Expert Systems With Applications 198 (2022) 116741

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Applying ontology learning and multi-objective ant colony optimization method for focused crawling to meteorological disasters domain knowledge

Jingfa Liu [a,b,1], Yi Dong [c,2,*], Zhaoxia Liu [d], Duanbing Chen [e,f,g,*]

[a] Guangzhou Key Laboratory of Multilingual Intelligent Processing, Guangdong University of Foreign Studies, Guangzhou 510006, China
[b] School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou 510006, China
[c] School of Computer & Software, Nanjing University of Information Science & Technology, Nanjing 210044, China
[d] Network and Information Center, Guangdong University of Foreign Studies, Guangzhou 510006, China
[e] Big Data Research Center, University of Electronic Science and Technology of China, Chengdu 611731, China
[f] The Center for Digitized Culture and Media, University of Electronic Science and Technology of China, Chengdu 611731, China
[g] Union Big Data Tech. Inc., Chengdu 610041, China

ARTICLE INFO

Keywords:
Focused crawler
Multi-objective ant colony optimization
Ontology
Ontology learning

ABSTRACT

The focused crawler based on semantic analysis is a research hotspot in the field of information retrieval. The domain ontology is generally applied to construct the topic model of the focused crawler. In order to overcome the limitations of builders' knowledge reserve and subjective consciousness in the process of constructing artificially ontology, a semi-automatic construction method of domain ontology based on ontology learning technology combining the latent Dirichlet allocation and the Apriori algorithm is proposed in this article. When evaluating the relevance between a hyperlink and a specific topic, the joint evaluation method considering both the web text and the link structure is usually used. However, the traditional weighted sum method is difficult to reasonably determine the optimal weights of these evaluating indicators. To solve this problem, a multi-objective optimization model for link evaluation and a subsequent multi-objective ant colony optimization algorithm (MOACO) are proposed. In the MOACO, a method of the nearest farthest candidate solution (NFCS) is combined with the fast non-dominated sorting to select a set of Pareto-optimal hyperlinks and guide the crawlers' search directions. The experimental results of the focused crawling on the domain knowledge of typhoon disasters and rainstorm disasters prove that the ability of the proposed focused crawlers to retrieve topic-relevant webpages.

1. Introduction

Meteorological disasters account for more than 70% of various natural disasters (Liu, & Yan, 2011), which has a huge impact on people's lives and property safety. For example, in August 2012, under the influence of typhoon "Dawei", a heavy rainstorm disaster occurred in Liaoning Province of China, and the direct economic loss exceeded 21.9 billion yuan. In 2016, Tianjin of China was affected by the "7•20" rainstorms, which affected more than 140,000 people, and the direct economic loss exceeded 250 million yuan (Chen, Gao, Li, Xie, Wang, Liu, & Han, 2019). In April 2019, rainstorms triggered floods in Pakistan, causing at least 49 deaths and 176 injuries. In June 2019, continuous rainfall in Myanmar triggered floods and landslides, killing at least 90 people and injuring 65 others. In response to the impact on meteorological disasters, in addition to emergency response when disasters occur and post-disaster reconstruction, effective early warning and preventive measures are also indispensable to avoid and reduce the losses caused by meteorological disasters and ensure the safety of people's lives and property. Therefore, it is extremely important to obtain early warning, preventive measures and emergency response information of meteorological disasters. Due to the Internet has huge data resources, it has become an important channel for obtaining large amounts of meteorological disaster information. At present, the scale of webpages on the Internet is massive and growing. The content of webpages is highly dynamic and complex. The information about webpages related to meteorological disasters is sparse and has the characteristics of big data. Traditional search engines (such as Google, Baidu) and web crawlers (such as Scrapy, Pyspider) face a huge challenge in accuracy

* Corresponding authors.
E-mail addresses: jfliu@gdufs.edu.cn (J. Liu), dy10101@126.com (Y. Dong), 554822022@qq.com (Z. Liu), dbchen@uestc.edu.cn (D. Chen).
[1] ORCID: 0000-0002-0407-1522.
[2] ORCID: 0000-0002-2039-9798.

https://doi.org/10.1016/j.eswa.2022.116741
Received 1 November 2020; Received in revised form 2 December 2021; Accepted 22 February 2022
Available online 1 March 2022

rate of information retrieval. Unlike these methods, focused crawlers filter webpages based on the values of webpages on the Internet. For a specific search topic of users, the results returned by focused crawlers are more streamlined and more accurate. Therefore, the development of focused crawler shows an increasing trend.

At present, methods of focused crawlers mainly include classic heuristic focused crawlers, focused crawlers based on semantic analysis, and focused crawlers based on intelligent optimization algorithms.

(1) The classic heuristic focused crawlers are divided into search strategies based on content evaluation and link structure evaluation. The main representatives of the search strategies based on content evaluation are the fish-search algorithm (Bra, Houben, Kornatzky, & Post, 1994) and the shark-search algorithm (Hersovici, Jacovi, Maarek, Pelleg, Shtalhaim, & Ur, 1998). The main idea of the fish-search algorithm stems from the concept of fish schools. If a school of fish finds food (webpage), the fish will breed and continue to search for more food. Of course, if a school of fish does not find food, the fish will die (stop searching webpages). The shark-search algorithm is an improvement on the fish-search algorithm. The selection of hyperlinks not only consider the relevance of the webpages where the hyperlink is located, but also the relevance of the anchor text and the anchor text context. Although the shark-search algorithm has improved the fish-search algorithm, both algorithms ignore the impact of the link structure on the crawler. The most representative algorithms for search strategies based on link structure evaluation are the PageRank algorithms (Brin, & Page, 1998; Wang, & Ji, 2016) and the hyperlink-induced topic search (HITS) algorithms (Asano, Tezuka, & Nishizeki, 2008). The basic idea of the PageRank algorithm is that if a page is referenced by many other pages, the page is likely to be an important page; a page that is not referenced multiple times but is referenced by an important page is likely to be an important page. Kleinberg (1999) proposed the HITS algorithm, which divides webpages into authority webpages and hub webpages. This kind of search strategy based on link structure evaluation has less consideration of topic relevance and is prone to cause “topic drift” phenomenon (Henzinger, 2001). Both search strategies based on content evaluation and link structure evaluation are methods for obtaining hyperlink’s value. In order to better capture the value of hyperlinks, the above two strategies are usually used in combination. Seyfi, Patel, and Júnior (2016) proposed a crawling method based on link and content that used a hierarchical structure called T-Graph to assign an appropriate priority score to each unvisited link to prioritize the links.

(2) In the process of crawling webpages, focused crawlers need to evaluate the relevance of webpages. The principle of the most widely used correlation judgment is to compare the difference between the target webpage and the topic model. Currently, concept context graph (CCG) (Peng, Du, Hai, Chen, & Gao, 2008; Du, Pen, & Gao, 2013) and ontology (Du, Li, & Wang, 2006) are the most frequently used topic models. Du, Li, Hu, Li and Chen (2016) built CCG based on the formal concept analysis (FCA) and proposed relation context graph (RCG), link context graph (LCG), concept similarity context graph (CSCG), and path trust knowledge graph (PTKG). The CCG is used to store the knowledge context based on the history of the user’s clicking on the webpage, and guide the crawler to obtain the webpages with high relevance about the topics of interest to the user. The construction methods based on context graph (CG) rely on the historical records of users’ search pages. Different people have different understanding of knowledge, which may lead to deviation in the description of topics in CG, reducing the accuracy of crawlers. Therefore, more focused crawlers use ontology as a topic model. Sharma and Khan (2015) proposed an ontology-based adaptive framework, which used ontology to calculate the concept similarity between terms. Daoui, Gherabi, and Marzouk (2017) combined mathematical factors such as the concept depth and the concept density to obtain semantic similarities between concepts of the ontology.

(3) In recent years, some scholars have introduced intelligent optimization algorithms into the focused crawlers to improve their global search abilities. Yan and Pan (2018) proposed a focused crawler strategy based on the improved genetic algorithm. They used the vector space model (VSM) to calculate topic relevance, used the improved PageRank algorithm to calculate topic importance, and optimized genetic operations based on user browsing behavior. Chen, Zhang, and Zhang (2011) proposed a focused crawler search strategy based on ant colony algorithm and improved the way of pheromone updating so that the algorithm can adjust pheromone dynamically and adaptively to avoid the local convergence of ant colony algorithm. Liu and Du (2014) proposed a focused crawler strategy based on the cell-like membrane computing optimization algorithm (CMCFC). CMCFC used evolution rules and communication rules in membrane to determine the weight factors of hyperlinks in relation to documents (including the full text of the page, anchor text, the title text of the page, and the surrounding text of the paragraph), and then calculated the topic relevance through the VSM. Zheng (2011) applied the combination of the genetic algorithm (GA) and the ant algorithm (AA) denoted by the genetic algorithm-ant algorithm (GAAA) to the focused crawler, which made full use of the GA’s fast, random and global convergence advantages and the AA’s parallelism, positive feedback and low time consuming properties. Dewanjee (2016) produced more accurate results by a recently formulated metaheuristic optimization algorithm called cuckoo searching. The basic motivation of the cuckoo search algorithm was to minimize the number of URLs when the crawler used the URL to find any particular content.

With the gradual maturity of the machine learning technology, machine-learning algorithms have been widely used in focused crawler. Some scholars utilized machine learning techniques such as learning automata (Suebchua, Manaskasemsak, Rungsawang, & Yamana, 2017), reinforcement learning (Han, Wuillemin, & Senellart, 2018), Naïve Bayes (Saleh, Abulwafa, & Rahmawy, 2017) and support vector machines (Hosseinkhani, Taherdoost, & Keikhaee, 2021) to enhance crawling technology. With regard to those crawlers under machine learning, the quality of the training sample data directly affects the results of the crawlers.

The main research issues of the focused crawler technology include the establishment of topic models and the design of crawler strategies. In the process of crawling webpages, focused crawlers need to evaluate the relevance of webpages. The principle of the most widely used correlation judgment method is to compare the difference between the target webpage and the topic model. The focused crawler strategy needs to consider the calculation of topic relevance and the crawling order of hyperlinks. The calculation of the topic relevance of hyperlinks is mainly based on the webpage text and the hyperlink structure. The hyperlinks with lower relevance are filtered through a preset threshold. Because different crawling sequences can cause the crawler to perform differently, the design of the crawler strategy is a key factor in designing the focused crawler.

In this article, we use domain ontology as the topic model. A semi-automatic domain ontology construction method based on ontology learning technology is used to improve the quality of domain ontology construction. In order to prevent the problem of “topic drift”, the relevancies of both webpage text and link structure are considered. In the process of applying the intelligent optimization algorithm to select the optimal hyperlink, the traditional single-objective optimization method based on the weighted summation has the disadvantage that it is difficult to determine the optimal weight coefficients reasonably. Therefore, we constructed a multi-objective optimization model based on multiple evaluation criteria of hyperlink relevance. Subsequently, the multi-objective ant colony optimization (MOACO) algorithm is introduced into the focused crawler to select a set of Pareto-optimal links for the first time.

The main contributions of this article embody in two aspects. (1) A semi-automatic construction method of domain ontology based on ontology learning combining latent Dirichlet allocation (LDA) and Apriori algorithm is proposed to construct topic model, where the LDA is used to obtain domain concept terminology from domain knowledge and the Apriori algorithm is applied to obtain the relationships between

concepts from the topic set trained by LDA. (2) A MOACO algorithm by combining the fast non-dominated sorting and the nearest farthest candidate solution (NFCS) method is proposed to select Pareto-optimal hyperlinks and guide the crawlers' search direction. The experimental results of the focused crawling on the domain knowledge of typhoon disasters and rainstorm disasters prove that the effectiveness of the proposed focused crawlers to retrieve topic-relevant webpages.

This article is structured as follows. Section 2 presents the construction method of domain ontology. Section 3 introduces the calculation method of topic relevance based on ontology. Section 4 describes a multi-objective optimization model and a focused crawling strategy based on ontology and MOACO algorithm. The experimental results and discussion of different focused crawlers on typhoon disasters and rainstorm disasters are shown in Section 5. The conclusion and future work are presented in Section 6.

2. Ontology learning-based ontology construction

In this section, we first introduce the ontology and the domain ontology. Then, the process of the domain ontology construction based on ontology learning is described. Finally, taking typhoon disaster as an example, the domain ontology of typhoon disaster is introduced and displayed.

