Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A new focused crawler using an improved tabu search algorithm incorporating ontology and host information*

Jingfa LIU[1,2], Zhen WANG[†‡1,3], Guo ZHONG[1,2], Zhihe YANG[1,2]

[1]*School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou 510006, China*

[2]*Guangzhou Key Laboratory of Multilingual Intelligent Processing, Guangdong University of Foreign Studies, Guangzhou 510006, China*

[3]*China Unicom Central South Research Institute, Changsha 410000, China*

[†]E-mail: 1007427607@qq.com

Received July 22, 2022; Revision accepted Jan. 6, 2023; Crosschecked

Abstract: To solve the problems of incomplete topic description and repetitive crawling of visited hyperlinks in traditional focused crawling methods, in this paper, we propose a novel focused crawler using an improved tabu search algorithm with domain ontology and host information (FCITS_OH), where a domain ontology is constructed by formal concept analysis to describe topics at the semantic and knowledge levels. To avoid crawling visited hyperlinks and expand the search range, we present an improved tabu search (ITS) algorithm and the strategy of host information memory. In addition, a comprehensive priority evaluation method based on Web text and link structure is designed to improve the assessment of topic relevance for unvisited hyperlinks. The experimental results on both tourism and rainstorm disaster domains show that the proposed focused crawlers overmatch the traditional focused crawlers for different performance metric indices.

Key words: Focused crawler; Tabu search algorithm; Ontology; Host information; Priority evaluation
https://doi.org/10.1631/FITEE.2200315 **CLC number:**

1 Introduction

Currently, Internet resources are growing explosively. The data update speed is increasing, and users' needs for web information are becoming more personalized. Traditional search engines can no longer satisfy the needs of customized information, so a focused crawler (Chakrabarti *et al*, 1999; Deng 2020) is presented to collect topical information. Compared with the traditional crawler, the focused crawler can retrieve larger quantities and higher quality topic-relevant webpages. Therefore, in recent years, the focused crawler has attracted the attention of many scholars (Yu and Liu, 2015; Hosseinkhani *et al*, 2021).

At present, focused crawlers face three main issues: topic description, evaluation of the topic relevance of unvisited hyperlinks, and design of crawling strategies. The methods of topic description mainly include topic words (Fei and Liu, 2018), context graphs (CG) (Guan and Luo, 2016; Du *et al*., 2014), and domain ontology (Rani *et al*, 2017; Khan and Sharma, 2016). The topic words are collected through the experience of domain experts, but there is a problem of semantic ambiguity. The construction of CG relies on the user's historical crawling information and may deviate from the topic if the users lack topic-relevant knowledge. Because ontology can describe the specific domain at the semantic and knowledge level, most semantic-based crawlers (Khan and Sharma, 2016; Lakzaei and Shmasfard, 2021) use ontology to describe topics.

The methods of evaluating unvisited hyperlinks include hyperlink structure-based methods and webpage content-based methods. Hyperlink structure-based methods, such as the PageRank algorithm (Yuan *et al*, 2017) and the hyperlink induced topic search (HITS) algorithm

‡ Corresponding author

(Asano et al, 2007), focus on the structure itself and ignore the relevance of the topic, which may cause crawlers "topic drifting". Webpage content-based methods mainly evaluate priorities of unvisited hyperlinks by calculating and analyzing the relevance of the webpage text and the anchor text, such as the fish-search algorithm (Bra *et al*, 1994) and shark-search algorithm (Prakash and Kumar, 2015). These algorithms ignore the characteristics of the global hyperlink structure and only perform well when searching nearby webpages. Most researchers ignore the impact of combining these two methods, and the considered indices are not sufficiently comprehensive.

The crawling strategy determines the order in which hyperlinks with different priorities are visited. The traditional algorithms mainly include the breadth-first search (BFS) (Li *et al*, 2015) and the optimal priority search (OPS) (Rawat and Patil, 2013). The BFS neglects to evaluate the order of hyperlinks during crawling so that it has the worst performance. The OPS only takes the best value of priority into account, and the greedy strategy leads to a greater possibility of falling into a choice of a hyperlink with no prospects. To avoid the inherent flaws of greedy algorithms, many scholars have proposed intelligent focused crawler methods based on metaheuristic strategies. For instance, He *et al*., (2009) proposed a focused crawler strategy based on the simulated annealing algorithm (SA), allowing crawlers to obtain suboptimal hyperlinks for expanding the search range. Yan and Pan (2018) considered users' browsing behavior to optimize genetic operations and proposed a heuristic focused crawler strategy based on an improved genetic algorithm (GA). Tong (2008) considered the distribution characteristics of website resources and proposed a heuristic focused crawler strategy based on an adaptive dynamic evolutionary particle swarm optimization (PSO) algorithm. Xiao and Chen (2018) analyzed the priority of crawlers in global crawling and proposed a focused crawler strategy based on the gray wolf optimization algorithm (GWO). Recently, Liu *et al*. (2022a) proposed a heuristic focused crawler strategy combining ontology learning and the multiobjective ant colony algorithm (OLMOACO). In OLMOACO, a method of the nearest farthest candidate solution (NFCS) combined with fast nondominated sorting was used to select a set of Pareto-optimal hyperlinks and guide the crawlers' search directions. Liu *et al*. (2022b) built a multiobjective optimization model for evaluating unvisited hyperlinks based on Web text and link structure and proposed a focused crawler strategy combining the Web space evolution algorithm and domain ontology (FCWSEO). Both the OLMOACO and FCWSEO algorithms guide the crawling direction by building a multiobjective optimization model to select the next hyperlinks to visit. However, the OLMOACO and FCWSEO algorithms suffer from the tendency to crawl the visited hyperlinks under a few hosts, which causes the crawler to converge prematurely.

To overcome the above issues, in this paper, we propose a novel focused crawler based on an improved tabu search strategy by combining ontology and host information (FCITS_OH). The main contributions of this paper are as follows:

(1) Two domain ontologies of tourism and rainstorm disasters based on formal concept analysis (FCA) are constructed to describe topics at the semantic and knowledge levels.

(2) In the crawling process, an improved tabu search (ITS) strategy with host information is presented to select the next hyperlink, where the modified tabu object and acceptance principles are used to avoid crawling the visited hyperlinks in the focused crawler, and the host information memory of hyperlinks is proposed to prevent the crawler from cycling under a few hosts, which controls the convergence speed of the algorithm.

This paper is organized as follows: Section 2 gives the topic description involving the construction of the domain ontology and the computation of the topic semantic weighted vector. In Section 3, the comprehensive priority evaluation method of hyperlinks is presented. Section 4 proposes three focused crawler strategies based on an improved tabu search algorithm with ontology and host information. Experimental results and analysis on two domains of tourism and rainstorm disaster are displayed in Section 5. Finally, the conclusion is presented in Section 6.

2 Topic description

In this paper, we use domain ontology to describe the topic. This section first introduces the construction process of domain ontology based on the FCA method and then computes the topic semantic weighted vector based on domain ontology semantics.

2.1 Ontology construction

Formal concept analysis (FCA) (Zhu *et al.*, 2017) is a semiautomatic method of constructing ontology, whose main data structure is the concept lattice. The process of generating a concept lattice is concept clustering, which formalizes the hierarchical relationship between the concepts. The detailed steps of ontology construction in this paper are as follows. (1) Select five keywords for the determined domain and search for keywords through search engines such as Baidu and Google to obtain the top 50 webpages of each search engine. (2) Use the tool IK-Analyser (Wang and Meng., 2014) to perform word segmentation. (3) Extract document sets and term sets that describe the topic. (4) Build a document-term matrix, which is input into the tool ConExp[1]to generate a concept lattice and obtain a Hasse diagram. (5) Describe the hierarchical relations among concepts by ontology web language (OWL)[1]. (6) Visualize the ontology by Protégé[2].

Applying the above method, we constructed a tourism ontology and rainstorm disaster ontology. The tourism ontology includes seven branches: tourist attractions, tourism purpose, accommodation, service agencies, tourism routes, means of transportation, and tourist. The whole ontology includes 61 concepts and a 7-level hierarchical structure. The rainstorm disaster ontology includes three branches: disaster management, secondary disaster, and disaster grade. The whole ontology contains 50 concepts and a 6-level hierarchical structure.

2.2 Topic semantic weighted vector

Referring to the literature (Liu *et al.*, 2022a), we consider the five impact factors, including semantic distance (IF_{Dis}), concept density (IF_{Den}), concept depth (IF_{Dep}), concept coincidence degree (IF_{Coi}), and concept semantic relationship (IF_{Rel}), to measure the topic semantic similarity between concepts based on the constructed domain ontology. The calculation formula of the semantic similarity value $Sem(C_1, C_2)$ between concept C_1 and concept C_2 is shown as Eq. (1).

$$Sem(C_1,C_2)=k_1\times IF_{Dis}+k_2\times IF_{Den}+k_3\times IF_{Dep}+k_4\times IF_{Coi}+k_5\times IF_{Rel} \quad (1)$$

Here, the adjustment factors k_1, k_2, k_3, k_4, and $k_5 \geq 0$ and satisfy $k_1 + k_2 + k_3 + k_4 + k_5 = 1$. To obtain the topic semantic weighted vector, we first determine a topic concept C, which is tourism or rainstorm disaster in this paper. Suppose the topic word set $T = (t_1, t_2, \ldots, t_i, \ldots, t_n)$. Calculate the semantic similarity between each topic word t_i and topic concept C based on Eq. (1) to obtain the corresponding topic semantic weighted vector $W_T = \{w_{t1}, w_{t2}, \ldots, w_{ti}, \ldots, w_{tn}\}$, where w_{t_i} is the weight of the i-th topic word t_i in the set T. Thus, the topic semantic weighted vector between topic concept C and topic word vector T is shown as Eq. (2).

$$W_T = (Sem(C, t_1), Sem(C, t_2), \ldots, Sem(C, t_n)) \quad (2)$$

3 Comprehensive evaluation method of hyperlinks

We use the vector space model (VSM) (Farag *et al.*, 2018) to calculate the topic relevance of a webpage and propose a comprehensive priority evaluation method for predicting the topic relevance of the unvisited hyperlinks.

