Computing and Informatics, Vol. 35, 2016, 30–54

INTEGRATION OF LINK AND SEMANTIC RELATIONS FOR INFORMATION RECOMMENDATION

Qin Zhao, Yuan He, Changjun Jiang

The Key Laboratory of Embedded System and Service Computing
Ministry of Education
Tongji University, Shanghai 200092, China
e-mail: 2qzhao@tongji.edu.cn

Pengwei Wang

School of Computer Science and Technology
Donghua University, Shanghai 200051, China
e-mail: pwei.wang@gmail.com

Man Qi

Department of Computing
Canterbury Christ Church University, Canterbury, Kent, CT1 1QU, UK
e-mail: mq4@canterbury.ac.uk

Maozhen Li

Department of Electronic and Computer Engineering
Brunel University, London, Uxbridge, UB8 3PH, UK
e-mail: Maozhen.Li@brunel.ac.uk

Abstract. Information services on the Internet are being used as an important tool to facilitate discovery of the information that is of user interests. Many approaches have been proposed to discover the information on the Internet, while the

search engines are the most common ones. However, most of the current approaches of information discovery can discover the keyword-matching information only but cannot recommend the most recent and relative information to users automatically. Sometimes users can give only a fuzzy keyword instead of an accurate one. Thus, some desired information would be ignored by the search engines. Moreover, the current search engines cannot discover the latent but logically relevant information or services for users. This paper measures the semantic-similarity and link-similarity between keywords. Based on that, it introduces the concept of similarity of web pages, and presents a method for information recommendation. The experimental evaluation and comparisons with the existing studies are finally performed.

Keywords: Information retrieval, data mining, link similarity, information recommendation

Mathematics Subject Classification 2010: 68-P20

1 INTRODUCTION

Nowadays, the amount of information on Internet is growing rapidly. There are many types of information, e.g., web pages, applications and web services, where web pages account for the largest proportion. The data volumes on the Internet are in a multi-petabyte range. Information discovery has become a major theme of the current research efforts.

The conventional information search methods use keyword-matching techniques to discover information, which include the full or partial keywords-matching. However, such methods have some disadvantages. First of all, they work only if users clearly know what they are looking for. If users have little knowledge about what they want to know, a search engine is unable to recommend the information that has a latent semantic or logical relationship with the provided keywords. For example, a user may want to search for Apple's smartphone "iPhone", unfortunately the user forgets both the name of smartphone and its manufacturer but only remembers that the name of manufacturer is a kind of fruit. The conventional search engines cannot give the user correct information if the user only type in the keyword "smartphone" and "fruit". Secondly, current search engines consider only the similarity of the web contents, but not the similarity of web links. Two web pages are possibly similar if they both link to the same kind of web pages. For instance, *New York Times* and *Yahoo! News* both have the news page links on their homepages, and these news pages are similar, so we can consider they are the same kind of website. Conventional search engines cannot recommend information to users dynamically according to users' browsing history. Last but not least, existing search methods do not take into account the logical relationship that may exist among web pages. Briefly, they focus only on the direct links between web pages and the semantic relationship

between keywords, but ignore some logical relationships between web pages and keywords. A web page generally contains web page links which might be of user's interest. For instance, the airline websites generally contain the hyperlinks of hotel websites. These hyperlinks show that a logical relationship exists between air travel and hotel reservation. If one website is frequently linked to another website, they can be considered logically relevant. In addition, the logical relationship also exists among keywords, just like aforementioned keywords "airline" and "hotel". These two keywords have no semantic relationship, but we can see that there is a logical relationship between them in fact. The conventional search engines may recommend users some semantic-relative keywords and some direct-linked web pages, but cannot give users the aforementioned logical-relative keywords and links.

Some methods are proposed to take the hyperlink information into account. Jeh and Widom present SimRank in [1], which is a type of measure based on link similarity. The objects with common neighbours are considered similar. SimRank uses an iterative algorithm to calculate the similarity between them. Unfortunately, the huge data volume of the Internet makes it impractical to use this method to calculate the similarity between each and every pair of web pages. Hence, a computationally efficient method is needed for calculation of the similarities of web pages.

The existing methods to compute the similarity of keywords are mostly based on the semantic relationship of keywords. The work presented in this paper makes an improvement by introducing a logical relationship between keywords. It calculates the semantic similarity among web pages by using the semantic relevance of keywords, and prunes useless links by using the logical relevance of keywords. By considering both contents and links of web pages, this paper presents a novel method to calculate the similarity among web pages. It then gives an algorithm to combine the semantic and link similarity together to recommend the latent relevant keywords or web pages to users. The method to recommend keywords and web pages is also presented.

The remainder of the paper is structured as follows: Section 2 gives a brief introduction to the methods for calculating keyword relationships. Section 3 introduces the concept of the relevance of keywords and Section 4 introduces the link similarity and the similarity among web pages. Section 5 presents a method to recommend relevant words and web pages. Section 6 evaluates the proposed method and analyzes the experimental results. Section 7 discusses related works. Finally, the conclusion and future work is presented in Section 8.

2 BACKGROUND

In this section, we provide a brief background of the methods utilized for calculating keyword relationships. We first discuss the concepts of frequent itemset and mutual information. Then, we provide an overview of SimRank.

2.1 Frequent Itemset

We use frequent itemset [4] to calculate the statistical measure.

An itemset is a set of items that occur together. Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of distinct items. Consider $X \subseteq I$ an itemset. Its support $\text{supp}(X)$ is the ratio of the number of transactions that include all the items of X to the total number of transactions. If $\text{supp}(X) \geq L_k$, a user-specified minimum support where $k = |x|$, we call X a frequent itemset.

The support of an association rule $X_1 \rightarrow X_2$ is defined as $\text{supp}(X_1 \rightarrow X_2) = N_{X_1 \cup X_2}/N$, where $N_{X_1 \cup X_2}$ denotes the number of transactions including both X_1 and X_2, and N denotes the total number of transactions. It represents the probability that X_1 and X_2 appear together, denoted by p_{12}.

The confidence of association rule $X_1 \rightarrow X_2$ is defined as $\text{conf}(X_1 \rightarrow X_2) = \text{supp}(X_1 \cup X_2)/\text{supp}(X_1)$. It represents the probability that X_2 appears when X_1 appears, denoted by $p_{2|1}$.

If an association rule has its support and confidence equals or exceeds their respective minimum threshold, we call it a strong rule.

2.2 Mutual Information

In probability theory and information theory, the mutual information of two random variables is a quantity that measures their mutual dependence [2, 3]. It is a dimensionless quantity with units of bits, and can be thought of as the reduction in uncertainty about one random variable given knowledge of another. High mutual information indicates a large reduction in uncertainty; low one indicates a small reduction; and zero between two random variables means that they are independent.

Formally, the mutual information of two discrete random variables X and Y can be defined as:

$$I(X,Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{1}$$

where $p(x,y)$ is the joint probability distribution function of X and Y, and $p(x)$ and $p(y)$ are the marginal probability distribution functions of X and Y respectively.

