Applied Intelligence (2023) 53:7390–7407
https://doi.org/10.1007/s10489-022-03180-5

A semantic and intelligent focused crawler based on semantic vector space model and membrane computing optimization algorithm

Wenjun Liu[1] · Zurui Gan[1] · Tiejun Xi[1] · Yajun Du[1] · Jing Wu[1] · Yu He[1] · Pengjun Jiang[1] · Xing Liu[2] · Xia Lai[1]

Accepted: 4 January 2022 / Published online: 28 July 2022

Abstract
The focused crawler downloads web pages related to the given topic from the Internet. In many research studies, most of focused crawler predict the priority values of unvisited hyperlinks by integrating the topic similarities based on the text similarity model and equivalent weighted factors based on the manual method. However, in these focused crawlers, there are flaws in the text similarity models, and weighted factors are arbitrarily determined for calculating priorities of unvisited URLs. To solve these problems, this paper proposes a semantic and intelligent focused crawler based on the Semantic Vector Space Model (SVSM) and the Membrane Computing Optimization Algorithm (MCOA). Firstly, the SVSM method is used to calculate topic similarities between texts and the given topic. Secondly, the MCOA method is used to optimize four weighted factors based on the evolution rules and the communication rule. Finally, this proposed focused crawler predicts the priority of each unvisited hyperlink by integrating the topic similarities of four texts and the optimal four weighted factors. The experiment results indicate that the proposed SVSM-MCOA Crawler improve the evaluation indicators compared with the other four focused crawlers. In conclusion, the proposed SVSM and MCOA method promotes the focused crawler to have semantic understanding and intelligent learning ability.

Keywords Focused Crawler · Semantic Vector Space Model · Membrane Computing · Semantic Similarity · Optimization Algorithm

1 Introduction

A web crawler is a program that searches and downloads network resources automatically. It is an important part of the general search engine. The main characteristic of the general search engine is to collect web resources from the Internet as much as possible, and the collection process does not consider the priorities of unvisited web pages and the similarities of visited web pages [1, 2]. With the rapid growth of network resources, it is necessary to add more storage space to store web pages, but this greatly increases system maintenance costs and text index costs. For a single user or an organization, they may only care about a single topic or several topics in a field, but the general search engine collects all web information in all fields [3, 4]. As a result, most of the information in the index database is valueless for these users or organizations. In addition, the response time of the list of retrieved results is prolonged and the throughput of the user query is reduced.

The topic-oriented search engine appears due to the above problems of the general search engine. This topic-oriented web crawler, called the focused crawler, downloads only the web pages related to the given topic, and the priorities of unvisited URLs are determined based on the topic similarities of retrieved web pages [5, 6]. The focused crawler firstly downloads web pages according to the given initial URLs, and extracts hyperlinks from these web pages. Then, the topic similarity of each unvisited URL is acquired based on the text content or link structure of web pages. These values are compared with the setting threshold to select URLs with high topic similarities into the queue of unvisited URLs in descending order. Finally, these unvisited URLs in the front of the queue are preferentially selected to continuously download web pages. The above processes are repeated until a certain number of web

✉ Yajun Du
duyajun@mail.xhu.edu.cn

[1] School of Computer and Software Engineering, XiHua University, Chengdu 610039, China

[2] Xihua Honors College, XiHua University, Chengdu 610039, China

pages is reached, or the storage space is full [7–9]. The focused crawlers take advantage of the semantic relationship between parent web pages and child web pages, that is, a parent web page contains hyperlinks pointing to other child web pages with the same topic [10, 11].

Most of focused crawlers utilize the text contents of web pages to predict the priorities of unvisited URLs. The Vector Space Model (VSM) is a classic model that is commonly used in focused crawlers [12]. The VSM method utilizes the term vectors to represent documents and topics, and these vector values are obtained through the Term Frequency Inverse Document Frequency (TF*IDF). Then the inner product between the document vector and the topic vector is regarded as the topic similarity of the document. The Semantic Similarity Retrieval Model (SSRM) is the typical method for the semantic crawlers. The SSRM method considers not only the weights of terms in web pages or topics, but also the semantic similarities between web page terms and topic terms [13]. Compared with the VSM model, the SSRM model further excavates the semantic similarity between document terms and topic terms except that it obtains the TF*IDF weights of document terms and topic terms. In other words, the SSRM model considers not only the TF*IDF weights of terms, but also the semantic similarities between terms.

The above text-based focused crawlers can obtain web pages related to a given topic from the Internet. However, at present, there are still some problems worth further research for text-based focused crawlers. These problems are described as follows:

(1) There are flaws in the topic similarity models between the text and the given topic. The VSM model requires that the document vector and the topic vector corresponds to the same term. If two terms are completely different, the two vectors will not consider TF*IDF values of the two terms. If the two terms are essentially the same semantic, namely, the semantic similarity of two terms is equal to one. Then TF*IDF values of the two terms can be considered as vector values on the same terms for the document vector and the topic vector. The SSRM model obtains the semantic similarities between document terms and topic terms, and it can further calculate the similarity between the document and the topic. If the document term set is the same as the topic term set, or the two term sets are synonyms, the similarity between the document and the topic is still the maximum one although the TF*IDF values of the document terms are quite different from TF*IDF values of the corresponding topic terms. Therefore, the above two topic similarity models exist obviously certain defects.

(2) The weighted factors are arbitrarily determined for calculating priorities of unvisited URLs. In order to get more accurate priorities of unvisited URLs, more influence factors must be taken into account, and weighted factors are correspondingly increased to show the different influence degrees that these factors have on priorities. These weighted factors are set by the manual method for most of focused crawlers. This manual method can not objectively reflect the influence degrees that weighted factors have on priorities of unvisited URLs, and may result in that there is a large deviation for the priorities. Therefore, weighted factors based on the manual method may reduce the accuracy and recall of the focused crawler.

In recent studies, there are many metaheuristic optimization algorithms to intelligently determine weighted factors. The Monarch Butterfly Optimization (MBO) simulates the migration behavior of the monarch butterflies in nature, and is mainly determined by the migration operator and butterfly adjusting operator [14]. The Slime Mould Algorithm simulates the process of producing positive and negative feedback generated by slime mould during foraging, thus forming the optimal path for connecting food with excellent exploratory ability and exploitation propensity [15]. The Hunger Games Search (HGS) is designed according to the hunger-driven activities and behavioural choice of animals, and incorporates the concept of hunger into the feature process to simulate the effect of hunger on each search step [16]. The Runge Kutta optimizer (RUN) uses a specific slope calculation concept based on the Runge Kutta method as an effective search engine for global optimization, and consists of two main parts including a search mechanism and an enhanced solution quality mechanism [17]. The Membrane Computing Optimization Algorithm (MCOA) simulates high energy molecules passing through biological membranes, and includes the evolution rule and the communication rule [18]. Compared with the Genetic Algorithm, the MCOA can utilize the communication rule to further improve the population diversity and prevent this method falling into a local optimal solution prematurely. Therefore, the MCOA method is used to obtain the optimal weighted factors in this paper.

To solve the above-mentioned problems, this paper proposes a semantic and intelligent focused crawler based on the Semantic Vector Space Model (SVSM) and the Membrane Computing Optimization Algorithm (MCOA). This crawling strategy firstly acquires four topic similarities of full texts, anchor texts, title texts and context texts of each unvisited hyperlink by the Semantic Vector Space Model (SVSM). Then, the corresponding four weighted factors are optimized based on the Membrane Computing Optimization Algorithm (MCOA) for above-mentioned four different texts. Finally,

the priority of each unvisited hyperlink is calculated by integrating the four topic similarities of four texts and the corresponding four weighted factors. The experiment results indicate that the proposed SVSM-MCOA Crawler improves the evaluation indicators compared with the BF Crawler, the VSM Crawler, the SSRM Crawler and the MCOA Crawler. In conclusion, the proposed method improves the performance of the focused crawler and makes focused crawlers have the ability of semantic understanding and intelligent learning by utilizing the SVSM and MCOA method.

The contributions of this paper are as follows:

(1) This paper proposes a semantic and intelligent focused crawler based on the Semantic Vector Space Model (SVSM) and the Membrane Computing Optimization Algorithm (MCOA). This focused crawler combines the SVSM method and the MCOA method. The topic similarities of four texts are obtained based on the SVSM method. The optimal four weighted factors are obtained based on the MCOA method.
(2) The experiment has implemented the five focused crawlers including the BF Crawler, the VSM Crawler, the SSRM Crawler, the MCOA Crawler and the SVSM-MCOA Crawler. The performance of the five focused crawlers is evaluated based on three indicators including the relevant number, the harvest rate and the average topic similarity.

The remainder of this paper is constructed as follows: Section 2 introduces three kinds of focused crawlers. In Section 3, the semantic and intelligent focused crawler based on the SVSM and the MCOA is proposed, and the experiment results for five focused crawlers are compared and analyzed in Section 4. Finally, Section 5 presents the conclusions and future works.

2 Related works

The focused crawler grabs web pages based on the priorities of unvisited URLs. The focused crawler tries to grab web pages related to the given topic, and reduces the consumption of computer resources and the waste of running time. This paper divides focused crawlers into three categories: Classic Focused Crawlers, Semantic Focused Crawlers and Intelligent Focused Crawlers. These three categories of focused crawlers are described in the following.

2.1 Classic focused crawler

The classic focused crawler utilizes the text content and link structure of web pages to calculate the topical similarities of unvisited URLs. This classic focused crawler can be divided into three categories: focused crawlers based on text content, focused crawlers based on the link structure and focused crawlers based on text content and the link structure. Focused crawlers based on text contents utilize text contents of web pages to calculate the topical similarities of unvisited URLs [19, 20]. Focused crawlers based on the link structure utilize the link structure of web pages to calculate the topic similarities of unvisited URLs [21]. In addition, one kind of focused crawlers combines with the text contents and the link structure of web pages to calculate the topic similarities of unvisited URLs, which is called focused crawlers based on text content and link structure [22]. This focused crawler utilizes the text content to construct the classifier and the link structure to establish all kinds of topic hierarchies.

