file name that causes the bit to be set is considered too popular in the neighborhood to provide any helpful routing hint. About 300,000 files are populated over the network of about 6,500 peers. File names are split into 64,112 constituent words with duplicates removed. The word set follows Zipf distribution with 'the' ranked at the top with the occurrences of 60,215 times. Also, about 30,000 words that amount to 45% of the total words appears just once in the data set. We choose 10 words as query terms that represent each of the popularity bands from the most replicated to the least: *of* (20,340), *you* (13,010), *live* (8,792), *mix* (4,145), *man* (4,033), *end* (1,029), *have* (1,001), *coffee* (98), *sunflower* (33), and *equal* (10). The numbers in parenthesis represent their respective frequency.

B. Comparison of Semantic Overlays

We first look at query distance which is defined as the number of hops a query has to travel until a match. Fig. 2 shows the results of two cases that represent popular and unpopular query words: *live* and *equal*. As expected, three schemes do not show significant differences for the heavily populated query word *live*. However, given an unpopular keyword *equal*, our popularity-aware semantic overlay outperforms its popularity-unaware version and Tribler [7] by 1 and 3 hops, respectively. It is also mentioned that the preference list size of Tribler is 50 for the simulation, which means that up to 50 files can be listed in one message exchanged with neighbor peers.



(a)



(b)

Figure 2. Query distance comparison of popular and unpopular terms. (a) Query distance for *live*. (b) Query distance for *equal*.

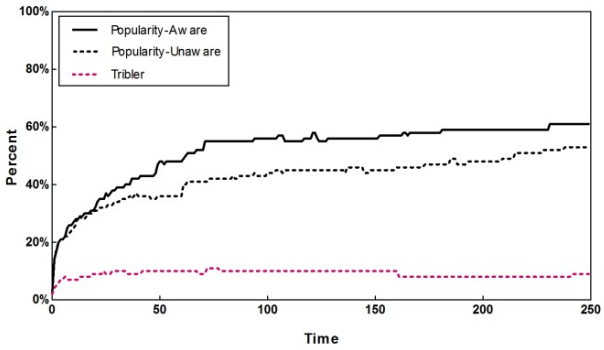


Figure 3. Co-occurrence comparison.

One good way to assess the effectiveness of semantic overlays is to look at how many files are shared among overlay neighbors, which we define as *co-occurrence*. More specifically, it indicates the number of files that simultaneously appear at both a peer and its semantic neighbors. The higher the metric, the better clustered from a semantic closeness standpoint. Fig. 3 compares the effectiveness of each algorithm against an optimal overlay network topology where a peer is associated with 10 best semantic neighbors out of the entire network from the content similarity perspective. Each protocol's co-occurrence is calculated for the whole network; individual node's co-occurrence is first calculated, and then averaged for the network. According to the graph, Tribler's performance remains low at about 0.1, which is attributed to the limit of Tribler's message size (i.e., preference list size that Tribler can exchange with its neighbors at a time.) Popularity-unaware and -aware protocols achieve

performance around 0.47 and 0.6, respectively, which corroborates our initial hypothesis that heavily populated words overshadow otherwise would-be-useful keywords. Removal of the dominance can capture healthier semantic neighbor relationships, as demonstrated in the result.

IV. Related Work

Although P2P search attracted extensive studies over the past years, it still remains one of the most challenging problems [14]. Notable among others are those that capture and exploit semantic relationships between peers for better search results. The idea behind them is that semantic neighbors, i.e., semantically close peers, are more likely to be able to satisfy a semantically related query. A semantic overlay is first queried, before turning to a fail-over search mechanism if no enough answer is produced by the overlay. Several semantic shortcuts have been proved effective in providing high search efficiency for P2P networks, whether the links are either statically or dynamically created [3],[15-19]. Semantic relationships between peers may be captured either implicitly (e.g., from observing user download patterns) or explicitly (e.g., from the information about the type of files being searched and peer profiles).

The idea of content discrimination based on its popularity is also found elsewhere [4],[9],[11],[20]. However, it is embodied in a different way. Rather unpopular items are favored over popular documents, when building semantic-based routing index [4]. More specifically, semantic neighbors for rare documents are given a preference by having highly replicated document links to be evicted from a semantic link cache to make room for a new entry. Also interesting is the hybrid search technique where a structured overlay is employed to locate rare items, and flooding over an unstructured network is used for searching popular files [11]. This is because structured networks incur higher overheads than unstructured overlay networks for popular documents. Search is first performed via query flooding over the unstructured network. With not enough results being returned, the DHT can be queried as a fail-over search mechanism.

Content popularity is represented by Bloom filters that summarize the content of peers in our work. However, it should also be mentioned that Bloom filters are used to measure the overlap between peer documents in the form of compact synopses [9]. More specifically, Bloom filters are used to estimate mutual overlap between collections, so that the most useful collection can be added to the current set for P2P search to achieve a good recall with a minimal number of peers to contact. The work is similar, in spirit, to ours in that Bloom filters are exploited to determine which peer's documents are more beneficial for content searches.