2.3 SimRank

Co-citation denotes that if two items are cited together by other items. It was widely used in document retrieval. SimRank [1] is an extension of Co-Citation that considers an entire graph structure. It is a kind of measure based on link similarity, and can be applied to any domain by object-to-object relationships. It considers the co-citation relationship as the similarity of two items. The intuition behind SimRank is that two objects are similar if they are referenced by similar objects.

SimRank uses a recursive computation to calculate the similarity between two objects. Given objects a and b, their SimRank is given below:

$$s(a,b) = \begin{cases} 1, & a = b, \\ \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} s(I_i(a), I_j(b)), & a \neq b, \end{cases} \tag{2}$$

SimRank uses the recursive computation to get the similarity score. Initially,

$$R_0 = \begin{cases} 1, & a = b, \\ 0, & a \neq b. \end{cases} \tag{3}$$

Then, the similarity iteration equation is

$$R_{k+1}(a,b) = \begin{cases} 1, & a = b, \\ \frac{C}{|I(a)||I(b)|} \sum_{i=1}^{|I(a)|} \sum_{j=1}^{|I(b)|} R_k(I_i(a), I_j(b)), & a \neq b. \end{cases} \tag{4}$$

SimRank has several disadvantages. First of all, it considers only the in-neighbors of items, because the out-links of Web pages are given by their designers and may contain some spams. Note that out-links may provide more information to improve the accuracy of similarity. Moreover, SimRank ignores the similarity between direct links. For instance, there are three items a, b, and c, and both b and c are linked by a. SimRank considers that b and c are similar, but considers a is not relevant to b and c. Actually, a may have the same topic as b and/or c. Furthermore, SimRank ignores the semantic relationships between Web pages and their links. It considers only the same links to compute the similarity, but does not take the semantically similar links into consideration. Finally, the time complexity of SimRank is $O(kn^2d^2)$ where k is the iteration count, n is the number of objects (number of the web pages on the Internet), and d is the average in-degree of all objects. SimRank will take a huge amount of computation time when n is large.

3 RELEVANCE OF KEYWORDS

The conventional and popular search engines like Google, Baidu and Bing use only the keyword matching to retrieve the web pages containing the input keywords. Obviously, users can obtain accurate information when the web pages have the queried keywords. However, the information returned by the search engines may not meet users' needs, when the web pages do not explicitly have the proper keywords.

Some methods have been proposed to calculate the relevance of keywords, mainly their semantic relevance. Semantic relevance can not only denote the semantic relation between two items, but also present the logical relation between them.

For instance, a news page about Apple's smartphone *iPhone* may link to another page about Samsung's *Galaxy*. Next, we discuss how to measure the relevance of keywords.

3.1 Support

The concept of support is proposed to measure the probability that two items occur together. It can represent the relationship between two items. For instance, if customers often buy items A and B at the same time, A and B are supposed to be relevant. The support are often used in data mining to find out frequent patterns [20]. Yang et al. use it to measure the statistical similarity of keywords [7]. Paliwal et al. measure the similarity between Web services [19] by calculating the support of their input and output. In Information Retrieval, the support can measure the similarity of objects. Two keywords frequently appearing in the same web page have some latent relationships between them. Hence, we use the support to measure the semantic relevance of keywords.

Given two keywords w_1 and w_2, their support can be calculated by:

$$S(w_1, w_2) = \frac{N_{w_1 \cup w_2}}{N} \tag{5}$$

where $N_{w_1 \cup w_2}$ denotes the number of pages in which w_1 and w_2 appear together, and N denotes the number of the total web pages.

The normalized support $\overline{S}(w_1, w_2)$ can be calculated by:

$$\overline{S}(w_1, w_2) = \frac{S(w_1, w_2) - S_{min}}{S_{max} - S_{min}} \tag{6}$$

where S_{max} is the maximum support of all pairs of the keywords, and S_{min} is the minimum one.

3.2 Mutual Information

The mutual information is a quantity that measures the mutual dependence of two random variables, and it has been introduced into association rule mining by Mei et al. [8]. Different from the support, it does not only denote the positive correlations between keywords, but also give the negative ones. Thus, it can provide more information about a relation between keywords than the support does. However, it requires more computation.

Given two keywords w_1 and w_2, their mutual information can be calculated by:

$$M(w_1, w_2) = \sum_{x_1 \in \{0,1\}} \sum_{x_2 \in \{0,1\}} p(x_1, x_2) \times \log \frac{p(x_1, x_2)}{p(x_1) \times p(x_2)} \tag{7}$$

where $\forall i \in \{1,2\}$, $x_i = 1$ means that w_i appears, and

$$p(x_i) = \begin{cases} \frac{N_{w_i}}{N}, & x_i = 1, \\ 1 - \frac{N_{w_i}}{N}, & x_i = 0. \end{cases} \tag{8}$$

The normalized mutual information $\overline{M}(w_1, w_2)$ can be calculated by:

$$\overline{M}(w_1, w_2) = \frac{M(w_1, w_2) - M_{min}}{M_{max} - M_{min}} \tag{9}$$

where M_{max} is the maximum mutual information of all pairs of the keywords, and M_{min} is the minimum one.

3.3 Confidence

As aforementioned, the confidence can represent the logical relationship from one item to another [4]. Different from the two discussed measures, we use the confidence to measure the link relation that is hidden in the keywords instead of the direct relevance among keywords. The web pages containing one keyword A may frequently link to the web pages containing another keyword B. For example, the websites about travel often have keyword "airline", and often link to the websites containing keyword "hotel". Obviously, "airline" has a latent logical relationship with "hotel". We assume that two keywords A and B are logically relevant if the web pages where A appears often link to other pages where B appears. Therefore, different from the conventional methods calculating the relevance of keywords in one web page, we use the confidence to measure this kind of logical relationship among keywords in different web pages.The confidence from keywords w_1 to w_2 is calculated by:

$$C(w_1 \to w_2) = \frac{N_{w_1 \cup w_2}}{N_{w_1}} \tag{10}$$

where $N_{w_1 \cup w_2}$ denotes the number of pages where w_1 appears if they link the pages containing w_2, N_{w_1} denotes the number of the web pages where w_1 appears.

The normalized confidence $\overline{C}(w_1, w_2)$ is calculated by:

$$\overline{C}(w_1 \to w_2) = \frac{C(w_1 \to w_2) - C_{min}}{C_{max} - C_{min}} \tag{11}$$

where C_{max} is the maximum confidence of all pairs of the keywords, and C_{min} is the minimum one.

3.4 Relevance of Keywords

We propose two methods to calculate the relevance of keywords. These two methods are used to calculate the semantic similarity between web pages and to prune weak links respectively.

The first method is to combine the support and mutual information together to measure the semantic relevance between keywords:

$$r_s(c_i, c_j) = \alpha \times \overline{S}(c_i, c_j) + (1 - \alpha) \times \overline{M}(c_i, c_j) \tag{12}$$

where $\alpha \in [0, 1]$ is a parameter specified by a user, $\overline{S}(a, b)$ and $\overline{M}(a, b)$ are support and mutual information between a and b, which are presented in Equations (6) and (9), respectively. The semantic relevance $r_s(a, b)$ denotes that how close the relationship between keywords a and b is. Users can increase the value of α to obtain more similar relevant keywords, because the support represents the similarity among the keywords. They can decrease the value of α to reveal more negative correlative keywords.