The following will describe the representative methods for the three types of the classic focused crawler. The Vector Space Model (VSM) is the main model of focused crawlers based on text content. The VSM mainly constructs the document vector and the topic vector by the Term Frequency Inverse Document Frequency (TF*IDF), and the cosine similarity between the two vectors is used as the topic similarity of the document [12]. The PageRank algorithm and the HITS algorithm are the main models of focused crawlers based on link structure. The PageRank algorithm is a common link structure analysis algorithm. The PageRank algorithm is an iterative method and can measure the importance of web pages from levels of the link structure [23]. The HITS algorithm utilizes the number of incoming links and outgoing links of web pages to iteratively calculate the authority and center values of web pages, and then ranks web pages according to these values [24]. The PageRank algorithm considers all the web pages from the Internet while the HITS algorithm only considers the set of retrieved web pages for a given query.

The focused crawlers based on the context graph and the relevancy context graph combine the text content and the link structure of web pages. The focused crawler based on the context graph utilizes the link structure to construct the context graph and the text content to construct the classifier [25]. The focused crawler based on the relevancy context graph improves the focused crawler based on the context graph. According to the semantic relationship between web pages, the focused crawler sets the constant between the levels of the context graph to form the relevancy context graph [26]. According to the constant, the focused crawler can get the correlation degrees between the web pages of each level and the topic in the relevancy context graph.

2.2 Semantic focused crawler

The semantic focused crawler utilizes the semantic similarity between terms to calculate the topic similarities of unvisited URLs. In the VSM, a web page and the given topic

must have the same term in order to calculate the similarity between the web page and the given topic [12]. If they have no common terms, the web page is considered to be irrelevant to the given topic and the topic similarity of the web page is 0. Although there are no common items between the web page and the given topic, the semantic meaning between web page terms and topic terms may be similar or even the same. However, the web page is still judged to be irrelevant to the given topic according to the VSM so that the focused crawler can not collect web pages with similar semantics. Therefore, the semantic focused crawler emerges as the times require. Even if there are no common terms between the web page and the given topic, the similarity between the web page and the given topic can still be determined by calculating the semantic similarity between terms. The semantic focused crawler can retrieve the collection of web pages with similar semantics and obtain more relevant web pages [27, 28].

The Semantic Similarity Retrieval Model (SSRM) is a typical method of semantic crawlers. It not only considers the weights of items in web pages or topics, but also considers the semantic similarity between web page terms and topic terms [13, 29]. Compared with the VSM model, the SSRM model does not require that the web page and the given topic have common terms. The similarity between the web page and the given topic can still be calculated by calculating the semantic similarities between web page terms and topic terms. Therefore, the focused crawler based on the SSRM can retrieve more web pages related to the given topic, and the SSRM method can improve the accuracy and recall of the focused crawler.

2.3 Intelligent focused crawler

The intelligent focused crawler utilizes the artificial intelligence technology to calculate the topic similarities of unvisited URLs. The artificial intelligent technology can effectively guide focused crawlers to collect web pages related to the given topic. The artificial intelligence technology refers to the calculation model which is abstracted from the structure, function and behavior of organisms, such as Ant Colony Algorithm, Genetic Algorithm, Neural Network Algorithm, Membrane Computing, DNA Computing and so on [30]. By using the idea of artificial intelligence, the intelligent focused crawler has a certain learning ability. The intelligent method can effectively guide the focused crawler to approach to web pages related to the given topic and be away from web pages unrelated to the given topic [31, 32].

The intelligent focused crawler utilizes the artificial intelligence technology to train the crawling process to download web pages. The focused crawler based on the online semi-supervised theory utilizes Q-value learning, Naive Bayes and Genetic Algorithm to calculate the topic similarities of unvisited URLs [33]. In the training stage, the online semi-supervised focused crawler uses Q-value learning to construct a classifier and learn the process that the crawler grabs the web pages. Meanwhile, the Genetic Algorithm is used to obtain the optimal fuzzy membership matrix in the crawling stage to reduce the classification error. The idea of artificial intelligence can make the focused crawler have a certain learning ability. To a certain extent, this intelligence idea improves the overall performance of the focused crawler.

As mentioned above, there are three kinds of focused crawlers including the classic focused crawler, the semantic focused crawler and the intelligent focused crawler from previous literatures in Table 1. The classic focused crawler utilizes text content and link structure of web pages to calculate topical similarities of unvisited URLs. The semantic focused crawler utilizes the semantic similarity between terms to calculate topic similarities of unvisited URLs. The intelligent focused crawler utilizes the artificial intelligence technology to calculate topic similarities of unvisited URLs.

Table 1 Focused crawlers based on different methods from previous literatures

Authors	Methods	Classification
G. Salton et al	Focused crawler based on the VSM method [12]	Classic Focused Crawler
S. Brin et al	Focused crawler based on the PageRank algorithm [23]	Classic Focused Crawler
J. M. Kleinberg	Focused crawler based on the HITS algorithm [24]	Classic Focused Crawler
M. Diligenti et al	Focused crawler based on the context graph [25]	Classic Focused Crawler
C. C. Hsua et al	Focused crawler based on the relevancy context graph [26]	Classic Focused Crawler
G. Varelas et al	Focused crawler based on the SSRM method [13]	Semantic Focused Crawler
J. Hernandez et al	Focused crawler based on the knowledge representation schema [27]	Semantic Focused Crawler
A. Hliaoutakis et al	Focused crawler based on the semantic similarity method [29]	Semantic Focused Crawler
W. Wang et al	Focused crawler based on the URL knowledge base [31]	Intelligent Focused Crawler
H. Dong et al	Focused crawler based on the ontology learning [32]	Intelligent Focused Crawler
H. X. Zhang et al	Focused crawler based on the semi-supervised clustering [33]	Intelligent Focused Crawler

3 Focused crawler based on SVSM and MCOA

In this part, this paper proposes a semantic and intelligent focused crawler based on Semantic Vector Space Model (SVSM) and Membrane Computing Optimization Algorithm (MCOA). This focused crawler divides the text contents of unvisited URLs into four parts including full texts, anchor texts, title texts, and context texts. The SVSM constructs text semantic vectors of the above four texts and the topic semantic vector, and utilizes the cosine similarity between the text semantic vector and the topic semantic vector as the topic similarity of the text. The MCOA takes the four weighted factors corresponding to the four texts as one object and utilizes the evolution rules and communication rules to obtain the optimal object which corresponds to the four optimal weighted factors. This proposed algorithm can guide the focused crawler to choose better unvisited URLs to continuously collect more web pages related to the given topic from the Internet.

Figure 1 shows the flow chart of the focused crawler based on SVSM and MCOA. In Fig. 1, the flow chart of this focused crawler is divided into three main modules including Sematic Vector Space Model, Membrane Computing

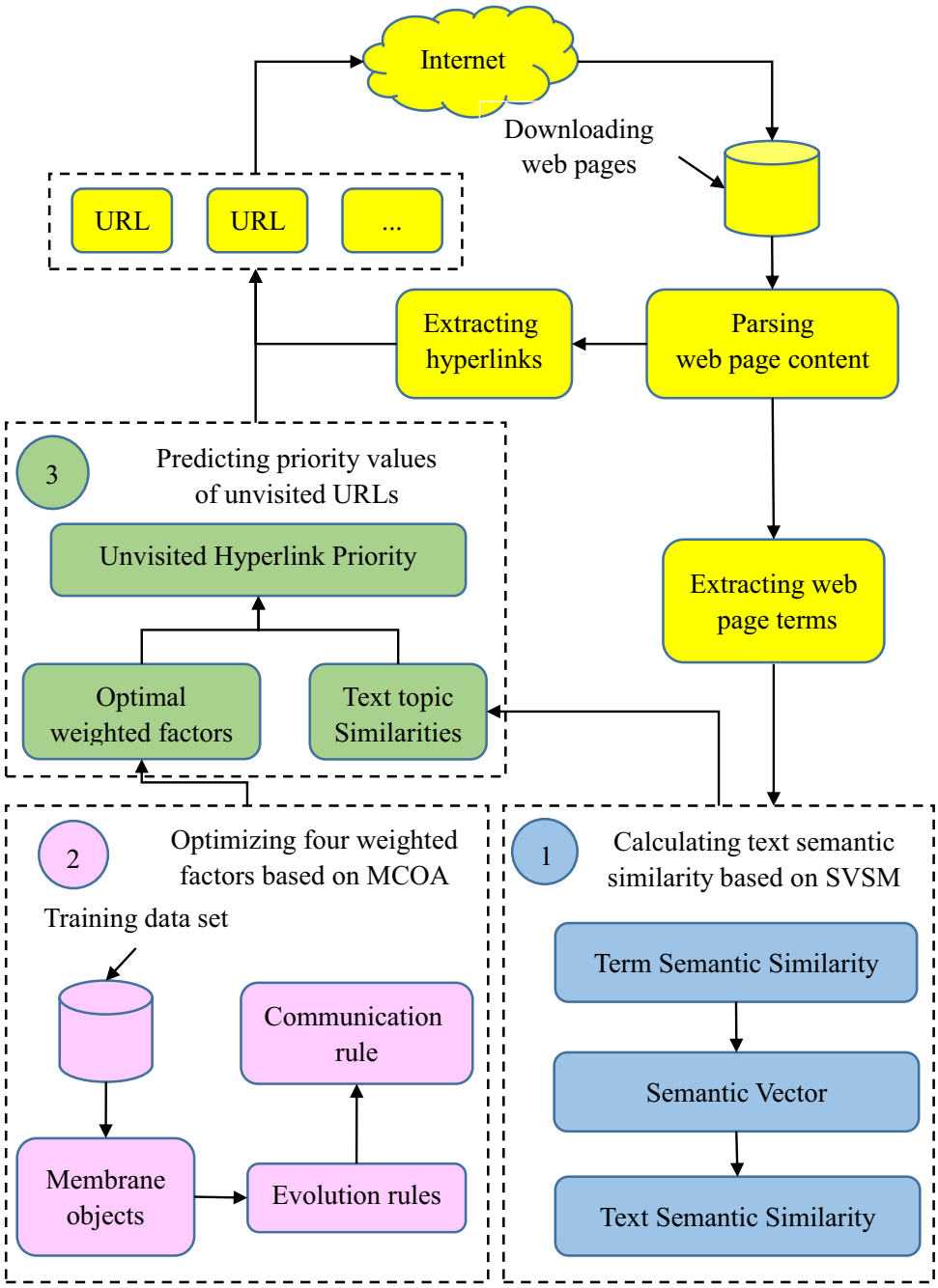

Fig. 1 The flowchart of a focused crawler based on SVSM and MCOA

Optimization Algorithm and Priority Prediction Method. In this focused crawler, the texts of hyperlinks include full texts, anchor texts, title texts and context texts. Firstly, the SVSM method obtains the topic similarities of four texts. Secondly, the MCOA method obtains the optimal four weighted factors. Finally, the priority prediction method obtains the topic similarity of each unvisited hyperlink by linearly combining optimal four weighted factors and four topic similarities of four texts for the unvisited hyperlink which is taken as the priority of each unvisited hyperlink. The above three main modules will be described in the following.

