Interest-Aware Content Discovery in Peer-to-Peer Social Networks

YONGHONG GUO, Jiangsu University, Changzhou College of Information Technology
LU LIU, University of Derby
YAN WU, Jiangsu University
JAMES HARDY, University of Derby

With the increasing popularity and rapid development of Online Social Networks (OSNs), OSNs not only bring fundamental changes to information and communication technologies, but also make extensive and profound impact on all aspects of our social life. Efficient content discovery is a fundamental challenge for large-scale distributed OSNs. However, the similarity between social networks and online social networks leads us to believe that the existing social theories are useful for improving the performance of social content discovery in online social networks. In this paper, we propose an interest-aware social-like peer-to-peer (IASLP) model for social content discovery in OSNs by mimicking ten different social theories and strategies. In the IASLP network, network nodes with similar interests can meet, help each other and co-operate autonomously to identify useful contents. The presented model has been evaluated and simulated in a dynamic environment with an evolving network. The experimental results show that the recall of IASLP is 20% higher than the existing method SESD while the overhead is 10% lower. The IASLP can generate higher flexibility and adaptability and achieve better performance than the existing methods.

Categories and Subject Descriptors: Networks~Network protocol design
General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Online Social Networks, Content Discovery, Self-organization

1. INTRODUCTION

Online Social Networks (OSNs) such as Facebook, Twitter, LinkedIn, Google+, etc. have become popular Internet platforms where people around the world can share their social contents. Currently most popular OSNs have been designed as centralized system architectures. However, events have shown the current OSN may face serious security problems with user data storage within the OSN hosting environment [Chowdhury et al. 2014; Paul et al. 2014]. As a result, user privacy may be compromised leading to loss of data or data theft and misappropriation. Furthermore, the centralized nature of OSN infrastructure is frequently prone to generating single point failures. In an attempt to address the problems associated with the centralized OSN, there are growing researches into decentralized architectures for social networks [Chowdhury et al. 2014]. The inherent nature of interaction of people with each other in social networks makes P2P architecture suitable for building the decentralized OSN [Guidi et al. 2013; Paul et al. 2014; Kourtellis et al. 2015]. Social networks provide useful services such as social communication, content-sharing, virtual communities, etc. However, the massive and expanding scale of available information makes the task of locating useful information increasingly difficult. Therefore, the ability of users to be able to find desired social contents from social networks remains a critical issue.

Social content discovery in traditional social networks is mainly related to locating social content and social information relevant to a specific person. Research methods of traditional content discovery can be divided into collaborative filtering methods and community discovery methods. Collaborative filtering methods are aiming at searching for social contents of target users by analyzing similar preferences of users in social networks. Making full use of the information held by friends with similar interests can improve the efficiency of content

This work was partially supported by the National Natural Science Foundation of China under Grants No. 61502209 and 61502207, Natural Science Foundation of Jiangsu Province under Grant BK20170069, UK-China Knowledge Economy Education Partnership and Jiangsu Provincial Project for Brand Specialty Construction under Grant No. PPZY2015A090.
Author's addresses: Y. Guo and Y. Wu are with the School of Computer Science and Telecommunication Engineering, Jiangsu University, China; L. Liu (corresponding author) and J. Hardy are with the Department of Electronics, Computing and Mathematics, University of Derby, Derby, UK.

discovery [Liu and Lee 2010]. Community discovery methods take advantage of the fact that users of social networks usually connect to communities based on their interests and preferences [Newman 2004; Lancichinetti et al. 2009]. Users can find highly relevant contents by identifying relevant communities.

Although traditional content discovery methods have obtained better efficiency in centralized OSNs, they are not suitable for decentralized P2P social networks. Currently, the mechanisms for content discovery in P2P social networks are derived from techniques of P2P networks. Existing solutions for content discovery of P2P systems can be classified into two categories: structured and unstructured. Structured P2P systems have a dedicated network structure which establishes a link between stored content and node addresses by using Distributed Hash Tables (DHTs) for content discovery, like Chord [Stoica et al. 2001]. In contrast to structured systems, unstructured systems do not have to maintain network structure. The flooding method [Ripeanu 2001], random walks method [Kalogeraki et al. 2002] and super-node routing method [Lo et al. 2005] are used for content discovery. These methods can, however, generate massive amounts of traffic or have a lower performance. Some studies [Liao et al. 2010; Hu et al. 2014] assign peer nodes to different groups or communities according to the interests of users. These studies exploit similarity of interests of users in the same group, theories of small-world [Watts and Strogatz 1998; Kleinberg 2000] and scale-free [Barabási and Albert 1999] features to optimize the search algorithms for P2P networks and improve performance of content discovery. Studies of Liu et al. [2009] propose an efficient social-like P2P (ESLP) model for content discovery in self-organized unstructured P2P networks according to social network theories presented by Watts et al. [2002]. Liu et al. [2016] further improve ESLP and propose the SESD model which supports multi-topics search for contents discovery. Guidi et al. [2016] propose a P2P Distributed Online Social Network (DiDuSoNet) model to exploit friend relationships to store data copies and resolve data query. Yuan et al. [2016] use the homophily-based user model to create knowledge index, and use the olfactory sensitive search algorithm to search service content in social networks. Margaris et al. [2017] employ social network information and collaborative filtering techniques to present a query personalization algorithm to search social contents.

Some search techniques described above provide useful solutions to the problem of content discovery in P2P systems. However, these methods are flawed when used for content discovery in P2P social networks. Structured P2P methods require a higher overhead for network structure maintenance. Existing P2P methods based on communities use extra communication overhead to build communities. Although self-organized method in studies of Liu et al. [2009; 2016] has a lower overhead, it neither considers forming content communities according to contents of peer nodes with similar interests nor exploits the number of contents of peer nodes to improve search efficiency.

In this paper, we present an interest-aware social-like P2P (IASLP) model for social content discovery in P2P social networks. The privacy issues and problems of finding content to download will not be discussed in this paper. Unlike previous models, the IASLP model is not going to intentionally use extra messages to construct communities, but adopt human interaction strategies in social networks to self-organizationally form content communities and knowledge networks similar to the ESLP and SESD schemes. However, the ESLP and SESD models only leverage the search topic and related topics in the interest area of the search topic to form knowledge networks. In contrast to the ESLP and SESD models, the IASLP model not only exploits search topics to group knowledge networks but also takes into account interest attributes of users to form content communities spontaneously. In the IASLP model, each peer utilizes a local social knowledge index composed of an interest index and a knowledge index to collect knowledge of previous queries to enhance future routing decisions. In the social knowledge index, the interest index is associated with communities of contents taxonomy and the knowledge index is associated with knowledge networks relevant to search topics. Moreover, in the IASLP network, knowledge structures of the social knowledge index are different from existing methods. In the interest index, the knowledge consists of associations between interest-keywords of peer nodes and related peer nodes with the number of documents matching with these interest-keywords. In the knowledge index, the knowledge is composed of associations between search topics and related peer nodes with the number of documents matching with these search topics. Furthermore, IASLP has a different routing forwarding strategy when compared to existing methods. Each peer node of IASLP utilizes an adaptive node selection algorithm to select forwarded neighbors. In the query forwarding process, requesting peer nodes of IASLP search for social contents from neighbors within an interest community with a probability p and search for social contents from peer nodes outside of the community with probability (1-p).

In this paper, we made the following contributions:

— We developed the IASLP model for social content discovery in P2P social networks. In the IASLP model, content communities are self-organized according to declared interest attributes of peer nodes.
— We proposed an adaptive node selection algorithm to select forwarded neighbours. Queries are forwarded to neighbours with more matching social contents.
— We proposed a probabilistic method to forward queries. Queries are usually sent to peers within a content community and a knowledge network with high forwarding probability, but occasionally sent to randomly selected peers outside of this content community and the knowledge network with a low forwarding probability.
— We simulated the proposed model and evaluated its performances compared to existing state-of-art methods.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 presents the IASLP model. The evaluation methodology and simulation results are introduced in Section 4 and Section 5. The work is concluded in Section 6.

2. RELATED WORK

There are many existing P2P solutions, which can be applied to address problems of content discovery in P2P social networks.

The Random Breadth-First-Search (RBFS) [Kalogeraki et al. 2002] is a distributed algorithm for content discovery without index mechanisms in P2P networks. RBFS is an extension of the Gnutella protocol. When receiving a query, each peer node sends the query to peers in a random selected subset of neighbors until the TTL (Time-to-Live which is the number of times a query message can be forwarded before it is discarded) reaches 0. The duplicated queries are discarded in each hop. In contrast to flooding-based search query, RBFS mitigates network overload. The RBFS algorithm is more efficient than Gnutella. However, RBFS has a relatively lower query performance.