3.1 Topic relevance of webpages

Most webpages are represented as HTML files, and the content of the webpage is presented in the form of tags. Different positions of tags display different importance degrees in the entire webpage. We choose main tags from HTML files and divide them into five groups. Each tag group is assigned a specific weight W_k, k=1, 2, 3, 4, and 5, as shown in Table 1.

Table 1 Division of labels and their weights

Groups	Labels	Meanings	W_k
Group 1	<title>、<keyword>、<description>、<h1>	title, keyword, description, first-level headline	2
Group 2	<h2>、<h3>	secondary-level headline, third-level headline	1.5
Group 3	<h4>、<h5>、	fourth-level headline, fifth-level headline, bold text	1.2

[1] https://sourceforge.net/projects/conexp/
[2] https://www.w3.org/TR/owl-features/
[3] https://protege.stanford.edu/

Group 4	<p>、<td>、	body information	1.0
Group 5	other labels	nonbody information	0.2

We map the webpage text into a webpage feature vector $D = \{d_1, d_2,..., d_i,..., d_n\}$ and obtain the corresponding webpage feature weighted vector $W_D = \{w_{d1}, w_{d2}, \ldots, w_{di}, \ldots, w_{dn}\}$, where w_{di} represents the weight of the i-th feature word and is computed by the improved term frequency-inverse document frequency (TF-IDF) (Wu *et al*., 2017). Its expression is shown as Eq. (3).

$$w_{d_i} = \sum_{k=1}^{K} tf_{i,k} \times W_k = \sum_{k=1}^{K} \left(\frac{f_{i,k}}{\max f_{i,k}} \times W_k\right) \tag{3}$$

Here, max $f_{i,k}$ represents the maximum term frequency of the i-th topic word in all label groups, and W_k represents the weight of the k-th group of labels. We adopt the vector space model (VSM) (Farag *et al*., 2018) to calculate the topic relevance $R(p)$ of webpage p. Its expression is shown in Eq. (4).

$$R(p) = Sim(T, D) = \frac{W_T \times W_D}{\|W_T\| \times \|W_D\|} = \frac{\sum_{i=1}^{n} \left(w_{t_i} \times w_{d_i}\right)}{\sqrt{\sum_{i=1}^{n} w_{t_i}^2} \times \sqrt{\sum_{i=2}^{n} w_{d_i}^2}} \tag{4}$$

VSM is a well-known measure of cosine and transforms a language problem into a mathematical problem. The cosine similarity between two vectors is considered the similarity of the text related to the given topic. When the angle between two vectors is equal to 0°, the relevance between them is maximum and equals 1, indicating that they are the most relevant. When the angle is equal to 90°, the relevance is minimal and equals 0, indicating that they are irrelevant. Assume that the threshold of the webpage topic relevance is α. If $R(p) > \alpha$, then webpage p is considered to be topic-relevant.

3.2 Topic relevance of anchor text

The anchor text usually has only a few words or phrases, but it is an important resource to predict the relevance of the webpage to which the hyperlink points. Generally, the TF-IDF (Wu *et al*., 2017) model is used to evaluate the importance of keywords. However, it is not comprehensive to use term frequency (TF) to measure the importance of a word in the whole anchor text. Therefore, we use the improved BM25 model (Wu, 2018) to evaluate the importance of keywords in anchor text. It retains the important indicator of IDF in the TF-IDF model and improves the computation of TF. The BM25 algorithm is generally used to evaluate the relevance of words and documents. In this paper, we use the BM25 algorithm to obtain the weight of words in the anchor text. The weight w_{a_i} of the i-th topic word in the anchor text is calculated as Eq. (5).

$$\begin{aligned} w_{a_i} &= IDF(N_i) \cdot \sum_{i=1}^{m} \frac{f_{i,j}(k+1)}{f_{i,j} + k \cdot (1 - b + b\dfrac{dl_j}{avgdl})} \\ &= \log_a\left(\frac{N}{N_i} + 0.01\right) \cdot \sum_{j=1}^{m} \frac{f_{i,j}(k+1)}{f_{i,j} + k \cdot (1 - b + b\dfrac{dl_j}{avgdl})} \end{aligned} \tag{5}$$

Here, N is the number of crawled webpages, N_i denotes the number of webpages containing the i-th topic word, and a>1. m represents the number of webpages containing the anchor text of the considered hyperlink. k=2 and b=0.75 represent adjustment factors. dl_j represents the length of the j-th webpage (i.e., the number of words) containing the anchor text, *avgdl* is the average length of all crawled webpages, and $f_{i,j}$ denotes the frequency of the i-th topic word in the anchor text located in the j-th webpage. After obtaining the anchor text feature weighted vector $W_A = \{w_{a1}, w_{a2},...,w_{ai},...,w_{an}\}$, we calculate the cosine similarity between the topic semantic weighted vector W_T and the anchor text feature weighted vector W_A to obtain the topic relevance $R(A_l)$ of anchor text A_l. The topic relevance of the anchor text A_l is computed as Eq. (6).

$$R(A) = Sim(T, A) = \frac{W_T \times W_A}{\|W_T\| \times \|W_A\|} = \frac{\sum_{i=1}^{n} \left(w_{t_i} \times w_{a_i}\right)}{\sqrt{\sum_{i=1}^{n} w_{t_i}^2} \times \sqrt{\sum_{i=1}^{n} w_{a_i}^2}} \tag{6}$$

where $A=\{a_1, a_2,..., a_i,..., a_n\}$ denotes the anchor text feature set.

3.3 Improved PageRank value computation

The PageRank algorithm is an essential algorithm for evaluating unvisited hyperlinks. For a webpage p, the traditional calculation formula of the PageRank (PR) value is Eq. (7).

$$PR(p) = (1 - d) + d \times \sum_{i=1}^{h} \frac{PR(p_i)}{C(p_i)} \tag{7}$$

Here, d is the damping factor and is set to 0.85, h represents the total number of all in-links of webpage p, p_i is the i-th in-link webpage of webpage p, $PR(p_i)$ denotes the PR value of webpage p_i, and $C(p_i)$ represents the total number of out-links of webpage p_i. To avoid the topic drifting of traditional PR calculation, by referring to paper (Ma *et al*., 2016), we integrate the anchor text topic relevance into the PR value calculation and propose an improved PR value calculation method for webpage p, which is shown as Eq. (8).

$$PR(p)=(1-d)+d\times\sum_{i=1}^{h}\left[\frac{PR(p_i)}{C(P_i)}\times(1+\omega\times R(A_i))\right] \quad (8)$$

Here, ω represents an adjustment factor and is set to 0.6 in this paper, and $R(A_i)$ represents the topic relevance of the anchor text A_i of the i-th in-link of webpage p (see Section 3.2).

3.4 Topic relevance evaluation of hyperlinks

A comprehensive priority evaluation method is given to evaluate the topic relevance of unvisited hyperlink l. Its expression is shown in Eq. (9).

$$P(l)=r_1\times PR(p_l)+r_2\times\frac{1}{m}\sum_{i=1}^{m}R(p_i)+r_3\times R(A_l) \quad (9)$$

Here, r_1, r_2, and r_3 represent weighted factors and satisfy $r_1+r_2+r_3=1$. $P(l)$ represents the comprehensive priority value of the unvisited hyperlink l, $R(A_l)$ represents the topic relevance of the anchor text A_l of hyperlink l, $R(p_i)$ represents the topic relevance of webpage p_i that contains hyperlink l, and m is the number of webpages containing hyperlink l. $PR(p_l)$ is the PR value of webpage p_l containing hyperlink l. To filter irrelevant hyperlinks, we set a comprehensive priority threshold β. If $P(l)\geq\beta$, the unvisited hyperlink l is considered topic-relevant and is added to the waiting queue (Q_{wait}).

4 Focused crawler based on tabu search with ontology and host information

In this section, we first introduce the tabu search (TS) algorithm and subsequently propose the improved tabu search (ITS) algorithm by modifying the tabu object and acceptance principles. Finally, by incorporating the domain ontology and the host information memory into the focused crawler strategy based on ITS, a new focused crawler using an improved tabu search algorithm with ontology and host information (FCITS_OH) algorithm is proposed.

4.1 Tabu search algorithm

The tabu search (TS) algorithm was first proposed by Fred Glover. The TS algorithm (Liu *et al.*, 2021) is a random heuristic algorithm based on local search in essence. It generates some new candidate solutions in the neighborhood of the current solution. The basic flow of TS is as follows: (1) Given an initial solution, select some candidate solutions from the neighborhood of the current solution. (2) If the objective function value of the optimal candidate solution is better than the objective function value of the current optimal solution, its tabu property will be ignored. Displace the current solution and the current optimal solution with the optimal candidate solution. Add it into the tabu list and simultaneously update the term of each object in the tabu list. (3) Otherwise, select the nontabu optimal solution from the candidate solutions as the new current solution, add it into the tabu list and update the term of each object in the tabu list. (4) Repeat the above process until the algorithm meets the ending condition. The TS algorithm involves some related elements, such as the objective function, neighborhood, tabu list, and aspiration criterion, which will directly affect the optimization performance of the algorithm.