2.1. Ontology

Ontology (Gruber, 1995) is a knowledge system shared, conceptualized, and formalized in a certain field. It can be divided into domain ontology, general or commonsense ontology, knowledge ontology, linguistic ontology, and task ontology. Among them, the domain ontology is a specialized ontology that describes the knowledge of the specified domain. It gives the rich relations between concepts. The main components in the ontology include classes, instances, attributes, relations and axioms (Gruber, 1993; Khadir, Aliane, & Guessoum, 2021), where classes are types of objects that define a category of entities and represent the concepts of the ontology, instances (individuals) are instantiations of classes and represent the ontology population, attributes (properties) are associated with classes or objects and may include the statements of datatypes and their values, relations (relationships) define the relatedness between objects and generally have taxonomic and non-taxonomic relations, and axioms (rules) represent formal definitions of the ontology knowledge. The above ontology components are not all necessary to the creation of every ontology. It can consist of only part of ontology entities (Khadir, Aliane, & Guessoum, 2021).

The methods of the ontology construction are mainly divided into three types: artificially constructing ontology, semi-automatically building ontology, and automatically constructing ontology. The methods of artificially building ontology are time-consuming and laborious, and the constructed ontology is bound by the builders' knowledge reserve and subjective consciousness. The method of automatically constructing ontology draws on relevant technologies of knowledge acquisition, such as natural language rules and machine learning methods based on statistical analysis. In the process of using machine-learning methods, due to various reasons such as word segmentation and semantic expression, a large amount of noise data may be generated during automatic extraction. The concepts of extraction are loose and the credibility cannot be guaranteed. This will affect the consistency of the ontology to a certain extent, and the process of semantic reasoning is generally affected by the noise data. The methods of constructing the ontology automatically are less, difficult to implement, and the quality of the constructed ontology is not high. Ontology learning is a semi-automatic process of constructing the ontology under the guidance of the user. The semi-automatically building ontology method considers multiplexing existing ontology, such as WordNet, Ontology Engineering Group, and so on. Therefore, in this article, we adopt the semi-automatically building ontology method to construct ontology. In the ontology construction process, we use the formal concept analysis (FCA) to construct the skeleton of the ontology and enrich the concepts of ontology through the latent Dirichlet allocation (LDA) and the Apriori algorithm.

2.2. Formal concept analysis

FCA is a mathematical method based on data analysis and knowledge formal expression (Peng, Du, Hai, Chen, & Gao, 2009; Du, Pen, & Gao, 2013). It analyzes data from the formal context, extracts the rules between concepts and concepts, and realizes the discovery, sorting and display of concepts. Its main data structure "concept lattice" is a graph model composed of multiple concept nodes. The concept node consists of two parts: extension and connotation. In FCA, the collection of all objects in a concept is called the extension of the concept. The set of attributes shared by the objects contained in a concept is called the connotation of the concept. In the following, we introduce the definition of the formal context (Li, Mei, & Lv, 2013; Jiang, 2019; Rocco, Hernandez-Perdomo, & Mum, 2020).

Definition 1. *The formal context can be defined as a triplet* $F = (O, A, I)$, *where* $O=\{O_1, O_2,\ldots, O_b\}$ *and* $A=\{A_1, A_2,\ldots, A_c\}$ *are sets, and* $I \subseteq O \times A$ *is a binary relationship. The elements of O are called objects, and the elements of A are called attributes. If* $(O_i, A_j) \in I$ $(1 \le i \le b, 1 \le j \le c)$, *object* O_i *has attribute* A_j.

A formal context is a simple way of specifying which objects have which attributes. It is a data matrix where rows correspond to objects and columns to attributes. The binary relationship is represented by 1 or 0 to reflect the presence or the absence of the attribute for that object. In general, replace 1 s by "X" and leave a blank space for 0 s.

Example 1. *Through the IK-Analyzer* (Wang, & Meng, 2014), *an open-source word segmentation tool, we segment the documents, extract and count terms to obtain the formal context composed of document set and term set. We create a formal context* $F = (Documents, Terms, R)$, *as shown in* Table 1, *where "Documents" represents a set of objects, "Terms" represents a set of attributes, and "R" represents a binary relationship between an object and an attribute. We use the concept explorer ConExp tool (freely available at URL:* http://sourceforge.net/projects/conexp) *to mine conceptual relationships from formal backgrounds and automatically generate concept lattices. The concept lattice is displayed in the form of a Hasse graph, as shown in* Fig. 1. *In* Fig. 1, *a term represents an attribute, and a doc represents an object. Term* 1 *through* 6 *represent the attribute set, and doc* 1 *through* 5 *represent the object set. The relationship between terms is defined as the hyponymy relationship of the attributes. After building the concept lattice, we use the ontology web language (OWL) to formalize the concept hierarchy. In this process, the visualization of OWL is written and implemented by the development tool Protégé* (Noy, Sintek, Decker, Crubezy, Feigerson, & Musen, 2005).

In Fig. 1, the upper half of the node circle represents attributes, and the lower half represents objects. If the attribute part of a node is blue, it means that there is a new attribute linked to the node. If the object part of a node is black, it means that a new object is linked to the node. The attribute set of each concept node is the sum of all the attributes of the upper level of the node (inheriting the parent concept attributes), and

Table 1
The binary relation of the formal context.

Documents	Terms					
	term 1	term 2	term 3	term 4	term 5	term 6
doc 1	×			×	×	×
doc 2	×	×	×			
doc 3			×		×	
doc 4	×					×
doc 5	×	×	×	×	×	

"×" means that the doc has the term.

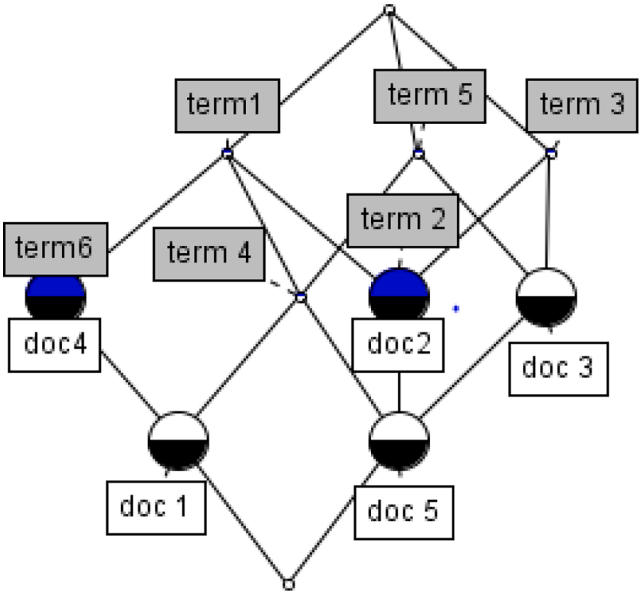


Fig. 1. Hasse graph of the concept lattice that corresponds to the relations in Table 1.

the object set is the sum of all the objects at the lower level of the node (covering the sub-concept objects). For example, the attribute set of the node "doc 2, term 2" in Fig. 1 is {term 1, term 2, term 3}, and its object set is {doc 2, doc 5}. As reported by Rios-Alvarado, Lopez-Arevalo, and Sosa-Sosa (2013), a hyponym is defined as a word of more specific meaning than a general or superordinate term applicable to it, and a hypernym as a word with a broad meaning constituting a category under which more specific words fall. Thus, "term 1" and "term 3" are the hypernym of "term 2", and "term 2" is the hyponymy of "term 1" and "term 3". The hyponymy of other terms in the figure can be obtained similarly.

2.3. Latent dirichlet allocation

Latent Dirichlet allocation (LDA) (Jelodar, Wang, Yuan, Feng, Jiang, Li, & Zhao, 2019) is a probabilistic topic model for modeling discrete data sets. In LDA, for a document, each word in the document is obtained through a process of "selecting a certain topic with a certain probability and selecting a certain word from this topic with a certain probability". The probability of each word appearing in a document is:

$$P(word|document) = \sum_{topic} P(word|topic) \times P(topic|document) \quad (1)$$

Eq. (1) can be represented by the matrix of Fig. 2. In Fig. 2, the "document-word" matrix represents the probability of occurrence of each word in each document; the "topic-word" matrix represents the probability of occurrence of each word for each topic; the "document-topic" matrix represents the probability of occurrence of each topic in each document. The LDA model calculates the word frequency of each word in each document from a set of documents that have been acquired to obtain a "document-word" matrix. Through this matrix training, the "topic-word" matrix and the "document-topic" matrix are obtained. We assume that there are D documents, T topics, and a total of V words. Fig. 3 shows the generation process of the LDA model.

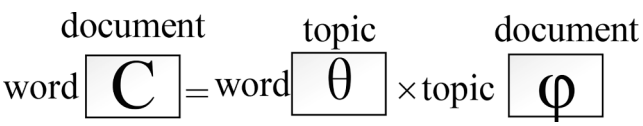


Fig. 2. Matrix representation of the LDA model.

In Fig. 3, N_d indicates that the d-th document is composed of N_d words; α is a hyperparameter of the Dirichlet polynomial distribution, which is a T-dimensional vector; β is a hyperparameter of the Dirichlet polynomial distribution, which is a V-dimensional vector.

- Take a corresponding T-dimensional topic distribution θ_d (the topic of the d-th document) from the probability density function of the α-controlled Dirichlet polynomial distribution.
- Generate T corresponding word distributions φ_t from the probability density function of the β-controlled Dirichlet distribution.
- $Z_{d,n}$ represents the n-th topic of the d-th document. When $n = 2$, it means that the second topic of the d-th document is taken, and the word distribution φ_t (the word distribution of the corresponding topic) of the second topic generated by the corresponding β is obtained.
- Randomly pick a word from φ_t as the value of $W_{d,n}$ (the word corresponding to the n-th topic of the d-th document).
- Loop through the above steps to get the words corresponding to each topic.

In LDA, it is difficult to determine the number of topics. In this article, we first set a Q value for the initial number of topics, and obtain the initial model through training. According to the initial model, the similarity between topics is calculated. The average similarity of these topics is obtained. By adjusting (increasing or decreasing) the Q value, the target model is retrained to calculate the similarity between topics and their average similarity. Compare all average similarities and find the minimum average similarity. The Q value corresponding to the minimum average similarity is regarded as the number of topics in LDA.

2.4. Apriori algorithm

Association rules reflect the interdependence and correlation between a thing and other things. As an important technology of data mining, it is used to mine the correlation between valuable data items from a large amount of data. Two thresholds of support and confidence are often used to measure the degree of association.

Definition 2. *Support refers to the proportion of records in the data set that contain the item set, such as the probability that both item-sets {X,Y} appear simultaneously. Set a threshold as the minimum support, eliminate the item-sets with lower frequency, and keep the more frequent item-sets. These reserved item-sets are called frequent item-sets.*

Definition 3. *Confidence is the probability that another data will appear after one data appears, or the probability of event B occurring on the base of event A. For example, in the supermarket shopping data, there are beer versus diapers. Given support* = 60%, *confidence* = 60%, *which means that in all data, beer and diapers appear together at a frequency of* 60%, *and in the purchase of beer,* 60% *also purchase a diaper. Set a threshold as the minimum confidence. A rule that satisfies both the minimum support threshold and the minimum confidence threshold is called a strong rule* (Sornalakshmi et al., 2020; Wang, & Yang, 2018). *In this article, the support and confidence are obtained through many experiments. The support is set to* 0.5 *and the confidence to* 0.6.