3.1 Sematic vector space model

The Semantic Vector Space Model (SVSM) is used to calculate the topic similarities of four texts. The SVSM method combines the cosine similarity based on the VSM method and the semantic similarity based on the SSRM method [34]. The SVSM method is divided into three parts including Term Semantic Similarity, Semantic Vector and Text Sematic Similarity. Firstly, the term semantic similarity is the semantic similarity between text terms and topic terms, and is calculated based on the WordNet. Secondly, the text and topic semantic vectors are constructed by combining the term TF*IDF weights and the term semantic similarities. Finally, the text semantic similarity is the topic similarity of the text obtained by calculating the cosine similarity between the text semantic vector and the topic semantic vector. The SVSM method can make the focused crawler has the semantic understanding ability.

3.1.1 Term semantic similarity

WordNet can accurately describe the meaning of concepts and the internal relations between concepts. WordNet can be used to obtain the semantic similarity between terms according to the structural hierarchy relationship among concepts [35]. The concept is expressed through synonym sets, each of which includes synonymous terms. Meanwhile, the relationships of hypernyms and hyponyms, parts and whole, synonyms and antonyms are expressed in the form of linked lists in WordNet [36]. The semantic similarity model believes that there are common points among concepts, and meanwhile each concept has its own information capacity [37]. This model indicates that the semantic similarity between two concepts should be measured by the ratio of common information and total information. The calculation formula is as follows:

$$sim(c_i, c_j) = \frac{2 \times IC(lso(c_i, c_j))}{IC(c_i) + IC(c_j)} \tag{1}$$

where $sim(c_i, c_j)$ is the semantic similarity between two concepts c_i and c_j, $lso(c_i, c_j)$ is the common parent node of two concepts c_i and c_j, $IC(c_i)$ and $IC(c_j)$ are the information contents of two concepts c_i and c_j respectively.

3.1.2 Semantic vector

This section constructs the semantic vectors for texts and the given topic. The two semantic vectors are obtained by using TF*IDF weights of terms and the semantic similarities between terms. For the text semantic vector, each vector score projected to each double-term is calculated by multiplying the TF*IDF weight of the text term by the semantic similarity between the text term and the topic term. In contrast, for the topic semantic vector, each vector score projected to each double-term is calculated by multiplying the TF*IDF weight of the topic term by the semantic similarity between the text term and the topic term. The text and topic semantic vectors are established as follows:

$$\begin{aligned}\overrightarrow{DSV}_k &= (w_{k1} \cdot sem^k_{11}, w_{k1} \cdot sem^k_{12}, \ldots, w_{k1} \cdot sem^k_{1n}, w_{k2} \cdot sem^k_{21}, w_{k2} \cdot sem^k_{22}, \ldots, w_{k2} \cdot sem^k_{2n} \\ &\quad , \ldots, w_{km} \cdot sem^k_{m1}, w_{km} \cdot sem^k_{m2}, \ldots, w_{km} \cdot sem^k_{mn} \\ \overrightarrow{TSV}_k &= (w_{t1} \cdot sem^k_{11}, w_{t1} \cdot sem^k_{21}, \ldots, w_{t1} \cdot sem^k_{m1}, w_{t2} \cdot sem^k_{12}, w_{t2} \cdot sem^k_{22}, \ldots, w_{t2} \cdot sem^k_{m2} \\ &\quad , \ldots, w_{tn} \cdot sem^k_{1n}, w_{tn} \cdot sem^k_{2n}, \ldots, w_{tn} \cdot sem^k_{mn}\end{aligned} \tag{2}$$

where $\overrightarrow{DSV_k}$ and $\overrightarrow{TSV_k}$ are the text and topic semantic vectors respectively, $w_{ki}(1 \le i \le m)$ is the TF*IDF weight of term i in the text , d_k, $w_{tj}(1 \le j \le m)$ is the TF*IDF weight of term j in the topic t, sem^k_{ij} is the semantic similarity between term i and term j in the text d_k and topic t respectively.

3.1.3 Text Sematic Similarity

The text semantic similarity between a text and the given topic is calculated based on semantic vectors. In the SVSM, the text and the given topic are represented by semantic vectors. The cosine similarity between the text and topic semantic vectors can be considered as the text semantic similarity called as the topic similarity of the text. The topic similarity of the text is calculated as follows:

$$Sim(d_k, t) = \overrightarrow{DSV_k} \cdot \overrightarrow{TSV_k} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} w_{ki} w_{tj} (sem^k_{ij})^2}{\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} (w_{ki} sem^k_{ij})^2} \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} (w_{tj} sem^k_{ij})^2}} \tag{3}$$

where $Sim(d_k, t)$ is the topic similarity of the text d_k. The rest of the symbols in this formula have the same meaning as the symbols in the formula (2).

Figure 2 gives a simple example to calculate the text semantic similarity for the SVSM. In Fig. 2, topic t has

three terms and text d_1 has two terms. The dimension of the semantic similarity matrix between terms of text d_1 and terms of topic t is 2×3. In Fig. 2, the text vector $\overrightarrow{d_1}$ and the topic vector $\vec{t}$ are respectively (1.493, 1.182) and (0.119, 0.106, 0.096) which are mapped to the term space. In addition, there are six double-terms between terms of text d_1 and terms of topic t. The text semantic vector $\overrightarrow{DSV_1}$ and the topic semantic vector $\overrightarrow{TSV_1}$ are respectively (1.478, 1.07, 1.07, 0.945, 0.721, 0.721) and (0.118, 0.08, 0.141, 0.095, 0.076, 0.119) which are mapped to the semantic space. The text semantic similarity between the text d_1 and the topic t is 0.96 by using the formula (3).

3.2 Membrane computing optimization algorithm

The Membrane Computing Optimization Algorithm (MCOA) is used to obtain the optimal four weighted factors. The MCOA method utilizes evolution rules and the communication rule to optimize objects in membranes [18]. The MCOA method is divided into three parts including Membrane Object, Membrane Evolution Rule and Membrane Communication Rule. Firstly, the membrane object is a four dimensional vector, and each dimension corresponds to a weighted factor. Secondly, the membrane evolution rules include the selection rule, mutation rule and crossover rule to evolve objects and maintain the population diversity of objects. Finally, the membrane communication rule exchange objects between membranes and send better objects to the outer membrane. The MCOA method can make the focused crawler has the intelligent learning ability.

3.2.1 Membrane Object

The membrane objects exist in different types of membranes forming the membrane structure. The membrane structure consists of three kinds of membranes including the superficial

(1) Term Semantic Similarity

The topic t and text d_1 are as follows:

Suppose that terms of topic t are c3, c6 and c7, $\vec{t} = (0.119,\ 0.106,\ 0.196)$

Suppose that terms of text d_1 are c2 and c5, $\overrightarrow{d_1} = (1.493,\ 1.182)$

The semantic similarity matrix for topic t and text d_1 is as follows:

	c3	**c6**	**c7**
c2	0.99	0.72	0.72
c5	0.80	0.61	0.61

(2) Semantic Vector

The semantic vectors for topic t and text d_1 are as follows:

(c2, c3), (c2, c6), (c2, c7), (c5, c3), (c5, c6), (c5, c7)

$\overrightarrow{DSV_1} = (1.478,\ 1.07,\ 1.07,\ 0.945,\ 0.721,\ 0.721)$

$\overrightarrow{TSV_1} = (0.118,\ 0.08,\ 0.141,\ 0.095,\ 0.076,\ 0.119)$

(3) Text Semantic Similarity

$Sim(d_1, t) = \overrightarrow{DSV_1} \cdot \overrightarrow{TSV_1} = 0.96$

Fig. 2 A example to calculate the text semantic similarity for the SVSM

membrane (SM), the middle membrane (MM) and the elemental membrane (EM) [38, 39]. The initial objects of different membranes are randomly generated, and each object is a four dimensional vector in which the range of each dimension is [0,1]. There are eight membranes including five elemental membranes 4, 5, 6, 7, 8, two middle membranes 2, 3 and one superficial membrane 1 in the MCOA method. In addition, the middle membrane 2 contains three elemental membranes 4, 5, 6, the middle membrane 3 contains two elemental membranes 7, 8, and the superficial membrane 1 contains two middle membranes 2, 3. The object in the membrane is a real number vector composed of four weighted factors. Each object has a fitness value computed based on the fitness function. The fitness function of objects is expressed as the reciprocal of the root measure square error between the training topic similarities and the estimated topic similarities of all training hyperlinks. The higher the fitness value of an object is, the better an object is. The object in each membrane and the fitness function are expressed as follows:

$$\begin{gathered} o_{ij} = (\lambda_{i1}^{j}, \lambda_{i2}^{j}, \lambda_{i3}^{j}, \lambda_{i4}^{j}) \quad 1 \le i \le N_j, \quad 1 \le j \le 8 \\ Sim_{ek} = \lambda_{i1}^{j} \cdot Sim_{k1} + \lambda_{i2}^{j} \cdot Sim_{k2} + \lambda_{i3}^{j} \cdot Sim_{k3} + \lambda_{i4}^{j} \cdot Sim_{k4} \\ fit(o_{ij}) = 1 \Big/ \sqrt{\sum_{k=1}^{N} (Sim_{tk} - Sim_{ek})^2} \end{gathered} \tag{4}$$

where o_{ij} is the object i in membrane j, λ_{i1}^{j}, λ_{i2}^{j}, λ_{i3}^{j} and λ_{i4}^{j} are the four weighted factors corresponding to object o_{ij} which are considered as the contribution degrees of full texts, anchor texts, title texts and context texts for the given topic, N_j is the number of objects in membrane j, Sim_{tk} and Sim_{ek} are respectively the training topic similarity and the estimated topic similarity of the training hyperlink k, Sim_{k1}, Sim_{k2}, Sim_{k3} and Sim_{k4} are respectively the training topic similarities of full texts, anchor texts, title texts and context texts of the training hyperlink k, $fit(o_{ij})$ is the fitness function, i.e. $fit(o_{ij})$ is the fitness value of object o_{ij}, N is the number of the training hyperlinks.