The index engineering of P2P search methods can help improve the efficiency of search algorithms. The concept of the Routing Induce (RI) [Crespo and Garcia-Molina 2002] is introduced to discover contents in P2P systems. Each peer node keeps a Compound RI (CRI) which stores the number of files along each search path and the number of files on each topic interest. The routing algorithm selects the “best” neighbors with the maximum number of files and sends queries to them. This avoids the flooding problem, but creating and updating CRI can generate a higher cost and an increase in network overhead. In contrast, the routing indices of IASLP can be maintained spontaneously without extra messages.

NeuroGrid [Joseph 2002] is an adaptive decentralized search system with a knowledge base at each peer. The routing algorithm of NeuroGrid builds the knowledge base of each peer to store associations between search topic and related peers in the query process. NeuroGrid uses the location of historic contents in the knowledge base to find files. In the NeuroGrid network, each peer maintains a routing table to support distributed searches by sematic routing. Query messages are forwarded to at most M neighbors matching search topics in the knowledge base. When an insufficient number of matching peer nodes are found, query messages are forwarded to neighbors randomly selected from the pool of connected nodes until the number of queried peer nodes reaches a lower bound N (M>N). NeuroGrid has a better performance when compared to RBFS. However, NeuroGrid has a low efficiency in P2P social networks where peer nodes continuously join and leave.

Watts et al. [2002] suggest that ordinary people are capable of directing messages to reach a distant person through their network acquaintance in a few steps. The social network searchability can be applied to many network search problems, including content discovery in P2P social networks. The ESLP model [Liu et al. 2009] is presented to search social contents by mimicking the behavior of people in human society. In ESLP, each uses a knowledge index to store associations of topics and related peer nodes. The ESLP algorithm uses ordinary and active query mechanisms to find social contents. Liu et al. [2016] further developed ESLP and proposed the SESD model which can also support multi-topics search. The ESLP and SESD networks are self-organized and have a lower overhead. However, the ESLP and SESD models have not considered improving search efficiency by building content-sharing communities according to attributes of peers' interests relevant to social contents. The IASLP model in this paper will address this problem.

Social network features can be useful for content discovery of P2P social networks [Han et al. 2014; Bellavista et al. 2014]. The authors [Han et al. 2014] map Facebook users' information into P2P networks to build a P2P social network model and present the social-DRWR-P2P search algorithm to extract the latent friendships and compute friend scores. In this model, each node assigns a different weight to each friend according to social features (e.g. knowledge, similarity, etc.). Then, each node computes scores for each friend in terms of weights and latent relationships. The query message is forwarded to the top M friends with higher scores. However, the algorithm has a high computing overhead.

Guidi et al. [2016] propose the DiDuSoNet, a P2P Distributed Online Social Network where users can exercise full access control on their data. In DiDuSoNet, data objects of each node are copied and stored on elected friendly nodes called Points of Storage (PoSs). the DiDuSoNet use a DHT to maintain the knowledge about PoSs. A request node can search the list of PoSs to find requested data. Whenever a PoS of a node changes, the list of PoSs is updated based on the DHT. Maintaining DHT may require additional network and computational overhead.

Yuan et al. [2017] propose a self-organized decentralized social network (SDOSN) model to discover service content in social network. The SDOSN model uses a homophily-based user model to capture the homophily similarity that integrates social relationship and user interest. The olfactory sensitive search algorithm proposed in the model utilizes the collective swarm intelligence to discover the shortest paths with maximum desired services. The SDOSN model does not address the problem of service unavailability under a high network churn.

3. IASLP MODEL

Human society is a self-organizing system where daily social interactions of people gradually form social networks. In social networks, a group of individuals with similar interests spontaneously form social communities with common goals or responsibilities. In these communities, people may obtain social contents from acquaintances that either possess the documents of contents or have knowledge about the content location. Similar to human social networks, in P2P social networks, peer nodes are regarded as persons, connections of peer nodes are regarded as relationships, and the autonomous peer nodes can create connections by virtue of human strategies in social networks [Liu et al. 2009, 2016]. The similarity between human social networks and P2P social networks contributes to our belief that social theories will be helpful in the design of an enhanced performance content discovery method specifically aimed at P2P social networks. In this section, we propose the IASLP model based on ten different social strategies which are derived from social theories.

3.1 Model Design

Social strategy 1: In social networks, the social attributes and social behaviours of people affect the selection of network partners. People with similar interests gradually connect to each other and form communities spontaneously according to interests. The intensity of people's relationship is influenced by the intensity of their interests [Lönnqvist and Itkonen 2016].

Social strategy 2: People develop strong relationships and weak relationships according to the frequency of their interactions [Wellman 1997]. Strong relationships tend to provide more social support than weak relationships and tend to connect people who provide similar social contents. Weak relationships tend to link people to other social worlds providing new social contents [Wellman 1997].

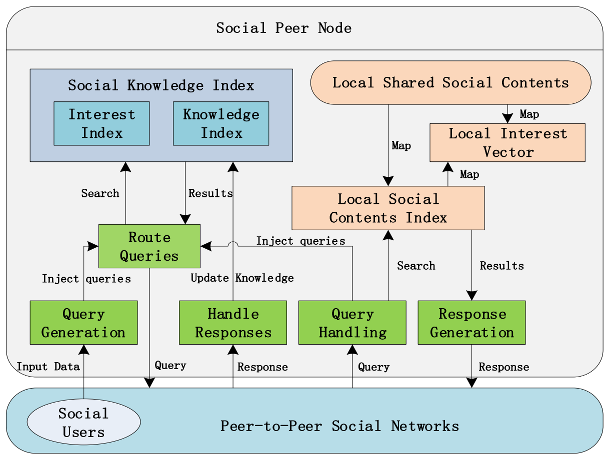


Fig.1. The structure of IASLP model.

As with people in social networks, each peer node in an IASLP network shares an interest attribute (local interest vector) associated with its own social contents to help form content communities based around common interests (as shown Fig. 1). Each peer has a social knowledge index to save acquired knowledge. The social knowledge index includes an interest index and a knowledge index. The interest index is a collection of knowledge according to strong relationships. The knowledge index is a collection of knowledge according to weak relationships. The interest index of each peer node records declared interest topics in the local interest vector and associated nodes as determined by the results of searches. Social peer nodes form strong relationships based on their interest indices. The interest index clusters social peers with the similar social contents into content communities. Peers sometimes attempt to search for new social contents which are irrelevant to its existing interests. The knowledge index of each peer node records search topics unrelated to local interest vector and associated nodes as determined by the results of searches. Social peer nodes form weak relationships by their knowledge indices. The knowledge index groups peers into knowledge networks based on the relevance of search topic. In Fig. 1, when a peer node receives a query, it will first check local social content index to find matched files. If the query needs to be further forwarded, the peer node will search local social knowledge index to find the associated peer nodes using the IASLP routing algorithm and multicast the query to these peer nods.

Social strategy 3: In social networks, people remember and update potentially useful knowledge from social interactions. This leads to the apparently random and diffuse information obtained from initial contacts to gradually become highly organized [Newcomb 1975].

In the IASLP network, a newly joined peer will randomly select a subset of its connection nodes and send its first query message to nodes in the subset (as shown in Fig. 1). The query message includes the interest and search topic of the request peer. A target peer will respond the requesting peer node with the contents matching with query. When the requesting peer node receives a response message from a target peer node and they have a common interest, the requesting peer node will update the interest index in the local social knowledge index to associate the target peer according to interest-keywords. Then, the request peer creates a connection to the target peer with the same interest. If the requesting peer node and the target peer node do not have a common interest, or the search topic is irrelevant to the existing interests of the requesting peer node, the requesting peer node will update the knowledge index in the local social knowledge index to associate the target peer node with the search keyword. The level of knowledge that can be acquired from the results of previous searches increases in proportion to the number of searches performed. The newly obtained knowledge is stored in the local social knowledge index, and the invalid knowledge is removed. Each peer can acquire knowledge from the results of previous searches, and the acquired knowledge can help peer nodes to quickly find other peer nodes who possess the desired contents in the future.

Social strategy 4: In social networks people are motivated in a selective manner toward specific goals. People tend to manipulate circumstances, so that they can benefit in socializing with the people they choose [Newcomb 1975]. When people join a new society, they will not only learn knowledge that they directly want, but also actively collect potentially useful knowledge that they have an interest in.