The second one is to combine the support and confidence together to measure the logical relevance between keywords:

$$r_l(c_i, c_j) = \beta \times \overline{S}(c_i, c_j) + (1 - \beta) \times \overline{C}(c_i \to c_j) \tag{13}$$

where $\beta \in [0, 1]$ is an user-specified parameter, $\overline{S}(a, b)$ and $\overline{M}(a, b)$ are support and confidence between a and b, which are presented in Equations (6) and (11), respectively. Similar to the role of α in Equation (12), decreasing β allows users to reveal more keywords with hidden relations to give search word. Note that the support can also be replaced with the mutual information, which can provide more accurate results but cost more computation.

4 SIMILARITY OF WEB PAGES

The conventional and popular methods of measuring the similarity between web pages mostly use the similarity between keywords or keyword vectors. The most common methods are term frequency-inverse document frequency (TF-IDF) [3, 23] and cosine similarity [3]. These two methods both consider the similarity of contents to measure the similarity between web pages. However, they can only measure one side of similarity. Two web pages may not be similar even if they have similar contents. For instance, some information on Apple's "iPhone" is referenced by both a shopping website "ebay.com" and an IT website "engadget.com", while they serve different purposes and exhibit different functions.

Same to the document retrieval, readers would like to read some relevant web pages on the current one to acquire more information. SimRank measures the similarity between items by their links. We can use SimRank to calculate the link similarity between web pages. But different from the document retrieval, it is hard to find two web pages having the same links. Consequently, most of web pages would be irrelevant if we used SimRank to calculate their similarities.

Actually, the semantic similarity among web pages can be considered as a kind of virtual links. By using the latent links, we can calculate the link similarity among the web pages that in fact have no same links.

As illustrated in Figures 1 and 2, the items "Apple" and "Microsoft" are irrelevant and unable to compute their similarity without using the concept of virtual links.

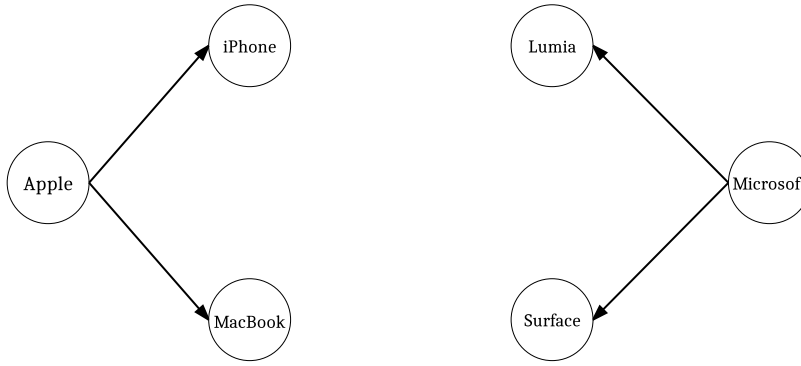


Figure 1. Actual links

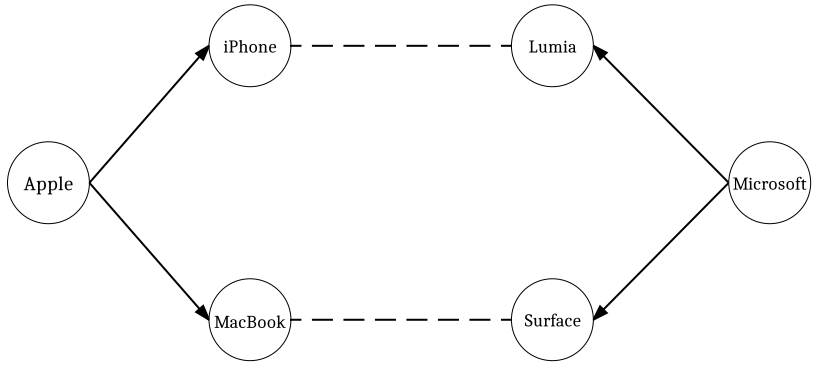


Figure 2. Virtual links

Next, we discuss how to compute the similarity of web pages from two aspects: semantic and links. Since some spams may exist in hyperlinks, we propose a method to prune the weak relevant information from the set of hyperlinks.

4.1 Semantic Similarity of Web Pages

To measure two web pages' semantic similarity, we make the following assumption: If two web pages have the same or relevant keywords, they are semantic-similar. For instance, a web page containing keywords "MacBook" is semantic-similar to another containing "iPhone", because "MacBook" and "iPhone" are relevant.

We use the top k appearing keywords to calculate the similarity among web pages. We extract the most appearing keywords into a keyword vector from a web page. Then we calculate the relevance between each pair of keywords, and find out the most relevant pairs.

Given two web pages P_1 and P_2 whose keyword vectors are $W_1 = \{w_{11}, w_{12}, \ldots, w_{1m}\}$ and $W_2 = \{w_{21}, w_{22}, \ldots, w_{2n}\}$, respectively, their semantic similarity can be calculated by:

$$\mathrm{Sim}_s(P_1, P_2) = \frac{\sum_{i=1}^{m} r_s(w_{1i}, P_2) + \sum_{j=1}^{n} r_s(w_{2j}, P_1)}{m+n} \tag{14}$$

where $r_s(w, P)$ denotes the semantic similarity of a keyword and a web page. The function of $r_s(w, P)$ is represented as:

$$r_s(w, P) = \max_{w_i \in W} r_s(w, w_i) \tag{15}$$

where $w_i \in W = \{w_1, w_2, \ldots, w_n\}$ is the keyword vector of P.

4.2 Link Similarity of Web Pages

In document retrieval, two articles are supposed to be similar if they have the same references and are referenced by the same articles. Similarly, we assume that the web pages with the same hyperlinks and linked by the same web pages are similar.

However, different from the keywords, there may be some useless information contained in the links, due to spams on the web. In addition, some links may be directed to outdated pages. These pages may have already been modified or deleted. Therefore, we must prune some links among the web pages before calculating the link similarity.

In this section, a link l actually represents its directly linked web page.

4.2.1 Prune Weak Links

As aforementioned, the confidence from one word to another can represent their logical relationship. If a web page containing a keyword links to a web pages containing another keyword frequently, these two pages are supposed to be logically relevant. On the contrary, if a link directs to a web page that does not contain any relevant keyword, it is supposed to be weakly relevant. If two web pages have a link between them, but their contents are totally logically irrelevant, we consider this link as a weakly relevant link as well.

To prune weak links, we propose the logical relevance of two web pages as follow:

$$r_l(P_1, P_2) = \max_{w_i \in W_1, w_j \in W_2} r_l(w_i, w_j) \tag{16}$$

where W_1 and W_2 are keyword vectors of P_1 and P_2, respectively.