The basic idea of the Apriori algorithm (Sornalakshmi et al., 2020) is to calculate the support of the item set by scanning the data set multiple times and find all the frequent item-sets to generate the association rules. In the Apriori algorithm, if a set is a frequent item-set, all its subsets are frequent item-set. If a set is not a frequent item-set, then all of its supersets are not frequent item-set. The flow of the Apriori algorithm is as follows:

- First search for the candidate 1 item-sets and compute the corresponding support degree, and remove the candidate 1 item-sets

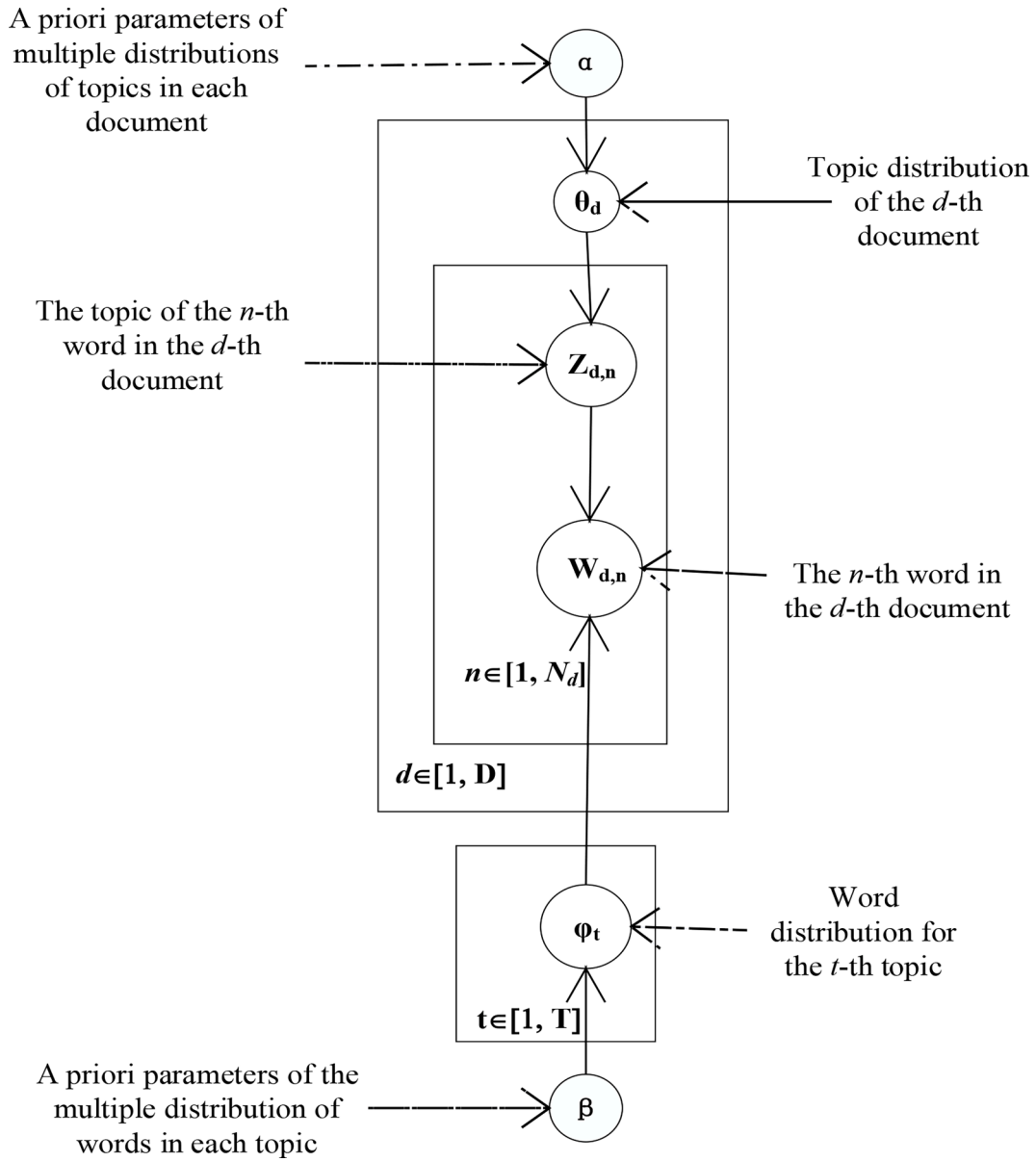


Fig. 3. Generation process of the LDA model.

below the minimum support degree to obtain the frequent 1 item-sets.

- Connect the frequent 1 item-sets and itself to obtain the candidate frequent 2 item-sets, and filter out the candidate frequent 2 item-sets below the minimum support degree to obtain the frequent 2 item-sets.
- Use frequent 2 item-sets to find frequent 3 item-sets, and so on, until it can't find frequent $k + 1$ item-sets. All rules above the minimum confidence are extracted from the frequent item-sets.

The Apriori algorithm requires two steps of join and pruning when generating frequent k-item sets L_k from frequent k-1 item-sets $L_{k\text{-}1}$.

- Join step: To find L_k (a set of all frequent k item-sets), a set C_k of candidate k item-sets is generated by concatenating $L_{k\text{-}1}$ (a set of all frequent k-1 item-sets) with itself. Let l_1 and l_2 be the item-sets in $L_{k\text{-}1}$. $l_i[j]$ represents the j-th item in the item set l_i ($i = 1, 2$). If ($l_1[1] = l_2[1]$)&&($l_1[2] = l_2[2]$)&&… && ($l_1[k\text{-}2] = l_2[k\text{-}2]$) && ($l_1[k\text{-}1] < l_2[k\text{-}1]$), it is considered that l_1 and l_2 are connectable. The results produced by the connections l_1 and l_2 are {$l_1[1]$, $l_1[2]$, …, $l_1[k\text{-}1]$, $l_2[k\text{-}1]$}.
- Pruning step: Since C_k is a superset of L_k, members of C_k may or may not be frequent. The degree of support for each candidate member in the C_k is determined by scanning all the data. If a candidate's support is not less than the minimum support, the candidate item set is considered to frequent. In order to compress C_k, the Apriori property can be utilized: all non-empty subsets of any frequent item-sets must also be frequent. Conversely, if a candidate non-empty subset is not frequent, then the candidate is certainly not frequent, so it is removed from the C_k.

2.5. Domain ontology construction

Due to there are few ontologies that can be reused in the field of meteorological disasters, we use the formal concept analysis (FCA) and the Chinese classification thesaurus with its English version to construct the skeleton of the domain ontology. Subsequently, compare it with the concepts in WordNet to construct an initial ontology. The specific ontology construction process is as follows:

(1) Search for 20 related papers from the CNKI (China National Knowledge Internet) database with certain a theme, where the CNKI digitizes 80% of China's knowledge and information resources including papers and dissertations, which people can download through the cross-databases. The titles, abstracts, and keywords in these papers are extracted as a set of domain term candidates. The candidate set is segmented using the IK-analysis (Wang, & Meng, 2014) tokenizer and the word frequency is counted.

(2) Use formal analysis based on these terms and their word frequencies to construct a concept lattice. A term in the concept lattice is taken as an attribute, and the upper and lower relationships of the term are placed at the upper and lower positions of the attribute, and a simple initial ontology is constructed by using the concept lattice and the Chinese classification thesaurus (with its English version).

(3) Compare the concepts in the initial ontology with the concepts in WordNet. If the concept A appears on both the initial ontology and the WordNet ontology tree, we find the concept nodes near the concept A on the WordNet ontology tree, and use the nonlinear combination function proposed by Li, Bandar and David (2003) to calculate the conceptual similarity between these nearby concept nodes and concept A. We set a threshold γ. If the correlation of concept B that exists on the WordNet ontology tree is greater than γ, we add concept B to the initial ontology. Calculate the conceptual relevance between concept B and every conceptual node near concept A on the initial ontology. Place concept B next to the conceptual node with the greatest correlation, and obtain the adjusted the initial ontology.

(4) Search for keywords through Google, Bing, Baidu, and other search engines, and select 30 top-ranked webpages.

(5) Analyze and segment the text of each webpage, and use the IK-Analyzer to select the nouns in the webpage text as the concept candidate set.

(6) Use the topic model LDA (Jelodar, Wang, Yuan, Feng, Jiang, Li, & Zhao, 2019) to obtain 20 topic sets from the concept candidate set.

(7) Set the minimum support threshold Q, and use the Apriori algorithm in association rule to obtain strong rule concept pairs greater than Q from 20 topic sets. After these concept pairs are extended to the adjusted initial ontology, a domain ontology is finally formed. The construction process of domain ontology based on ontology learning is shown in Fig. 4.

According to the above ontology construction process, we construct two domain ontologies based on typhoon disasters and rainstorm disasters. Take the ontology of typhoon disasters as an example to display the ontology skeleton. Because this ontology has a 7-level hierarchical structure and contains 97 concepts, in order to display plainly the structure of the domain ontology, the extendible parts (which are framed by rectangles in the figure) of the ontology are displayed in different sub-graphs (see Fig. 5).

3. Topic relevance calculation based on ontology

To compute the topic relevance of the webpage, we first extract all the concepts from the constructed ontology and construct the topic vector $TK=\{tk_1, tk_2, \ldots, tk_h\}$, where h denotes the number of topic words, and then compute the concept semantic similarity between the topic concept and any concept from the topic vector based on the ontology. Afterward, based on the concept semantic similarity, the calculation methods of the topic relevance of the webpage text and the hyperlink are described. Finally, the comprehensive priority of the unvisited hyperlink is given to direct the search of the focused crawler.

3.1. Concept semantic similarity calculation

To compute the semantic similarity of two concepts, we comprehensively consider the five impact factors of similarity (Ma, Li, Lian, Liang, & Chen, 2016): semantic distance (IF_{Dis}), concept density (IF_{Den}), concept depth (IF_{Dep}), concept coincidence degree (IF_{Coi}), and concept semantic relationship (IF_{Rel}). The definitions of IF_{Dis}, IF_{Den}, IF_{Dep}, and IF_{Coi} can refer to the literature (Ma, Li, Lian, Liang, & Chen, 2016). However, when analyzing IF_{Rel}, we give a different definition from the literature as follows.

Definition 4. *In the domain ontology, we consider five concept semantic relations that are "synonym", "induced-by", "is-a", "part-of", and "coordinate-term-of", respectively. With reference to the opinions of experts in the field, the weights of the five relationships are set as* 1, 1/2, 1/3, 2/7 *and* 1/4, *respectively. The impact factor IF_{Rel} of concept semantic relations between two concepts C_1 and C_2 on the semantic similarity can be expressed by Eq. (2).*

$$IF_{Rel} = \frac{\sum_{i=1}^{Ns} Value(C_i, F_i)}{Ns} \tag{2}$$

where

$$Value(C_i, F_i) = \begin{cases} 1 & Synonym(C_i, F_i) \\ 1/2 & Induced\text{ - }By(C_i, F_i) \\ 1/3 & Is\text{ - }a(C_i, F_i) \\ 2/7 & Part\text{ - of}(C_i, F_i) \\ 1/4 & Coordinate\text{ - term - of}(C_i, F_i) \end{cases}$$

Here, Ns represents the number of edges on the shortest path between concept C_1 and concept C_2. C_i represents the i-th concept node on the shortest path, and F_i represents the parent concept of C_i. $Value\ (C_i, F_i)$ represents the directed edge weight between the concept node C_i and its parent concept node F_i.

Combining the above five impact factors, the concept semantic similarity $sim(C_1, C_2)$ between concept C_1 and concept C_2 is calculated as follows:

$$sim(C_1, C_2) = k_1 \times IF_{Dis} + k_2 \times IF_{Den} + k_3 \times IF_{Dep} + k_4 \times IF_{Coi} + k_5 \times IF_{Rel} \tag{3}$$

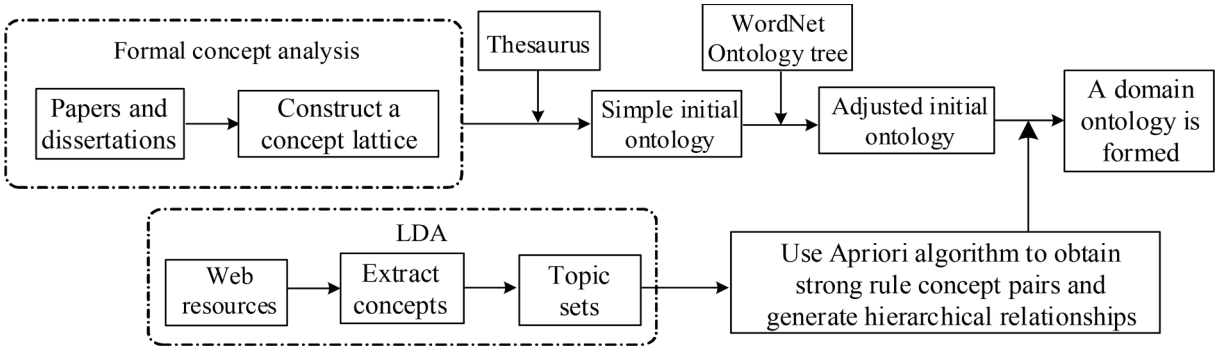


Fig. 4. Construction process of domain ontology.

(a)

Fig. 5. Domain ontology about typhoon disasters: (a) Main part of typhoon ontology; (b) The precaution part of the typhoon ontology; (c) The typhoon forecast part of the typhoon ontology; (d) The secondary disaster part of the typhoon ontology; (e) The direct disaster part of the typhoon ontology; (f) The emergency part of the typhoon ontology; (g) The government part of the typhoon ontology; (h) The reconstruction part of the typhoon ontology; (i) The early warning part of the typhoon ontology; (j) The flood part of the typhoon ontology; (k) The rain disasters part of the typhoon ontology.

Here, k_1, k_2, k_3, k_4, and k_5 are adjustable factors, obtained according to expert experience, and satisfy $k_1 + k_2 + k_3 + k_4 + k_5 = 1$.