3.2.2 Membrane evolution rule

The membrane evolution rule includes the selection rule, the crossover rule and the mutation rule in the MCOA method. The membrane evolution rules are similar as the three operators of the Genetic Algorithm. The selection rule selects objects with the high fitness values for the next generation to enhance convergence and improve computational efficiency. The MCOA method utilizes the Roulette Wheel Selection to realize the selection rule. The crossover rule and mutation rule produce new objects to improve the diversity of the population and avoid premature convergence. The MCOA method utilizes the arithmetic crossover between two objects and the arithmetic variation in a single object to realize the crossover rule and the mutation rule respectively. The selection rule, the crossover rule and the mutation rule in the MCOA method are shown as follows:

$$\begin{gathered} R_{selection}: \; P_{si} = \frac{fit_{ij}}{\sum_{n=1}^{N_j} fit_{nj}} \\ if \;\; P_{srand} \le P_{si}, \;\; then \;\; Current\;\; Generation[o_{ij}]_j \to Next\;\; Generation[o_{ij}]_j \\ R_{crossover}: \; P_{ci}, P_{ck} = \begin{cases} r_c \dfrac{fit_{\max} - fit_{ij}}{fit_{\max} - \overline{fit_j}}, \; r_c \dfrac{fit_{\max} - fit_{kj}}{fit_{\max} - \overline{fit_j}} & fit_{ij}, fit_{kj} \ge \overline{fit_j} \\ r_c & fit_{ij}, fit_{kj} < \overline{fit_j} \end{cases} \\ if \;\; P_{crand} \le P_{ci}, P_{ck}, \quad then \;\; [o_{ij}, o_{kj}]_j \to [o'_{ij}, o'_{kj}]_j \\ o'_{ij} = \alpha o_{ij} + (1-\alpha) o_{kj} \qquad o'_{kj} = \alpha o_{kj} + (1-\alpha) o_{ij} \\ R_{mutation}: \; P_{mi} = \begin{cases} r_m \dfrac{fit_{\max} - fit_{ij}}{fit_{\max} - \overline{fit_j}} & fit_{ij} \ge \overline{fit_j} \\ r_m & fit_{ij} < \overline{fit_j} \end{cases} \\ if \;\; P_{mrand} \le P_{mi}, \quad then \;\; [o_{ij}]_j \to [o'_{ij}]_j \qquad o'_{ij} = o_{ij} + \beta(1,1,1,1)_{1\times 4} \\ r_c = \frac{P_{c\max} - P_{c\min}}{1 + \exp(-aU(t))} + P_{c\min} \quad r_m = \frac{P_{m\max} - P_{m\min}}{1 + \exp(-aU(t))} + P_{m\min} \quad U(t) = \frac{2H}{\log_2(1+t)\log_2 N_j} - 1 \\ H = -\sum_{d=1}^{C_j} P_d \log_2 P_d \;\; (C_j \le N_j) \quad P_d = \frac{n_d}{N_j} \qquad \overline{fit_j} = \frac{1}{N_j} \sum_{n=1}^{N_j} fit_{nj} \qquad fit_{\max} = \max_{1 \le n \le N_j} fit_{nj} \end{gathered} \tag{5}$$

where $R_{selection}$ is the selection rule, P_{si} is the selection probability of object o_{ij}, P_{srand}, P_{crand} and P_{mrand} are random numbers within the range [0,1], fit_{ij}, fit_{kj} and fit_{nj} are the fitness values of three objects o_{ij}, o_{kj} and o_{nj} in membrane j respectively, N_j is the number of objects in membrane j, object o_{ij} is selected from the current generation to the next generation if P_{si} is no less than P_{srand}, $R_{crossover}$ is the crossover rule, P_{ci} and P_{ck} are the crossover probabilities of two objects o_{ij} and o_{kj} respectively, r_c is the parameter in the crossover rule, $\overline{fit_j}$ is the average fitness value for all objects in membrane j, $fit_{\max}$ is the maximum fitness value for all objects in membrane j, $P_{c\max}$ and $P_{c\min}$ are the maximum crossover probability and the minimum crossover probability, two objects o_{ij} and o_{kj} are crossed to generate two new objects o'_{ij} and o'_{kj} if P_{ci} and P_{ck} are no less than P_{crand}, a is a random number within the range [0,1], $R_{mutation}$ is the mutation rule, P_{mi} is the mutation probability of object o_{ij}, r_m is the parameter in the mutation rule, $P_{m\max}$ and $P_{c\min}$ are the maximum mutation probability and the minimum mutation probability, object o_{ij} is mutated to generate a new object o'_{ij} if P_{mi} is no less than P_{mrand}, β is a random number within the range [0,1], $U(t)$ is the function value, a is the given constant, t is the current evolution generation in membrane j, H is the population entropy of all objects in membrane j, C_j is the number of categories for objects in membrane j at the evolution generation t and the same objects are classified as one category, P_d is the category probability which is equal to n_d divided by N_j, n_d is the number of objects in the category d.

3.2.3 Membrane Communication Rule

The membrane communication rule transmits the better objects of the inner membrane to the outer membrane or external environment. The communication rule improves the diversity of objects in the membrane, and prevents the MCOA method falling into a local optimal solution prematurely. In the MCOA method, the best object in the elemental membrane is transmitted to the middle membrane. The middle membrane transmits the best object and better objects to the superficial membrane. The best object in the superficial membrane is transmitted to the external environment and is considered as the optimal object. The communication rule for all membranes transmits the best object with the highest fitness to the outer membrane or the external environment. In addition, the communication rule for the middle membrane transmits the better objects to the outer membrane based on the communication probability. The communication rule in the MCOA method is shown as follows:

$$
R_{communication}:
$$

$$
EM\begin{cases}
(1)\ [o_{ij}]_{j=4,5,6} \rightarrow [o_{i2}]_2 & fit_{ij} = \max\limits_{1\le n\le N_j} fit_{nj} \quad j=4,5,6 \\
(2)\ [o_{ij}]_{j=7,8} \rightarrow [o_{i3}]_3 & fit_{ij} = \max\limits_{1\le n\le N_j} fit_{nj} \quad j=7,8
\end{cases}
$$

$$
MM\begin{cases}
(1)\ [o_{ij}]_{j=2,3} \rightarrow [o_{i1}]_1 & fit_{ij} = \max\limits_{1\le n\le N_j} fit_{nj} \quad j=2,3 \\
(2)\ if\ \ fit_{ij} \ge fit_j^{father-\max} & then\ \ [o_{ij}]_{j=2,3} \rightarrow [o_{i1}]_1 \quad j=2,3 \\
(3)\ if\ \ fit_{ij} < fit_j^{father-\max}\ \ and\ P_{rand} \le P_{ei} & \\
P_{ei} = \exp((fit_{ij} - fit_j^{father-\max})/cg) & then\ \ [o_{ij}]_{j=2,3} \rightarrow [o_{i1}]_1 \quad j=2,3
\end{cases}
$$

$$
SM\quad [o_{ij}]_{j=1} \rightarrow []_{j=1} o_{ij} \qquad fit_{ij} = \max_{1\le n\le N_j} fit_{nj} \qquad j=1 \tag{6}
$$

where *EM* is the elemental membrane, the best objects with highest fitness values in the elemental membrane 4, 5, 6 are transmitted to the middle membrane 2, the best objects with highest fitness values in the elemental membrane 7, 8 are transmitted to the middle membrane 3, fit_{ij} and fit_{nj} are the fitness values of two objects o_{ij} and o_{nj} in membrane j respectively, N_j is the number of objects in membrane j, *MM* is the middle membrane, the best objects with highest fitness values in the middle membrane 2, 3 are transmitted to the superficial membrane 1, $fit_j^{father-\max}$ is the maximum fitness value for the father generation objects in the middle membrane j, object o_{ij} in the middle membrane j is transmitted to the superficial membrane 1 if fit_{ij} is no less than $fit_j^{father-\max}$, P_{rand} is a random number within [0,1], P_{ei} is the communication probability of object o_{ij}, object o_{ij} in the middle membrane j is transmitted to

the superficial membrane 1 if fit_{ij} is less than $fit_j^{father-\max}$ and P_{ei} is no less than P_{rand}, cg is the current communication generation, SM is superficial membrane, the best object o_{ij} with highest fitness value in the superficial membrane 1 is transmitted to the external environment and is considered as the optimal object.

1 is transmitted to the external environment and is considered as the optimal object.

Algorithm 1: The MCOA method optimizing four weighted factors

Input: (1) The topic similarities of the web pages corresponding to the N training hyperlinks are described as Sim_{tk} $(1 \le k \le N)$.