In an IASLP network, when a social peer node searches for social content, it not only uses the local knowledge index to save knowledge directly associated with the search topic, but also utilizes the local interest index to actively collect knowledge relevant to its own interests from neighbors with the same interests.

The interest index and the knowledge index have the same structure which is a two-dimensional map, as is shown in Fig. 2. In this structure, the topic-list contains a maximum of topic keywords y . In the interest index, the topic keywords are interest-keywords. In the knowledge index, the topic keywords are search-keywords. Each topic keyword is associated with a link-list which has a maximum of nodes x . Each node of the link-list is an entry composed of a neighbor $Node\_n_j$ $(1\leq j\leq x)$ and the number of its documents m_j $(1\leq j\leq x)$ matching with the topic keyword $Topic\_n_i$ $(1\leq i\leq y)$.

The memory overhead of the proposed method is low as the size of the knowledge index is limited. For example, if the maximum size of each entry is 256 bytes, the storage overhead of a 60*60 knowledge index is no more than 900k. The computational overhead to process this small index is also very low. Moreover, when processing a query, a forwarding node only needs to search the knowledge index to find the topic which is matching with the requested topic. Only the entries related to the topic need to be traversed. For example, in a 60*60 knowledge index, no more than 15k content needs to be processed.

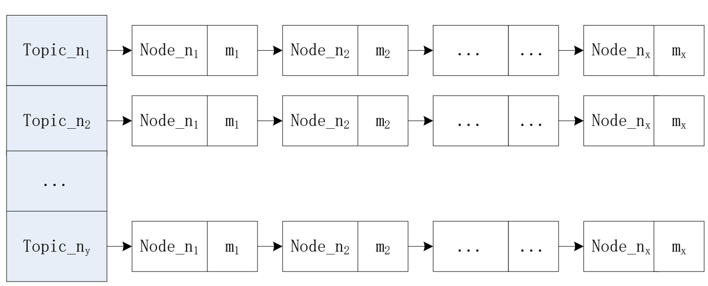
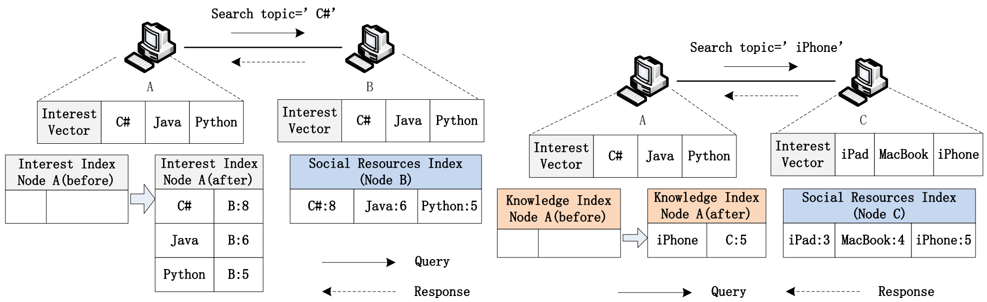


Fig.2. The structure of interest index or knowledge index.

The process of creating social knowledge index is shown in Fig. 3. Node A and node B have the same interest ‘programing language: Java, C#, Python’. Node B has 8, 6, and 5 documents about ‘Java’, ‘C#’, and ‘Python’. When node A receives a response message pack ‘{B, {Java:8, C#:6, Python:5}}’ from node B, it creates its interest index according to this pack. The interest-keywords ‘Java’, ‘C#’, and ‘Python’ are stored in topic-list of interest index. The data pack {B:8}, {B:6}, and {B:5} are added into the link-list relevant to interest-keywords. The process of creating interest index and connecting a target peer is shown in Fig. 3(a).



(a) An example of creating interest index (b) An example of creating knowledge index

Fig.3. Creation of an interest index and a knowledge index

Fig. 3(b) shows an example of creating knowledge index. Node C has declared its interest 'Apple Inc. Products: iPad, MacBook, iPhone' and has 3, 4, and 5 documents about 'iPad', 'MacBook', and 'iPhone'. Node A requests documents about the topic 'iPhone'. Node C receiving the query message will return a response message pack '{C, {iPhone:5}}' to node A. Node A has a different interest from node C, so that node A creates its knowledge index in terms of the response message pack. The search keyword 'iPhone' is saved in the topic-list of knowledge index, and {C:5} is added into the link-list relevant to search topic 'iPhone'.

Social strategy 5: In social networks, some events with associated people will fade from memory with time [Cowan et al. 1999]. In the same way, personal networks evolve and adapt with changing experience and environment [Fisher and Lipson 1985].

Similarly, each peer in the IASLP network updates its knowledge from daily search results. The old knowledge will be replaced by some newly learned knowledge using a Least Recently Used (LRU) strategy. The old and invalid knowledge will be discarded when the number of knowledge items reaches the maximum storage capacity.

In the social knowledge index of IASLP network, the topic-list is maintained by using the LRU algorithm. The most recently used topic is at the top and the least recently used topic is at the bottom. When the number of topic reaches the maximum list size y, the least recently used topic will be discarded. The nodes in the link-list of each topic are also updated by using the LRU algorithm. The most recently used node is at the head of link-list, and the least recently used node is at the end of link-list. When the number of neighboring nodes in the link-list reaches a maximum size x, the least recently used node is dropped from link-list.

In the IASLP network, suppose that there is a target peer D that receives a forwarded message from peer C, and the query message is generated by a request peer A, if D has some documents matching with the search topic, target peer D returns a response message with its documents information to peer A. Peer A updates its social knowledge index (interest index or knowledge index) according to response message of D and connects to D. Then, for the future query on the same topic, the peer A can directly select neighbor D from local social knowledge index and sends query message to D rather than to C.

The IASLP model adopts very well to the short-term social events, like block buster movies or major social events. Those flash crowd events will attract many searches and attentions by the Internet users including the user's friends and neighboring nodes, which will enable the user to easily find the desired content.

3.2 Routing Query algorithm

When a peer generates a query, it will search the interest index or the knowledge index from its local social knowledge index in terms of search topic to get neighbors to be forwarded and send the query to selected neighbors. Neighbors receiving the query will update their interest indices or knowledge indices, and then the query will be forwarded until TTL=0.

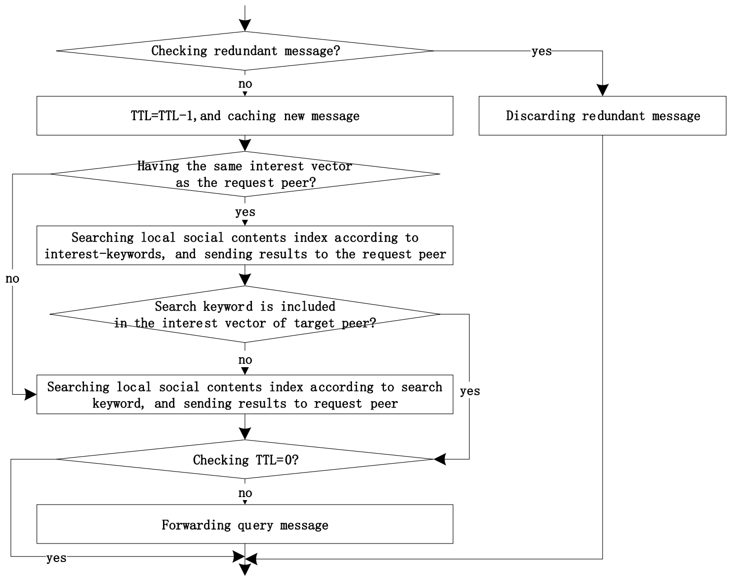


Fig.4. The flowchart for query process handling.

Social strategy 6: Watts and Strogatz [1998] rewire each edge in a regular network at random with probability p to form a small world network. This model is adaptable for social networks. Communities in social network are self-organized with common interests. In the IASLP network, the knowledge of each peer is collected in query procedure, and the links between peers are built according to search results. When peers search social contents, they have a higher probability of connecting to other peers with the same interests or knowledge, the effect of which is the spontaneous creation of virtual content communities and knowledge networks.

When a peer node receives a query message, it will check whether this message has been received, redundant message will be discarded, and new message will be stored in a message cache. The IASLP uses the TTL to prevent infinite propagation of query messages and leverages Globally Unique Identifiers (GUIDs) to limit duplicate queries. If a query message should be propagated further, i.e. TTL>0, the message receiver will send this message to selected neighbors by using nodes selection algorithm. If the query message is valid, the target peer will compare own interest with that of the request peer. If two peers have the same interest, the target peer looks for documents from local social contents index according to interest-keywords in local interest vector (as shown in Fig. 1). The results will be returned to the request peer. The request peer will update the interest index according to responded messages. If the target peer has a different interest vector with the request peer, or the search keyword is not included in the local interest vector of the target peer, the target peer will search local social contents index to find documents in terms of the search keyword, and send search results to the request peer. Then the request peer will update its knowledge index according to feedback. The flowchart of query processing is shown in Fig. 4.