The logical relevance denotes the relationship between two web pages. If it is less than a user-specified threshold θ, we consider the link is weak and delete the target link from the link set.

4.2.2 Calculate Link Similarity

In document retrieval, the conventional methods calculate probability of the same reference of two documents to represent their similarity. Different from the documents, the web pages are unusual to have the same links. However, the links that have the same topic are supposed to be useful to reflect the similarity. For instance, a page talking about MacBook may be similar to another page talking about iPhone, because they have the same topic 'Apple'. We assume that two pages containing links directed to semantic similar pages are link-similar.

Given web pages P_1 and P_2 whose linked page vectors are $\{l_{11}, l_{12}, \ldots, l_{1m}\}$ and $\{l_{21}, l_{22}, \ldots, l_{2n}\}$, respectively, their link similarity can be calculated by:

$$\mathrm{Sim}_l(P_1, P_2) = \frac{\sum_{i=1}^{m} \mathrm{Sim}_s(l_{1i}, P_2) + \sum_{j=1}^{n} \mathrm{Sim}_s(l_{2j}, P_1)}{m+n} \tag{17}$$

where $\mathrm{Sim}_s(l, P)$ denotes the semantic similarity of a linked page and a web page. The function of $\mathrm{Sim}_s(l, P)$ is

$$\mathrm{Sim}_s(l, P) = \underset{l_i \in L}{MAX}\, \mathrm{Sim}_s(l, l_i) \tag{18}$$

where $L = \{l_1, l_2, \ldots, l_n\}$ is the link set of P.

For instance, to calculate the link similarity between two items showed in Figure 1, we calculate the semantic similarity between their links as shown in Table 1, where $m = n = 2$:

	Lumia	Surface
iPhone	0.7	0.1
MacBook	0.2	0.6

Table 1. Semantic similarity

Then, we calculate the link similarity between "Apple" and "Microsoft" $\mathrm{Sim}_l(\mathrm{Apple}, \mathrm{Microsoft}) = 0.65$ using Equations (17) and (18).

Furthermore, the accuracy can be improved if we divide a link set into two sets: Link-ins and Link-outs. The similarities of link-in and link-out are calculated respectively:

$$\mathrm{Sim}_I(P_1, P_2) = \frac{\sum_{i=1}^{m} \mathrm{Sim}_s(I_{1i}, P_2) + \sum_{j=1}^{n} \mathrm{Sim}_s(I_{2j}, P_1)}{m+n}, \tag{19}$$

$$\mathrm{Sim}_O(P_1, P_2) = \frac{\sum_{i=1}^{m} \mathrm{Sim}_s(O_{1i}, P_2) + \sum_{j=1}^{n} \mathrm{Sim}_s(O_{2j}, P_1)}{m+n}, \tag{20}$$

where I_i and O_i are the items of link-in set and link-out set of web page P respectively. The similarity based on links is finally calculated as follows:

$$\mathrm{Sim}_l(P_1, P_2) = \frac{1}{2}\mathrm{Sim}_I(P_1, P_2) + \frac{1}{2}\mathrm{Sim}_O(P_1, P_2). \tag{21}$$

4.3 Similarity of Web Pages

This work considers two kinds of similarity, i.e. semantic and link ones. They can be combined to compute the similarity of two web pages as follows:

$$Sim(P_1, P_2) = \alpha \times \mathrm{Sim}_s(P_1, P_2) + \beta \times \mathrm{Sim}_l(P_1, P_2) \tag{22}$$

where $\alpha, \beta \in [0, 1]$ with $\alpha + \beta = 1$, which are specified by users.

5 RECOMMENDATION AND RANKING OF THE RELATED INFORMATION

After calculating the relevance of keywords and similarity of web pages, we can use them to recommend users some useful information, which includes the similar content and the possible information which might be of their interests.

5.1 Main Keywords

There are many keywords on a web page. However, they are not equally important. If one has more relevance than the others, then it can be considered as a main keyword.

For a web page p, we can extract a vector $W = (w_1, w_2, \ldots, w_n)$, which contains the most appearing keywords in a descending order from w_1 to w_n. Then, we can calculate the relevance of each keyword to the others and compute the overall relevance of keyword to find out the main keyword w:

$$M(p) = w \tag{23}$$

where $\sum r_s(w_i, w) = MAX\,[\sum r_s(w_i, w_j)]$, $w, w_i, w_j \in W, i, j \in N_n = \{1, 2, \ldots, n\}$, $r_s(a, b)$ is the relevance between keywords which is presented in Equation (12).

The keyword with the highest overall relevance value denotes the main keyword of the web page. Note that we may extract multiple keywords as the main keywords, if one keyword is not enough from a user's viewpoint.

5.2 Recommendation of Relevant Keywords

We recommend the similar keywords according to their relevance. The keywords with the highest value of semantic relevance are recommended, and considered as the most similar concepts to a user-specified keyword.

Sometimes, a user provides a group of keywords $W = \{w_1, w_2, \ldots, w_m\}$, $K = \{k_1, k_2, \ldots, k_n\}$ is set of all keywords in the Internet. We propose the following function to fetch the most similar keyword for W:

$$S(W) = k \tag{24}$$

where $\sum_{i=1}^{m} r_s(w_i, k) = MAX\left[\sum_{i=1}^{m} r_s(w_i, k_j)\right]$, $w_i \in W$, $k_j \in K$, $r_s(a, b)$ is the relevance between keywords which is presented in Equation (12).

If only one keyword cannot meet the user's need, we can obtain multiple keywords that have the highest values.

Furthermore, we can give the logically relevant keywords according to their confidence values. The keywords with the highest values of confidence are recommended and considered as the most likely keywords that meet the users' interests.

Same to the aforementioned similar keywords, when a user provides a group of keywords, we propose the below function to retrieve concepts:

$$L(W) = k \tag{25}$$

where $\sum_{i=1}^{m} r_l(w_i, k) = MAX\left[\sum_{i=1}^{m} r_l(w_i, k_j)\right]$, $w_i \in W$, $k_j \in K$, $r_l(a, b)$ is the relevance between keywords which is presented in Equation (13).

After the most relevant keyword is found, users can use a search engine to search for the information they want.

5.3 Recommendation of Similar Web Pages

When users locate their focus on a web page, we can recommend them more web pages based on the feature of the web page. Similar to the recommendation of keywords, we can recommend similar web pages from both semantic and link perspectives. As we already have Sim_s and Sim_l, the aforementioned semantic and link similarity can denote these two aspects respectively. We can find out the most semantically relevant web pages by fetching the web pages with the highest semantic similarity. The link similarity is the same, except that we can also combine the semantic information with it to attain a better result.

6 EXPERIMENTAL EVALUATION

To evaluate the performance of the proposed method, two real datasets were used in the tests:

1. The CP (Crawled Pages) dataset is a set of web pages crawled from the Internet. We used a spider to crawl 100 000 web pages from the Internet, and extracted

most appearing 20 keywords on each page to calculate their relevance with other keywords.