4.2 Objective function

The objective function is also called the fitness function, which is used to compute the objective value of the solution. In the focused crawler, the objective function is expressed by the comprehensive priority of hyperlink l (see Eq. (9)), and $P(l)$ represents the objective function value..

4.3 Neighborhood set and extended neighbor-hood set

Definition 1. Neighborhood set. The set of all hyperlinks in the webpage to which the current hyperlink *Plink* points is called the neighborhood set of *Plink*, denoted as $N(Plink)$.

Definition 2. Candidate neighborhood set. The set of hyperlinks with a comprehensive priority higher than the threshold β, located in the webpage to which the current hyperlink *Plink* points, is called the candidate neighborhood set of *Plink*, denoted as $C(Plink)$. Obviously, $C(Plink)\subseteq N(Plink)$.

Definition 3. Extended neighborhood set. The set of hyperlinks whose comprehensive priority is higher than the threshold β in the webpage where the current hyperlink *Plink* is located is called the extended neighborhood set of *Plink*, denoted as $E(Plink)$.

In the entire crawling process, the traditional neighborhood search range only considers hyperlinks in the webpage to which the current hyperlink *Plink* points, i.e., neighborhood set or candidate neighborhood set. To expand the search range of the crawler, our improved tabu search (ITS) algorithm extends the neighborhood set to the extended neighborhood set. After access to the candidate neighborhood set of the current hyperlink *Plink* for a specified number of times and at every time if there is no a suitable hyperlink to be found, the next hyperlink will be selected from the extended neighborhood set.

4.4 Tabu list

The tabu list contains the tabu object and tabu length. The tabu object is the object in the tabu list. When updating the crawler queue based on the neighborhood set $N(Plink)$, it is possible for the crawler to repeatedly select a certain hyperlink *Plink* with the highest comprehensive

priority. To avoid this, in the traditional TS algorithm, if the comprehensive priority of *Plink* is higher than the priority of the current optimal hyperlink, the algorithm will ignore its tabu property and replace the current optimal hyperlink and the current hyperlink by *Plink* and at the same time set it as the tabu object and put it into the tabu list; otherwise, the nontabu hyperlink with the highest comprehensive priority from *N*(*Plink*) will be selected as the current hyperlink and will be regarded as a new tabu object. However, in the improved tabu search (ITS) algorithm, we do not consider whether the current hyperlink *Plink* is a tabu object or not. As long as each of the comprehensive priorities of five randomly selected hyperlinks from *C*(*Plink*) is lower than *Plink's* comprehensive priority, we will set *Plink* as a tabu object, put *Plink* into the tabu list, and then select a nontabu hyperlink with the highest comprehensive priority from *E*(*Plink*) as the current hyperlink. Obviously, when the hyperlink is selected from *E*(*Plink*), *Plink* is not selected again. This improved tabu object strategy not only gives the current hyperlink more opportunities to select the next hyperlink with better comprehensive priority from the candidate neighborhood set but also effectively extends the search range of the crawler by the extended neighborhood set.

Tabu length denotes the maximum number of times by which tabu objects are not picked out from the tabu list without considering the aspiration criteria. In this paper, the tabu length is set to five.

4.5 Aspiration criterion and improved acceptance principles

The aspiration criterion means that when a tabooed hyperlink has higher comprehensive priority than the current optimal hyperlink, the tabu property of this tabooed hyperlink will be ignored, and it will be accepted as the current hyperlink. In the traditional TS algorithm, when the tabooed hyperlink does not satisfy the aspiration criterion, the nontabu hyperlink with the highest comprehensive priority will be selected from the neighborhood set as the current hyperlink (ignoring its comparison with the current hyperlink). This method easily accepts hyperlinks with a low comprehensive priority. The improved tabu search (ITS) algorithm refines the acceptance principles by the following steps while retaining the aspiration criterion:

1) If hyperlink *Glink* selected from *C*(*Plink*) is a tabu object and satisfies the aspiration criterion, *Glink* will be released and accepted as the current hyperlink *Plink*.

2) If hyperlink *Glink* is a tabu object and does not satisfy the aspiration criterion, *Glink* will not be accepted as the current hyperlink. Thereafter, a new hyperlink is randomly selected from *C*(*Plink*). If its comprehensive priority is larger than that of the current hyperlink, it will be accepted as the new current hyperlink; otherwise, another hyperlink will be selected from *C*(*Plink*) and judged whether it is accepted. This process is repeated five times until a selected hyperlink is accepted. If each of the five cannot be accepted, we set the hyperlink *Plink* as a tabu object and put it into the tabu list. Then, select a nontabu hyperlink with the highest comprehensive priority from *E*(*Plink*) as the current hyperlink *Plink*. Update the tabu list and release the object whose term is 0.

3) If hyperlink *Glink* is not a tabu object and its comprehensive priority is higher than the comprehensive priority of the current hyperlink *Plink*, *Glink* will be accepted as the current hyperlink *Plink*.

4) If hyperlink *Glink* is not a tabu object and its comprehensive priority is not higher than the comprehensive priority of the current hyperlink *Plink*, *Glink* will not be accepted as the current hyperlink. Thereafter, the five different hyperlinks are selected from *C*(*Plink*), similar to the above step 2). If the comprehensive priority of a selected hyperlink is larger than that of the current hyperlink, this hyperlink will be accepted as a new current hyperlink. If each of them cannot be accepted as the current hyperlink, we set the hyperlink *Plink* as a tabu object and put it into the tabu list. Then, select a nontabu hyperlink with the highest comprehensive priority from *E*(*Plink*) as the current hyperlink *Plink*. Update the tabu list and release the object whose term is 0.

4.6 Focused crawler based on improved tabu search algorithm

The improved tabu search (ITS) algorithm is obtained by improving the tabu object and acceptance principles of the traditional TS algorithm. The ITS algorithm is applied to determine the next hyperlink to be visited from the waiting queue Q_{wait}.

First, initialize the tabu list H_1. Suppose that *Hlink* is the current optimal hyperlink and *Plink* is the current hyperlink selected randomly from Q_{wait}. Construct a candidate neighborhood set *C*(*Plink*) and an extended neighborhood set *E*(*Plink*) based on the current hyperlink *Plink*. Randomly select a hyperlink *Glink* from *C*(*Plink*) as a candidate hyperlink. Then, judge whether *Glink* is accepted according to the improved acceptance principles. If it is accepted, replace the current hyperlink *Plink* by *Glink*, and output the hyperlink *Plink*. If it is not accepted, we select another candidate hyperlink *Glink* from *C*(*Plink*) and continue the judgment process. If five different candidate hyperlinks are selected, and at every time, the selected candidate hyperlink is not accepted, then we set *Plink* as a tabu object and put it into tabu list H_1. Reselect a nontabu hyperlink with the highest comprehensive priority from the extended neighborhood set *E*(*Plink*) as

the current hyperlink. Update the tabu list H_1 by subtracting 1 for the term of each tabu object in the tabu list, and release the object whose term is 0. The above iterative process is repeated until a hyperlink is accepted. The detailed process of the ITS(Q_{wait}) algorithm is presented in Algorithm 1 of Appendix A.

4.7 Focused crawler combining ontology and improved tabu search algorithm

By introducing the ITS algorithm into the focused crawler and using the domain ontology to describe the topic, we design a focused crawler strategy combining ontology and the ITS algorithm (FCOITS), which is used to fetch topic-relevant webpages from the Internet.

First, determine the topic and build the domain ontology about this topic, and add the seed URLs to Q_{wait}. Suppose that α is the threshold of topic-relevant webpages and β is the threshold of the hyperlink's comprehensive priority. Then, the ITS(Q_{wait}) algorithm is used to select the next hyperlink *phead* to visit and download the webpage *phead-page* to which the hyperlink *phead* points. If R(*phead-page*) > α, it is considered a topic-relevant webpage; otherwise, it is considered an irrelevant webpage. Subsequently, all hyperlinks in the webpage *phead-page are extracted* and added to the set of *child-links*. Calculate the comprehensive priority of every hyperlink *child-link*$_i$ in *child-links* based on Eq. (9). If P(*child-link*$_i$) > β, add *child-link*$_i$ to Q_{wait}; otherwise, discard it. The above iteration process is repeated until the end conditions are met. Fig. 1 shows the flowchart of the proposed FCOITS algorithm, where *DP* represents the number of current downloaded webpages, *LP* is the number of current downloaded topic-relevant webpages, and Q_{wait} represents the waiting queue. The detailed process of the FCOITS algorithm is presented in Algorithm 2 of Appendix A.

4.8 Focused crawler combining FCOITS algorithm and host information

It is possible that the crawler recursively crawls under a few hosts, resulting in premature convergence of the crawler and limitation to retrieve more topic-relevant webpages. Hostgraph (Jiang and Zhang, 2007) reveals the connection between hyperlinks and common hosts. For example, "klme.nuist.edu.cn" in the hyperlink "http://klme.nuist.edu.cn/Show.aspx? AI = 1937" is the host, and any hyperlink that contains it can be under the same host. In this paper, we analyse the hyperlink's syntactic structure, leverage the host of the hyperlink, and propose a new focused crawler that integrates host information into FCOITS, called FCITS_OH.