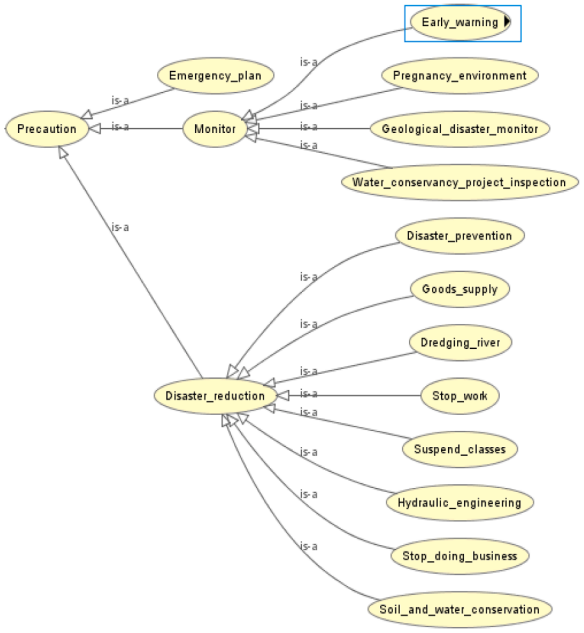
In order to obtain the topic semantic weight vector, we first determine a topic concept C, and then calculate the concept semantic similarity between topic concept C and each topic word in the topic vector TK=$\{tk_1, tk_2, \ldots, tk_h\}$ according to Eq.(3), where h is the number of topic words. Finally, we obtain the topic semantic weight vector $W_{TK} = \{w_{tk_1}, w_{tk_2}, \ldots w_{tk_i} \ldots, w_{tk_h}\}$ by the Eq.(4).

$$W_{TK} = (sim(C, tk_1), sim(C, tk_2), \ldots, sim(C, tk_h)) \quad (4)$$

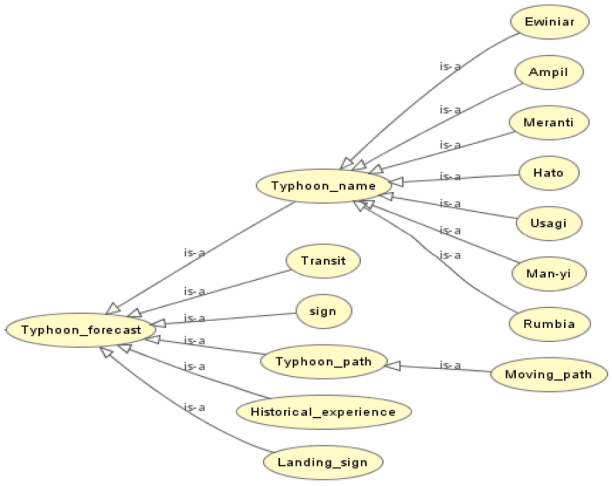
Here, w_{tk_i} represents the topic semantic weight of the i-th topic word in the topic vector, that is, the semantic similarity value between the topic concept C and the topic word tk_i.

3.2. Topic relevance calculation

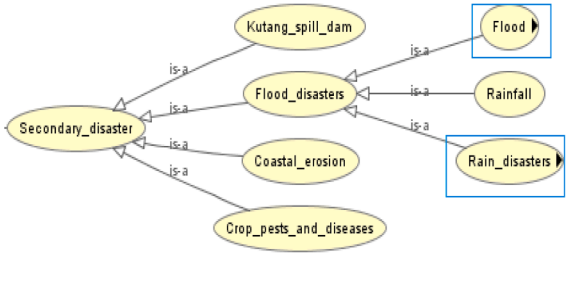
The calculation methods of the topic relevance of the webpage text



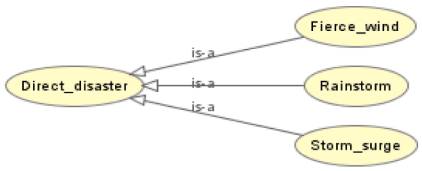
(b)



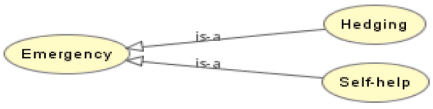
(c)



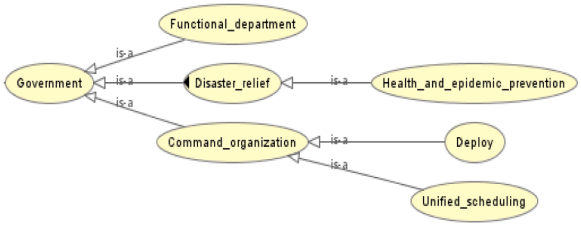
(d)



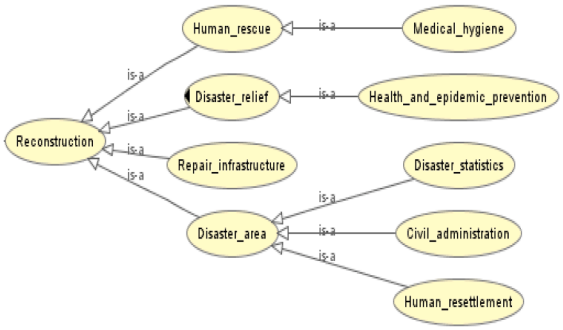
(e)



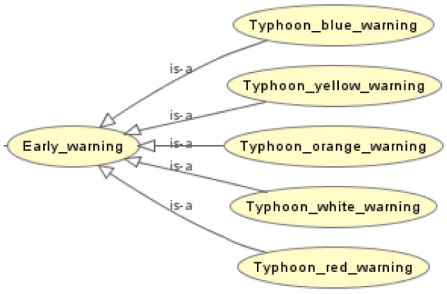
(f)



(g)



(h)



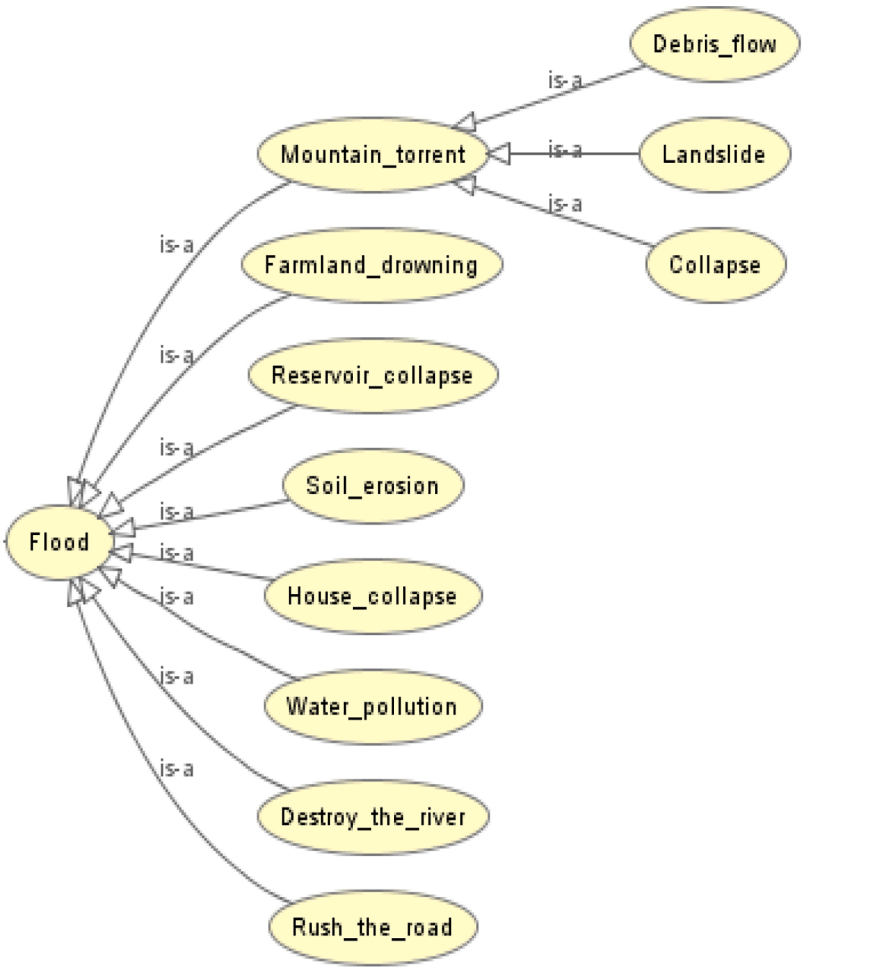
(i)

Fig. 5. (*continued*).

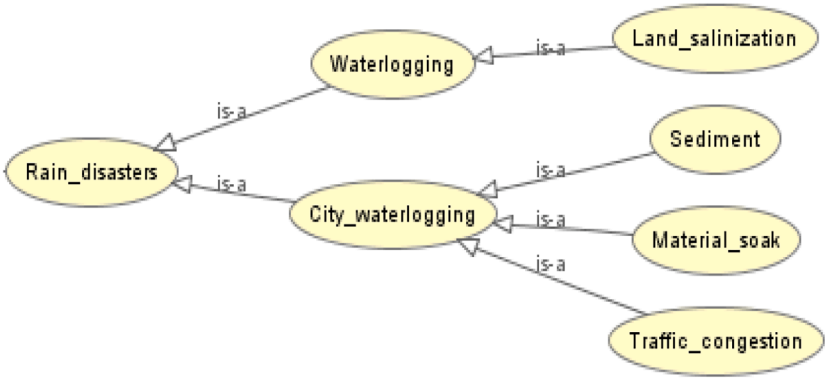
and the hyperlink based on the conceptual semantic similarity are given in this section.

3.2.1. Topic relevance of webpage text

Hypertext markup language (HTML) webpages are widely used on the World Wide Web because of their simplicity, scalability, platform independence, and versatility. HTML uses tag symbols to mark the content of various parts of a webpage that needs to be displayed. Since the content of different labels has different influences on the topic relevance of the entire webpage, different labels are given different



(j)



(k)

Fig. 5. (*continued*).

weights. We divide the main labels selected into 5 groups: G_1 = (title, keyword, description, first-level title), G_2 = (second-level title, third-level title), G_3 = (fourth-level title, fifth-level title, sixth-level title, bold text), G_4=(body information), G_5=(non-body information). At the same time, we give different weights to the labels of different groups: W=(2, 1.5, 1.2, 1.0, 0.2).

A webpage text is mapped into a feature vector DK={dk_1, dk_2, …, dk_h}, and the corresponding feature weight vector W_{DK}={w_{dk_1}, w_{dk_2}, …w_{dk_i}…, w_{dk_h}} is computed by using the following equation:

$$w_{dk_i} = \sum_{l=1}^{L} tf_{i,l} \times W_l = \sum_{l=1}^{L} \left(\frac{f_{i,l}}{max f_{i,l}} \times W_l\right) \tag{5}$$

Here, $tf_{i,l}$ is the term frequency (TF) after the i-th subject word is normalized in the l-th position of the webpage text. $f_{i,l}$ is the TF of the i-th topic word at the l-th position of the webpage text. max $f_{i,l}$ is the maximum TF of the groups in which the i-th topic word appears in all

occurrences of the webpage text. L is the number of tag groups (here, $L = 5$). W_l is the weight of the l-th label.

We use the vector space model (VSM) to calculate the topic relevance of the webpage text, that is, calculate the cosine of the topic word weight vector W_{TK} and the web text feature weight vector W_{DK}. The calculation of the topic relevance of webpage P is given by Eq. (6).

$$R(P) = Sem(TK, DK) = \frac{W_{TK} \times W_{DK}}{||W_{TK}|| \times ||W_{DK}||} = \frac{\sum_{i=1}^{h}(w_{tk_i} \times w_{dk_i})}{\sqrt{\sum_{i=1}^{h} w_{tk_i}^2} \times \sqrt{\sum_{i=1}^{h} w_{dk_i}^2}} \tag{6}$$

The value range of $R(P)$ is [0, 1]. The smaller the angle between the topic word weight vector and the webpage feature weight vector is, the larger the topic relevance $R(P)$ of webpage text is. We set a threshold σ. If $R(P) \geq \sigma$, webpage P is considered to be related to a pre-selected topic.

3.2.2. Topic relevance of hyperlinks

In the process of the focused crawling, we use the topic relevance of the hyperlink anchor text, the topic relevance of the webpage to which the hyperlink points, and the average value of the topic relevance of all web pages containing the hyperlink, to determine whether the hyperlink is related to the topic.

For the anchor text of hyperlink l, we use an improved Term Frequency × Inverse Document Frequency (TF × IDF) (Liu, & Du, 2014) to calculate the feature weight of the anchor text of the hyperlink. The improved TF × IDF reflects the importance of a word to an anchor text in the text set. The importance of a word increases in proportion to the number of times it appears in the anchor text, but decreases in inverse proportion to the frequency of its occurrence in the text set. The weight w_{ak_i} of the i-th topic word in an anchor text is computed by equation (7).

$$w_{ak_i} = TF_i \times IDF_i = \frac{f_i}{\sum_{m=1}^{h} f_m} \times log_k\left(\frac{N}{N_i} + 0.01\right) \quad i = 1, 2, ..., h \tag{7}$$

Here, f_i represents the TF of the i-th topic word appearing in anchor text A_l of hyperlink l, N the total number of webpages crawled, h the number of topic words, N_i the number of webpages containing the i-th topic word, and k (greater than1) a real number.

