

Research Article

An Efficient Web Usage Mining Approach Using Chaos Optimization and Particle Swarm Optimization Algorithm Based on Optimal Feedback Model

Lu Dai,¹ Wei Wang,² and Wanneng Shu³

¹ Computer School, Hubei University of Technology, Wuhan 430068, China

² College of Electronics and Information Engineering, Sichuan University, Chengdu 610064, China

³ College of Computer Science, South-Central University for Nationalities, Wuhan 430074, China

Correspondence should be addressed to Wei Wang; ww0830@gmail.com

Received 22 July 2013; Revised 18 August 2013; Accepted 21 August 2013

Academic Editor: Baochang Zhang

Copyright © 2013 Lu Dai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The dynamic nature of information resources as well as the continuous changes in the information demands of the users has made it very difficult to provide effective methods for data mining and document ranking. This paper proposes an efficient particle swarm chaos optimization mining algorithm based on chaos optimization and particle swarm optimization by using feedback model of user to provide a listing of best-matching webpages for user. The proposed algorithm starts with an initial population of many particles moving around in a D -dimensional search space where each particle vector corresponds to a potential solution of the underlying problem, which is formed by subsets of webpages. Experimental results show that our approach significantly outperforms other algorithms in the aspects of response time, execution time, precision, and recall.

1. Introduction

With the rapid development in Internet technology, the number of webpages and the volume of information content have led to an explosion in the amount of available information. While there may be some webpages that are more relevant, popular, or authoritative than others, web users look forward to easily, search the most interesting and significant website by specifying relevant keywords [1]. When a user enters a query into a search engine by using keywords, the engine will provide a list of best-matching webpages according to its criteria [2]. It becomes increasingly important to help user find useful information more easily and quickly.

It is well known that web search is one of the most universal and influential applications on the Internet. Web search engines can support users on a wide variety of topics across a comprehensive range of websites [3]. In order to achieve the typical massive content collections rapid response to a specific query form, web mining technology appears in the technical background and social environment [4]. Web

mining is a comprehensive data mining method that uses data mining techniques from web-related resources and extracts interesting behavior, useful patterns and implicit information, involving web technology, data mining, computational linguistics, artificial intelligence, machine learning and other fields.

Web mining can be viewed as the extraction of structure from an unlabeled, semistructured data set containing the characteristics of users and information [5]. Web mining is divided into web content mining and web usage mining. Web content mining is mining the web page content and the background of transactional database and obtaining useful knowledge from the web document and the description of the content information regarding the websites [6]. Web usage mining is done by mining the appropriate website log files and related data to discover frequent browsing patterns based on clickstream data analysis.

A number of novel optimization methods have been proposed to optimize the web usage evaluation function. The conventional web mining approach makes a type of relevance

ranking, whereby a webpage may be relevant to a topic or theme. Taking into account the amount of available information, the processing essentially requires adequate approaches suitable for extracting only the relevant, sometimes hidden, knowledge as the final result of the problem under consideration. A heuristic intelligent mining approach can be derived using particle swarm optimization (PSO) for addressing the web mining.

This paper focuses on feedback model of user by using chaos optimization and particle swarm optimization to help user find useful information as fast as possible. The dynamic feature of web information as well as the continuous changes in the information demands of the users has made it very difficult to provide efficient and effective approaches for data mining techniques. To realize the goal of searching useful information effectively and efficiently, we have developed an efficient particle swarm chaos optimization mining algorithm (PSCOMA) based on chaos optimization and particle swarm optimization by using feedback model of user to provide a list of best-matching webpages for user. We compare our approach with PSO, PCS, and HITS algorithms, and, as far as we know, it is currently the best method for the problem considered. Experimental results show that our approach significantly outperforms other algorithms in most cases.

The remainder of the paper is organized as follows. A brief survey is given in Section 2. We study the user's feedback behavior and formalize it as a mathematical optimization model in Section 3. Section 4 explains the details of the PSCOMA. Section 5 discusses the experimental results and compares them with other Web mining algorithms. Finally, Section 6 concludes the paper and discusses some future research directions.