At the beginning of the FCITS_OH, the hyperlinks that are located under different hosts and have higher comprehensive priorities are selected as seed hyperlinks to avoid premature convergence of the algorithm. Then, put the selected hyperlinks into Q_{wait}. Apply the ITS(Q_{wait}) algorithm to obtain the head hyperlink *phead* of Q_{wait}, whose host is marked by *phead_host*. Suppose that the number of hosts in Q_{wait} is *QN_host*. The number of downloaded webpages is *DP*. The algorithm ends when *DP* reaches 15,000. The algorithm completion rate is defined by *com-rate* = *DP*/15,000. During crawling, if some hyperlinks are visited many times under the same host, the crawler will select another hyperlink located at different hosts in Q_{wait} from the current host. To avoid the crawler circularly crawling under a few hosts, a tabu list H_2 for hosts is defined. Continue the following four steps: (1) If *phead_host* is a tabooed host, call the ITS(Q_{wait}) algorithm to obtain another head hyperlink *phead* of Q_{wait}; (2) If *com_rate* < 0.3 and *QN_host* < 10, select three hyperlinks according to descending order of comprehensive priorities from the discarded hyperlinks whose hosts do not belong to the set of hosts in Q_{wait} and add them into Q_{wait}. This is conducive to expanding the number of hosts of hyperlinks in Q_{wait}. (3) If the number of visited links under the current host *phead_host* is less than 50, continue to visit other links under the current host *phead_host*; otherwise, compute the percentage *ph_ratio* of topic-relevant webpages to all visited webpages under the current host *phead_host*. (4) If the number of visited links under the current host *phead_host* is less than 100 and *ph_ratio* > 0.8, continue to visit other links under the current host *phead_host*; otherwise, set *phead_host* as a tabooed host and put it into H_2. After the head hyperlink *phead* is obtained, continue the remaining steps of Algorithm 2 until the end conditions are met. The detailed process of the FCITS_OH algorithm is presented in Algorithm 3 of Appendix A

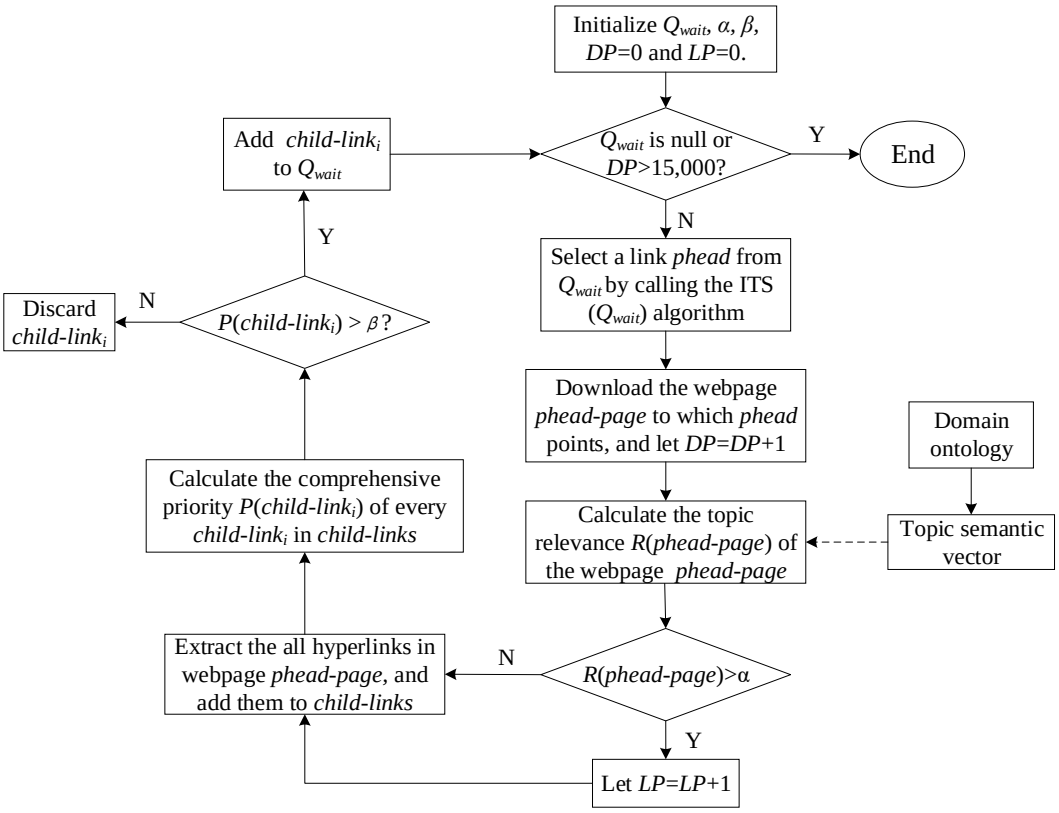

Fig. 1. Flowchart of the proposed FCOITS algorithm

5 Experimental results and analysis

In this paper, the initial seed hyperlinks are acquired from Baidu, which is the most authoritative and widely used search engine in China. We obtained some webpages by searching the keywords tourism and rainstorm disaster, respectively. We choose 30 top-ranked webpages as the initial seed hyperlinks in the tourism domain and rainstorm disaster domain.

In addition, some important parameters α and β have a great impact on the experimental results. For example, if the topic relevance threshold α is too high, the crawled topic-relevant webpages will be reduced because some topic-relevant webpages are filtered. If the threshold α is too low, some irrelevant webpages will be wrongly considered topic-relevant webpages. This paper has conducted parameter experiments on different values of α between 0.5 ~ 0.8 based on the lattice search method under different domains by referring to the literature (Liu and Du, 2014). The results show that when α_1 = 0.7 in the tourism domain and α_2 = 0.62 in the rainstorm disaster domain, the crawler could correctly capture topic-relevant webpages and achieve the best performance. The other parameters are set similarly. Here, β=0.19 for the tourism domain, and β=0.15 for the rainstorm disaster domain. r_1 =0.55, r_2 = 0.25, and r_3 = 0.20.

5.1 Performance metric indices

The effectiveness of the focused crawlers can be generally evaluated by the accuracy (*AC*) and recall (*RC*). *AC* equals the ratio of the number *LP* of downloaded topic-relevant webpages to the total number *DP* of downloaded webpages. *RC* equals the ratio of the number of downloaded topic-relevant webpages to the total number of all topic-relevant webpages on the Internet. Because it is difficult to count the total number of all topic-relevant webpages on the Internet, in this study, we do not use *RC* as the evaluation metric. In addition, we also use the average topic relevance (*AR*) and the standard deviation (*SD*) of downloaded webpages as evaluation metrics. The three metric indices in this paper are as follows.

$$AC = \frac{LP}{DP} \tag{10}$$

$$AR = \frac{1}{DP}\sum_{i=1}^{DP} R(p_i) \tag{11}$$

$$SD = \sqrt{\frac{1}{DP}\sum_{i=1}^{DP}\left(R(p_i) - AR\right)^2} \tag{12}$$

Here, *AC* represents the accuracy, *LP is* the number of downloaded topic-relevant webpages,

DP is the number of downloaded webpages, $R(p_i)$ is the topic relevance of webpage p_i, and *AR is* the average topic relevance of all downloaded webpages. *SD* is the standard deviation of all downloaded webpages compared to *AR* and is used to measure the spread of the topic relevance of all downloaded webpages. The value of *SD* is in [0,1].

5.2 Experimental results of different crawlers

In this paper, we first test seven focused crawling algorithms in the tourism and rainstorm disaster domains under the same experimental environment, including the breadth-first search algorithm (BFS) (Li *et al*, 2015), best-first search algorithm (OPS) (Rawat and Patil, 2013), focused crawler based on simulated annealing algorithm (FCSA) (Liu *et al*., 2019), focused crawler combining web space evolutionary algorithm and ontology (FCWSEO) (Liu *et al*., 2022b), focused crawler combining ontology learning and multiobjective ant colony algorithm (OLMOACO) (Liu *et al*., 2022a), focused crawler combining ontology and improved tabu search algorithm (FCOITS), and focused crawler using improved tabu search algorithm with domain ontology and host information (FCITS_OH). The last two algorithms are proposed in this paper. We implement all crawling algorithms in Java language and run them on an Intel Core i7-7700 PC with 3.6 GHz CPU and 8.0 GB RAM. When the number of downloaded webpages reaches 15,000, all algorithms tend to be stable and end. The same evaluation indices are used to test different crawler algorithms on the two topics of tourism and rainstorm disasters. This is conducive to investigating the validity, superiority and adaptability of each algorithm.

5.2.1 Experimental results in the tourism domain

Experimental results of the number *LP* of downloaded topic-relevant webpages, the accuracy (*AC*), the average topic relevance (*AR*) and the standard deviation (*SD*) of downloaded webpages by six different crawling algorithms including the BFS, OPS, FCSA, FCWSEO, FCOITS and FCITS_OH in the tourism domain are shown in Figs. 2-5 for comparison. Fig. 2 shows the results of the *LP* obtained by six crawling algorithms in the tourism domain. With the increase in the number of downloaded webpages, the *LP* of all the other five crawling algorithms also increase rapidly except for the BFS. Obviously, the *LP* obtained by FCITS_OH is significantly greater than that of the other five crawling algorithms. The *LP* obtained by FCITS_OH is 13,082 when *DP* reaches 15,000. Fig. 3 shows the results of the *AC* obtained by six crawling algorithms in the tourism domain. It is not hard to see from the figure that the *AC* of FCITS_OH becomes higher than that of the other five crawler algorithms after the *DP* exceeds 8,000. The *AC* of the BFS, OPS, FCSA, FCWSEO, FCOITS and FCITS_OH crawling algorithms are 37.4%, 68.2%, 75.03%, 80.86%, 84.53% and 87.21%, respectively, when *DP* reaches 15,000.