3.3 Routing Forwarding Algorithm

In an IASLP network, each peer receives queries and forwards these queries to a subset of its neighbors. The IASLP utilizes an adaptive forwarding algorithm to forward query messages. The forwarding degree of each peer can be adaptively adjusted based on knowledge learned from previous query processes.

3.3.1 Adaptive Forwarding. In an IASLP network, the number of peer nodes to be forwarded to can be adaptively adjusted between a minimum $D_{\min}$ and a maximum $D_{\max}$ in each hop according to the number of selected peers associated with the search topic. The forwarding strategy is somewhat similar to ESLP [Liu et al. 2009], SESD [Liu et. 2016] and NeuroGrid [Joseph 2002]. However, the algorithm for computing forwarding degree d in an IASLP network differs from that of either SESD or NeuroGrid. The forwarding degree of NeruoGrid d is directly related to knowledge matching search topic. In ESLP and SESD networks, the d is related to either direct knowledge matching with the search topic or the indirect knowledge in the interest area relevant to the search topic. But the forwarding algorithms of ESLP, SESD and NeuroGrid do not take into account the number of neighbor's documents matching with the search topic. In contrast, the IASLP selects peers according to the number of documents of neighboring peers directly related to search topics in the social knowledge index. The number of neighbor's documents matching with a search topic is used to calculate the forwarding degree of each peer associated with the search topic. The algorithm selects peers having high forwarding degrees as receivers. The adaptive algorithm of IASLP can provide better performance than ESLP, SESD and NeuroGrid.

In the local social content index, if the number of documents of a selected peer related to search topic is larger, the more desired contents are shared by the peer. Therefore, the probability of forwarding a query message to the peer should also be high by defining a high cut-off. In contrast, if the number is smaller, a lower cut-off of the peer should be set to constrain query message propagation.

Social strategy 7: In the social network, social contents of the same category are distributed in a common community according to interests of users. However, there also are some similar social contents which are held by people outside of the community. People commonly search for contents from friends within a community associated with their interests, and they only occasionally look for the same contents from people that are outside of their established community.

In an IASLP network, the distribution of contents is unbalanced. The content communities formed by the interests of social peers cannot include all social peers who have the same interests or similar contents. Therefore, in order to find more social contents and acquire more knowledge, each peer node usually forwards query messages to neighbors selected from local social knowledge index with the probability p ($0 \le p \le 1$), or occasionally to other social peers randomly chosen from the pool of connected neighbors with the probability $(1-p)$. When $p=1$, all peers to be forwarded are selected from the local social knowledge index, whereas when $p=0$, all of the peers are selected from the pool of connected neighbors which are not included in the local social knowledge index.

3.3.2 Calculating forwarding degree. **Social strategy 8:** A personal network is a special kind of social network that is centered on a person [McCarty 2002]. The personal network is usually used for singularly personal benefit. When a person is initially assigned to a project with a topic, they would normally search contacts for people who know the topic or can provide background information or give advice on how to proceed [Waloszek 2002]. For contents discovery in a social network, people seek contact with persons who have large amounts of useful contents rather than those who have few contents. For example, Bob wants to find Java documents. John has a large amount of Java documents. But Alice only has a small amount of Java documents. Therefore, Bob will have a preference to contact John rather than Alice.

Similarly, in an IASLP network, when a social peer searches for social contents, it prefers to forward query messages to neighbors with more similar contents rather than ones with a few contents.

In the IASLP network, the forwarding degree of a peer associated with the search topic is calculated in terms of the number of this peer's documents related to search topic. Given that an ordered list consists of n peer nodes ($node_i, i=1,2,\cdots,n$) matching with the search topic. The number of documents of each peer node in the list is m_i ($m_i \geq m_{i+1}, i=1,2,\cdots,n$). Then, the correlation degree of a peer node can be calculated by the following formula,

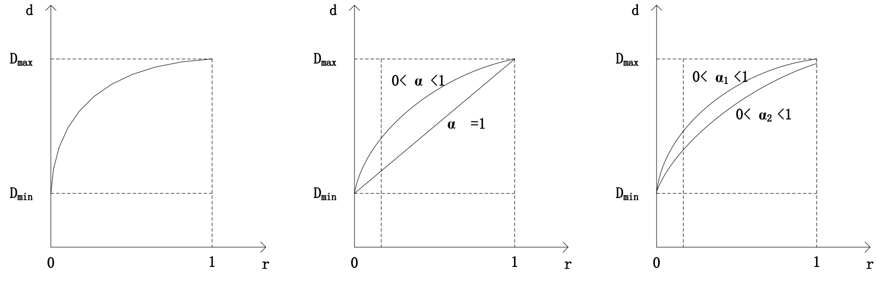
$$r_i = m_i \Big/ \sum_{j=1}^{n} m_j \,, \tag{1}$$

where the correlation degree of the peer r_i is in the range 0 to 1. The forwarding degree of each peer d is determined by its r. The cut-off criterion d is between the minimum number of peers to be forwarded $D_{\min}$ and the maximum number of peers to be forwarded $D_{\max}$, and different peers have different d. During the process of selecting peers, d of a higher correlated peer should be enlarged to make this peer easier to be chosen, and d of lower correlated peer should be decreased to reduce the likelihood of selecting this peer, as is shown in Fig. 5(a). This process is similar to olfactory fatigue. Olfactory fatigue is the temporary, normal inability to distinguish a particular odor after a prolonged exposure to airborne compound [Olfactory fatigue 2013]. For example, when entering a restaurant initially, an odor of food is often perceived as being very strong, but after some time, the awareness of the odor fades out quickly. The d of a selected peer is depicted by the following equation:

$$d = round(r^{\alpha} \times (D_{\max} - D_{\min})) + D_{\min} \,, \tag{2}$$

where the function round(x) returns the closest integer to the given value x, and the parameter $\alpha(\alpha \in (0,1))$ is an adjustment factor. When the forwarding degree of a peer d is higher, this peer has more desired contents. The query message should be sent to peers with higher forwarding degree d. In an IASLP network, the query message will be sent to the selected peers only when the number of these selected peers n is smaller than its cut-off d ($n<d$).

In equation (2), when the correlation degree of a selected peer r is low, there are few contents matching with the search topic in the local content index. Therefore, the forwarding degree of the peer d should also be low. When r tends to zero, d will be close to $D_{\min}$. In contrast, when a selected peer has lots of contents matching with the search topic $(r \approx 1)$, the forwarding degree of this peer d should be high $(d \approx D_{\max})$, as is shown in Fig. 5. According to the principle of olfactory fatigue, using an adjustment factor α ($0<\alpha<1$) controls the size of d. When $\alpha \to 0$, then $d \to D_{\max}$, whereas $\alpha \to 1$, and then $d \to r \times (D_{\max} - D_{\min}) + D_{\min}$. The relationship between the forwarding degree of a selected peer d and the correlation degree of this peer r with α is shown in Fig. 5. In Fig. 5(b), d ($0<\alpha<1$) is higher compared with d ($\alpha=1$) while the correlation degree r remains constant. Fig. 5(c) shows that when the value of α ($0 < \alpha_1 < \alpha_2 < 1$) is smaller, the d of a peer is higher, and then this peer is more likely to be selected. This should ensure that the request peer will obtain more contents matching with the search topic. Thus, selecting a lower α will get a higher recall in the IASLP network.



(a) $0<\alpha<1$ (b) $0<\alpha<1$ or $\alpha=1$ (c) $0<\alpha_1<\alpha_2<1$

Fig.5. For a selected peer node, the relationship between forwarding degree d and the correlation degree r with different adjustable factor α.

3.3.3 Node Selection Procedure. **Social strategy 9**: In social networks, when people want to look for social contents, they usually find previously contacted acquaintances by recalling information from their memory. For example, Bob wants to borrow some books about a programing language and remembers Alice previously loaned some books to him. Therefore, he directly contacts Alice again.

Social strategy 10: In social networks, well-connected people tend to be connected with well-connected people [Newman 2002]. For instance, people form professional relationships with people who have a common interest and also have access to desired communities. Within a given community, those people who have large connections are most likely to be able to provide requested contents.