2. Related Works

In this section, we focus our discussion on the prior research on web mining. The web mining is a very important problem and has attracted much attention of many researchers.

Drs. Yin and Guo proposed a new formulation for the website structure optimization (WSO) problem based on a comprehensive survey of existing works and practice considerations [7]. In [8], the greedy-add algorithm with backtracking was introduced for obtaining the initial solution, and a better solution is sought through VNS as well as tabu search. Dr. Chen et al. presented a page clipping synthesis (PCS) search method to extract relevant paragraphs from other web search results [9]. Spink et al. reported the results of a major study examining the overlap among four major web search engines for results retrieved for the same queries [10]. In [11], proposed hybrid algorithm makes use of the strong global search ability of particle swarm optimization and the strong local search ability of tabu search to obtain high quality solutions in web mining. Herrera et al. presented a study about the impact of using several features extracted from the document collection and query logs on the task of automatically identifying the users' goals behind their queries [12]. Ling and Van Schaik reported findings from two experiments that explored the influence of font type and line length on a range of performance and

subjective measures [13]. Moawad et al. proposed a new multiagent system based approach for personalizing the web search results. The proposed approach introduces a model to build a user profile from initial and basic information and maintain it through implicit user feedback to establish a complete, dynamic, and up-to-date user profile [14]. Zhang and Dong proposed a novel effective approach to exploit the relationships among users, queries and resources based on the search engine's log [15].

The rank value indicates the importance of a particular page. A hyperlink to a page counts as a vote of support. Link analysis is a method for determining which pages are good for particular topics based on both the quantity and quality of links pointing to that document. PageRank is a link analysis algorithm, named after Larry Page and used by the Google Internet search engine, that assigns a numerical weighting to each element of a hyperlinked set of documents [16]. hyperlink-induced topic search (HITS) is a link analysis algorithm that developed by Jon Kleinberg. HITS provides a new method of searching the web that returns a list of the most valuable sites on a given topic, plus a list of sites that index the subject. In [17], a comparison among the rankings of results for identical queries retrieved from several search engines is made. The method was based only on the set of URLs that appear in the answer sets of the engines being compared.

3. Optimal Feedback Model

In this section, we design the optimal feedback model of users for the evaluation of the weight of webpage. The explicit feedback is the one in which the user is asked to fill up a feedback form after he has obtained searching results [18]. Since the user's feedback behavior reflects the user's preferences, we need to monitor the response of the user to the search results presented before him. Here is the general definition of the problem.

Definition 1. Assume that $FU = \{V, T, L, M, R\}$ represents the feedback model of user, $V = \{v_1, v_2, \dots, v_i, \dots, v_N\}$ denotes the order that user browses the webpages, $T = \{t_1, t_2, \dots, t_i, \dots, t_N\}$ denotes the time in which the user browses the webpages, $L = \{l_1, l_2, \dots, l_i, \dots, l_N\}$ denotes the clicks of the webpages, $M = \{m_1, m_2, \dots, m_i, \dots, m_N\}$ denotes the behavior that the user evaluates the webpages and $R = \{r_1, r_2, \dots, r_i, \dots, r_N\}$ denotes the user's behavior in responding to the webpages.

If v_i is the k th webpage visited by the user, then the value of v_i is set k . If the i th webpage is not browse by the user at all before the next query is submitted, then we set $v_i = 0$, and the corresponding value of t_i is set 0. One of the mostly used measures for evaluating the web usage is the number of clicks needed for accessing the target webpage [19, 20]. The click frequency can be easily derived by scanning the user sessions, and the initial value of l_i is 0; if the user clicks the v_i one time, then set $l_i = l_i + 1$. M indicates the level of importance that webpage holds for the specified query; according to the rules and strategies for evaluation of difference, the value of m_i is not the same, $i = 1, 2, 3$. The initial value of r_i is 0; if the

user participates in the discussion of the i th webpage, then the value of r_i is set 1.

When the feedback model of the user is complete, we propose to define the weight w_i of the i th webpage.