(2) The topic similarities of the four texts corresponding to full texts, anchor texts, title texts and context texts including these N training hyperlinks are respectively described as Sim_{k1}, Sim_{k2}, Sim_{k3} and Sim_{k4} $(1 \le k \le N)$.

Output: Four optimal weighted factors λ_1, λ_2, λ_3 and λ_4

01. Initialize $N_j (1 \le j \le 8)$, $P_{c\min}$, $P_{c\max}$, $P_{m\min}$, $P_{m\max}$, $nEvoGen$, $nComGen$;
02. for(j = 1; j <=8; j++)
03. O_j = RandomGenerateInitialObjects( );
04. for(i = 1; i <= N_j ; i++)
05. fit_{ij} = ComputeFitnessValue( );
06. end for
07. end for
08. while($nComGen <= nMaxComGen$)
09. for(j = 1; j <=8; j++)
10. while($nEvoGen <= nMaxEvoGen$)
11. O_j = UtilizeSelectionRule(O_j);
12. O_j = UtilizeCrossoverRule(O_j);
13. O_j = UtilizeMutationRule(O_j);
14. $nEvoGen$++;
15. end for
16. for(j = 8; j >=1; j--)
17. O_j = UtilizeCommunicationRule(O_j);
18. end for
19. $nComGen$++;
20. end while
21. $(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = \arg\max_{1 \le i \le N_1} fit_{i1}$
22. Return four optimal weighted factors $(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$

The pseudo code of the MCOA method optimizing four weighted factors is given in Algorithm 1. Line 1 initializes parameters including the numbers of initial objects $N_j(1 \leq j \leq 8)$ in all eight membranes, the minimum and maximum crossover probability $P_{c\min}$ and $P_{c\max}$, the minimum and maximum mutation probability $P_{m\min}$ and $P_{m\max}$, the evolution generation *nEvoGen*, and the communication generation *nComGen*. In lines 2 to 7, initial object sets are randomly generated in all eight membranes, O_j is currently the object set in membrane j, and each object corresponds to four weighted factors each of which range from 0 to 1. In addition, the fitness values of initial objects are computed based on the formula (4) in all eight membranes. In lines 8 to 20, objects in all eight membranes are optimized based on the evolution rule and the communication rule. In addition, the evolution generation *nEvoGen* is no more than the maximum evolution generation *nMaxEvoGen*, and the communication generation *nComGen* is no more than the maximum communication generation *nMaxComGen*. In lines 9 to 15, objects in all eight membranes are evolved based on the evolution rule which includes the selection rule, the crossover rule and the mutation rule by using the formula (5). In lines 16 to 18, after the evolution of objects in all eight membranes, these objects are transmitted from the inner membrane to the outer membrane based on the communication rule by using the formula (6). Line 21 obtains four optimal weighted factors with the maximum fitness value in the superficial membrane 1. Line 22 returns the four optimal weighted factors.

The time complexity of Algorithm 1 is mainly determined by the maximum evolution generation *nMaxEvoGen*, the maximum communication generation *nMaxComGen* and the number N_j of objects in membrane j. Therefore, Suppose $N_{\max}$ is the maximum number of objects for all eight membranes, and then the time complexity of this algorithm is $O(nMaxEvoGen * nMaxComGen * N_{\max})$.

3.3 Priority Prediction Method

The priority prediction method predicts the priorities of unvisited hyperlinks by calculating the topic similarities of unvisited hyperlinks. The priority of each unvisited hyperlink is the linear combination value between the topic similarities of four texts and the four optimal weighted factors which is the topic similarity of the unvisited hyperlink. The SVSM method is used to the topic similarities of four texts by calculating the cosine similarities between the text semantic vectors and the topic semantic vectors. The MCOA method is used to obtain the optimal weighted factors by using the evolution rules and the communication rule. The linear combination values between the topic similarities of four texts and the corresponding optimal weighted factors are considered as the priorities of unvisited hyperlinks. The priority of each unvisited hyperlink is calculated as follows:

$$P(k) = \lambda_1 Sim(f_k, t) + \lambda_2 Sim(a_k, t) + \lambda_3 Sim(t_k, t) + \lambda_4 Sim(c_k, t) \quad (7)$$

where $P(k)$ is the priority of the unvisited hyperlink k, λ_1, λ_2, λ_3 and λ_4 are four optimal weighted factors corresponding to the topic contribution degrees of four texts, $Sim(f_k, t)$ is the topic similarity of the full text f_k, $Sim(a_k, t)$ is the topic similarity of the anchor text a_k, $Sim(t_k, t)$ is the topic similarity of the title text t_k, $Sim(c_k, t)$ is the topic similarity of the context text c_k.

The proposed method can improve the performance of the focused crawler. The focused crawler based on SVSM and MCOA predicts the priorities of unvisited hyperlinks by using the linear combination values between optimal four weighted factors and topic similarities of four texts. The SVSM method utilizes the cosine similarity of the text semantic vector and the topic semantic vector as the topic similarity of the text. The MCOA method obtains the optimal four weighted factors to show that different texts contribute to the priorities of unvisited hyperlinks more objectively. The proposed method can guide the focused crawler to select unvisited hyperlinks with high priorities to download more web pages related to the given topic from the Internet.

4 Experiment

The experiment system for focused crawlers is constructed to indicate that the SVSM and MCOA method can improve the performance of focused crawlers. The experiment designs five focused crawlers, prepares initial data, and provides evaluation indicators. The experiment results are obtained to compare this proposed focused crawler with other five focused crawlers according to evaluation indicators. The experiment results explain that the SVSM and MCOA method can guide the focused crawler to grab more quantity and better quality web pages related to the given topic from the Internet.

4.1 Experimental design

The experiment design is the preparation work of the whole experiment. The experiment design includes the experimental focused crawlers, experimental initial data and experimental evaluation indicators. The above three contents will be described in the following.

Table 2 The initial URLs for 10 different topics

Topics	Initial URLs
1. Network Public Opinion	https://onlinelibrary.wiley.com/doi/10.1002/cpe.4212
	https://ui.adsabs.harvard.edu/abs/2019IJMPB..3350393Z/abstract
	https://www.scientific.net/AMM.416-417.1533
2. Artificial Intelligence	https://www.ibm.com/watson-health/learn/artificial-intelligence-medicine
	https://www.sciencenews.org/topic/artificial-intelligence
	https://cs.illinois.edu/research/areas/artificial-intelligence
3. Big Data	https://datasciencedegree.wisconsin.edu/data-science/what-is-big-data
	https://www.statista.com/topics/1464/big-data
	https://www.dataversity.net/brief-history-big-data
4. Machine Learning	https://www.trendmicro.com/vinfo/us/security/definition/machine-learning
	https://www.investopedia.com/terms/m/machine-learning.asp
	https://online-learning.harvard.edu/course/data-science-machine-learning
5. Deep Learning	https://www.mygreatlearning.com/blog/what-is-deep-learning
	https://www.tutorialkart.com/deep-learning
	https://simplicable.com/new/deep-learning
6. Neural Networks	https://www.solver.com/training-artificial-neural-network-intro
	https://data-flair.training/blogs/neural-network-algorithms/
	https://www.guru99.com/backpropogation-neural-network.html
7. Smart Medical	https://smarthealthamsterdam.com/
	https://smart-visa.boi.go.th/smart/pages/medical-hub.html
	https://www.medicaldevice-network.com/projects/impulse-dynamics-optimizer-smart-system/
8. Blockchain	https://www.nist.gov/blockchain
	http://blockchain.mit.edu/how-blockchain-works
	https://blockchain.ieee.org/
9. Knowledge Graph	https://www.ibm.com/cloud/learn/knowledge-graph
	https://deepai.org/machine-learning-glossary-and-terms/knowledge-graph
	https://www.cambridgesemantics.com/
10. Speech Recognition	https://research.mozilla.org/machine-learning/
	https://answers.microsoft.com/en-us/windows/forum/windows_10-desktop/speech-recognition-not-working-windows-10/317f7371-a09c-430c-bfa4-31ecd1abdb26
	https://www.aware.com/voice-authentication/

4.1.1 Experimental focused crawler

The experiment designs five focused crawlers according to the texts of hyperlinks and the calculation methods of the topic similarities of texts. The five focused crawlers include the BF Crawler, the VSM Crawler, the SSRM Crawler, the MCOA Crawler and the SVSM-MCOA Crawler. The BF Crawler grabs web pages based on the Breadth First Search algorithm. The VSM Crawler computes the topic similarity of the text based on the VSM method, and the texts of hyperlinks include the full texts and anchor texts. The SSRM Crawler computes the topic similarity of the text based on the SSRM method, and the texts of hyperlinks also include the full texts and anchor texts. The MCOA Crawler computes the topic similarity of the text based on the VSM method, acquires optimal weighted factors based on the MCOA method, and the texts of hyperlinks include full texts, anchor texts, title texts and context texts. The SVSM-MCOA Crawler combines the SVSM method which is used to calculate the topic similarities of the texts and the MCOA method which is used to acquire optimal weighted factors, and the texts of hyperlinks also include full texts, anchor texts, title texts and context texts. Compared with the MCOA Crawler, the SVSM-MCOA Crawler computes the topic similarity of the text based on the SVSM method in difference, but acquires optimal weighted factors based on the MCOA method in same.