2. The CiteSeer [24] dataset is a dataset about science papers which can be downloaded from the Internet. CiteSeer has the information of papers which contains keywords, references and classifications.

After calculating the keyword relevance of these datasets, we saved the result into a database. Then we used the results of the relevance to calculate the similarity of every pair of web pages in every dataset.

6.1 Complexity Analysis

We measure the algorithm complexity from both space and time aspects.

Given n web pages, the theoretical space complexity is n^2. However, we can use the pruning technique to prune most of the records. We can modify the threshold of similarity to control the number of records, or give a fixed number of each web page.

The time complexity depends on two factors: the number of keywords and the number of hyperlinks. We divide the calculation into two parts: semantic similarity and link similarity. They can be calculated in parallel. The time complexity to compute semantic similarity is $O(n^2k^2)$, where k is the average number of keywords a web page has. Computing link similarity takes $O(n^2d^2)$, where d is the average number of hyperlinks. As aforementioned, the time complexity of SimRank is $O(kn^2d^2)$. Compared with SimRank, the proposed method is much more efficient. Also it does not need iterative computations, and thus the similarities can be individually computed by different computers.

6.2 Evaluation Methods

We use two different methods to evaluate the aforementioned two kinds of datasets. The first one uses a subjective metric, and the other uses an objective metric.

6.2.1 The CP Dataset

Because of we do not have the classification of the web pages in the CP dataset, we cannot evaluate the accurate rate of the relevance of the keywords and the similarity of the web pages. So, we selected some related keywords and web pages manually to judge whether they are eligible for recommendations.

6.2.2 The CiteSeer Datasets

The CiteSeer dataset has the information about the classification of web pages, so we can use the objective method to evaluate the results on this dataset.

Let $top_{A,N}(v)$ denote the set of top N similar objects to object v retrieved by algorithm A (A can be any algorithm to computing similarity between web pages like SimRank or our method), and similar(v) denotes the set of papers whose class labels are the same as that of v. We use precision, recall and F-measure to evaluate the performance of algorithm A:

$$\text{precision}_{A,N}(v) = \frac{|\text{top}_{A,N}(v) \cap \text{similar}(v)|}{|\text{top}_{A,N}(v)|}, \tag{26}$$

$$\text{recall}_{A,N}(v) = \frac{|\text{top}_{A,N}(v) \cap \text{similar}(v)|}{N}, \tag{27}$$

$$F\_\text{score}_{A,N}(v) = 2 \cdot \frac{\text{precision}_{A,N}(v) \cdot \text{recall}_{A,N}(v)}{\text{precision}_{A,N}(v) + \text{recall}_{A,N}(v)}. \tag{28}$$

Moreover, we use the average of these metric over this dataset to measure the overall quality:

$$\Delta_{\text{precision}}(A, N) = \frac{\sum_{v \in V} \text{precision}_{A,N}(v)}{||V||}, \tag{29}$$

$$\Delta_{\text{recall}}(A, N) = \frac{\sum_{v \in V} \text{recall}_{A,N}(v)}{||V||}, \tag{30}$$

$$\Delta_{F\_\text{score}}(A, N) = \frac{\sum_{v \in V} F\_\text{score}_{A,N}(v)}{||V||}. \tag{31}$$

6.3 Experimental Results

6.3.1 Results on the CP Data Set

We manually selected some examples to show the effort of our method on the CP set. We do not calculate the similarity on this dataset by SimRank. The time complexity of SimRank is in the same order as our method, however, SimRank uses recursive computing, and the number of recursion is at least 10 times, so SimRank will theoretically spend at least 5 times more time than our method. To calculate the similarities among 100 000 pages by our method, we spent nearly one month. So the time cost of SimRank on the CP dataset is unacceptable. Furthermore, it is difficultly to find a path between two random web pages, so it is hard to calculate the similarity between web pages by SimRank.

Relevance of Keywords. We extracted 54 280 words from 100 000 crawled web pages, and calculated their relevance respectively. A stopword set is given for pruning the clearly non-keywords like "is". The number of words is limited, and thus the volume of relevance data in a database is acceptable. We used the support and confidence element to calculate the relevance of keywords. According to empirical

value, we set $\alpha = \beta = 0.5$. We obtain 3 555 718 records finally after deleting the redundant information.

Table 2 illustrates the five most relevant keywords given some keywords respectively.

Keyword	Relevant Words	Relevance
NBA	NFL	0.152
	basketball	0.136
	NCAA	0.135
	football	0.133
	scout	0.110
iPhone	apple	0.145
	iPad	0.137
	iOS	0.087
	maps	0.072
	store	0.060
Google	hub	0.272
	search	0.252
	apple	0.145
	facebook	0.144
	twitter	0.128

Table 2. Relevant words

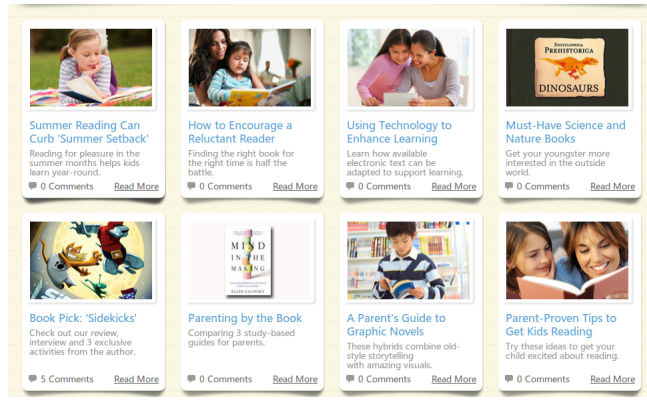
Similarity of Web Pages. We calculated both semantic and link similarity for each pair of web pages. It is easy to store the similarity data among 100 000 pages, but storing the similarity data among the whole pages in Internet will occupy much more storage space. Thus we give a threshold to filter the data. The similarity values being less than the threshold are deleted from the database to save the storage space.

There are some advantages to use the proposed method. Different from SimRank, the similarity among keywords needs no iterative computations, thereby enabling its parallel computation.

We use an inverted index to store the keywords' ID and the web pages' ID. Because the results are related to the keywords only, the computation of each web page's similarity can be performed individually, or in parallel.

Some examples are given below to illustrate the similarity of web pages.

Figure 3 shows that a web page talking about book readings for parents is the most relevant to another web page about book readings on Christmas. Both pages belong to class "reading" but focus on different themes. Figure 4 shows that the news about high school sports is the most relevant to the roster of a college basketball team. In both cases, the web pages are relevant but not simply similar. Considering that the data volume is not enough to represent the relationship between all the web pages completely, this result can be improved by crawling more data from the Internet.



a)

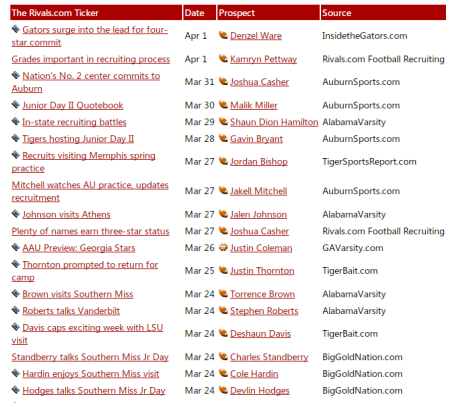
b)

Figure 3. A recommendation on books

6.3.2 Results on the CiteSeer Dataset

The results are presented in Figure 5.