In an IASLP network, the learned knowledge in the local social knowledge index can help a peer node to obtain social contents from its neighbors. The interest index of the peer node causes clusters of large numbers of peer nodes with similar interests and these homogeneous peer nodes can provide similar social contents. The peer nodes chosen as forwarding targets are first selected from the interest index. When the desired number of peer targets cannot be satisfied by the interest index, the rest of peer nodes are selected from the knowledge index or chosen at random from the rest of connected neighbors.

Three phases are used to select peers as forwarding query targets in each hop: selecting neighbor peers associated with the search topic from the interest index, selecting neighbor peers associated with the search topic from the knowledge index, and randomly selecting peers from the rest of neighbors. When a target peer receives a query message, if the search topic in the message is included in the receiver's local interest vector, the receiver will look for neighboring peers associated with the search topic from the local interest index, and add them to a target list. The selected peers in the target list are ranked by the number of their contents. The receiver will calculate the forwarding degree of each peer *d* using equation (2), and forward query messages to target neighbor peers which have high forwarding degree *d*. If the number of target peers is smaller than $D_{\max}$, the algorithm will move to the second phase. Neighboring peers associated with the search topic in knowledge index are chosen to be added to the target list. The receiver selects peers with high forwarding degree *d* from the target list and forwards query messages to them. In the previous two phases, the total number of target peers is at most $D_{\max}$. If the desired number of target peers is still not satisfied and is smaller than $D_{\min}$, the algorithm will move to the third phase. The receiver will randomly select peers as targets from the rest of its neighbors.

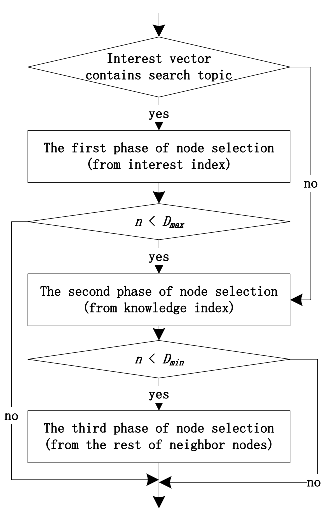


Fig.6. Node selection procedure.

If the interest vector of the receiver does not contain the search topic, the algorithm will directly move to the second phase of node selection procedure. The receiver will select peers associated with the search topic from the local knowledge index rather than the local interest index. At most $D_{\max}$ peers are selected to receive a query message. If the number of selected peers is smaller than $D_{\min}$, the selection process will move to the third phase. In the third phase of node selection procedure, target peers are randomly selected from the rest of neighbors until the number of selected peers reaches $D_{\min}$. The flowchart of node selection is shown in Fig. 6.

3.3.4 An example for Forwarding Query Message. An example for selecting neighboring nodes and forwarding query messages to these nodes is shown in Fig. 7. Peer node A has an interest about programing languages, and declares its interest vector as $\{C\#, Java, Python\}$. The minimum number of selected peers $D_{\min}$ is set to 2, the maximum number of selected peers $D_{\max}$ is set to 4 and the adjustment factor α is 0.7. Suppose that node A receives a message with the search topic 'C#'. It firstly scans its interest vector for keyword 'C#'. Because there is an interest-keyword matching with 'C#', peer A searches its interest index in the first phase. Peer B, C and D are associated with 'C#' and are therefore selected. Peer A ranks B, C and D with the number of contents (B=6, C=5, D=1). The total number of contents of these peers is 12 (12=6+5+1). Then, the peer A calculates the correlation degree of peer B, C and D ($r_B=\frac{6}{12}$, $r_C=\frac{5}{12}$, $r_D=\frac{1}{12}$). The cut-off of peer B is $d_B=round((\frac{6}{12})^{0.7}(4-2))+2\approx 3$ according to equation (2), and due to $d_B>n$ (n=0), the query message is forwarded to peer B. The number n of peers to be forwarded is increased by one: n=0→n=1. The cut-off of peer C is $d_C\approx 3$ by using equation (2), and the query message will be sent to C, because the number of selected peers (n=1) is smaller than the cut-off of peer C ($d_C>n$). Then n=1→n=2. Peer D is not selected, because $d_D=n$ ($d_D\approx 2$, $n=2$). Due to $n<D_{\max}$ ($D_{\max}=4$, $n=2$), the algorithm moves to the second phase of the node selection procedure, and further searches for peers matching with the search topic 'C#' from the knowledge index. However, none of peers are selected from the knowledge index, and the number of selected peers (n=2) is equal to $D_{\min}$ ($D_{\min}=2$), the selection process completes. In this loop query procedure, the number of target peers is two.

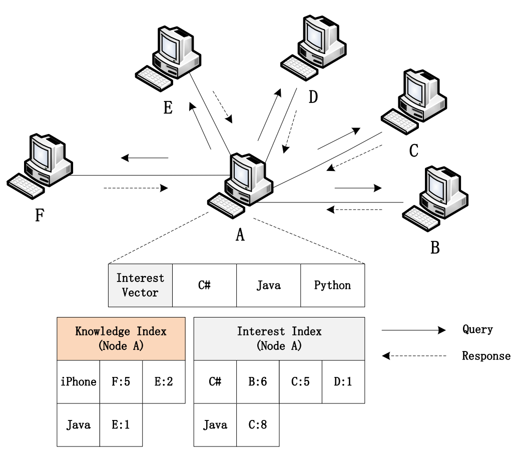


Fig.7. An example of selecting peer nodes to be forwarded from local interest index and knowledge index.

In Fig. 7, when the search topic is "Java", the peer node C selected from the interest index will be the first target peer. Then n=0→n=1. Since there is only one peer in the interest index and $n<D_{\max}$ ($n=1$, $D_{\max}=4$), the algorithm moves to the second phase of node selection procedure. Because $d_E>n$ ($d_E=4$, $n=1$), the peer node E will be selected from the knowledge index. Then n=1→n=2. The node selection procedure will end because of $n=D_{\min}$ ($D_{\min}=2$, $n=2$). Peer C and E are selected as target nodes.

If the search topic is “iPhone” which is not in the interest index of node A, the algorithm will search the knowledge index to find node F and E are associated with “iPhone”. Peer F is selected as the first target ($d_F \approx 4$), then n=0→n=1. Finally, the peer node E is chosen as a target because $d_E > n$ ($d_E \approx 3$, $n=1$). Then n=1→n=2, at which point the node selection procedure completes ($n=2$, $D_{\min}=2$).

In the case of no matching keywords found in the local social knowledge index with a search topic, node A will randomly select two neighbors from the rest of connected nodes as target nodes ($D_{\min}=2$).

4. EVALUATION METHODOLOGY

4.1 Simulator Design

The simulation environment of the IASLP model has been developed using Java. The main components of the simulator are illustrated in Fig. 8.

The simulator generates 1280 topics of social contents and 10,000 documents. Each document is assigned by three topics. These topics are classified into 40 interest areas and each interest area is associated with 32 topics. The interest areas can be found from the Open Directory Categories [ODP 2016], which is a widely distributed database with hierarchical structure. The interest vector of each peer node is generated by randomly selecting an interest area. Each peer node shares a number of documents to the network. These documents are relevant to the interest vector of peer node with a probability of 90%. For the documents relevant to the interest vector, at least one of topics of each document should be included in the interest vector. As there have been no interactions between peer nodes at the beginning of each simulation, each node has an empty social knowledge index.

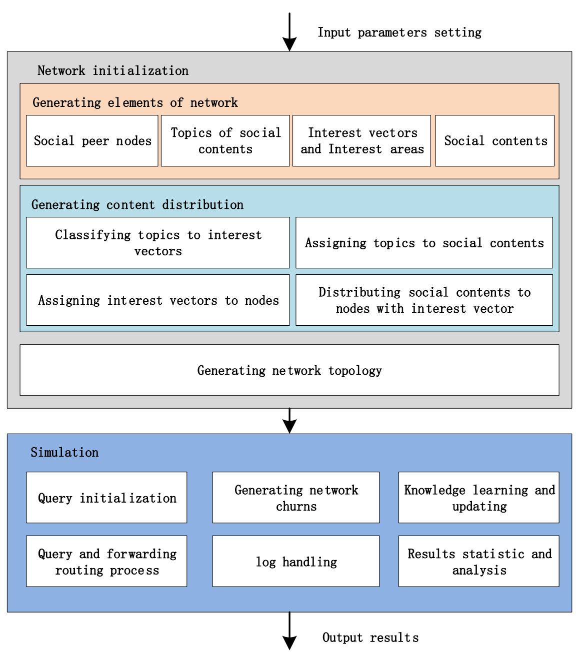


Fig.8. Simulator structure.