4.1.2 Experimental initial data

The above five focused crawlers are given the same topic set and initial data to compare the performance of five focused crawlers. The more topics the topic set contains, the more persuasive the experimental results are. In the experiment,

Table 3 Optimal weighted factors for all 10 topics based on the MCOA method

Topics	λ_1	λ_2	λ_3	λ_4
1.Network Public Opinion	0.5989	0.0546	0.1506	0.1048
2. Artificial Intelligence	0.3967	0.4396	0.0514	0.4463
3. Big Data	0.6079	0.3979	0.1032	0.3363
4. Machine Learning	0.7579	0.1184	0.0540	0.0006
5. Deep Learning	0.4070	0.3365	0.0029	0.2454
6. Neural Networks	0.3532	0.2311	0.2580	0.4715
7. Smart Medical	0.5151	0.1030	0.2685	0.2418
8. Blockchain	0.4403	0.0093	0.0565	0.4493
9. Knowledge Graph	0.6087	0.0361	0.1284	0.2016
10. Speech Recognition	0.4203	0.3188	0.8215	0.2818

the topic set includes 10 different topics: Network Public Opinion, Artificial Intelligence, Big Data, Machine Learning, Deep Learning, Neural Networks, Smart Medical, Blockchain, Knowledge Graph and Speech Recognition. The initial data includes the topic web page set, the crawling data set and the training data set. The topic web page set describes the topic information and the contents of web pages are related to the topics. The crawling data set includes initial URLs of different topics, and initial URLs of each topic start focused crawlers to grab web pages related to the topic. The training data set includes training data, test data and training parameters for different topics. In addition, the content size of each web page should not exceed 100 kb and the max number of downloaded web pages is 5000.

The topic web page set can be used to calculate topic similarities between web pages and topics. This web page set can be acquired by the web crawler. In the experiment, the size of the topic web page is set as 20 to reduce the time complexity. Firstly, 10 different topics are respectively inputted into the general search engine like Bing or Google to acquire a large of web pages related to the topics. These retrieved web pages have been sorted by topic similarities, and the most relevant web page will be at the top of the result list. Then, the experiment directly records the top 20 URLs from the result list to the specified file. Finally, the web crawler is used to download the topic web page set through the specified file for each topic.

The crawling data set includes initial URLs of different topics. This data set can be used to extract new hyperlinks for focused crawlers to continuously download web pages from the Internet. Table 2 shows initial URLs of 10 different topics and each topic has 3 different initial URLs.

The training data set includes the training data, test data and training parameters for different topics. The training data set is used to acquire the optimal four weighted factors. The training data and test data of each topic include the topic similarities of the web pages corresponding to the training and test hyperlinks and the topic similarities of the four texts corresponding to full texts, anchor texts, title texts and context texts including these hyperlinks. The experiment obtains 1000 groups of topic similarities for all training and test hyperlinks. Each hyperlink corresponds to a group of topic similarities which has five values for five different texts. In addition, the five different texts of each hyperlink contains the web page corresponding to the hyperlink and four texts corresponding to full texts, anchor texts, title texts and context texts in the parent web page including the hyperlink. The 1000 groups of topic similarities are divided into training data and testing data, and the first 500 data and the last 500 data are considered as training data and testing data respectively.

The training parameters are set to acquire the optimal weighted factors by using training data in the MCOA method [18]. In the experiment, all training parameters are given as follows: $N_1 = 150$, $N_2 = N_3 = 150$, $N_j = 50\ (4 \le j \le 8)$, $P_{c\max} = 0.7$, $P_{c\min} = 0.3$, $P_{m\max} = 0.5$, $P_{m\min} = 0.1$, $nEvoGen \in [1, 3000]$, $nEvoGen \in [1, 3]$, $\alpha = 9.903438$ in Algorithm 1 and formula (5). The maximum evolution generation is set as 3000 which can reach the convergence for the MCOA method. The max communication generation is set as 3 which can reduce the time

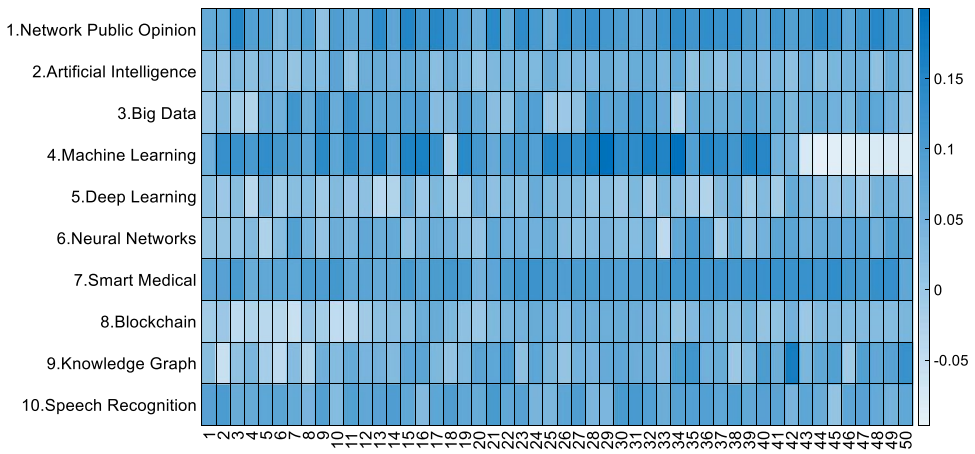

Fig. 3 The heat map of the difference values between absolute errors based on the manual method and the MCOA method for all different topics

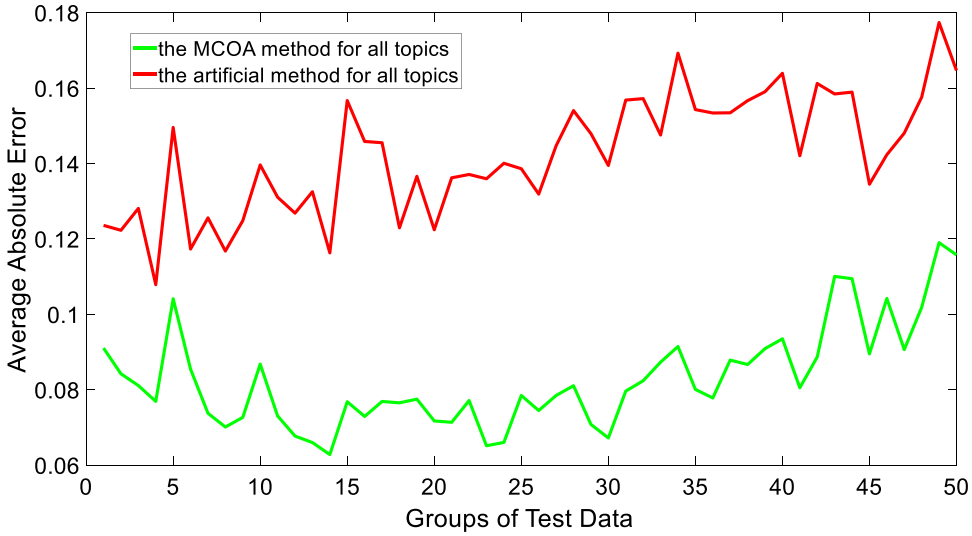

Fig. 4 The average absolute errors for all different topics based on the MCOA method and the manual method

complexity for the MCOA method. In addition, initial objects in each membrane are randomly generated and each object corresponds to four weighted factors each of which range from 0 to 1.

4.1.3 Experimental evaluation indicator

In the experiment, the performance of all focused crawlers is evaluated based on three indicators including the relevant number, the harvest rate and the average topic similarity [18, 34]. The relevant number refers to the number of web pages related to the given topic in all retrieved pages. The relevant number can indicate the number of topic-relevant web pages which the focused crawler can grab. The harvest rate refers to the proportion of the number of topic-relevant pages to the number of all retrieved pages. The harvest rate can indicate the speed at which the focused crawler grabs topic-relevant web pages. The average topic similarity refers to the average value of topic similarities of all topic-relevant web pages. The average topic similarity can indicate the quality of topic-relevant web pages retrieved by the focused crawler.

Table 4 The crawling results including RN, HR and AS for five focused crawlers

Numbers of retrieved pages	BF Crawler			VSM Crawler			SSRM Crawler			MCOA Crawler			SVSM-MCOA Crawler		
	RN	HR	AS	RN	HR	AS	RN	HR	AS	RN	HR	AS	RN	HR	AS
100	49	0.491	0.434	43	0.434	0.452	45	0.452	0.442	48	0.478	0.457	48	0.482	0.475
200	96	0.481	0.436	93	0.467	0.448	91	0.453	0.447	99	0.496	0.458	105	0.526	0.475
300	138	0.46	0.433	139	0.464	0.448	140	0.467	0.448	151	0.502	0.461	166	0.552	0.472
400	182	0.454	0.434	185	0.462	0.447	182	0.454	0.447	195	0.489	0.46	216	0.54	0.475
500	235	0.47	0.435	228	0.456	0.446	223	0.446	0.445	247	0.494	0.458	267	0.534	0.475
600	279	0.465	0.435	275	0.459	0.444	268	0.446	0.444	290	0.484	0.457	312	0.52	0.473
700	329	0.47	0.435	316	0.452	0.446	312	0.446	0.442	337	0.483	0.456	352	0.502	0.473
800	375	0.469	0.435	364	0.455	0.446	357	0.447	0.443	382	0.478	0.457	394	0.492	0.471
900	419	0.466	0.435	408	0.454	0.445	401	0.445	0.443	425	0.474	0.456	446	0.495	0.469
1000	465	0.465	0.435	455	0.455	0.446	444	0.444	0.443	475	0.476	0.457	502	0.502	0.469
1500	671	0.447	0.432	693	0.462	0.445	684	0.456	0.443	712	0.475	0.454	755	0.503	0.471
2000	895	0.447	0.43	889	0.444	0.445	926	0.463	0.455	936	0.469	0.462	995	0.498	0.474
2500	1082	0.433	0.428	1113	0.445	0.444	1171	0.468	0.456	1184	0.475	0.463	1223	0.489	0.474
3000	1307	0.436	0.427	1360	0.453	0.442	1387	0.462	0.455	1416	0.473	0.458	1465	0.488	0.473
3500	1532	0.438	0.427	1594	0.455	0.442	1581	0.452	0.456	1649	0.471	0.461	1715	0.49	0.472
4000	1737	0.434	0.426	1827	0.457	0.442	1787	0.447	0.457	1865	0.466	0.459	1928	0.482	0.472
4500	1968	0.437	0.425	2066	0.459	0.443	2031	0.451	0.456	2095	0.465	0.458	2143	0.476	0.473
5000	2197	0.439	0.425	2310	0.462	0.442	2247	0.449	0.456	2316	0.462	0.458	2342	0.468	0.473

The above three indicators can evaluate the performance of focused crawlers. The evaluation indicators are calculated as follows:

$$H = \frac{n}{N} \quad \bar{R} = \frac{1}{n}\sum_{i=1}^{n} R_1 \quad R_1 \geq th \quad (1 \leq i \leq n) \tag{8}$$

where H is the harvest rate, n is the number of topic-relevant pages, N is the number of all retrieved pages, $\bar{R}$ is the average topic similarity for all topic-relevant pages, R_i is the topic similarity of topic-relevant page i, th is the parameter to judge whether the web page is related to the topic,. In the experiment, the topic similarities of retrieved web pages are acquired by uniformly using the VSM. In addition, due to that the size of the topic web page set is 20, there are the 20 different topic similarities of each retrieved web pages. In the experiment, the maximum of these 20 values is considered as the topic similarity of the web page.