Precision. The proposed integrated similarity performs better than SimRank on precision. By introducing the semantic similarity, our method improves the performance in the range from 12 % to 17 %.



The Rivals.com Ticker	Date	Prospect	Source
Gators surge into the lead for four-star commit	Apr 1	Denzel Ware	InsidetheGators.com
Grades important in recruiting process	Apr 1	Kamryn Pettway	Rivals.com Football Recruiting
Nation's No. 2 center commits to Auburn	Mar 31	Joshua Casher	AuburnSports.com
Junior Day II Quotebook	Mar 30	Malik Miller	AuburnSports.com
In-state recruiting battles	Mar 29	Shaun Dion Hamilton	AlabamaVarsity
Tigers hosting Junior Day II	Mar 28	Gavin Bryant	AuburnSports.com
Recruits visiting Memphis spring practice	Mar 27	Jordan Bishop	TigerSportsReport.com
Mitchell watches AU practice, updates recruitment	Mar 27	Jakell Mitchell	AuburnSports.com
Johnson visits Athens	Mar 27	Jalen Johnson	AlabamaVarsity
Plenty of names earn three-star status	Mar 27	Joshua Casher	Rivals.com Football Recruiting
AAU Preview: Georgia Stars	Mar 26	Justin Coleman	GAVarsity.com
Thornton prompted to return for camp	Mar 25	Justin Thornton	TigerBait.com
Brown visits Southern Miss	Mar 24	Torrence Brown	AlabamaVarsity
Roberts talks Vanderbilt	Mar 24	Stephen Roberts	AlabamaVarsity
Davis caps exciting week with LSU visit	Mar 24	Deshaun Davis	TigerBait.com
Standberry talks Southern Miss Jr Day	Mar 24	Charles Standberry	BigGoldNation.com
Hardin enjoys Southern Miss visit	Mar 24	Cole Hardin	BigGoldNation.com
Hodges talks Southern Miss Jr Day	Mar 24	Devlin Hodges	BigGoldNation.com

a)

Scout.com College Basketball Recruiting Search

Modify Existing Search OR New Search

Page 1 >>

Pos	Nat'l Rank	Nat'l Rating	Name School, Hometown	Video	HT/ WT/ PPG	Schools of Interest
PG	1	★★★★★	**Andrew Harrison** (Travis HS) **Richmond, TX**		6-5/205	Committed to **Kentucky**
SG	1	★★★★★	**Aaron Harrison** (Travis HS) **Richmond, TX**		6-5/205	Committed to **Kentucky**
SF	1	★★★★★	**Andrew Wiggins** (Huntington Prep) **Huntington, WV**		6-8/205	Florida State, Kansas, Kentucky, North Carolina
PF	1	★★★★★	**Julius Randle** (Prestonwood Christian) **Plano, TX**		6-8/225	Committed to **Kentucky**
C	1	★★★★★	**Dakari Johnson** (Montverde Academy) **Montverde, FL**		6-10/240	Committed to **Kentucky**
PG	2	★★★★★	**Kasey Hill** (Montverde Academy) **Montverde, FL**		6-0/160	Committed to **Florida**
SG	2	★★★★★	**Robert Hubbs** (Dyer County HS) **Newbern, TN**		6-4/170	Committed to **Tennessee**
SF	2	★★★★★	**Jabari Parker** (Simeon Vocational HS) **Chicago, IL**		6-7/215	Committed to **Duke**
PF	2	★★★★★	**Aaron Gordon** (Archbishop Mitty HS) **San Jose, CA**		6-8/215	Arizona, Kentucky, Oregon, Washington
C	2	★★★★★	**Marcus Lee** (Deer Valley HS) **Antioch, CA**		6-9/220	Committed to **Kentucky**
PG	3	★★★★★	**Terry Rozier** (Hargrave Military Academy) **Chatham, VA**		6-1/170	Committed to **Louisville**

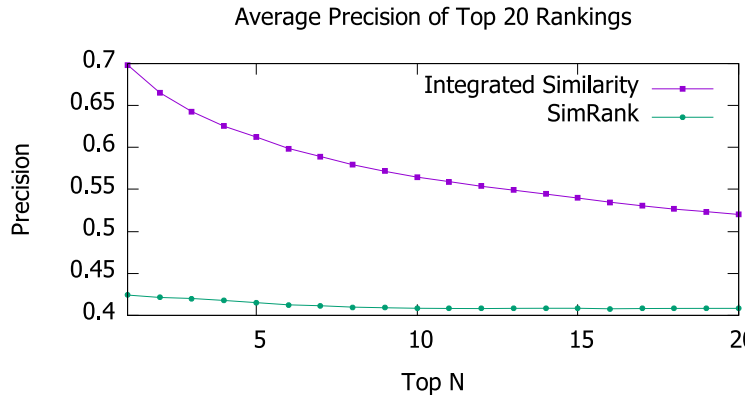
b)

Figure 4. A recommendation on sports

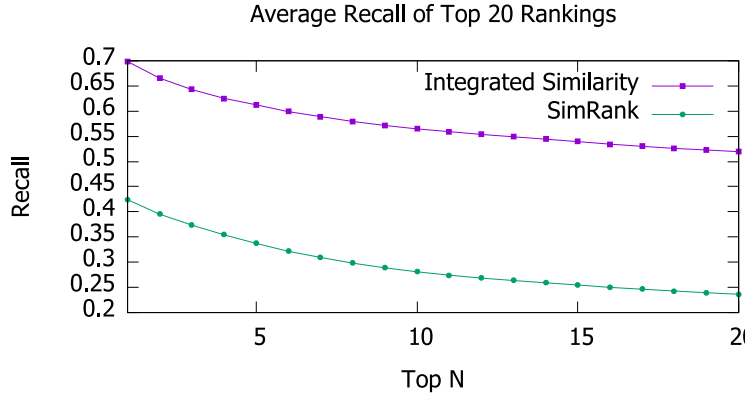
Recall. Integrated similarity performs much better than SimRank on recall, because our method can calculate the similarity by latent semantic links. The performance gains in the range from 27 % to 29 %.

F-Score. The result of the proposed integrated similarity is also much better than SimRank. The F-Score increases in the range from 26 % to 27 %.