There was a 100-nodes small network at the beginning of the simulation. 30 nodes were added into the network each day. This process continues until the end of the first month and the network reached 1000 nodes at the end of the first month. When the network becomes a mature network with 1000 nodes, the simulation continues to run 30 days in the second month. At the beginning of the simulation, each node randomly selects four other nodes to create bi-directional connections to generate a random topology at the simulation start-up.

In the dynamic Internet environment, network churns are usually caused by peer nodes frequently going online and offline. The simulation followed the availability distribution of peer nodes in the study [Bhagwan et al. 2003], where about 50% of peer nodes are presented on the network as less than 30% of time. An online node is randomly picked up as the requesting node to send the query message with a topic. Each search topic is randomly chosen from the interest vector of the requesting node with a probability of 90%, and occasionally selected from other interest areas irrelevant to the interest vector. Query messages are forwarded to peer nodes selected from the social knowledge index with a probability of p (p=0.9). The value of adjustment factor α is 0.3 (α=0.3). Each query is tagged by TTL to limit the lifetime of message to four hops in simulations.

In each time step of the simulations, we randomly chose an online node as the requesting node and started a search with a topic. Even though request frequency is variable for different users in different periods, the study [Gummadi et al. 2003] observers that each peer node generates an average of two requests each day. This has been implemented in our simulations. Simulations are performed to trace the results over two months (60 days). Each request generates an experimental result. The weighted average result is generated from experimental results of each day. In two months, the simulation runs 92,100 times to obtain 60 statistical results.

4.2 Evaluation Metrics

Performance is evaluated with the following measures.

- Recall: a ratio of the number of successfully found documents to the number of all matched documents in network.
- Average recall: a weighted average recall (the weight of each recall is a ratio of the recall to the sum of all recalls).
- Number of query messages: a weighted average number of all query messages to be forwarded in the network.
- Number of visited nodes: an average number of nodes having received query messages (each peer to be forwarded is accumulated only once).
- Number of found documents: an average number of successfully found documents.
- Recall per query message: a ratio of average recall to the number of query messages.
- Recall per visited peer node: a ratio of average recall to the number of visited nodes.

5. SIMULATION RESULTS

The IASLP network has been simulated in a dynamic environment based on different scenarios. The simulation results are analyzed in this section.

5.1 The Performance Comparison to Relevant Methods

IASLP was compared with three relevant methods: RBFS, NeuroGrid and SESD.

- RBFS: query messages are forwarded to randomly selected $D_{\min}$ ($D_{\min}=2$) neighbor nodes in each hop.
- NeuroGrid: query messages are forwarded to peer nodes directly associated with the search topic. At most, $D_{\max}$ ($D_{\max}=5$) nodes are forwarded. If the number of nodes to be forwarded is less than $D_{\min}$ ($D_{\min}=2$), query messages are forwarded to peers randomly selected from the rest of connected neighbors until the number of selected peers reaches $D_{\min}$.
- SESD: the single-topic query of SESD is the same as that of ESLP, query messages are forwarded to not only peers directly associated with the search topic and but also peers

relevant to interest area with the search topic; the number of peers to be forwarded is between $D_{\min}$ ($D_{\min}=2$) and $D_{\max}$ ($D_{\max}=5$); the threshold ratio of active queries is 80%.

The performances of the IASLP network are observed and compared to SESD, NeuroGrid and RBFS. It is observed in Fig. 9 that the recalls of IASLP and SESD are higher than the recalls of NeuroGrid and RBFS; this is because IASLP and SESD leverage knowledge relevant to interests of peers to help enhance the recall. However, in the first month, the recalls of all peers are gradually decreasing owing to the network churn. In the first month, new peers constantly join the network each day and peer nodes in network are frequently going online and offline. This gives rise to generally low performance. In the search process, SESD uses interest areas associated with the search topics to quickly learn knowledge, which displays better performance than IASLP in the early days of simulations. However, as the network becomes more mature, the average recall of IASLP outperforms that of SESD. With the network growing, the peers in the IASLP network can learn more knowledge with the same interest according to a declared interest attribute and store learned knowledge in the local interest index. In the search process, the IASLP also saves knowledge which is irrelevant to the declared interest attribute but matches the search topic with the knowledge index. Meanwhile, the knowledge in the social knowledge index of IASLP includes information about the number of documents of neighbor peers. When forwarding a query message, the peers of IASLP select neighbors directly associated with the search topic by considering the number of matching documents. However, RBFS has no semantic strategy, and both SESD and NeuroGrid only include peers matching the search topic but without considering the number of matching documents. Therefore, the IASLP can obtain more documents matching with the search topics (as shown in Fig. 10) and has better performance (as shown in Figs. 11-12) when compared with other search methods.

In the SESD network, query messages are not only forwarded to peers directly associated with the search topics but also to peers relevant to interest areas associated with the search topics. Therefore, the number of visited nodes for query messages and the number of transferring query messages are highest in the SESD network (as shown in Figs. 13-14). In the IASLP network, query messages are sent to peers directly associated with the search topics using high forwarding probability and occasionally forwarded to randomly selected peers using low forwarding probability. This means that the number of nodes queried and the number of transferring query messages are less than those of SESD. When compared to SESD, the IASLP obtains a higher number of relevant documents with a smaller number of visited nodes (as shown in Fig. 10 and Fig. 13). Fig. 13 and Fig. 14 show that RBFS has the lowest query overhead and the smallest number of visited nodes for query messages but the recall and number of found documents also is smallest (as shown in Figs. 9-10). This is mainly because RBFS only utilizes the minimum forwarding degree ($D_{\min}=2$) and selects random neighbors as target nodes.

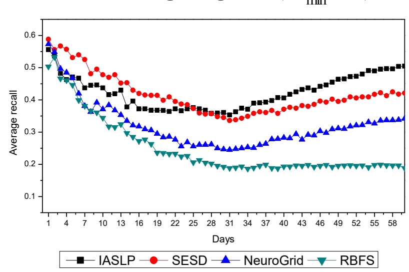


Fig.9. Comparison for average recall in IASLP, SESD, NeuroGrid and RBFS.

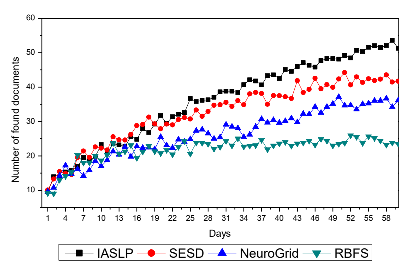


Fig.10. Comparison for number of found documents in IASLP, SESD, NeuroGrid and RBFS.

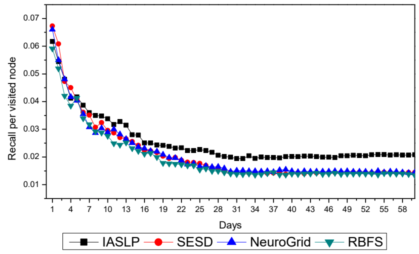


Fig.11. Comparison for recall per visited node in IASLP, SESD, NeuroGrid and RBFS.

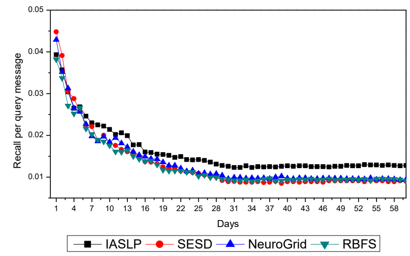


Fig.12. Comparison for recall per query message in IASLP, SESD, NeuroGrid and RBFS.

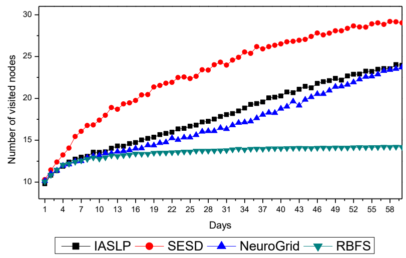


Fig.13. Comparison for number of visited nodes in IASLP, SESD, NeuroGrid and RBFS.

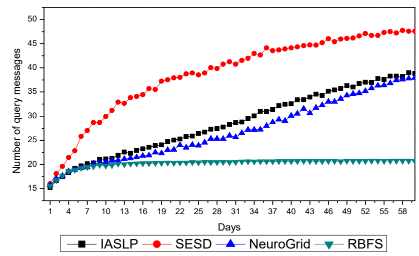


Fig.14. Comparison for number of query messages in IASLP, SESD, NeuroGrid and RBFS.