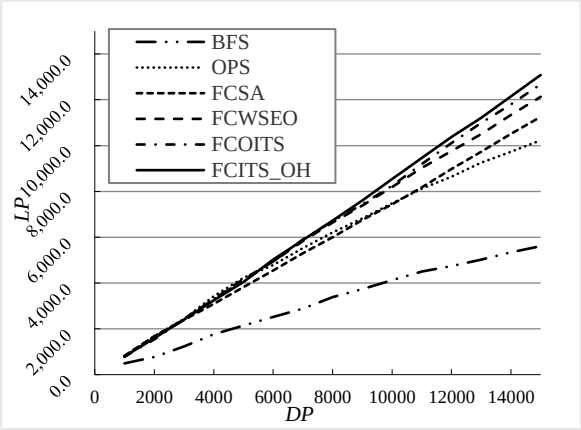

Fig. 2 Results of *LP* obtained by six crawling algorithms in the tourism domain (the same below).

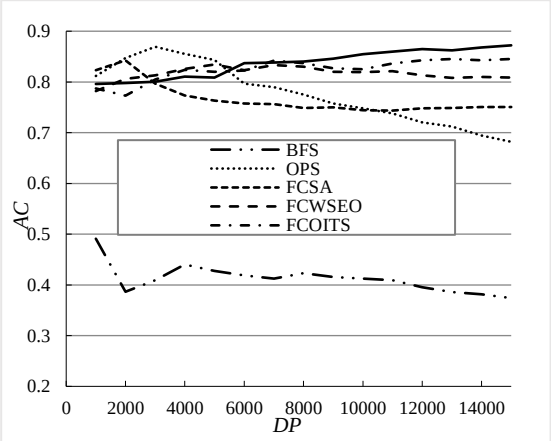

Fig. 3 Results of *AC* obtained by six crawling algorithms in the tourism domain.

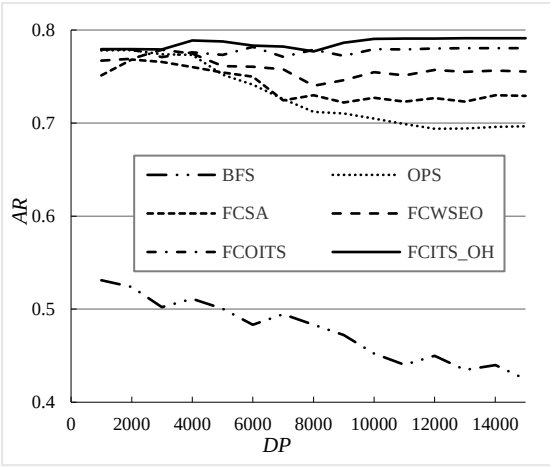

Fig. 4 Results of *AR* obtained by six crawling algorithms in the tourism domain.

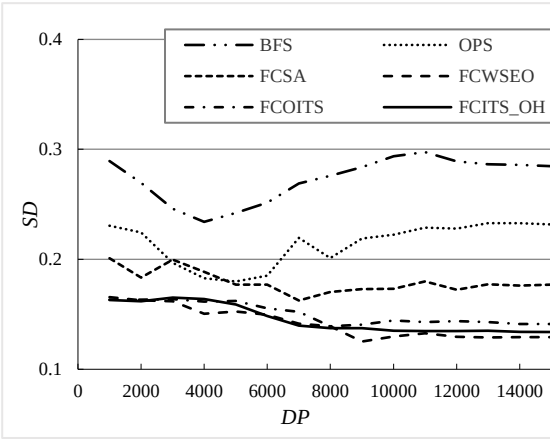

Fig. 5 Results of *SD* obtained by six crawling algorithms in the tourism domain.

Fig. 4 shows the results of the average topic relevance (*AR*) obtained by six crawling algorithms in the tourism domain. According to Fig. 4, the *AR* of FCITS_OH is obviously higher than that of the other five crawling algorithms after the *DP* exceeds 8,000. The *AR* of the BFS, OPS, FCSA, FCWSEO, FCOITS and FCITS_OH crawling algorithms are 0.4247, 0.6966, 0.7292, 0.7553, 0.7806 and 0.7912, respectively, when *DP* reaches 15,000. Fig. 5 shows the results of the *SD* obtained by six crawling algorithms in the tourism domain. From Fig. 5, the FCITS_OH maintains a low *SD* throughout the whole crawling process. The *SD* of the BFS, OPS, FCSA, FCWSEO, FCOITS and FCITS_OH crawling algorithms are stable at 0.2848, 0.2317, 0.1769, 0.1293, 0.1413 and 0.1340, respectively, when *DP* reaches 15,000. The *SD* reflects the stability of the topic relevance of webpages captured by the algorithm. The lower the standard deviation is, the more stable the algorithm. Although the *SD* of the FCITS_OH is slightly higher than that of the FCWSEO, the FCITS_OH outperforms the FCWSEO in the other evaluation metrics.

5.2.2 Experimental results in the rainstorm disaster domain

Experimental results by seven different crawling algorithms including the BFS, OPS, FCSA, FCWSEO, OLMOACO, FCOITS and FCITS_OH in the rainstorm disaster domain for four evaluation indices *LP*, *AC*, *AR*, and *SD* are shown in Figs. 6-9, respectively. Fig. 6 shows the results of the *LP* obtained by seven crawling algorithms in the rainstorm disaster domain. From Fig. 6, we find that when *DP* reaches 15,000, FCITS_OH obtains 12,393 topic-relevant webpages. These values indicate that FCITS_OH can collect more topic-relevant webpages than the other six crawling algorithms. Fig. 7 shows the results of the *AC* obtained by seven crawling algorithms in the rainstorm disaster domain. From Fig. 7, we find that the *AC* of FCITS_OH tends to stabilize gradually after the *DP* exceeds 10,000. Finally, the *AC* of the BFS, OPS, FCSA, FCWSEO, OLMOACO, FCOITS and FCITS_OH crawling algorithms are 23.66%, 65.42%, 70.04%, 81.03%, 74.17%, 79.69% and 82.62%, respectively. Compared with the other six crawling algorithms, FCITS_OH has a higher *AC*.

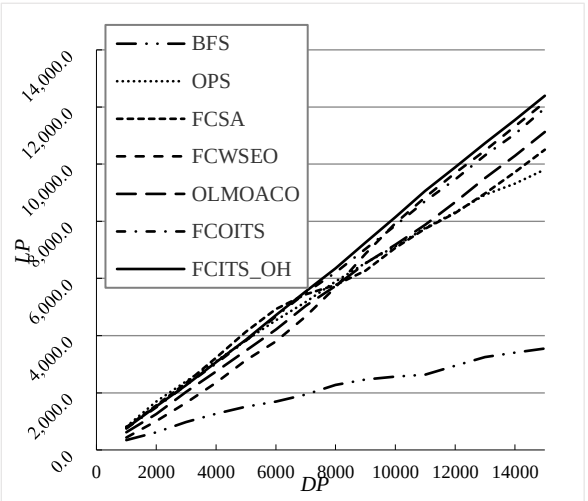

Fig. 6 Results of *LP* obtained by seven crawling algorithms in the rainstorm disaster domain.

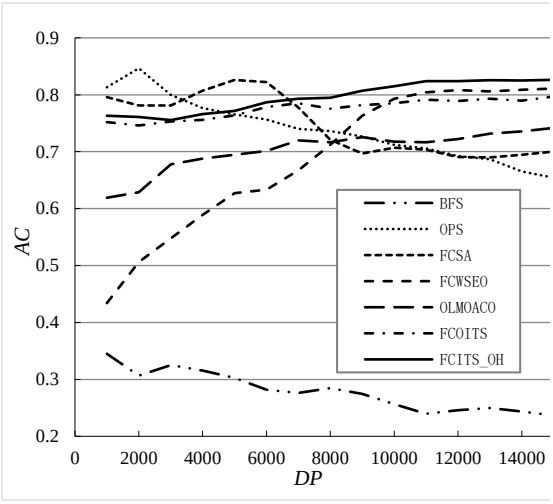

Fig. 7 Results of *AC* obtained by seven crawling algorithms in the rainstorm disaster domain.

Fig. 8 shows the results of the average topic relevance (*AR*) obtained by seven crawling algorithms in the rainstorm disaster domain. Throughout the entire crawler's process, the average relevance of FCITS_OH is relatively high and flat in seven crawling algorithms. Finally, the *AR* of the BFS, OPS, FCSA, FCWSEO, OLMOACO, FCOITS and FCITS_OH crawling algorithms are stable at 0.2947, 0.6376, 0.6627, 0.8200, 0.7781, 0.7306 and 0.7421, respectively. Fig. 8 shows that although the *AR* of FCWSEO and OLMOACO are slightly higher than that of FCWITS_OH, the effect of FCITS_OH to grab topic-relevant webpages is relatively stable.

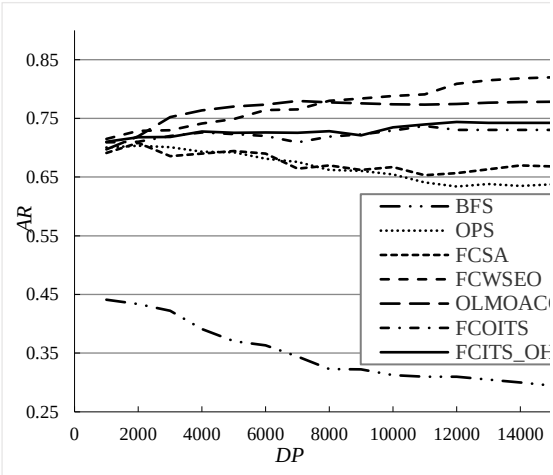

Fig. 8 Results of *AR* obtained by seven crawling algorithms in the rainstorm disaster domain.