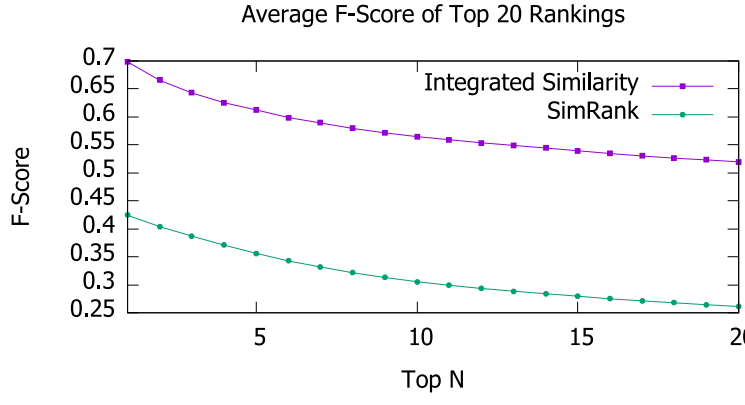
Compared to SimRank, the method of integrated similarity improves the accuracy of recommendation while giving users more related objects. Moreover, as what mentioned before, recursive computing will spend much more time than our



a)



b)



c)

Figure 5. Average precision, recall and F-score on the CiteSeer dataset

method, and the time cost is unacceptable on the scale of Internet. Thus we made a tradeoff between time cost and accuracy of link similarity to let both of them at the acceptable level.

7 RELATED WORKS

Prior methods are proposed to compute the similarity of two objects in both content and link aspects. The content similarity of items have widely been applied to data mining, while the link similarity of items to the field of document retrieval. Statistical methods are often used to discover the frequent patterns. Some methods focus on the link similarity of web pages.

Yang et al. propose a method to search for interesting association rules by keywords [7]. It combines semantic and statistical information together to measure the relation between two objects. Like conventional methods, this method neglects the effect of links on the web pages. It needs an ontology to reflect the semantic relationship between objects, which is hard to build for the keywords on the Internet.

Zhao et al. propose P-Rank [5]. Different from SimRank, P-Rank introduces the out-link relationship to gain more information. It is reasonable to consider the out-links that have useful information. However, it also takes spams into account, so the accuracy will decrease by this reason. Furthermore, it considers only the similarity of items by the links of the same item but ignores the relationship among similar items. P-Rank has the same high computational complexity as SimRank does.

BlockSimRank [6] divides the original graph into m blocks. Each block represents a domain consisting of the similar objects. Objects in the same block are usually more similar than those from different blocks. Its complexity is lower than that of SimRank's. However, it needs preprocessing before calculation. How to divide the block optimally is another question. It may need a domain expert's help, or use some clustering algorithm to do so.

Lin et al. propose PageSim to calculate the similarity among web pages [9]. A web page is similar to another if they have hyperlinks with each other. In this method, the web pages are not equally important. The hyperlinks from important web pages are also important. The importance of web pages can be measured by other methods like PageRank [16]. This method uses an iterative algorithm to compute the similarity between each pair of web pages. Same as SimRank, this method does not take the semantic similarity into account. Moreover, its complexity is the same as SimRank's.

MatchSim [21] can calculate the similarity of web pages. Its idea is the same as SimRank's, but it takes only direct neighbours into consideration to reduce the complexity. This method overcomes a loophole of computation in SimRank, thereby improving the computational efficiency and accuracy. However, it also ignores the semantic similarity among items. Furthermore, different from our method, it has not pruned the useless links and take this useless links into account as its shortcomings.

Li et al. propose Single-Pair SimRank [25] to improve the computation efficiency of SimRank. It costs less time when we only need to assess similarity of one or a few node-pairs. However, besides that it also ignores the semantic similarity between objects, this method does not meet the need of information recommendation on Internet which is needed to compute all pairs of web pages.

Du et al. propose Probabilistic SimRank [26] to compute similarity between objects in uncertain graphs. It defines the similarity measure on probabilistic graphs. It solves the problem of computation on uncertain graph. Same as SimRank, it can not overcome the aforementioned drawbacks of SimRank, too.

Table 3 shows a comparison between the proposed method and the aforementioned methods.

Method	Semantic	Prune Spams	Complexity
SimRank	No	Yes	High
P-Rank	No	No	High
BlockSimRank	No	Yes	Low
PageSim	No	No	High
MatchSim	No	No	High
Single-Pair SimRank	No	No	Low
Probabilistic SimRank	No	Yes	High
Our Method	Yes	Yes	Low

Table 3. Comparison with representative methods

To the best of our knowledge, the proposed method is the only one that takes into account semantic similarity and spam pruning with a low computational complexity.

8 CONCLUSION AND FUTURE WORKS

In this paper, we have proposed a novel method to discover relevant information on the Internet. We calculate the relevance of keywords, and use it to calculate the semantic similarity among web pages. Moreover, logical relevance of keywords is used to prune the useless links of web pages. We consider the hyperlink as the latent logical relationship, and combine both semantic and link similarity to compute the relationship among web pages. Furthermore, a method to recommend latently relevant keywords or web pages also is proposed in this paper. According to the experiment, our method performs better than SimRank on both the accuracy and number of returned objects.

In the future, we are planning to improve the proposed method of recommendation to make the related information more accurate, especially the link similarity value. We will also research how to use some clustering algorithm to cluster web pages into different classes, in order to recommend some classes of websites instead of only some relevant web pages.

Acknowledgement

This work was supported by the Major Research Plan of the National Natural Science Foundation of China under Grant No. 91218301 and HongKong, Macao and Taiwan Science and Technology Cooperation Program of China under Grant No. 2013DFM10100.

REFERENCES

[1] Jeh, G.—Widom, J.: SimRank: A Measure of Structural-Context Similarity. Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02), 2002, pp. 538–543.

[2] Shannon, C. E.: A Mathematical Theory of Communication. The Bell System Technical Journal, Vol. 27, 1948, pp. 379–424, pp. 623–656.

[3] Singhal, A.: Modern Information Retrieval: A Brief Overview. IEEE Data Engineering Bulletin, Vol. 24, 2001, No. 4, pp. 35–43.

[4] Agrawal, R.—Imielinski, T.—Swami, A.: Mining Association Rules Between Sets of Items in Large Databases. Proceedings of the International Conference on Management of Data (SIGMOD '93), 1993, pp. 207–216.

[5] Zhao, P. X.—Han, J. W.—Sun, Y. Z.: P-Rank: A Comprehensive Structural Similarity Measure over Information Networks. Proceeding of the 18th ACM Conference on Information and Knowledge Management (CIKD '09), 2009, pp. 553–562.

[6] Li, P.—Cai, Y. Z.—Liu, H. Y.—He, J.—Du, X. Y.: Exploiting the Block Structure of Link Graph for Efficient Similarity Computation. Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD '09), 2009, pp. 389–400.

[7] Yang, G. F.—Mabu, S.—Shimada, K.—Hirasawa, K.: A Novel Evolutionary Method to Search Interesting Association Rules by Keywords. Expert System with Applications, 2011, pp. 13378–13385.

[8] Mei, Q.—Xin, D.—Cheng, H.—Han, J. W.: Generating Semantic Annotations for Frequent Patterns with Context Analysis. Proceedings of the 12th ACM International Conference on Knowledge Discovery and Data Mining (KDD '06), 2006, pp. 337–346.

[9] Lin, Z. J.—King, I.—Lyu, M. R.: PageSim: A Novel Link-Based Measure of Web Page Similarity. Proceedings of the 15th International Conference on World Wide Web (WWW '06), 2006, pp. 1019–1020.