5.2 Effect of Relevance for Contents

In order to observe the influence of relevance for resources on performance of the IASLP network, different probabilities (P_d=70%, 80%, 90%) of relevance for resources to local interest vector of a peer are used in simulations. With P_d decreasing rapidly, the recall of IASLP decreases slowly in Fig. 15. The recall of IASLP(P_d=70%) is higher than that of SESD(P_d=90%), NeuroGrid(P_d=90%) and RBFS(P_d=90%). Furthermore, the search overhead does not increase with P_d decreasing, as shown in Fig. 16. The IASLP network has a better performance in different probabilities of relevance for contents.

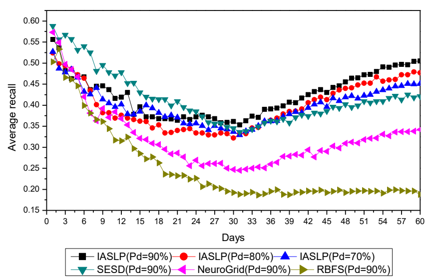


Fig.15. Comparison for recall with different relevance for contents.

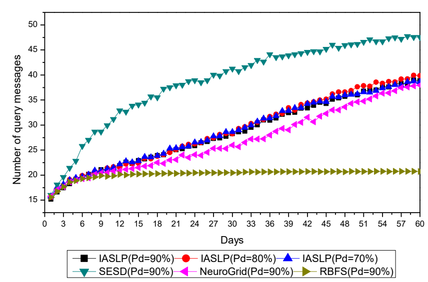


Fig.16. Comparison for number of query messages with different relevance for contents.

5.3 Effect of Adjustment Factor

In order to observe the influence of adjustment factor α on the performance of the IASLP network, different factors (α=0.1, 0.3, 0.5, 0.7, 0.9) are used in the simulations. With a smaller adjustment factor, the average recall is higher in a mature network, as shown in Fig. 17. The average recall increases with a decreasing adjustment factor. According to equation (2), a peer with a smaller adjustment factor has a higher forwarding degree. A peer node with a high forwarding degree holds a greater number of documents. In the IASLP network, a node prefers to forward queries to neighbors with the highest forwarding degree as this returns the highest number of relevant documents. Higher average recall is obtained with smaller adjustment factors, but the overhead related to query forwarding increases with small adjustment factors (as shown in Fig. 18) while performance is hardly affected by the different adjustment factor values (as shown Fig. 19). An appropriate adjustment factor should therefore be selected to maintain a balance between higher recall and lower forwarding overhead. According to experiments in this section, the adjustment factor of 0.3 (α=0.3) offers the best compromise and is therefore the most suitable choice.

5.4 Effect of Forwarding Probability

In this section, the effect of forwarding probability is determined by simulation. In the IASLP network, peers with the similar interest about homogenous contents may form a content community while peers with the same search interest may form a knowledge network. However, there are some desirable contents that are held by persons who are not part of either the content community or the knowledge network. In social contents searching process, a receiver forwards query messages to its neighbors who are selected from local social knowledge index with a probability of *p*, and the query messages also are forwarded to other randomly selected neighbors who are not in local social knowledge index with a probability of (1-*p*). The different probabilities *p* (*p*=0.0, 0.3, 0.5, 0.7, 0.9, 1.0) are used as parameters for simulations. A peer node utilizes the minimum number of selected peers ($D_{min}=2$) to randomly select peer nodes outside of social knowledge index with probability (1-*p*). When $p=0.0$, the least number of nodes ($D_{min}=2$) are always selected. In this case, although the query overhead is smallest, the average recall and performance also are lowest (as shown in Figs. 20-22). In contrast, when *p*=1.0, the message receiver selects all neighbor peers from its social knowledge index. If there are not enough selected peers in local social knowledge index, then the receiver randomly chooses peers from the rest of neighbors. Fig. 20 and Fig. 22 show that the average recall and performance of IASLP are highest with *p*=1.0. The average recall and performance with *p*=0.9 show only a minor reduction below *p*=1.0 (as shown in Fig. 22) but benefits from the opportunity to search outside of the existing community. The choice of 0.9 as the forwarding probability of 0.9 (*p*=0.9) is considered optimal as it delivers a high recall.

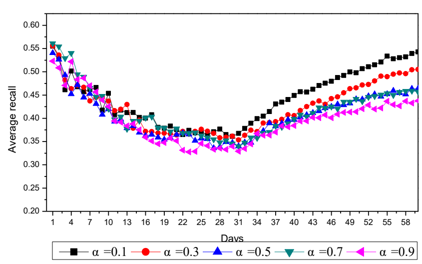


Fig.17. Average recall with different adjustment factors (α=0.1, 0.3, 0.5, 0.7, 0.9).

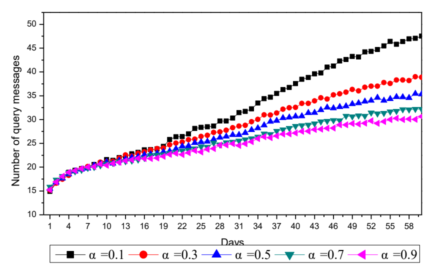


Fig.18. Number of query messages with different adjustment factors (α=0.1, 0.3, 0.5, 0.7, 0.9).

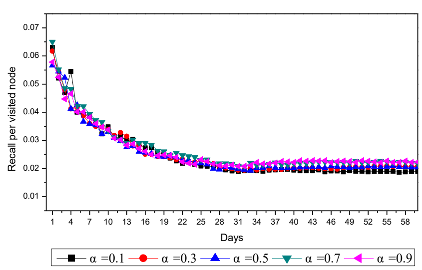


Fig.19. Recall per visited node with different adjustment factors (α=0.1, 0.3, 0.5, 0.7, 0.9).

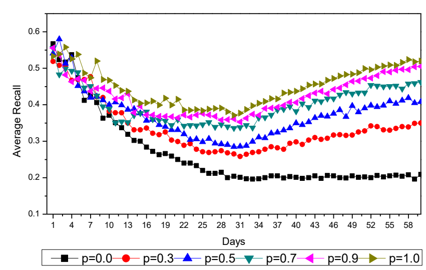


Fig.20. Average recall of different forwarding probabilities (p=0.0, 0.3, 0.5, 0.7, 0.9, 1.0).

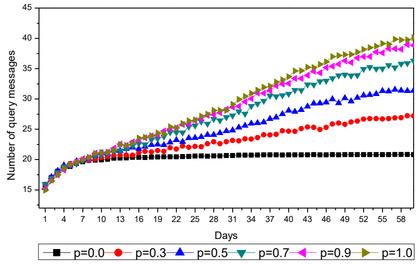


Fig.21. Number of query messages of different forwarding probabilities (p=0.0, 0.3, 0.5, 0.7, 0.9, 1.0).

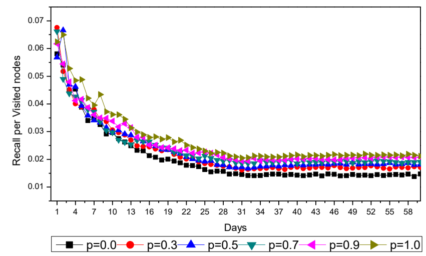


Fig.22. Recall per visited nodes of different forwarding probabilities ((p=0.0, 0.3, 0.5, 0.7, 0.9, 1.0)).

6. CONCLUSIONS

In social networks, people can search for social contents by directly contacting friends or acquaintances that may potentially possess, or have knowledge about, the desired social contents. The social strategies for human interactions in social networks can be leveraged to design a content discovery algorithm for online P2P social networks. In this paper, we have presented an interest-aware social-like P2P mode (IASLP) for social content discovery in distributed online P2P social networks. In the IASLP network, autonomous peers can interact with each other, form relationships and help each other by mimicking human behaviors in social networks. Each peer in the IASLP network can actively declare their own interest attribute associated with their own social contents to help other peers with the same interest form a homogenous content community. In an IASLP network, peers can also engage in knowledge networks based around search topics in a social content discovery procedure. Content communities as well as knowledge networks are formed by self-organized peer groups undertaking a content search process without any extra communication overhead. IASLP helps requesting peers to gain knowledge from each query procedure, which makes future search events more efficient.