4.2 Experimental Training Results

This experiment provides the training data and utilizes the MCOA method to obtain the optimal four weighted factors. The MCOA method utilizes the evolution rules and the communication rule to optimize the four weighted factors for four different texts based on the training data and the convergent evolution generation in the experiment. The MCOA method is given the convergent evolution generation for all different topics. In the experiment, when the evolution generation reaches 3000, the optimal object is considered as the optimal four weighted factors for each topic. Table 3 shows the optimal weighted factors for all 10 topics based on the MCOA method. In Table 3, for the topic "Network

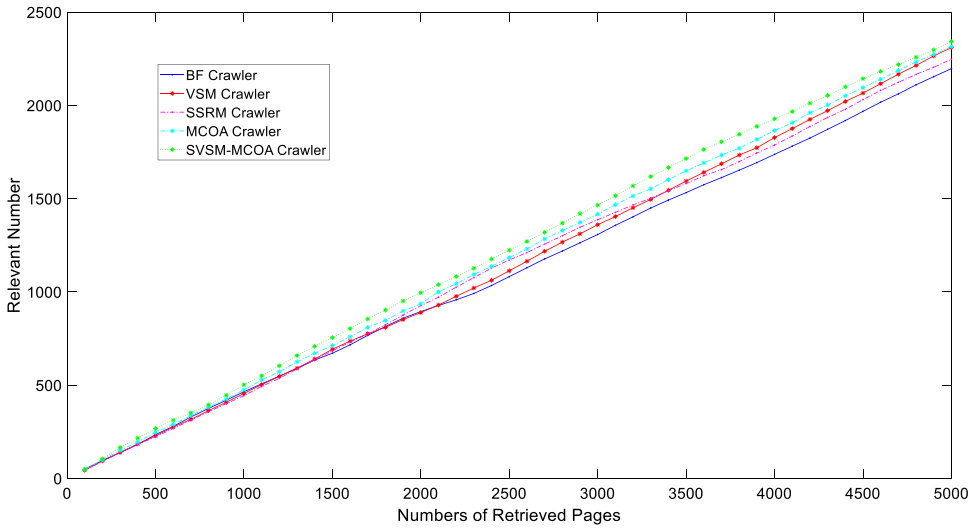

Fig. 5 The comparison of the relevant number for five focused crawlers based on the crawling results

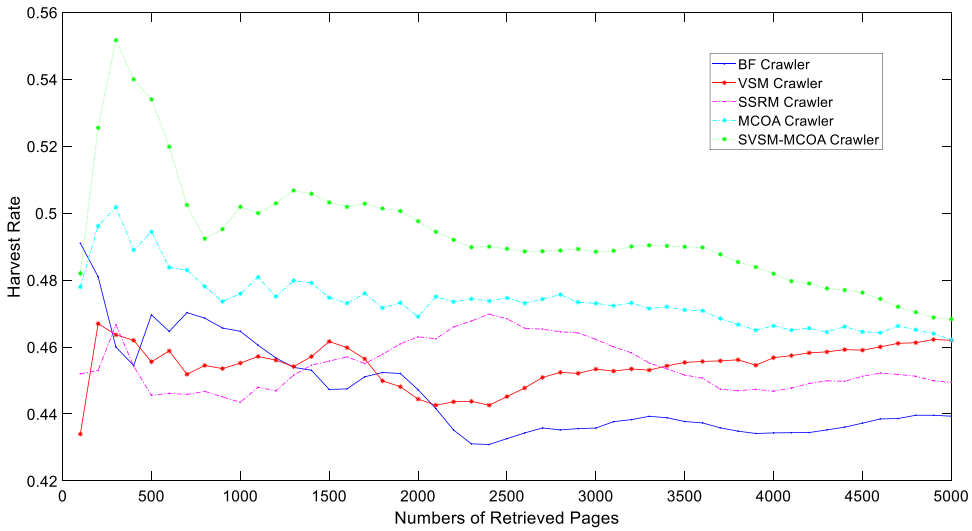

Fig. 6 The comparison of the harvest rate for five focused crawlers based on the crawling results

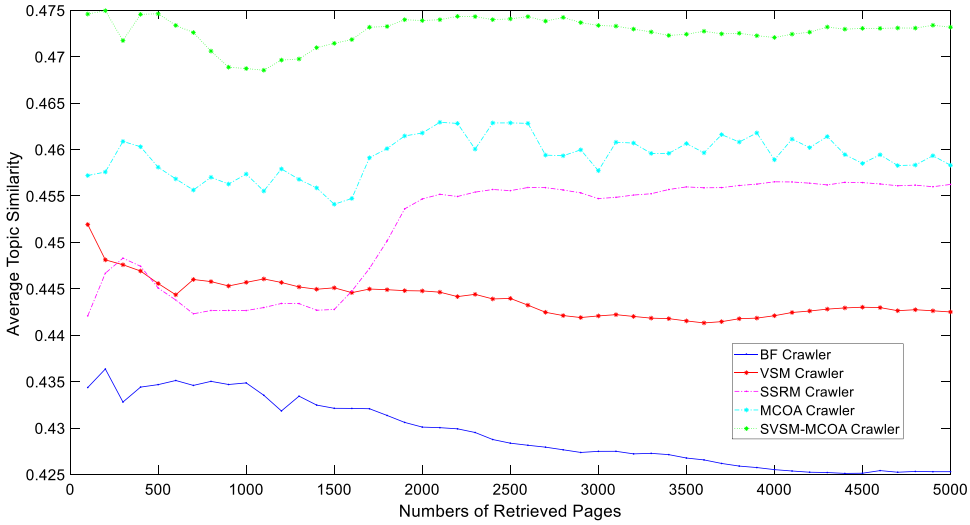

Fig. 7 The comparison of the average topic similarity for five focused crawlers based on the crawling results

Public Opinion", the optimal weighted factors are (0.5989, 0.0546, 0.1506, 0.1048) corresponding to the topic contribution degrees of full texts, anchor texts, title texts and context texts.

The experiment utilizes the test data to compare the above optimal four weighted factors determined by the MCOA method with the four weighted factors determined by the manual method. The manual method which is arbitrary and subjective utilizes the experience of researchers to determine the four weighted factor. The four weighted factors determined by the MCOA method are showed for all 10 topics in Table 3, while the four weighted factors determined by the manual method are divided equally in general, namely (0.25, 0.25, 0.25, 0.25). The test data is used to compare the four weighted factors determined by the MCOA method and the manual method respectively. The test data contains 500 groups, and each method has 500 absolute errors for each topic. In order to compare the two methods more clearly, the experiment divides the above 500 groups of the test data into 50 groups for each topic, i.e. every 10 groups of the test data are combined into one group for each topic. Then, each group of the test data after the combination corresponds to 10 absolute errors, and the average of these absolute errors is regarded as the average absolute error of this group for each topic. Finally, each method has the 50 average absolute errors for each topic after the combination of the test data.

Figure 3 shows the heat map of the difference values between absolute errors based on the manual method and the MCOA method for all different topics. In Fig. 3, the horizontal label represents 50 groups of test data for all different topics, and the vertical label represents 10 different topics. The color bar on the right represents the different colors corresponding to the difference values between the absolute errors based on the manual method and the absolute errors based on the MCOA method, and the color of each cell represents a difference value in the heat map. Obviously, the smaller the absolute error based on a method is, the better the method is. If the difference value is higher than 0, the absolute error based on the MCOA method is smaller than the absolute error based on the manual method, so the MCOA method is obviously better than the manual method. On the contrary, the MCOA method is obviously worse than the manual method. In Fig. 3, the colors of most cells display that most of the difference values are higher than 0, and indicate that the MCOA method is obviously better than the manual method for determining optimal four weighted factors. For example, the color of the cell in the first row and the third column represents the difference value 0.1489 in Fig. 3. This difference value is higher than 0, and indicates that the absolute error based on the MCOA method is smaller than based on the manual method for the 3-th group test data in the first topic "Network Public Opinion", so the MCOA method is obviously better than the manual method for this group test data.

Figure 4 shows the average absolute errors based on the two methods for all different topics. In Fig. 4, the average absolute errors determined by the MCOA method are obviously smaller than the average absolute errors determined by the manual method for all 50 groups of the test data. Therefore, the test results indicate that the four weighted factors determined by the MCOA method are more effective than the manual method.

The training results in the experiment is used to obtain the optimal weighted factors. First of all, the training results show the optimal weighted factors for all different topics by using the MCOA method. In addition, the test results show that the MCOA method can obtain more accurate four weighted factors than the manual method. The optimal weighted factors obtained based on the

MCOA method will be used to predict the priorities of unvisited hyperlinks for all different topics.

4.3 Experimental crawling results

The crawling results are obtained by all five focused crawlers using the initial URLs for all 10 topics in the experiment. Table 4 displays the average results of the crawling results obtained by all five focused crawlers for all 10 topics. In Table 4, there are three evaluation indicators including the relevant number (RN), the harvest rate (HR), and the average topic similarity (AS) for the crawling results. In addition, the number of retrieved pages starts with 100 and consecutively increases by 100 until 5000 for all 10 topics in Table 4. Figure 5, 6, and 7 respectively show the comparison of the three evaluation indicators for all five focused crawlers based on Table 4.