Similar to the method of calculating the topic relevance of webpage texts, we also use the VSM method to calculate the topic relevance of the anchor text. According to Eq. (6), the topic relevance $R(A_l)$ of anchor text A_l is as follows.

$$R(A_l) = Sem(TK, AK) = \frac{W_{TK} \times W_{AK}}{||W_{TK}|| \times ||W_{AK}||} = \frac{\sum_{i=1}^{h}(w_{tk_i} \times w_{ak_i})}{\sqrt{\sum_{i=1}^{h} w_{tk_i}^2} \times \sqrt{\sum_{i=1}^{h} w_{ak_i}^2}} \tag{8}$$

The value of $R(A_l)$ is between 0 and 1, the closer its value is to 1, the higher the topic relevance of the anchor text of hyperlink l is. AK={ak_1, ak_2,…, ak_h} is the feature vector of anchor text A_l.

For the webpage P_u pointed to by hyperlink l, UK={uk_1, uk_2, …uk_i, …, uk_h} is used to represent the text feature vector of the webpage P_u, and the corresponding text feature weight vector W_{uk}={w_{uk_1}, w_{uk_2}, …w_{uk_i}…, w_{uk_h}} is calculated by the TF-IDF model (see Eq. (5)). Here, w_{uk_i} represents the weight of the i-th topic word in the webpage text P_u. According to Eq. (6), the relevance of the webpage P_u pointed to by hyperlink l is as follows.

$$R(P_u) = \text{Sem}(TK, UK) \tag{9}$$

Based on the above analysis, the comprehensive priority $Priority(l)$ of the unvisited hyperlink l is computed by the equation (10).

$$Priority(l) = t_1 \times R(A_l) + t_2 \times \frac{1}{m}\sum_{i=1}^{m} R(P_i) + t_3 \times R(P_u) \tag{10}$$

Here, t_1, t_2, and t_3 are the weight coefficients of the topic relevance of anchor text A_l of hyperlink l, the average value of the topic relevance of the webpages P_i (i = 1, 2,…, m) where hyperlink i is located, and the topic relevance of the webpage P_u to which hyperlink l points, respectively. Satisfy $t_1 + t_2 + t_3 = 1$. In this article, we set $t_1 = 0.6$, $t_2 = 0.15$, and $t_3 = 0.25$. Here, P_i (i = 1, 2, …, m) are all webpages containing hyperlink l. We set a threshold η for the topic relevance of the hyperlink. If $Priority(l) \geq \eta$, we add this hyperlink into the waiting queue.

4. Focused crawling based on ontology and MOACO algorithm

In this section, we first build a multi-objective optimization model for evaluating unvisited hyperlinks, and then a focused crawler strategy based on ontology learning and multi-objective ant colony optimization (OLMOACO) is proposed. The non-dominated sorting method (Huang, Ye, & Cao, 2017; Deb, Pratap, Agarwal, & Meyarivan, 2002) and the nearest farthest candidate solution (NFCS) method (Liu, Liu, Liu, & Li, 2020) are used to select unvisited hyperlinks to guide the crawler's search direction.

4.1. Multi-objective optimization model for evaluating hyperlinks

The topic relevance of a hyperlink mainly depends on the link structure and the content of the webpages related to the hyperlink. Thus, the PageRank (PR) value and the topic relevance of the webpage to which the hyperlink points and the topic relevance of the webpage where the hyperlink is located can be used to evaluate the topic relevance of the hyperlink. In addition, because the anchor text of the hyperlink tends to display clearly the feature words of the webpage to which the hyperlink points, the topic relevance of the anchor text of the hyperlink is also an important ingredient of evaluating the hyperlink. Therefore, in this article we use the topic relevance of the anchor text, the average topic relevance of the webpages where the hyperlink is located, the PageRank (PR) value and the topic relevance of the webpage to which the hyperlink points as four objective functions, and establish a multi-objective optimization model for evaluating unvisited hyperlinks, as shown in Eqs.(11)-(14). When the evaluation of the hyperlinks involves more than one objective, which is conflicting, the selection of the hyperlinks is treated as a multi-objective optimization problem.

$$\max\ F_1(l) = R(A_l) \tag{11}$$

$$\max\ F_2(l) = \frac{1}{m}\sum_{i=1}^{m} R(P_i) \tag{12}$$

$$\max\ F_3(l) = R(P_u) \tag{13}$$

$$\max\ F_4(l) = PR(P_u) \tag{14}$$

Here, $F_1(l)$ represents the topic relevance of anchor text A_l of hyperlink l, $F_2(l)$ the average topic relevance of webpages that contain hyperlink l, $F_3(l)$ the topic relevance of the webpage P_u pointed to by hyperlink l, and $F_4(l)$ the PR value of the webpage P_u pointed to by hyperlink l. In Eq.(12), m is the number of webpages containing hyperlink l. $PR(P_u)$ is an improved PR value of P_u adopted by the literature (Ma, Li, Lian, Liang, & Chen, 2016), shown in Eq. (15).

$$PR(P_u) = (1 - d) + d \times \sum_{i=1}^{s}\left[\frac{PR(P_i)}{C(P_i)} \times (1 + \omega \times R(A_i))\right] \tag{15}$$

Here, d is the damping coefficient. s is the total number of all in-links of P_u in the crawled webpage set. $PR(P_i)$ is the PR value of the i-th in-link webpage of the webpage P_u. $C(P_i)$ represents the total number of out-

links of the webpage P_i. ω is an adjustable factor. $R(A_i)$ is the topic relevance of anchor text A_i of the i-th in-link of P_u.

4.2. Multi-objective ant colony optimization algorithm for focused crawler

In the multi-objective ant colony optimization (MOACO) algorithm (Angus, 2007; Liu, & Liu, 2019), the search process of the ant is not only affected by the pheromones that left on the path, but also by the optimal experience of the entire group. In focused crawling based on the MOACO, the crawling path construction of the ants, the update of the pheromones, and the selection of the optimal hyperlinks are three important factors affecting the algorithm.

4.2.1. Path construction

For the k-th ant, suppose that at time t, the webpage where it is located is P_i. If there is a hyperlink in the P_i pointing to the webpage P_j, the ant in P_i will decide whether to move from P_i to P_j according to certain a rule. Suppose that V represents a new set of pages pointed to by all the hyperlinks in the P_i, and the pseudo-random ratio selection rule is used to calculate the probability with which the k-th ant arrives at the webpage P_j from the current webpage P_i. The pseudo-random ratio selection rule is shown in Eq. (16).

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\psi}[\mu_{ij}]^{\zeta}}{\sum_{l(i,j)\in E}[\tau_{ij}(t)]^{\psi}[\mu_{ij}]^{\zeta}}, & P_j\in V\\ 0, \quad otherwise\end{cases} \tag{16}$$

Here, $p_{ij}^{k}(t)$ denotes the probability with which the k-th ant at the current page P_i selects the page P_j as the next crawling object at time t, $\tau_{ij}(t)$ the pheromone quantity on the hyperlink (edge) $l(i, j)$ from page P_i to page P_j, and E all hyperlinks by which P_i can reach the new webpage P_j ($P_j \in V$). μ_{ij} represents the heuristic information which in this article is the value obtained by linearly weighted summing of the anchor text topic relevance of the hyperlink $l(i, j)$ and the topic relevance of the webpage P_j. ψ and ξ are the parameters for pheromone and heuristic information, respectively.

4.2.2. Pheromone update

In the MOACO, the pheromones on every path are updated every time when an ant passes, and the pheromones on one path decrease as time t increases. The pheromone update on the hyperlink $l(i, j)$ from page P_i to page P_j is as follows.

$$\tau_{ij}(t+1)=(1-\rho)\times\tau_{ij}(t)+\sum_{k=1}^{w}\tau_{ij}^{k}(t) \tag{17}$$

Here, $\rho(0 < \rho \leq 1)$ is the pheromone evaporation rate, w is the number of ants, and $\tau_{ij}^{k}(t)$ is the pheromone left by the k-th ant based on the hyperlink $l(i, j)$ from webpage P_i to webpage P_j at time t, whose value is C/D. C denotes the topic relevance of the webpage P_j and D the length of the path constructed by the k-th ant in this cycle.

4.2.3. Selection of the optimal hyperlink

For a set of p Pareto optimal hyperlinks obtained by the MOACO, we use the fast non-dominated sorting (Huang et al., 2017; Deb, Pratap, Agarwal, & Meyarivan, 2002) and the nearest farthest candidate solution (NFCS) method (Liu, Liu, Liu, & Li, 2020) to select q ($q \leq p$) hyperlinks to join the optimal hyperlink set to guide the crawler's search direction. In the NFCS method, the objective function distance $Dis(X_1, X_2)$ between any two solutions (URLs) X_1 and X_2 is calculated by Eq.(18).

$$Dis(X_1,X_2)=\sqrt{\sum_{i=1}^{g}(F_i(X_1)-F_i(X_2))^2} \tag{18}$$

Here, $F_i(X_1)$ and $F_i(X_2)$ represent the i-th objective function values of X_1 and X_2, respectively. g is the number of objective functions and $g = 4$ in this article.

4.3. Process of focused crawling

We propose a focused crawler using ontology learning and the MOACO (OLMOACO) strategies for meteorological disaster domain knowledge. We first use topic-relevant words as keywords to search 100 top ranking topic-relevant webpages through browsers such as Google, Firefox, and Bing. Then, URLs that correspond to 30 webpages selected from 100 webpages by the domain expert are set as the initial seed hyperlinks *Link_Init*. In this article, we search the typhoon related webpages on the Internet by keyword set "typhoon, storm surge, wind, tropical cyclone, hurricane". At the same time, keywords such as "rainstorm, rainfall, urban waterlogging, flood" are used to search the webpages related to rainstorms.

Algorithm 1: OLMOACO

Input: Seed URLs.**Output**: Downloaded webpages.

1: Obtain the topic vector by constructing the domain ontology (see Section 2.5);
2: Add the seed URLs to the initial queue *Link_Init*. Initialize S_{rel}=∅, S_{down}=∅, $DP = 0$, $SP = 0$. Set σ, η and T. Let *Link_Wait* and *Child_Hyperlinks* to empty; *//DP denotes the number of downloaded webpages and SP denotes the number of downloaded topic-relevant webpages.*
3: Place randomly the positions of w ants on the selected w webpages (correspond to seed URLs) and initialize pheromone C_0 of every hyperlink in *Link_Init*. Put these webpages into a *Tabu* list and set $t = 1$;
4: **For** $k = 1$ to w **do**
  Extract all sub-hyperlinks from the current webpage P_k of the k-th ant, and put them into *Child_Hyperlinks*. Filter duplicate hyperlinks;
  For $i = 1$ to q **do** *// q is the number of sub-hyperlinks on the page P_k after deduplication*
    Calculate the comprehensive priority *Priority*(H_i) of the i-th sub-hyperlink H_i according to Eq.(10);
    If *Priority*(H_i) > η **then**
      Add the webpage P_i pointed to by the sub-hyperlink H_i to S_{down}, and put the sub-hyperlink H_i into the waiting queue *Link_Wait*;
      Let $DP = DP + 1$;
      If $R(P_i) > \sigma$ **then** *//$R(P_i)$ is the topic relevance of the webpage P_i*
        Add webpage P_i to S_{rel}; *//Save the topic-relevant webpages*
        Let $SP = SP + 1$;
      End If
      If *DP*>=15000 **then**
        The algorithm ends;
      End If
    Else give up H_i;
    End If
  End For
  According to Eq. (16) and the roulette method, choose webpage P_j (P_j is not in *Tabu*) as the next page of the k-th ant;
  Put webpage P_j into *Tabu* list and clear *Child_Hyperlinks*;
End For
5: **If** $t > T$ **then** *// In one cycle, every ant goes T steps, where T is set to 9 in this article*
    Clear *Tabu* list and *Link_Init*, and go to step 6;
Else
    Let $t = t + 1$ and go to step 4;
End If
6: Update the pheromones on all paths according to Eq. (17);
7: **For** $i = 1$ to L **do** *// L is the number of all hyperlinks in Link_Wait*
    Calculate the four objective function values for each hyperlink in *Link_Wait* according to Eqs.(11)-(14);
End For
8: **If** number of non-dominated hyperlinks in *Link_Wait* is more than 30 **then**
  Select 30 hyperlinks from *Link_Wait* by the NFCS;
Else
  Select all non-dominated hyperlinks from *Link_Wait*;
Put all selected hyperlinks into *Link-Init* and Reset w by the number of selected non-dominated hyperlinks.
9: Clear *Link_Wait* and go to step 3.