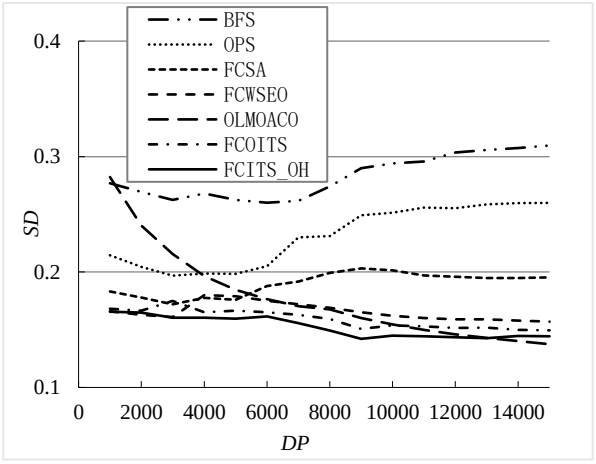

Fig. 9 Results of *SD* obtained by seven crawling algorithms in the rainstorm disaster domain.

Fig. 9 shows the results of the standard deviation (*SD*) obtained by seven crawling algorithms in the rainstorm disaster domain. The *SD* of FCITS_OH maintains a downwards trend in the whole crawler's process. Finally, the *SD* of the BFS, OPS, FCSA, FCWSEO, OLMOACO, FCOITS and FCITS_OH crawling algorithms are 0.3096, 0.2599, 0.1953, 0.1570, 0.1375, 0.1495 and 0.1444, respectively. Although the *SD* of FCITS_OH is slightly higher than that of OLMOACO, they are comparable.

5.3 Analysis and discussion of experimental results

From Figs. 2, 3, 4, 6, 7, and 8, it is not hard to find that the OPS algorithm has a better performance in the early crawling stage on the tourism and rainstorm disaster domains, but it decreases in the later crawling stage, resulting from its greedy strategy. The OPS algorithm always selects the highest priority hyperlink from the waiting queue to crawl the webpage. When it falls into a choice of a hyperlink with no prospects, the webpage it points to may contain few valuable hyperlinks, which is not conducive to the expansion of the search range. The FCSA algorithm is also a kind of greedy strategy but changes the optimal search by adopting a certain probability to receive hyperlinks with relatively low priority. However, the performance of the FCSA algorithm highly depends on its parameters, such as the initial temperature and annealing speed, which are difficult to determine. Therefore, the ability of the FCSA algorithm to grab topic-relevant webpages is only slightly higher than that of the BFS and OPS algorithms.

Figs. 3 and 7 show that the FCITS_OH algorithm overmatches the OLMOACO and FCWSEO algorithms on *AC* in the tourism and rainstorm disaster domains. The FCWSEO and OLMOACO algorithms grow fast in the early stage and tend to stabilize without improvement later in the rainstorm disaster domains. This is because the FCWSEO algorithm is a kind of multiobjective optimization algorithm that produces nondominant hyperlinks within circular regions. However, as the circular regions expand, it will easily catch some hyperlinks with no prospects by contrasting with the adjacent hyperlinks and affect the crawling performance. For the OLMOACO algorithm, it is easier to accumulate pheromones to find an optimal search path at the beginning, and as the crawler proceeds, it is affected by the feedback mechanism that makes it difficult to improve the pheromone of the optimal path again. As a result, it is challenging to continue to enhance the ability to fetch more topic-relevant webpages. The FCITS_OH algorithm uses tabu object and aspiration criteria to avoid crawling visited hyperlinks and introduces host information to expand the search range, so it is easier to find the optimal crawling path and fetch more topic-relevant hyperlinks in the entire crawling process.

Table 2 displays the specific values of all evaluation metrics and running times of the abovementioned seven crawling algorithms in the tourism domain and rainstorm disaster domain when *DP* reaches 15,000. The optimal value of every metric index is marked by bold font. From Table 2, we can find that the running time of the BFS is the shortest, while the FCWSEO and OLMOACO require longer running times than the other crawling algorithms in the tourism domain and rainstorm disaster domain, respectively. This is because FCWSEO and OLMOACO are multiobjective optimization crawling algorithms, where the optimization process of hyperlink selection based on a multiobjective optimization model increases the time consumption. The running time of FCITS_OH is slightly longer than that of the other crawling algorithms except FCWSEO and OLMOACO. This is because it takes more running time to construct the ontology and extract host information.

Table 2 Comparison of results obtained by nine crawling algorithms in the tourism and rainstorm disaster when *DP* reaches 15,000

Algorithms	Tourism domain					Rainstorm disaster domain				
	LP	*AC*	*AR*	*SD*	*Time/h*	*LP*	*AC*	*AR*	*SD*	*Time/h*
BFS (2015)	5610	37.4	0.4247	0.2848	**7.78**	3549	23.66	0.2947	0.3096	**8.24**
OPS (2013)	10230	68.2	0.6966	0.2317	8.56	9813	65.42	0.6376	0.2599	8.93
FCSA (2019)	11255	75.03	0.7292	0.1769	10.72	10506	70.04	0.6627	0.1953	11.08
FCWSEO (2022)	12129	80.86	0.7553	**0.1293**	13.94	12162	81.03	**0.8200**	0.1570	11.64
OLMOACO (2022)	-	-	-	-	-	11126	74.17	0.7781	**0.1375**	16.00
FCTS	10822	72.15	0.7066	0.1726	10.11	10254	68.36	0.6523	0.1962	9.98
FCITS	11581	77.21	0.7534	0.1446	11.26	11054	73.69	0.7002	0.1589	10.29
FCOITS	12679	84.53	0.7806	0.1413	11.27	11954	79.69	0.7306	0.1495	11.24
FCITS_OH	**13082**	**87.21**	**0.7912**	0.1340	11.97	**12393**	**82.62**	0.7421	0.1444	11.51

Furthermore, to further investigate the effectiveness of improved strategies of tabu object and acceptance principles in the improved tabu search (ITS) algorithm, we design the focused crawler based on the improved tabu search algorithm (FCITS) and the focused crawler based on the traditional tabu search algorithm (FCTS). For convenience of presentation, Table 2 also shows the *LP*, *AC*, *AR*, *SD* and running time of the FCITS and FCTS in the tourism and rainstorm disaster domains when *DP* reaches 15,000. We find that the experimental results of the FCITS for all evaluation metrics except the running time are better than those of the FCTS. This further confirms the effectiveness of the improved strategies in the ITS algorithm. With regard to the running time of the ITS algorithm, we analyse the time complexity of the ITS algorithm and the TS algorithm in the crawler process as follows.

Suppose that there are m hyperlinks in the waiting queue Q_{wait}. The time complexity of selecting a hyperlink *Plink* from Q_{wait} is O(m). The time consumption of selecting a hyperlink *Glink* from the neighborhood C(*Plink*) is assumed to be $k_1 \sim$O(m). The time consumption for determining the taboo object is constant, and the time complexity of computing the topic relevance of the hyperlink is $k_2 \times$O($DP \times n$), where k_2 is the time consumption of

word segmentation, word frequency statistics and link extraction from webpages; O($DP\times n$) represents the time complexity of calculating $R(p_i)$, $PR(p_l)$, and $R(A_l)$; DP and n are the number of downloaded webpages and the number of topic words, respectively. The time consumption of selecting a hyperlink from the extended neighborhood set is assumed to be $k_3 \sim O(m)$. Therefore, the time complexity of the ITS algorithm can be expressed as $O(m)\times[k_1\times k_2\times O(DP\times n)\times k_3]$. Because $k_1\sim O(m)$, $k_2\sim O(n)$, and $k_3\sim O(m)$, the time complexity of the ITS algorithm is $O(m^3\times DP\times n^2)$.

Table 3. Friedman ranks of nine crawling algorithms for the four representative evaluation indices in the tourism and rainstorm disaster when *DP* reaches 15000.

Friedman	Tourism domain					Rainstorm disaster domain				
	LP	*AC*	*AR*	*SD*	*Average*	*LP*	*AC*	*AR*	*SD*	*Average*
BFS (2015)	8	8	8	8	8	9	9	9	9	9
OPS (2013)	7	7	7	7	7	8	8	8	8	8
FCSA (2019)	5	5	5	6	5.25	6	6	6	6	6
FCWSEO (2022)	3	3	3	1	2.5	2	2	1	4	2.25
OLMOACO (2022)	-	-	-	-	-	4	4	2	1	2.75
FCTS	6	6	6	5	5.75	7	7	7	7	7
FCITS	4	4	4	4	4	5	5	5	5	5
FCOITS	2	2	2	3	2.25	3	3	4	3	3.25
FCITS_OH	1	1	1	2	**1.25**	1	1	3	2	**1.75**

Different from the ITS algorithm, the TS algorithm selects a nontabu link with the best comprehensive priority from neighborhood $C(Plink)$ when the tabooed hyperlink does not satisfy the aspiration criterion, so its time complexity is $O(m^2\times DP\times n^2)$. By analysing the time complexity of ITS and TS, it can be seen that the time complexity of the ITS algorithm is higher than that of the TS algorithm. This results in a longer running time of the FCITS algorithm than the FCTS algorithm.