V. Conclusion

The main contribution of this paper is a novel overlay scheme that connects semantically related peers together. Based on Bloom filter-based content summary exchanges, a peer measures and ranks the semantic closeness of its neighbor peers, among which top k peers are selected as the semantic neighbors. Our proposed popularity-aware overlay scheme substantially improves semantic link quality by preventing highly-replicated documents from overshadowing modestly-replicated-to-rare items that are often more critical for overall search performance. Our proposal is validated through a comparative simulation study, which demonstrates superior semantic overlay quality and query routing performance in terms of query distance and co-occurrence.

References

[1] J. Risson and T. Moors, "Survey of research towards robust peer-to-peer networks: search methods," *Computer Networks*, vol. 50, no. 17, pp. 3485-3521, Dec. 2006.

[2] M. Bender et al., "Peer-to-peer information search: semantic, social, or spiritual?", *IEEE Data Engineering Bulletin*, vol. 30, no. 2, pp. 51-60, Jun. 2007.

[3] S. B. Handurukande, A. -M. Kermarrec, F. L. Fessant, and L. Massoulie, "Exploiting semantic clustering in the eDonkey P2P network," in *Proc. 11th ACM SIGOPS European Workshop*, Leuven, Belgium, 2004, pp. 1-6.

[4] S. Voulgaris, A. -M. Kermarrec, L. Massoulie, and M. van Steen, "Exploiting semantic proximity in peer-to-peer content searching," in *Proc. 10th International Workshop Future Trends in Distributed Computing Systems*,Suzhou, China, 2004, pp. 238-243.

[5] S. Voulgaris and M. van Steen, "Epidemic-style management of semantic overlays for content-based searching," *Lecture Notes in Computer Science*, vol. 3648, pp. 1143-1152, 2005.

[6] J. P. Callan, Z. Lu, and W. B. Croft, "Searching distributed collections with inference networks," in *Proc. 18th Annu. International ACM SIGIR Conf. Research and Development in Information Retrieval*, Seattle, WA, USA, 1995, pp.21-28.

[7] J. A. Pouwelse et al., "Tribler: a social-based peer-to-peer system," *Concurrency and Computation: Practice & Experience*, vol. 20, no. 2, pp. 127-138, Feb. 2008.

[8] F. Saihan and V. Issarny, "Scalable service discovery for MANET," in *Proc. 3rd International Conf. Pervasive Computing and Communications*, Kauai Island, HI, USA, 2005, pp. 235-244.

[9] M. Bender, S. Michel, P. Triantafillou, G. Weikum, and C. Zimmer, "Improving collection selection with overlap awareness in P2P search engines," in *Proc. 28th Annu. International ACM SIGIR Conf. Research and Development in Information Retrieval*, Salvador, Brazil, 2005, pp. 67-74.

[10] G. Koloniari, Y. Petrakis, and E. Pitoura, "Content-based overlay networks for XML peers based on multi-level Bloom filters," *Lecture Notes in Computer Science*, vol. 2944, pp. 232-247, 2004.

[11] B. T. Loo, R. Huebsch, I. Stoica, and J. Hellerstein, "The case for a hybrid P2P search infrastructure," in *Proc. 3rd International Workshop Peer-to-Peer Systems*, San Diego, CA, USA, 2004, pp. 141-150.

[12] A. Broder and M. Mitzenmacher, "Network applications of Bloom filters: a survey," Internet Mathematics, vol. 1, no. 4, pp. 485-509, Oct. 2004.

[13] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422-426, Jul. 1970.

[14] G. Sakaryan, M. Wulff, and H. Unger, "Search methods in P2P networks: a survey," Lecture Notes in Computer Science, vol. 3473, pp. 59-68, 2006.

[15] Y. Aytas, H. Ferhatosmanoglu, and O. Ulusoy, "Link recommendation in P2P social networks," in *Proc. 1st International Workshop Online Social Systems*, Istanbul, Turkey, Aug. 2012.

[16] R. Zhang and Y. C. Hu, "Assisted peer-to-peer search with partial indexing," *IEEE Trans. Parallel and Distributed Systems*, vol. 18, no. 8, pp. 1146-1158, Aug. 2007.

[17] F. Draidi, E. Pacitti, and B. Kemme, "P2Prec: a P2P recommendation system for large-scale data sharing," *Lecture Notes in Computer Science*, vol. 6790, pp. 87-116, 2011.

[18] K. Sripanidkulchai, B. M. Maggs, and H. Zhang, "Efficient content location using interest-based locality in peer-to-peer systems," in *Proc. IEEE INFOCOM*, San Francisco, CA, USA, 2003, p. 2166-2176.

[19] Y. Zhang, G. Shen, and Y. Yu, "LiPS: efficient P2P search scheme with novel link prediction techniques," in *Proc. IEEE International Conf. Communications*, Glasgow, Scotland, 2007, pp. 1875-1880.

[20] H. Han, J. He, and C. Zuo, "A hybrid P2P overlay network for high efficient search," in *Proc. 2nd IEEE International Conf. Information and Financial Engineering*, Chongqing, China, 2010, pp. 241-245.