Suppose that there are w ants (in this article the initial value of w is set to 30, and its value will change with the number of selected non-dominated hyperlinks but does not exceed 30). Place randomly the positions of w ants on the selected w webpages and make sure that there is only one ant

on a webpage. Initialize pheromone for every hyperlink in *Link_Init*. Put these webpages into a *Tabu* list. For each of w ants, extract all sub-hyperlinks from the current webpage P_k of the k-th ant and put them into *Child_Hyperlinks*. Filter duplicate hyperlinks. For each sub-hyperlink in *Child_Hyperlinks*, calculate its comprehensive priority according to Eq. (10). If the comprehensive priority of a sub-hyperlink is greater than the preset threshold η, the webpage pointed to by this sub-hyperlink is put into the set S_{down} of the downloaded webpages, and put this sub-hyperlink into the waiting queue *Link_Wait*. If the topic relevance of the downloaded webpage is greater than the threshold σ, this webpage is considered topic-relevant and is put into the set S_{rel}. Calculate the transition probability of the all webpages pointed to by the sub-hyperlinks of the current webpage according to the pseudo-random ratio selection rule, and use the roulette method to select the next webpage to be crawled from these webpages. Once every ant crawls T webpages, we update the pheromones on all paths according to Eq. (17). Thereafter, according to the Eqs. (11)-(14), calculate the four objective function values for each hyperlink in *Link_Wait*. Select w hyperlinks by the non-dominated sorting and NFCS, and put them into *Link-Init*. Repeat the above steps until the number of the downloaded webpage attains 15000. The concrete steps of the OLMOACO are shown in Algorithm 1.

5. Experimental results and discussion

To test the effectiveness of the proposed focused crawler using ontology learning and multi-objective ant colony optimization (OLMOACO) strategies for meteorological disaster domain knowledge, we run the crawler on two topics of typhoon disasters and rainstorm disasters, respectively. We compare the experimental results of OLMOACO with the results of the breadth-first search (BFS) (Wang, 2011), the optimal priority search (OPS) (Rawat & Patil, 2013), the simulated annealing (SA) (He, Cheng, & Cai, 2009), and the focused crawler strategy combining web space evolutionary algorithm and ontology (WSEO) (Liu, Li, & Jiang, 2019). All algorithms are compiled in Java, and run on a PC equipped with Intel Core i7-7700HQ, 2.80 GHz processor and 8.0 GB RAM.

5.1. Experimental evaluation indices

The commonly used metrics for evaluating the performance of the focused crawler are the accuracy (*AC*) rate and the recall (*RC*) rate. The *AC* (see Eq. (19)) is the ratio of the number *SP* of the topic-relevant webpages crawled by the focused crawler and the total number *DP* of webpages crawled. The *RC* (see Eq. (20)) is the ratio of the number of the topic-relevant webpages crawled by the focused crawler and the number W of the topic-relevant webpages in the entire network. Because the web resources related to the topic in the whole network are very huge and the web resources are likely to change at any time, the recall rate is difficult to calculate. Therefore, this article does not use the recall rate as the evaluation index of focused crawlers' performance.

$$AC = \frac{SP}{DP} \tag{19}$$

$$RC = \frac{SP}{W} \tag{20}$$

There is no standard Benchmark for focused crawlers' performance evaluation in the current researches on focused crawlers. In order to further analyze the performance of the focused crawler, in addition to the accuracy *AC* and the number *SP* of the topic-relevant webpages downloaded, we also use the average topic relevance and standard deviation of all crawled webpages to analyze the performance of the algorithm. The average relevance (AR_{DP}) and standard deviation (SD_{DP}) of the topic relevance of all the crawled webpages are shown in the following Eqs. (21) and (22), respectively.

$$AR_{DP} = \frac{1}{DP} \times \sum_{i=1}^{DP} R(P_i) \tag{21}$$

$$SD_{DP} = \sqrt{\frac{1}{DP} \times \sum_{i=1}^{DP} \left(R(P_i) - AR_{DP}\right)^2} \tag{22}$$

Here, $R(P_i)$ is the topic relevance of the webpage P_i, and satisfies $R(P_i) > \sigma$.

5.2. Experimental results of different algorithms

We assign the same set of subject words to the five algorithms to calculate the topic relevance. For the typhoon disasters, we set the topic relevance threshold $\sigma = 0.67$ and the comprehensive priority threshold $\eta = 0.12$, and for the rainstorm disasters, we set $\sigma = 0.7$ and $\eta = 0.15$, respectively. If the relevance of a webpage is greater than σ, the webpage is considered to be topic-relevant, otherwise, it is considered irrelevant. When the total number of webpages to be crawled reaches 15,000, the algorithm ends. Although every crawler has tended to be stable gradually before that, 15,000 downloads is to better reflect the trend of each crawler algorithm and to evaluate the performance indices of different crawlers. Five algorithms are tested under the same experimental environment. The same evaluation indices are used to test different crawler algorithms on two topics of typhoon and rainstorm disasters. This is conducive to investigating the validity, superiority and adaptability of each algorithm.

5.2.1. Experimental results on typhoon disasters

Experimental results of the accuracy (*AC*) rate, the number of the topic-relevant webpages (*SP*), the average relevance (AR_{DP}), and standard deviation (SD_{DP}) of the topic relevance of the webpages crawled by five different crawling methods including OLMOACO, WSEO, SA, OPS and BFS on typhoon disasters are shown in Figs. 6–9 for comparison. Fig. 6 shows the results of the *AC* obtained by five focused crawler methods on the typhoon theme. It can be clearly seen from the figure that the accuracy rate of OLMOACO is higher than that of the other four search strategies. In fact, the accuracy rate of OLMOACO is stable at about 75%, WSEO at 70%, SA at 35%, OPS at 8%, and BFS at 4%. From Fig. 6, it can be found that the *accuracy* of SA and OPS increases rapidly at the initial stage of crawling. When *DP* reaches 2,500, the *accuracy* of SA and OPS reaches about 54% and 39%, respectively. Thereafter, the downward trends of two crawlers are obvious, indicating their *accuracy* is unstable. Fig. 7 shows the results of the *SP* obtained by the five focused crawler methods on the typhoon theme. As the number of the downloaded webpages continues to grow, the number of topic-relevant

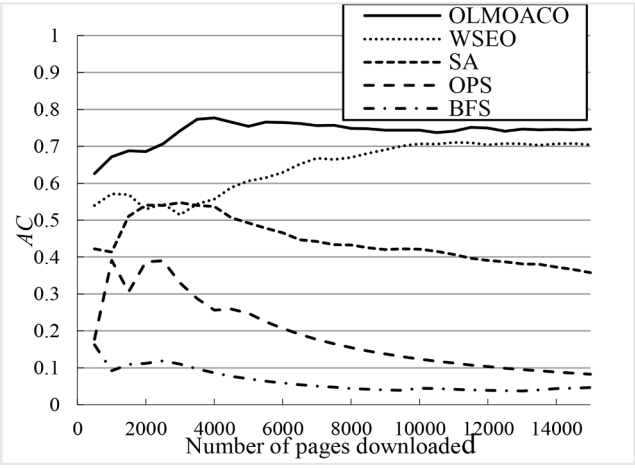


Fig. 6. Results of *AC* by five crawlers on typhoon disasters.

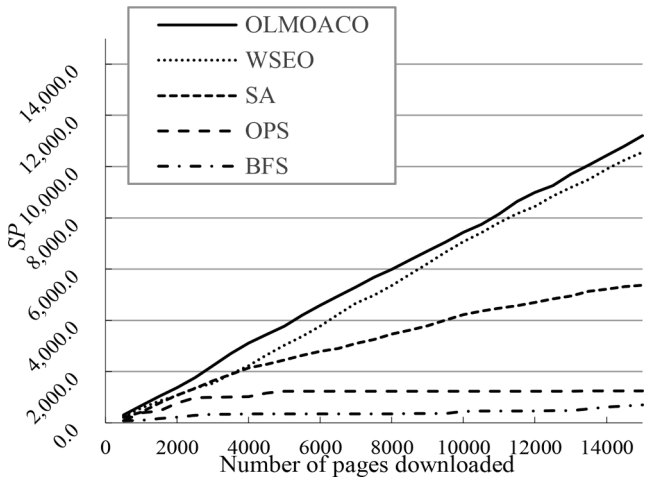


Fig. 7. Results of *SP* by five crawlers on typhoon disasters.

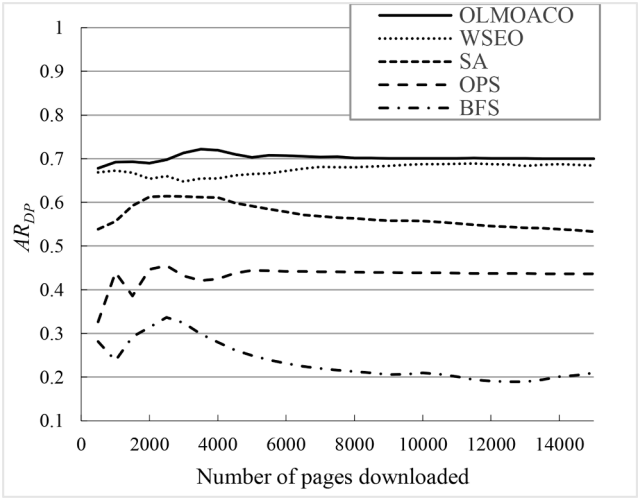


Fig. 8. Results of AR_{DP} by five crawlers on typhoon disasters.

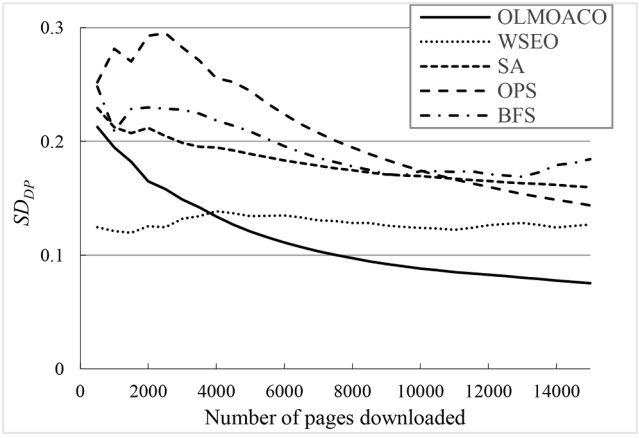


Fig. 9. Results of SD_{DP} by five crawlers on typhoon disasters.

webpages obtained by OLMOACO also increases. The number of the topic-relevant webpages downloaded by OLMOACO is more than that of other crawler methods. When the total number of the downloaded webpages is 15,000, the number of the topic-relevant webpages downloaded by OLMOACO is 11,201. Fig. 7 shows that before the number of retrieved webpages reaches 4,000, OLMOACO, WSEO, and SA acquire topic-relevant webpages quickly. Thereafter, the speed of SA crawling relevant webpages decrease significantly. However, OPS and BFS slowly acquire topic-relevant webpages during the whole crawling.

Fig. 8 shows the comparison of the average relevance (AR_{DP}) of the webpages downloaded by five focused crawlers on the typhoon theme. From Fig. 8, we can see that the average relevance curve of OLMOACO is relatively smooth, which is stable at about 70%. Compared with the average topic relevance of OLMOACO, WSEO is slightly lower at 68%, while SA at 53%, OPS at 43%, and BFS at 21%, which are obviously lower. Furthermore, the average relevance of OPS and BFS is unstable at the initial stage of crawling. Fig. 9 shows a comparison of standard deviations (SD_{DP}) of the topic relevance of the webpages downloaded by five focused crawlers based on the typhoon theme. Before the number of retrieved webpages reaches 4,000, standard deviation of OLMOACO is higher than that of WSEO. When the number of the downloaded webpages is 4,000, the standard deviation of OLMOACO is lower than that of the other four crawler methods. The standard deviation of OLMOACO has been steadily decreasing during the crawler's search for the topic-relevant webpages. The lower the standard deviation is, the better the stability of the algorithm is. Compared to the other four methods, the stability of OLMOACO is best.

5.2.2. *Experimental results on rainstorm disasters*

Experimental results of the *AC*, the *SP*, the AR_{DP}, and the SD_{DP} of the webpages crawled by five different crawling methods on rainstorm disasters are shown in Figs. 10–13 for comparison. Fig. 10 shows the results of the *AC* obtained by five focused crawler methods on the rainstorm theme. It can be seen from the figure that the accuracy rate of OLMOACO for obtaining the topic-relevant webpages is stable at about 74%, while WSEO is stable at about 73%, SA at about 58%, OPS at about 44%, and BFS at about 7%. When the number of the downloaded webpages reaches 6,500, the accuracy rate of OLMOACO tends to be stable, and thereafter the accuracy rate of OLMOACO is higher than that of the other four focused crawler methods. Fig. 11 shows the results of the *SP* obtained by the five focused crawler methods on the rainstorm theme. From the figure, one can find that OLMOACO and WSEO search the more topic-relevant webpages, and obtain 11,126 and 11,002 topic-relevant webpages, respectively. When the number of the downloaded webpages reaches 6500, the number of the topic-relevant webpages obtained by OLMOACO is more than that of the other four focused crawler methods.