It can be seen from Table 2 that not all evaluation metrics of FCITS_OH have optimal results. To better evaluate the effectiveness and superiority of the FCITS_OH, the Friedman test (Derrac *et al*., 2011), which is a nonparametric statistical test, is used to comprehensively evaluate the performance of these algorithms. In this paper, when DP=15,000, the results obtained by nine crawling algorithms for the four representative indices *LP*, *AC*, *AR* and *SD* are converted to average ranks. The best performing algorithm for each index should have the rank of 1, the second best ranks 2, and so on. The smaller the average rank is, the better the performance. Table 3 displays the experimental results of nine crawling algorithms based on four evaluation metrics by the Friedman test when DP reaches 15,000. From Table 3, we can find that the FCITS_OH algorithm for four indices is the best performing algorithm out of the nine algorithms in two domains. In summary, the experimental results show that FCITS_OH achieved impressive and satisfactory results in most performance evaluation indices, particularly prevailing over the other eight crawlers in the *LP* and *AC*. Therefore, we can conclude that the proposed FCITS_OH crawler is an effective semantic retrieval method.

6 Conclusions

The drawback of traditional crawlers is that they cannot provide enough topic-relevant information for a specific domain. To overcome the shortcomings of traditional crawlers, this paper focuses on focused crawlers. We propose a novel focused crawling algorithm, namely, FCITS_OH. Specifically, we construct a domain ontology based on the FCA method for topic description at the semantic and knowledge levels. The ITS strategy and host information are used to select the next hyperlink in the focused crawler. In addition, we design a comprehensive priority evaluation method for evaluating unvisited hyperlinks and preventing the problem of topic drifting. To demonstrate the effectiveness and superiority of the FCITS_OH algorithms, in the two domains of tourism and rainstorm disaster, we compare the experimental results of FCITS_OH and FCOITS with those of the BFS, OPS, FCSA, OLMOACO and FCWSEO in the literature under the same experimental environment. The experimental results show that FCITS_OH outperforms other focused crawling algorithms and has the ability to collect more quantity and higher quality webpages. Furthermore, we also compare the experimental results of the FCTS based on the original TS and the FCITS based on the improved TS. The experimental results confirm the effectiveness of the improvements of the proposed ITS.

However, the proposed FCITS_OH also has some disadvantages, such as no consideration of the tunnel crossing technique. It is possible for a hyperlink to cross an irrelevant webpage to a relevant webpage. In addition, the topic relevance evaluation of unvisited hyperlinks in the focused crawler adopts the traditional single-objective optimization method based on the weighted sum, which has the defect that it is difficult to determine the optimal weight coefficients reasonably. In future work, we intend to study focused crawlers based on the tunnel crossing technique and multiobjective intelligent optimization algorithms to improve our evaluation metrics.

Contributors

Jingfa LIU designed the research. Zhen WANG drafted the paper, implemented the software, and performed the experients. Guo ZHONG and Zhihe YANG revised and finalized the paper.

Compliance with ethics guidelines

Jingfa LIU, Zhen WANG, Guo ZHONG and Zhihe YANG declare that they have no conflict of interest.

References

Asano, Y., Tezuka, Y., Nishizeki, T., 2007. Improvements of HITS Algorithms for Spam Links. In: Dong, G., Lin, X., Wang, W., Yang, Y., Yu, J.X. (eds) Advances in Data and Web Management. APWeb WAIM 2007 2007. Lecture Notes in Computer Science, vol 4505. Springer, Berlin, Heidelberg, p.479-490. https://doi.org/10.1007/978-3-540-72524-4_50 .

Bra PD, Houben GJ, Kornatzky YR., 1994. Information retrieval in distributed hypertexts. In Proceedings of the 4th International Conference on Computer-Assisted Information Retrieval, Rockefeller University p. 481-493.

Chakrabarti S, Berg MVD, Dom B., 1999, Focused crawling: a new approach to topic-specific Web resource discovery. *Computer Networks*, 31 (11): 1623-1640. https://doi.org/10.1016/S1389-1286(99)00052-3

Deng SQ., 2020, Research on the Focused crawler of mineral intelligence service based on semantic similarity. *J. Phys.: Conf. Ser*. 1575 (1):1-8. https://doi.org/10.1088/1742-6596/1575/1/012142

Derrac J, García S, Molina D, et al., 2011. A practical tutorial-on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol Comput*, 1(1):3-18. https://doi.org/10.1016/j.swevo.2011.02.002

Du YJ, Du YJ, Hai YF, Xie CZ, et al., 2014. An approach for selecting seed URLs of focused crawler based on user-interest ontology. *Appl Soft Comput*, 14:663-676. https://doi.org/10.1016/j.asoc.2013.09.007

Farag MMG, Lee S, Fox EA, 2018. Focused crawler for events. *Int J Digit Libr*, 19(1):3-19. https://doi.org/10.1007/s00799-016-0207-1

Fei CJ, Liu BS., 2018, Focused crawler based on LDA extended topic terms. *Computer Applications and Software*. 35(4): 49-54 (in Chinese) http://dx.chinadoi.cn/10.3969/j.issn.1000-386x.2018.04.009

Guan WG, Luo YC, 2016. Design and implementation of focused crawler based on concept context graph. *Comput Eng Des*, 37(10):2679-2684 (in Chinese). https://doi.org/10.16208/j.issn1000-7024.2016.10.019

He S, Cheng JX, Cai XB, 2009. Focused crawler based on simulated anneal algorithm. *Comput Technol Dev*, 19(12): 55-58, 62 (in Chinese). https://doi.org/10.3969/j.issn.1673-629X.2009.12.015

Hosseinkhani J, Taherdoost H., Keikhaee S., 2021, ANTON Framework Based on Semantic Focused Crawler to Support Web Crime Mining Using SVM. *Ann Sci*, 8(2):227-240. https://doi.org/10.1007/s40745-019-00208-5

Jiang Q, Zhang Y, 2007, Siterank-based crawling ordering strategy for search engines. *7th IEEE International Conference on Computer and Information Technology*, p. 259-263. https://doi.org/10.1109/CIT.2007.35 .

Khan MA, Sharma DK., 2016, Self-adaptive ontology-based focused crawling: a literature survey. *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, IEEE, p. 595-601. https://doi.org/10.1109/ICRITO.2016.7785024 .

Lakzaei B, Shmasfard M., 2021, Ontology learning from relational databases. *Inf. Sci*, 577: 280-197 https://doi.org/10.1016/j.ins.2021.06.074

Li L, Zhang GY, Li ZW., 2015, Research on focused crawling technology based on SVM. *Comput. Sci*, 42(2):118-122 (in Chinese) https://doi.org/10.11896/j.issn.1002-137X.2015.2.025

Liu JF, Dong Y, Liu ZX., et al., 2022a, Applying ontology learning and multi-objective ant colony optimization method for focused crawling to meteorological disasters domain knowledge. *Expert Syst Appl*, 198 116741. https://doi.org/10.1016/j.eswa.2022.116741

Liu JF, Li F, Jiang SY, 2019, Focused Annealing Crawler Algorithm for Rainstorm Disasters Based on Comprehensive Priority and Host Information. *Comput. Sci*, 46(2):215-222 (in Chinese) https://doi.org/10.11896/j.issn.1002-137X.2019.02.033

Liu JF, Wang DW, Yan XM, 2021, Tabu search algorithm for dynamic facility layout problem, *J Huangzhong Univ of Sci & Tech (Natural Science Edition)*. 49(02):44-50 (in Chinese). https://doi.org/10.13245/j.hust.210206

Liu JF, Li X, Zhang SY, et al., 2022b. A novel focused crawler combining Web space evolutionary and domain ontology. *Knowledge-Based Systems*, 243 108495. https://doi.org/10.1016.j.knosys.2022.108495

Liu WJ, Du YJ., 2014. A novel focused crawler based on cell-like membrane computing optimization algorithm. *Neurocomputing*, 123:266-280.

https://doi.org/10.1016/j.neucom.2013.06.039

Ma LL, Li HW, Lian SW, et al., 2016. A strategy of disaster focused crawler based on ontology semantics. *Comput Eng*, 42(11):50-56 (in Chinese) https://doi.org/10.3969/j.issn.1000-3428.2016.11.009

Prakash J, Kumar R, 2015. Web crawling through shark-search using PageRank. *Procedia Comput Sci*, 48:210-216. https://doi.org/10.1016/j.procs.2015.04.172

Rani M, Dhar AK, Vyas OP., 2017, Semi-automatic terminology ontology learning based on topic modeling. *Eng Appl Artif Intel*, 63:108-125. https://doi.org/10.1016/j.engappai.2017.05.006

Rawat S, Patil DR, 2013. Efficient focused crawling base on best first search. Proc 3rd IEEE Int Advance Computing Conf, p.908-911. https://doi.org/10.1109/IAdCC.2013.6514347

Tong YL, 2008, Application of crawler using adaptive dynamical evolutional particle swarm optimization. *Geomat Inf Sci Wuhan Univ*, 33(12):1296-1299 (in Chinese). https://doi.org/*CNKI:SUN:WHCH.0.2008-12-022*

Wang ZG, Meng BJ, 2014. A comparison of approaches to Chinese word segmentation in hadoop. Proc IEEE Int Conf on Data Mining Workshop, p.844-850. https://doi.org/10.1109/ICDMW.2014.43

Wu YL, Zhao SL, Li CJ, et al., 2017, Text classification method based on TF-IDF and cosine similarity. Journal of Chinese Information processing, 31(5):138-145 (in Chinese).

Wu TY, 2018, Research on information retrieval technology based on Word2vec+BM25. *Electronics World*, 22: 136-136. https://doi.org/10.19353/j.cnki.dzsj.2018.22.080 .

Xiao JJ, Chen ZH. 2018, Focused Crawling Based on Grey Wolf Algorithms. *Comput Sci*, 45(S2):146-148+166(in Chinese).