[10] Liu, H. Y.—He, J.—Zhu, D.—Ling, X.—Du, X. Y.: Measuring Similarity Based on Link Information: A Comparative Study. IEEE Transactions on Knowledge and Data Engineering, Vol. 25, 2012, No. 12, pp. 2823–2840.

[11] Bishop, C.: Pattern Recognition and Machine Learning. Springer New York, 2006.

[12] Liu, H.—Bao, H.—Xu, D.: Concept Vector for Similarity Measurement Based on Hierarchical Domain Structure. Computing and Informatics, Vol. 30, 2011, pp. 881–900.

[13] Small, H.: Co-Citation in the Scientific Literature: A New Measure of the Relationship Between Two Documents. Journal of the American Society for Information Science, Vol. 24, 1973, No. 4, pp. 265–269.

[14] Kessler, M. M.: Bibliographic Coupling Between Scientific Papers. American Documentation, Vol. 14, 1963, No. 1, pp. 10–25.

[15] Amsler, R.: Applications of Citation-Based Automatic Classification. Linguistic Research Center, 1972.

[16] Brin, S.—Page, L.: The Anatomy of a Large Scale Hypertextual web Search Engine. Computer Networks ISDN Systems, Vol. 30, 1998, No. 1-7, pp. 107–117.

[17] Dean, J.—Henzinger, M. R.: Finding Related Pages in the World Wide Web. Computer Networks, Vol. 31, 1999, No. 11-16, pp. 1467–1479.

[18] Steinberger, J.—Jezek, K.: Evaluation Measures for Text Summarization. Computing and Informatics, Vol. 28, 2009, pp. 251–275.

[19] Paliwal, A. V.—Shafiq, B.—Vaidya, J.—Xiong, H.—Adam, N.: Semantics Based Automated Service Discovery. IEEE Transcations on Services Computing, Vol. 5, 2012, No. 2, pp. 260–275.

[20] Xin, D.—Yan, J.—Cheng, H.: Mining Compressed Frequent-Pattern Sets. Proceedings of VLDB '05, 2005, pp. 709–720.

[21] Lin, Z.—Lyu, M. R.—King, I.: Matchsim: A Novel Neighbor Based Similarity Measure with Maximum Neighborhood Matching. Proceeding of the 18th ACM Conference on Information and Knowledge Management (CIKM '09), 2009, pp. 1613–1616.

[22] Bouchiha, D.—Malki, M.—Alghamdi, A.—Alnafjan, K.: Semantic Web Service Engineering: Annotation Based Approach. Computing and Informatics, Vol. 31, 2012, No. 6+, pp. 1575–1595.

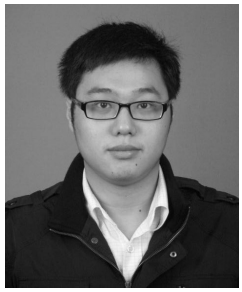
[23] Wu, J.—Zhao, H.—Li, Y.—Deng, S. G.: Web Service Discovery Based on Ontology and Similarity of Words. Chinese Journal of Computers, 2005, pp. 595–602.

[24] CiteSeer Dataset: `http://linqs.cs.umd.edu/projects//projects/lbc/index.html`.

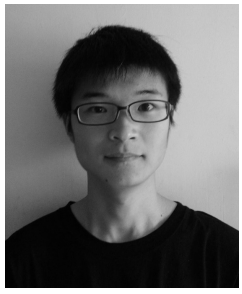
[25] Li, P.—Liu, H. L.—Yu, J. X.—He, J.—Du, X. Y.: Fast Single-Pair SimRank Computation. Proceedings of the 2010 SIAM International Conference on Data Mining, 2010, pp. 571–582.

[26] Du, L. X.—Li, C. P.—Chen, H.—Tan, L. W.—Zhang, Y. L.: Probabilistic SimRank Computation over Uncertain Graphs. Information Sciences, Vol. 295, 2015, pp. 521–535.

[27] Zhao, Q.—Wang, C.—Jiang, C. J.: HSim: A Novel Method on Similarity Computation by Hybrid Measure. Proceedings of the 6th International Conference on Information and Communication Systems, 2015, pp. 160–165.

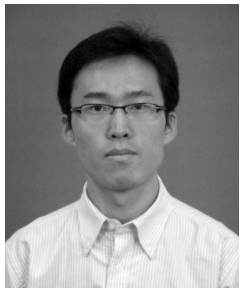


Qin Zhao is currently a Ph.D. student of computer software and theory at Tongji University. He received his B.Sc. degree in computer science and technology from Shanghai Ocean University, and his M.Sc. degree in software engineering from Tongji University. His research interests include service discovery, data mining, and information retrieval.



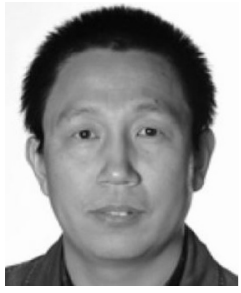
Yuan He received his B.Sc. degree from the Department of Computer Science and Technology, Tongji University. He is currently a Ph.D. student in Department of Computer Science at Tongji University in Shanghai, China. His research interests include topic modeling, text mining, convex optimization and machine learning.

Changjun Jiang is Professor and Supervisor of doctoral students. He was awarded his Ph.D. degree from Institute of Automation, Chinese Academy of Science in 1995 and postdoctoral degree from Institute of Computing Technology, Chinese Academy of Science in 1997. He has published more than 100 papers in domestic and international academic publications, such as "Chinese Science", "IEEE Transactions on Robotics & Automation", "IEEE Transactions on Fuzzy Systems", "International Journal of Computer Mathematics", "International Journal of Computer Systems Science and Engineering", "International Journal of Studies in Informatics Control" and "International Journal of Advances in Systems Science and Applications".



Pengwei Wang received his B.Sc. and M.Sc. degrees in computer science from Shandong University of Science and Technology, Qingdao, China, in 2005 and 2008, respectively, and his Ph.D. degree in computer science from Tongji University, Shanghai, China, in 2013. He finished his postdoctoral research work at the Department of Computer Science, University of Pisa, Italy, in 2015. Currently, he is Assistant Professor with the School of Computer Science and Technology, Donghua University, Shanghai, China. His research interests include service computing, cloud computing, and Petri nets.

Man Qi is Senior Lecturer in the Department of Computing at Canterbury Christ Church University, UK. Her research interests are in the areas of computer graphics, computer animation, multimedia and applications. She is a Fellow of the British Computer Society and also a Fellow of the Higher Education Academy.



Maozhen Li received his Ph.D. from Institute of Software, Chinese Academy of Sciences in 1997. He was a postdoctoral scholar in the School of Computer Science and Informatics, Cardiff University, UK in 1999–2002. He is currently Professor in the School of Engineering and Design at Brunel University, UK. His research interests are in the areas of high performance computing (grid and cloud computing) for big data analysis and intelligent systems. He has over 100 research publications in these areas. He is a Fellow of the British Computer Society.