Fig. 12 shows the comparison of the average relevance (AR_{DP}) of the webpages downloaded by five focused crawlers on the rainstorm

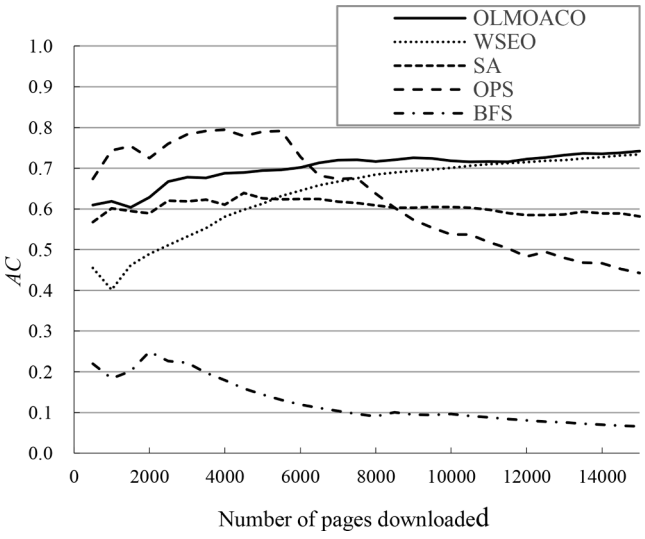


Fig. 10. Results of *AC* by five crawlers on rainstorm disasters.

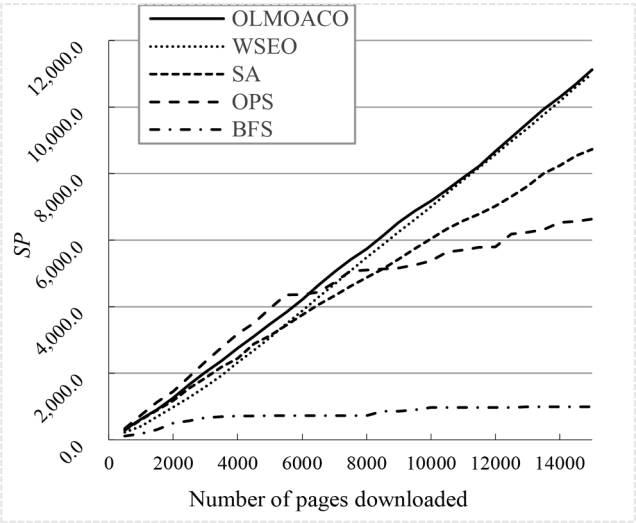


Fig. 11. Results of *SP* by five crawlers on rainstorm disasters.

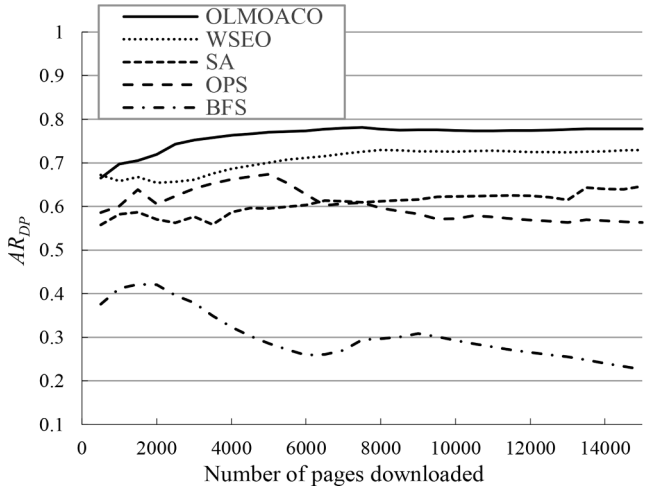


Fig. 12. Results of AR_{DP} by five crawlers on rainstorm disasters.

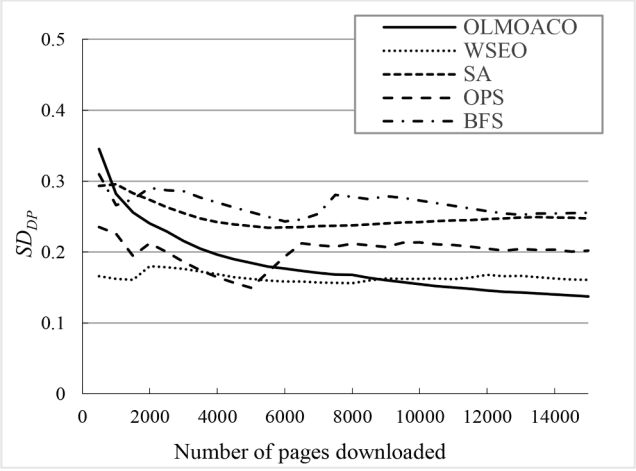


Fig. 13. Results of SD_{DP} by five crawlers on rainstorm disasters.

disaster theme. The average topic relevance of OLMOACO is around 78%, while WSEO is around 73%, SA around 64%, OPS around 56%, and BFS around 22%. Throughout the crawler's entire web search process, the average relevance curve of OLMOACO is relatively flat. The average relevance by OLMOACO is higher than that by the other four focused crawler methods. Fig. 13 shows a comparison of standard deviations (SD_{DP}) of the topic relevance of the webpages downloaded by five focused crawlers on the rainstorm disaster theme. The standard deviation of OLMOACO has maintained a downward trend throughout the whole process of crawling search. After the number of the downloaded webpages is greater than 9000, the standard deviation of OLMOACO is lower than that of the other four focused crawler methods, and eventually stabilizes at about 14%. According to the criterion of lower standard deviation corresponding to better algorithm stability, OLMOACO has the best stability.

From Figs. 10, 11 and 13, it is not hard to find that the OPS method has a better effect in the initial stage of webpage search on rainstorm disasters, but starts to diverge when the number of downloaded webpages increases. This is because that the OPS always downloads the most relevant webpages in the process of crawling. The greedy strategy of the OPS makes it obtain larger *AC* and *SP*, and smaller SD_{DP} in the early stage, but gradually fall into the local optimal search as the crawler continues. In addition, Figs. 9 and 13 show that the WSEO method overmatches the OLMOACO on standard deviation SD_{DP} in the early stage of webpage search on typhoon and rainstorm disasters. This is because that at the beginning, the WSEO is easy to generate more diverse non-dominated hyperlinks based on the smaller circular regions. As the radii of the circular regions are enlarged gradually, it is easy to grab the suboptimal hyperlinks by comparing with the nearby hyperlinks. However, in the OLMOACO, as the crawler continues, ants will accumulate more pheromones, and are easier to find the optimal crawling path and fetch more topic-relevant hyperlinks.

In addition, among the above five focused crawler methods, OLMOACO and WSEO adopt the domain ontology as the topic model and analyze the content of the search webpages from a semantic perspective to guide the crawling direction of the crawlers. From the above experimental results, we can find that the performance of OLMOACO and WSEO based on domain ontology is far superior to the other three focused crawler methods. At the same time, from the results of the above two groups of focused crawler experiments, it can also be found that OLMOACO has better performance and a stronger ability to search the topic-relevant webpages than the other four methods.

5.3. Experimental analysis and discussion of results

For focused crawling experiment setup of typhoon and rainstorm disasters, it is necessary that two different sets of keywords are used to locate initial seed URLs through browsers such as Google, Firefox, and Bing. This is because if the same keyword set are used, some irrelevant initial seed URLs may be returned, which is not conducive to the crawler algorithms to fetch the topic-relevant webpages during the later crawling process. For example, keyword "storm surge" in the typhoon disasters rarely appears in the rainstorm disasters domain. If starting to retrieve topic-relevant webpages from the initial seed URLs returned by this keyword, the effectiveness and efficiency of crawlers on rainstorm disasters will be reduced.

This article uses the VSM method to calculate topic relevance of a webpage text (or anchor text of hyperlink), that is, compute the cosine of the topic word weight vector obtained by constructing the domain ontology and the web text feature weight vector. There are also some other semantic similarity calculation methods in the literature. For example, Liu and Du (2014) used the semantic similarity retrieve model (SSRM) to measure the topic relevance of the text by associating term frequency and term semantic similarity. Taking advantage of VSM and SSRM, Du, Liu, Lv, and Peng (2015) researched further the calculation method of topic similarity, and proposed a semantic similarity vector space model (SSVSM), which may compare more accurately the results.

In order to further investigate the influence of the ontology approach to the performance of the proposed OLMOACO crawler, we execute the

MOACO crawler without the ontology approach (abbreviated as MOACO below). When retrieving 15,000 webpages, the results of *AC, SP, AR_{DP}, SD_{DP}* and running time by the MOACO on typhoon and rainstorm disasters are listed in Table 2, in comparison with those by the OLMOACO. Table 2 shows that experimental results by the OLMOACO for all five performance evaluation indices are better than those by the MOACO. This further confirms the significance of semantic approach to focused crawlers. Furthermore, we find that although the time of semantic similarity calculation based on the ontology is consumed in the OLMOACO, it does not occupy a great overhead in the whole crawling process. In fact, the most time spent in the crawling process is parsing the webpage content. Because the MOACO may miss some topic-relevant webpages, a more time overhead is spent to search topic-relevant webpages accompanying the parse of a large number of webpages. This is confirmed by the experimental results in Table 2, where the MOACO crawler spends more 4 h than the OLMOACO on experiments of both typhoon and rainstorm disasters. This further confirms the proposed OLMOACO crawler is an effective semantic retrieval method for fetching typhoon and rainstorm disasters domain knowledge.

It is a pity that the topic crawler method proposed in this article is not a real-time method. Although the crawler starts from the initial seed URLs, and once it grabs the topic-relevant webpages, they will be recorded in the text document in real time, but before that, it needs to spend more time to parse the webpages. Furthermore, the dynamic changes of the web resources at any time on the Internet also make it difficult for this method to crawl the webpages in real time.

In addition, effects of the focused crawlers are sensitive to the settings of some important parameters in experiments, such as topic relevance threshold σ and comprehensive priority threshold η. In order to obtain a good threshold, we adopt the grid search method. Take the evaluation of the *accuracy* of crawler as an example. We select some representative values of σ and η to execute the OLMOACO algorithm on typhoon and rainstorm themes. The threshold σ is set to 0.62, 0.67 and 0.72, respectively, for the typhoon disaster, and 0.65, 0.70 and 0.75, respectively, for the rainstorm disaster. The threshold η is set to 0.06, 0.09, 0.12 and 0.15, respectively, for the typhoon theme, and 0.09, 0.12, 0.15 and 0.18, respectively, for the rainstorm theme. Set the number of retrieved webpages to 15,000. The algorithm is run 10 times under each group of parameters independently. The average values of numerical results for the *accuracy* are shown in Table 3 and Table 4, respectively. In the tables, "-" means that *DP* has not reached 15,000 and the OLMOACO has ended prematurely under the σ and η thresholds.

Table 3 shows that when the value of η is 0.15, the crawler cannot retrieve 15,000 webpages because of too high threshold, which results in its excessive filtering capacities and the premature convergence of the algorithm. From Table 3, it is not hard to find when the threshold σ is smaller, the OLMOACO algorithm can obtain better *accuracy*. When σ is 0.62 and η is 0.12, the *accuracy* is the highest. However, a low topic relevance threshold may result in the calculation of some actually irrelevant webpages in the topic-relevant webpage collection. Therefore, in this article, we set σ to 0.67 and η to 0.12 for the typhoon theme according to the analysis of actually retrieved webpages.

Table 4 shows the comparison of the *accuracy* by the OLMOACO with different thresholds when the number of retrieved webpages reaches 15,000 for the rainstorm theme. When the value of η is 0.18, the crawler has ended prematurely and cannot retrieve 15,000 webpages. Through the analysis similar to the above and referring to the threshold value in Liu and Du (2014), we set σ to 0.70 and η to 0.15 for the rainstorm theme.

Table 3
Accuracy of the OLMOACO with different thresholds of σ and η when retrieving 15,000 webpages under the typhoon theme.

σ	η			
	0.06	0.09	0.12	0.15
0.62	0.64	0.74	0.79	—
0.67	0.53	0.70	0.75	—
0.72	0.45	0.66	0.68	—

Table 4
Accuracy of the OLMOACO with different thresholds of σ and η when retrieving 15,000 webpages under the rainstorm theme.