Figure 5 shows the comparison of the relevant number for five focused crawlers based on the crawling results. In Fig. 5, for the all numbers of retrieved web pages, the relevant number for the SVSM-MCOA Crawler is higher than for the other four focused crawlers. Figure 5 indicates that the SVSM-MCOA Crawler can collect more topic-relevant web pages than the other four focused crawlers. Figure 6 shows the comparison of the harvest rate for five focused crawlers based on the crawling results. In Fig. 6, for the all numbers of retrieved web pages, the harvest rate for the SVSM-MCOA Crawler is significantly higher than for other four focused crawlers. Figure 6 indicates that the SVSM-MCOA Crawler can collect topic-relevant web pages faster than other four focused crawlers. Figure 7 shows the comparison of the average topic similarity for five focused crawlers based on the crawling results. In Fig. 7, for the all numbers of retrieved web pages, the average topic similarity for the SVSM-MCOA Crawler is significantly higher than for the other four focused crawlers. Figure 7 indicates that the SVSM-MCOA Crawler can collect better topic-relevant web pages than other four focused crawlers.

The experiment obtains the crawling results to compare the performance of five focused crawlers. First of all, the crawling results indicate that the SVSM-MCOA Crawler can retrieve relevant web pages more, faster, and better than other four focused crawlers including the BF Crawler, the VSM Crawler, the SSRM Crawler and the MCOA Crawler from the Internet. Secondly, the crawling results indicate that the SVSM method can acquire the more accurate topic similarity between the text and the given topic than the VSM method and the SSRM method. Finally, the crawling results indicate that the SVSM-MCOA Crawler and the MCOA Crawler which utilize the optimized four weighted factors based on the MCOA method can predict more accurate priority values of unvisited hyperlinks than the VSM Crawler and the SSRM Crawler which utilize the equivalent four weighted factors based on the manual method. The crawling results indicate that the SSVSM and CMCOA method can improve the performance of the focused crawler.

5 Conclusion and future work

The SVSM and MCOA method is proposed to improve the performance of the focused crawler. The SVSM method is used to calculate topic similarities between four texts including full texts, anchor texts, title texts and context texts of each unvisited hyperlink and the given topic. The MCOA method is used to optimize four weighted factors based on the evolution rules and the communication rule. The SVSM-MCOA Crawler predicts the priority of each unvisited hyperlink by integrating the four topic similarities of four texts and the optimal four weighted factors. The experiment results indicate that the proposed SVSM-MCOA Crawler improves the evaluation indicators compared with the BF Crawler, the VSM Crawler, the SSRM Crawler, and the MCOA Crawler. In conclusion, the proposed method promotes the focused crawler to have semantic understanding and intelligent learning ability.

In the future, some research studies are still worth to further study. First of all, the semantic similarity between terms in this paper is obtained only through the content information of the concept nodes in WordNet. Both the structure information and the content information of these concept nodes can be used to obtain the more accurate semantic similarity between terms. In addition, the proposed method mainly calculates the topic similarity of each unvisited hyperlink by using different text contents of web pages. In addition to the text content information, the link structure information can be used to obtain the more accurate topic similarity of each unvisited hyperlink.

Acknowledgements This work is supported by the National Natural Science Foundation of China (Grant No. 61872298), the Science and Technology Department of Sichuan Province (Grant No. 2021YFQ0008), the College Student Innovation and Entrepreneurship Training Project of Sichuan Province (Grant No. S202110650044) and the Education and Teaching Reform Research Project of Xihua University (Grant No. xjjg2019026).

References

1. Pant G, Srinivasan P (2006) Link contexts in classifier-guided topical crawlers. IEEE Trans Knowl Data Eng 18(1):107–122
2. Tsikrika T, Moumtzidou A, Vrochidis S et al (2016) Focussed crawling of environmental web resources based on the combination of multimedia evidence. Multimedia Tools and Applications 75(3):1563–1587

3. Yang YK, Du YJ, Sun JY et al (2008) A topic-specific web crawler with concept similarity context graph based on FCA. Lect Notes Comput Sci 5227(1):840–847
4. Batsakis S, Petrakis EGM, Milios E (2009) Improving the performance of focused web crawlers. Data Knowl Eng 68(10):1001–1013
5. P Hegade N Lingadhal S Jain et al 2021 Crawler by Contextual Inference SN Computer Science 2 3 216 1 12
6. Lu HQ, Zhan DH, Zhou L et al (2016) An Improved Focused Crawler: Using Web Page Classification and Link Priority Evaluation. Math Probl Eng 2016(3):1–10
7. Rajiv S, Navaneethan C (2021) Keyword weight optimization using gradient strategies in event focused web crawling. Pattern Recogn Lett 142:3–10
8. Farag MMG, Lee S, Fox EA (2018) Focused crawler for event. Int J Digit Libr 19(1):3–19
9. Patel A, Schmidt N (2011) Application of structured document parsing to focused web crawling. Computer Standards & Interfaces 33(3):325–331
10. Li MM, Li CL, Wu C et al (2015) A Focused Crawler URL Analysis Algorithm based on Semantic Content and Link Clustering in Cloud Environment. International Journal of Grid and Distributed Computing 8(2):49–60
11. Prabha KSS, Mahesh C, Raja SP (2021) An Enhanced Semantic Focused Web Crawler Based on Hybrid String Matching Algorithm. Cybernetics and Information Technologies 21(2):105–120
12. Salton G, Wong A, Yang CS (1975) A vector space model for automatic indexing. Communications of the Association for Computing Machinery 18(11):613–620
13. Varelas G, Voutsakis E, Raftopoulou P et al (2005) Semantic similarity methods in WordNet and their application to information retrieval on the web. Proceedings of the 7th annual ACM international workshop on Web information and data management, Bremen, Germany. 10–16
14. Wang GG, Deb S, Cui ZH (2019) Monarch butterfly optimization. Neural Comput Appl 31:1995–2014
15. Li S, Chen H, Wang MJ et al (2020) Slime mould algorithm: A new method for stochastic optimization. Futur Gener Comput Syst 111:300–323
16. Yang YT, Chen HL, Heidari AA et al (2021) Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. Expert Systems With Applications 177:114864
17. Ahmadianfar I, Heidari AA, Gandomi AH et al (2021) RUN beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method. Expert Systems With Applications 181:115079
18. Liu WJ, Du YJ (2014) A novel focused crawler based on cell-like membrane computing optimization algorithm. Neurocomputing 123(1):266–280
19. Pavkovic M, Protic J (2019) SInFo - Structure-Driven Incremental Forum Crawler That Optimizes User-Generated Content Retrieval. IEEE Access 7:126941–126961
20. Lagopoulos A, Tsoumakas G (2020) Content-aware web robot detection. Appl Intell 50(11):4017–4028
21. Zhao W, Guan ZY, Cao ZW et al (2016) Mining and Harvesting High Quality Topical Resources from the Web[J]. Chin J Electron 25(1):48–57
22. Seyfi A, Patel A, Celestino J (2016) Empirical evaluation of the link and content-based focused Treasure-Crawler. Computer Standards & Interfaces 44:54–62
23. Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 30(1–7):107–117
24. Kleinberg JM (1999) Authoritative sources in a hyperlinked environment. J ACM 46(5):604–632
25. Diligenti M, Coetzee FM, Lawrence S et al (2000) Focused crawling using context graphs. Proceedings of the 26th International Conference on Very Large Database (VLDB), Cairo, Egypt 527–534
26. Hsua CC, Wu F (2006) Topic-specific crawling on the Web with the measurements of the relevancy context graph. Inf Syst 31(4–5):232–246
27. Hernandez J, Marin-Castro HM, Morales-Sandoval M (2020) A Semantic Focused Web Crawler Based on a Knowledge Representation Schema. Applied Science, 10(11): 3837, 1–21
28. Capuano A, Rinaldi AM, Russo C (2020) An ontology-driven multimedia focused crawler based on linked open data and deep learning techniques. Multimedia Tools and Applications 79(11–12):7577–7598
29. Hliaoutakis A, Varelas G, Voutsakis E et al (2006) Information retrieval by semantic similarity. Int J Semant Web Inf Syst 3(3):55–73
30. Zhang GX, Pan LQ (2010) A Survey of Membrane Computing as a New Branch of Natural Computing. Chinese Journal of Computers 2:208–214
31. Wang W, Yu LH (2021) UCrawler: A learning-based web crawler using a URL knowledge base. Journal of Computational Methods in Sciences and Engineering 21(2):461–474
32. Dong H, Hussain FK (2013) SOF: a semi-supervised ontology-learning-based focused crawler. Concurrecny and Computation-Practice & Experience 25(12):1755–1770
33. Zhang HX, Lu J (2010) SCTWC: An online semi-supervised clustering approach to topical web crawlers. Appl Soft Comput 10(2):490–495
34. Du YJ, Liu WJ, Lv XJ et al (2015) An improved focused crawler based on Semantic Similarity Vector Space Model. Appl Soft Comput 36(11):392–407
35. Prakoso DW, Abdi A, Amrit C (2021) Short text similarity measurement methods: a review. Soft Comput 25(6):4699–4723
36. Mohammed N, Mohammed D (2017) Experimental Study of Semantic Similarity Measures on Arabic WordNet. International Journal of Computer Science and Network Security 17(2):131–140
37. Lin D (1998) An Information-Theoretic Definition of Similarity. Proceedings of the 15th International Conference on Machine Learning, Madison, USA, 296–304
38. Li ZX, Zhang L, Su YS et al (2018) A skin membrane-driven membrane algorithm for many-objective optimization. Neural Comput Appl 30(1):141–152
39. Raghavan S, Chandrasekaran K (2021) Membrane-based models for service selection in cloud. Inf Sci 558:103–123