In simulations, the IASLP creates a social knowledge index which is composed of an interest index and a knowledge index for each peer. In the social knowledge index of a peer node, the interest index is used to collect knowledge about social contents that are possessed by neighbors with the same interests; the knowledge index is utilized to store knowledge which is directly relevant to the search topic but irrelevant to interest of the peer node. The interest indices of peers with the same interests form a content community with directly relevant interest attributes. The knowledge indices, which are irrelevant to the declared interest attributes of the peers, form a knowledge network. As a peer node collects more knowledge it becomes better educated. Each peer in an IASLP network knows the number of documents of its neighbors from the local social knowledge index. The index provides a method for the peers to identify the most knowledgeable nodes and encourages forwarding query messages to neighbors with most documents. Using simulations, the IASLP has achieved better performance when compared to existing content discovery methods.

In the future, we would like to simulate our model in a larger network with the support of HPC. Furthermore, in social networks, the variety of human interest displays features of diversification and complexity. Exploiting the complexity of human interests to build sematic content discovery strategies for P2P social networks is a worthy research subject. In future work, we will be analyzing human interest attributes to enhance sematic routing strategies to create a viable content discovery algorithm for online P2P social networks. The prototype system development will be also focused in the next step. The prototype system will be developed in the Android system and tested in the laboratory environment with student volunteers.

REFERENCES

Shihabur Rahman Chowdhury, Arup Raton Roy, Maheen Shaikh, and Khuzaima Daudjee. 2014. A taxonomy of

decentralized online social networks. Peer-to-Peer Networking and Applications, vol. 8, 367-383.
Thomas Paul, Antonino Famulari, and Thorsten Strufe. 2014. A survey on decentralized Online Social Networks. Computer Networks, vol. 75, Part A, 437-452.
Barbara Guidi, Conti Marco, and Ricci Laura. 2013. P2P architectures for distributed online social networks. In Proceedings of the 2013 International Conference on High Performance Computing and Simulation (HPCS 2013), 678-681.
Nicolas Kourtellis, Jeremy Blackburn, Cristian Borcea, and Adriana Iamnitchi. 2015. Special Issue on Foundations of Social Computing: Enabling Social Applications via Decentralized Social Data Management. ACM Trans. Internet Technol., vol. 15, 1-26.
Fengkun Liu and Hong Joo Lee. 2010. Use of social network information to enhance collaborative filtering performance. Expert Systems with Applications, vol. 37, 4772-4778.
M. E. J. Newman. 2004. Fast algorithm for detecting community structure in networks. Physical Review E, vol. 69, 066133.
Andrea Lancichinetti, Santo Fortunato, and Janos Kertész. 2009. Detecting the overlapping and hierarchical community structure in complex networks. New Journal of Physics, vol. 11, 033015.
Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan. 2001. Chord: A scalable peer-to-peer lookup service for internet applications. In ACM SIGCOMM 2001- Applications, Technologies, Architectures, and Protocols for Computers Communications, San Diego, CA, United states,149-160.
Matei Ripeanu. 2001. Peer-to-peer architecture case study: Gnutella network. In Proceedings-1st International Conference on Peer-to-Peer Computing (P2P 2001), 99-100.
Vana Kalogeraki, Dimitrios Gunopulos, and D. Zeinalipour-Yazti. 2002. A local search mechanism for peer-to-peer networks. In Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM 2002), McLean, VA, United states, 300-307.
Virginia Lo, Dayi Zhou, Yuhong Liu, Chris GauthierDickey, and Jun Li. 2005. Scalable supernode selection in peer-to-peer overlay networks. In 2nd International Workshop on Hot Topics in Peer-to-Peer Systems (HOT-P2P 2005), San Diego, CA, United states, 18-27.
Hao Liao, Kuo-Chan Huang, and Hung-Chang Hsiao. 2010. Small-world social relationship awareness in unstructured peer-to-peer networks. In 16th IEEE International Conference on Parallel and Distributed Systems (ICPADS 2010), Shanghai, China, 770-775.
Chih-Lin Hu, Yi-Hsun Chang, and Kuo-Fu Huang. 2014. Searching with feature similarity in unstructured peer-to-peer networks. In 2014 7th International Conference on Ubi-Media Computing and Workshops (U-MEDIA 2014), Ulaanbaatar, Mongolia, 65-71.
Duncan J. Watts and Steven H. Strogatz. 1998. Collective dynamics of "small-world" networks. Nature, vol. 393, 440-442.
Jon M. Kleinberg. 2000. Navigation in a small world. Nature, vol. 406, 845-845.
Albert-László Barabási and Réka Albert. 1999. Emergence of scaling in random networks. Science, vol. 286, 509-512.
Lu Liu, Nick Antonopoulos, Stephen Mackin, Jie Xu, and Duncan Russell. 2009. Efficient resource discovery in self-organized unstructured peer-to-peer networks. Concurrency Computation Practice and Experience, vol. 21, 159-183.
Lu Liu, Nick Antonopoulos, Minghui Zheng, Yongzhao Zhan, and Zhijun Ding, "A Socioecological Model for Advanced Service Discovery in Machine-to-Machine Communication Networks," Acm Transactions on Embedded Computing Systems, vol. 15, no. 2, pp. 38, 2016.
Duncan J. Watts, Peter Sheridan Dodds, and M. E. J. Newman. 2002. Identity and search in social networks. Science, vol. 296, 1302-1305.
Arturo Crespo and Hector Garcia-Molina. 2002. Routing indices for peer-to-peer systems. In 22nd International Conference on Distributed Systems, Vienna, Austria, 23-32.
Sam Joseph. 2002. NeuroGrid: Semantically routing queries in Peer-to-Peer networks. In International Workshop on Web Engineering and International Workshop on Peer-to-Peer Computing (NETWORKING 2002), Pisa, Italy, 202-214.
Xiao Han, Angel Cuevas, Noel Crespi, Ruben Cuevas, and Xiaodi Huang. 2014. On exploiting social relationship and personal background for content discovery in P2P networks. Future Generation Computer Systems, vol. 40, 17-29.
Paolo Bellavista, Carlo Giannelli, Luca Iannario, Laurent-Walter Giox, and Claudio Venezia. 2014. Peer-to-peer content sharing based on social identities and relationships. IEEE Internet Computing, vol. 18, 55-63.
Guidi Barbara, Amft Tobias, De Salve Andrea, Graffi Kalman, and Ricci, Laura. 2016. DiDuSoNet: A P2P architecture for distributed Dunbar-based social networks. Peer-to-Peer Networking and Applications 9, 1177-1194.
Bo Yuan, Lu Liu, Nick Antonopoulos. 2016. Efficient service discovery in decentralized online social networks. In Proceedings of the Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing, Applications and Technologies, Shanghai, China2016 ACM, 3006338, 73-78.
Dionisis Margaris, Costas Vassilakis, and Panagiotis Georgiadis. 2017. Query personalization using social network information and collaborative filtering techniques. Future Generation Computer Systems 78, 440-450.
Jan-Erik Lönnqvist and Juha VA Itkonen. 2016. Homogeneity of personal values and personality traits in Facebook social networks. Journal of Research in Personality, vol. 60, 24-35.
Barry Wellman. 1997. An electronic group is virtually a social network. Culture of the Internet, vol. 4, 179-205.
Theodore M. Newcomb. 1975. Social Psychology: the Study of Human Interaction (2nd. ed.). Revised Routledge and Kegan Paul, London.

Nelson Cowan, Lara D. Nugent, Emily M. Elliott, Igor Ponomarev and J. Scott Saults. 1999. The Role of Attention in the Development of Short - Term Memory: Age Differences in the Verbal Span of Apprehension. Child Development, vol. 70, 1082-1097.

K. M. Fisher and Joseph I. Lipson. 1985. Information processing interpretation of errors in college science learning. Instructional Science, vol. 14, 49-74.

Christopher McCarty. 2002. Structure in personal networks. Journal of social structure, vol. 3, 20.

Gerd Waloszek. 2002. Personal Networks. In SAP AG, Product Design Center.

M. E. J. Newman. 2002. Assortative mixing in networks. Physical review letters, vol. 89, 208701.

Olfactory fatigue. 2013. [Online]. Available: https://en.wikipedia.org/wiki/Olfactory_fatigue.

The Open Directory Project. 2016. Retrieved January 19, 2016 from http://www.dmoz.org/

Ranjita Bhagwan, Stefan Savage and Geoffrey M. Voelker. 2003. Understanding Availability. Lecture Notes in Computer Science 2735, 256-267.

Krishna P. Gummadi, Richard J. Dunn, Stefan Saroiu, Steven D. Gribble, Henry M. Levy, and John Zahorjan, "Measurement, modeling and analyis of a peer-to-peer file-sharing workload," Acm Sigops Operating Systems Review, vol. 37, no. 5, pp. 314-329, 2003.

Received June 2016; revised July 2017; accepted Dec 2017