σ	η			
	0.09	0.12	0.15	0.18
0.65	0.67	0.73	0.87	—
0.70	0.54	0.71	0.74	—
0.75	0.44	0.54	0.69	—

6. Conclusions and future work

Generic crawlers have the disadvantage of not being able to provide accurate search results for specified domain knowledge. Unlike generic crawlers, focused crawlers are tailored to specific topics. In order to avoid the problems encountered by the generic crawlers, the semantic-based topic relevance calculation and crawling strategy are the focus of research on the focused crawlers. In this article, we propose a semantic-based focused crawling strategy. Firstly, we construct the domain ontologies of typhoon disasters and rainstorm disasters based on ontology learning, and then use ontology as the topic benchmark model. When evaluating the relevance of hyperlinks to the specific topic, we find that the traditional weighted sum method has a shortcoming that it is difficult to determine the optimal weights reasonably. This study introduces the multi-objective optimization model for link evaluation of the focused crawler. Afterwards, a MOACO algorithm combining the non-dominated sorting and the NFCS is proposed to guide the search direction of the focused crawler by selecting a set of Pareto-optimal hyperlinks.

In order to prove the validity and superiority of the proposed OLMOACO, in the same experimental environment, we compare the results of OLMOACO with those of the BFS, OPS, SA, WSEO and MOACO without ontology approach in the literature. The experimental results show that the performance of OLMOACO is superior to that of the other five algorithms. Nevertheless, the proposed method also has some defects, such as no consideration of tunnel crossing technique and reduction in run time. Because it is possible for an irrelevant page to link a relevant page it is required to traverse the irrelevant pages to get more relevant pages. In future work, we will focus more on the tunneling techniques and development of Hadoop-based distributed systems to enhance further the effectiveness and efficiency of the focused crawlers. Furthermore, considering the strong optimization ability of differential evolution (DE) algorithm (Yu, Yu, Lu, Yen, & Cai, 2018) for multi-objective problems, we will introduce the competitive strategy based

Table 2
Comparison of performance of OLMOACO and MOACO crawlers on typhoon and rainstorm disasters when retrieving 15,000 webpages.

Algorithms	Typhoon disasters					Rainstorm disasters				
	AC	*SP*	AR_{DP}	SD_{DP}	Time/h	*AC*	*SP*	AR_{DP}	SD_{DP}	Time/h
MOACO	0.71	10,589	0.69	0.13	19	0.73	11,006	0.71	0.14	20
OLMOACO	0.75	11,201	0.70	0.07	15	0.74	11,126	0.78	0.13	16

on DE into focused crawler by replacing the MOACO by DE to optimize the multi-objective optimization model for evaluating unvisited hyperlinks in the next stage.

CRediT authorship contribution statement

Jingfa Liu: Conceptualization, Methodology, Investigation, Project administration, Writing – review & editing. **Yi Dong:** Writing – original draft, Software. **Zhaoxia Liu:** Supervision. **Duanbing Chen:** Investigation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is supported by the Special Foundation of Guangzhou Key Laboratory of Multilingual Intelligent Processing, China [No. 201905010008], the Major Program of the National Social Science Foundation of China [No. 16ZDA047], the Program of Science and Technology of Guangzhou, China [No. 202002030238], and Guangdong Basic and Applied Basic Research Foundation, China [No. 2021A1515011974].

References

Angus, D. (2007). Population-based ant colony optimisation for multi-objective function optimization. *Proceedings of the 3rd Australian conference on Progress in artificial life*, Springer-Verlag, Heidelberg, (pp. 232-244). Gold Coast, Australia.

Asano, Y., Tezuka, Y., & Nishizeki, T. (2008). Improvements of HITS algorithms for spam links. *IEICE Transactions on Information & Systems, 91*(2), 200–208.

Bra, P. D., Houben, G. J., Kornatzky, Y., & Post, R. (1994). Information retrieval in distributed hypertexts. *In Proceedings of the 4th International Conference on Computer-Assisted Information Retrieval*, (pp. 481-493). Rockefeller University, NY, USA.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems, 30*(1), 107–117.

Chen, J., Gao, Q., Li, P. Y., Xie, Y. Y., Wang, X. J., Liu, Z. J., & Han, Y. (2019). Evaluation of storm and flood disasters of small and medium-sized rivers in Xiqing district of Tianjin City based on storm waterlogging model. *Meteorological Technology, 47*(1), 147–153.

Chen, Y. B., Zhang, Z., & Zhang, T. (2011). A searching strategy in topic crawler using ant colony algorithm. *Microcomputers and applications, 30*(1), 53–56.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197.

Dewanjee, J. (2016). Heuristic approach for designing a focused web crawler using cuckoo search. *Journal of Computing and Information Science in Engineering, 4*(9), 59–63.

Doaui, A., Gherabi, N., & Marzouk, A. (2017). An enhance method to compute the similarity between concepts of ontology. *Advances in Intelligent Systems and Computing, 640*, 95–107.

Du, X. Y., Li, M., & Wang, S. (2006). Summary of ontology learning research. *Journal of Software., 17*(9), 1837–1847.

Du, Y. J., Li, C. X., Hu, Q., Li, X. L., & Chen, X. L. (2016). Ranking webpages using a path trust knowledge graph. *Neurocomputing, 269*(20), 58–72.

Du, Y. J., Liu, W. J., Lv, X. J., & Peng, G. L. (2015). An improved focused crawler based on semantic similarity vector space model. *Applied Soft Computing, 36*, 392–407.

Du, Y. J., Pen, Q. Q., & Gao, Z. Q. (2013). A topic-specific crawling strategy based on semantics similarity. *Data & Knowledge Engineering, 88*, 75–93.

Gruber, T. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5*, 199–220.

Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing? *International Journal of Human-Computer Studies, 43*(5–6), 907–928.

Han, M., Wuillemin, P. H., & Senellart, P. (2018). Focused crawling through reinforcement learning. *International Conference on Web Engineering* (pp 261-278), Cáceres, Spain.

He, S., Cheng, J. X., & Cai, X. B. (2009). Focused crawler based on simulated anneal algorithm. *Computer Technology and Development, 19*(12), 55–58.

Henzinger, M. R. (2001). Hyperlink analysis for the web. *IEEE Internet Computing, 5*(1), 45–50.

Hersovici, M., Jacovi, M., Maarek, Y. S., Pelleg, D., Shtalhaim, M., & Ur, S. (1998). The shark-search algorithm-an application: Tailored web site mapping. *Computer Networks & ISDN Systems, 30*(1–7), 317–326.

Hosseinkhani, J., Taherdoost, H., & Keikhaee, S. (2021). ANTON framework based on semantic focused crawler to support web crime mining using SVM. *Annals of Data Science, 8*(2), 227–240.

Huang, X., Ye, C. M., & Cao, L. (2017). Mixed variation weed optimization algorithm for multi-objective job shop scheduling problem. *Journal of Computer Applications, 34*(12), 3623–3627.

Jelodar, H., Wang, Y. L., Yuan, C., Feng, X., Jiang, X. H., Li, Y. C., & Zhao, L. (2019). Latent dirichlet allocation (LDA) and topic modeling: Models, applications, a survey. *Multimedia Tools and Applications, 78*(11), 15169–15211.

Jiang, Y. C. (2019). Semantifying formal concept analysis using description logics. *Knowledge-Based Systems, 186*, Article 104967.

Khadir, A., C., Aliane, H., & Guessoum, A. (2021). Ontology learning: Grand tour and challenges. *Computer Science Review*, 39, 100339.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of ACM, 46*(5), 604–632.

Li, J., Mei, C., & Lv, Y. (2013). Incomplete decision contexts: Approximate concept construction, rule acquisition and knowledge reduction. *International Journal of Approximate Reasoning, 54*(1), 149–165.

Li, Y. H., Bandar, Z. H. A., & David, M. L. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering, 15*(4), 871–882.

Liu, J. F., & Liu, J. (2019). Applying multi-objective ant colony optimization algorithm for solving the unequal area facility layout problems. *Applied Soft Computing, 74*, 167–189.

Liu, J. F., Li, X., & Jiang, S. Y. (2019). Focused crawler strategy of rainstorm disaster topic based on web space evolution algorithm. *Computer Engineering, 45*(2), 184–190.

Liu, J. F., Liu, S. Y., Liu, Z. X., & Li, B. (2020). Configuration space evolutionary algorithm for multi-objective unequal-area facility layout problems with flexible bays. *Applied Soft Computing, 89*, Article 106052.

Liu, T., & Yan, T. C. (2011). The main meteorological disasters and their economic losses in China. *Journal of Natural Disasters, 20*(2), 90–95.

Liu, W. J., & Du, Y. J. (2014). A novel focused crawler based on cell-like membrane computing optimization algorithm. *Neurocomputing, 123*, 266–280.

Ma, L. L., Li, H. W., Lian, S. W., Liang, R. P., & Chen, H. (2016). A disaster focused crawler strategy based on ontology semantics. *Computer Engineering, 42*(11), 50–56.

Noy, N. F., Sintek, M., Decker, S., Crubezy, M., Feigerson, R. W., & Musen, M. A. (2005). Creating semantic web contents with protege-2000. *IEEE Intelligent Systems, 16*(2), 60–71.

Peng, Q. Q., Du, Y. J., Hai, Y. F., Chen, S. M., & Gao, Z. Q. (2009). Topic-Specific crawling on the web with concept context graph based on FCA. *International Conference on Management & Service Science*. Wuhan, China. IEEE.

Rawat, S., & Patil, D. R. (2013). Efficient focused crawling based on best first search. *2013 3rd IEEE International Advance Computing Conference* (pp. 908-911), Ghaziabad, India, IEEE.

Rios-Alvarado, A. B., Lopez-Arevalo, I., & Sosa-Sosa, V. J. (2013). Learning concept hierarchies from textual resources for ontologies construction. *Expert Systems with Applications, 40*(15), 5907–5915.

Rocco, C., M., Hernandez-Perdomo, E., & Mum, J. (2020). Introduction to formal concept analysis and its applications in reliability engineering. *Reliability Engineering and System Safety*, 202, 107002.

Saleh, A. I., Abulwafa, A. E., & Rahmawy, M. F. A. (2017). A web page distillation strategy for efficient focused crawling based on optimized Naïve bayes (ONB) classifier. *Applied Soft Computing, 53*, 181–204.

Seyfi, A., Patel, A., & Júnior, J. C. (2016). Empirical evaluation of the link and content-based focused Treasure-Crawler. *Computer Standards and Interfaces, 44*, 54–62.

Sharma, D. K., & Khan, M. A. (2015). SAFSB: A self-adaptive focused crawler. *2015 1st International Conference on Next Generation Computing Technologies* (pp. 719-724), Dehradun, India, IEEE.

Sornalakshmi, M., Balamurali, S., Venkatesulu, M., Krishnan, M. N., Ramasamy, L. K., Kadry, S., … Muthu, B. A. (2020). Hybrid method for mining rules based on enhanced Apriori algorithm with sequential minimal optimization in healthcare industry. In *Neural Computing and Applications, Special Issue on New Trends in Brain Computer* (pp. 1–14).

Suebchua, T., Manaskasemsak, B., Rungsawang, A., & Yamana, H. (2017). Efficient topical focused crawling through neighborhood feature. *New Generation Computing, 36*(2), 95–118.

Wang, Z. G., & Meng, B. J. (2014). A comparison of approaches to Chinese word segmentation in Hadoop. 2014 IEEE International Conference on Data Mining Workshop (pp. 844-850), Shenzhen, China.

Wang, C., & Ji, X. H. (2016). Improved page rank algorithm based on user interest and topic. *Computer Science, 43*(3), 275–278.

Wang, X. Y., & Yang, B. (2018). Design and implementation of an Apriori-based recommendation system. *2018 International Conference on Engineering Simulation and Intelligent Control* (pp.372-375), Big Island, Hawaii, America, IEEE.

Wang, Y. (2011). *Design and implementation of focused crawler based on breadth-first*. Shanghai: Fudan University.

Wang, Z. G, Meng, B. J. (2014). A comparison of approaches to Chinese word segmentation in hadoop. In Proc. IEEE Int. Conf. Data Mining Workshop (pp. 844-850), Shenzhen, China.

Yan, W., & Pan, L. (2018). Designing focused crawler based on improved genetic algorithm. 2018 *Tenth International Conference on Advanced Computational Intelligence* (pp. 319-323). Xiamen, China, IEEE.

Yu, X. B., Yu, X. R., Lu, Y. Q., Yen, G. G., & Cai, M. (2018). Differential evolution mutation operators for constrained multi-objective optimization. *Applied Soft Computing, 67*, 452–466.

Zheng S. (2011). Genetic and ant algorithms based focused crawler design. *Proceedings of the 2011 Second International Conference on Innovations in Bio-inspired Computing and Application* (pp 374-378). Shenzhen, China.