Yan W, Pan L., 2018, Designing focused crawler based on improved genetic algorithm. *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, p. 319-323. https://doi.org/10.1109/ICACI.2018.8377476

Yu J, Liu G., 2015, Survey on topic-focused crawlers. *Comput Eng & Sci* 37(2):231-237 (in Chinese) https://doi.org/10.3969/j.issn.1007-130X.2015.02.007

Yuan ZQ, Zhang WH., Fu HJ., et al., 2017, A PageRank-improved ranking algorithm based on cheating similarity and cheating relevance. *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, p. 257-263. https://doi.org/10.1109/ICIS.2017.7960003

Zhu G, Yang JY, Wu XH, et al., 2017. Research on construction of hierarchy relationship and ontology of meteorological disaster based on FCA. *Mod Inf*, 37(5):79-88 (in Chinese). https://doi.org/10.3969/j.issn.1008-0821.2017.05.014

Appendix A:

Algorithm 1

ITS(Q_{wait})

Input: Q_{wait}

Output: a hyperlink.

Initialize the tabu list H_1. The tabu length of H_1 is set to 5.
Suppose that *Hlink* is the currently visited hyperlink with the highest comprehensive priority.
Randomly select a hyperlink from Q_{wait} as current hyperlink and denote it by *Plink*. Set j=1.
Randomly select a hyperlink *Glink* from C(*Plink*), and remove *Glink* from C(*Plink*).
If *Glink* is a tabu object **then**
    If P(*Glink*) > P(*Hlink*) **then**
        Accept *Glink* and delete *Glink* from H_1. Let *Hlink* = *Glink*, *Plink* = *Glink*, and go to step 8.
    Else
        Do not accept *Glink*. Let $j = j$+1, and go to step 6.
    End if
End if
If *Glink* is not a tabu object **then**
    If P(*Glink*) > P(*Plink*) **then**
        Let *Plink* = *Glink*.
        If P(*Glink*) > P(*Hlink*) **then**
            Let *Hlink* = *Glink*.
        End if
        Go to step 8.
    Else
        Do not accept *Glink*. Let $j = j$+1, and go to step 6.

End if

6: **If** $j > 5$ **then**

Set *Plink* as a tabu object and put it into H_1. Reselect a non-tabu hyperlink with the highest comprehensive priority from $E(Plink)$ as the current hyperlink *Plink*. Go to step 7.

Else

Keep *Plink* unchanged and go to step 4.

Else if

7: Update the tabu list H_1 by subtracting 1 for the term of each tabu object in the tabu list, and release the object whose term is 0. Go to step 4.

8: Output the hyperlink *Plink*.

Algorithm 2

FCOITS

Input: seed hyperlinks.

Output: downloaded webpages.

1: Determine the topic and construct domain ontology about this topic (see Section 2). Then, add the seed hyperlinks to Q_{wait}, and initialize thresholds α, β, $DP = 0$, and $LP = 0$. // *DP* is the number of downloaded webpages, and *LP* is the number of downloaded topic-relevant webpages.

2: **If** Q_{wait} is not empty or $DP < 15{,}000$ **then**

Let *phead* = ITS(Q_{wait}) and insert the hyperlink *phead* into the head of Q_{wait}.

Else the algorithm ends.

End if

3: Select the head hyperlink *phead* from Q_{wait}.

4: Download the webpage to which *phead* points, denote it by *phead-page*, and let $DP = DP+1$.

5: Analyze, and segment to obtain the feature vector of the webpage *phead-page*. Calculate the topic relevance R(*phead-page*) based on Eq (4).

6: **If** R(*phead-page*)$> \alpha$ **then**

Download the *phead-page*, and let $LP = LP + 1$.

End if

7: Extract all the *child-links* in webpage *phead-page*.

8: **For** $i = 1$ to x do // x is the number of child-links.

Calculate the comprehensive priority of $child\text{-}link_i$ based on Eq (9).

If $P(child\text{-}link_i) > \beta$ **then**

Add $child\text{-}link_i$ to Q_{wait}.

Else

Discard $child\text{-}link_i$.

End if

End For

9: Go to step2.

Algorithm 3

Alorithm1. FCITS_OH

Input: seed hyperlinks.

Output: downloaded webpages.

1: Determine the topic and construct domain ontology (see Section 2). Then, add the seed hyperlinks which are located under different hosts and have higher comprehensive priorities to Q_{wait}. Initialize thresholds α, β, $DP = 0$, and $LP = 0$.

// *DP* is the number of downloaded webpages, and *LP* is the number of downloaded topic-relevant webpages.

2: **If** Q_{wait} is not empty or $DP < 15{,}000$ **then**

Let *phead* = ITS(Q_{wait}) and insert the hyperlink *phead* into the head of Q_{wait}.

Else the algorithm ends.

End if

3: Select the head hyperlink *phead* from Q_{wait} and extract its host, denoted by *phead_host*. The tabu length of tabu list H_2 is set to 4. // The tabu object in tabu list H_2 is the host.

4: **If** *phead_host* is a tabooed host **then**
  Go to step 2.
End if
5: **If** *com_rate* < 0.3 and *QN_host* < 10 **then**
  Select three hyperlinks according to descending order of comprehensive priorities from the discarded hyperlinks whose hosts do not belong to the set of hosts in Q_{wait} and add them into Q_{wait}.
End if
6: **If** the number of visited links under the current host *phead_host* < 50 **then**
  Go to step 8;
Else
  Compute the percentage *ph_ratio* of topic-relevant webpages to all visited webpages under the current host *phead_host*.
End if
7: **If** the number of visited links under the current host *phead_host* < 100 and *ph_ratio* > 0.8 **then**
  Go to step 8;
Else
  Set the *phead_host* as a tabooed host and put it into H_2. Update the H_2 by subtracting 1 for the term of each tabooed host in tabu list. Release the tabooed host whose term is 0 and clear the visited links under the tabooed host whose term is 0. Keep the head hyperlink *phead* unchanged.
End if
8: Download the webpage to which *phead* points, denote it by *phead-page*, and let *DP* = *DP*+1.
9: Analyze, and segment to obtain the feature vector of the webpage *phead-page*. Calculate the topic relevance *R*(*phead-page*) based on Eq (4).
10: **If** *R*(*phead-page*)> α **then**
  Download the *phead-page*, and let *LP* = *LP* + 1.
End if
11: Extract all the *child-links* in webpage *phead-page*.
12: **For** *i* = 1 to *x* do // *x* is the number of child-links.
  Calculate the comprehensive priority of $child\text{-}link_i$ based on Eq (9).
  If $P(child\text{-}link_i) > \beta$ **then**
    Add $child\text{-}link_i$ to Q_{wait}.
  Else
    Discard $child\text{-}link_i$.
  End if
End For
13: Go to step2.

Appendix B:

TABLE A. Seed URLs in the rainstorm disaster domain

No	URL	No	URL
1	http://data.cma.cn	16	http://www.weather.com.cn/rain
2	http://www.weather.com.cn	17	http://www.ninhm.ac.cn
3	http://news.weather.com.cn	18	http://www.qixiangwang.cn/news
4	http://zhfy.xnl21.com	19	https://baijiahao.baidu.com/s?id=1599783711923902434&wfr=spider&for=pc
5	http://www.tianqi.com	20	http://www.whihr.com.cn/news.do?method=showNewsList&netyId=17
6	http://www.zaihai.cn	21	http://news.cctv.com/special/2016xq
7	http://news.cctv.com/special	22	http://js.weather.com.cn
8	https://15tianqi.cn	23	https://www.mem.gov.cn/kp/zrzh/hlzh
9	http://www.jsmb.gov.cn	24	http://www.qxkp.net/zhfy/byhl
10	http://www.nmc.cn	25	http://www.jieri6.com/baike/25506.html
11	http://www.ndrcc.org.cn	26	http://m.421688.com/news
12	http://www.cma.gov.cn	27	http://www.cma.gov.cn/2011xzt/kpbd/rainstorm/2018050901/201805/t20180509_468007.html
13	http://www.qgshzh.cn	28	http://www.lzjjdc.com/k/honglaozaihaidefangzhicuoshi
14	http://www.whihr.com.cn	29	http://www.cma.gov.cn/2011xzt/20120816/20130625/2013062504/201307.html
15	http://klme.nuist.edu.cn	30	http://www.csdata.org/p/618

TABLE B. Seed URLs in the tourism domain

No	URL	No	URL
1	http://www.029wyly.com/	16	https://www.chimelong.com/
2	http://travel.sina.com.cn/	17	http://www.gzcts03.cn/
3	http://www.cntour.cn/	18	https://travel.ifeng.com/
4	https://www.china-zjj.net/	19	https://tour.jxcn.cn/
5	http://www.ezjj.cn/	20	http://www.visitbeijing.com.cn/
6	https://www.qyer.com/	21	https://travel.sohu.com/
7	http://lffuye.com/	22	http://www.xjlymh.com/
8	http://www.springtour.com/	23	https://www.maigoo.com/maigoo/1343zxly_index.html
9	http://www.china1004.com/	24	http://www.chinazjy.com/
10	http://www.ctscd.com/	25	http://www.guanguang.net.cn/
11	http://www.dazijia.com/	26	http://www.landscape.cn/
12	http://www.huangjinlvyou.com/	27	http://www.sxwhlyw.com/
13	http://www.ljta.gov.cn/	28	http://www.huangshan.com.cn/home?m_id=730
14	https://www.mafengwo.cn/	29	http://wxpt.lyta.com.cn/h5/scenic/view/8a928c4841a657c10141ab7da62b0140
15	http://www.gdslyw.com/	30	http://www.sxjqtextile.com/