Definition 2. Assume that w_i represents the weight of the i th webpage; w_i represents the importance of the i th webpage and can be formulated as

$$w_i = \lambda_1 \times \frac{1}{v_i + 1} + \lambda_2 \times \frac{t_i}{t_{\max}^i} + \lambda_3 \times l_i + \lambda_4 \times m_i + \lambda_5 \times r_i, \quad (1)$$

where t_{\max}^i denotes the maximum time that the user is expected to spend examining the i th webpage, in which $\sum_{i=1}^5 \lambda_i = 1$, λ_i represents each of the five factors of the feedback vector. According to the user's preferences and practice considerations, the user can set different values for each of the five factors in the feedback vector. From (1), it is not difficult to see that the more time the user spent in browsing through the documents, the more important they must be for him. If the user evaluates the documents or responding to them, they must be having the significance for the user. Therefore, this paper combines of the above five components by simply taking their weighted sum and gives the overall importance of webpages.

4. The Application of PSCOMA

4.1. Particle Swarm Optimization and Chaos Optimization. Particle swarm optimization is an evolutionary computation technique based on swarm intelligence optimization algorithm inspired by the social behavior of birds flocking for food, which was first introduced to optimize various continuous functions by Kennedy and Eberhart. It is computationally effective, has fast convergence, and is easier to implement when compared with other mathematical and evolutionary algorithms while only a few parameters need to be adjusted.

Chaos is a universal phenomenon in many nonlinear systems. Chaos optimization can escape from local minima more easily compared with other stochastic optimization algorithms that escape from local minimum by accepting some wrong solutions according to a certain probability [21]. Chaos optimization can be within a certain range according to their own laws without repetition through all states. Recently, chaos optimization and PSO have been combined in different application fields for different purposes. Some of the works have intended to show the chaotic behaviors in PSO process [22].

4.2. The Basic Idea of PSCOMA. This paper proposes an efficient particle swarm chaos optimization mining algorithm that attempts to balance exploration and exploitation. The PSCOMA makes full use of the strong global search ability of PSO and the strong local search ability of chaos optimization to obtain high quality solution. PSCOMA uses the properties of ergodicity, stochastic property, and regularity of chaos to lead particles exploration.

The webpages that have higher weight are selected to compose an initial population that is then analyzed by PSCOMA. The basic idea of PSCOMA is as follows. The proposed algorithm starts with an initial population of N particles moving around in a D -dimensional search space where each particle vector corresponds to a potential solution of the underlying problem, which is formed by subsets of webpages, starting from the webpages with higher weight that have a high probability of being different from each other. We let the i th particle at the t th iteration be used to evaluate the quality of the particle and represent candidate solution, denoted by $x_t^i = (x_t^{i,1}, x_t^{i,2}, \dots, x_t^{i,D})$, representing a possible solution. The PSCOMA operates using a fitness function that considers the weight of a webpage. Then, all of the particles repeatedly move until a termination condition is satisfied. During each iteration, the particle individual best and swarm's best positions are determined. The particle adjusts its position based on the individual experience and swarm's intelligence. Each particle is further updated using a chaos optimization algorithm. Once the PSCOMA has been run, the best particle individual constitutes the subset of webpages to be returned to the user. When the algorithm terminates, the best particle is returned as a solution.

4.3. Fitness Function. Each particle is assigned a fitness value indicating the merit of the particle. Since all the particles represent candidate feasible solutions, we use (1) to assign a fitness value to each particle.

Definition 3. Assume that w_i represents the weight of the i th webpage that corresponds to a particle, x_t^i denotes the i th particle at the t th iteration, the fitness function of particle x_t^i is given by

$$f(x_t^i) = e^{w_i}. \quad (2)$$

4.4. Formulation of the Movement of the Particle. During the search process, the particle successively adjusts its position and updates its velocity toward the global optimum using two "best" values: p_{best} and g_{best} . The p_{best} represents the best position encountered by itself denoted as $p_{i,j} = (p_{i,1}, p_{i,2}, \dots, p_{i,D})$. The g_{best} represents the best position encountered by any particle in the population denoted as

$$p_g = (p_{g,1}, p_{g,2}, \dots, p_{g,D}). \quad (3)$$

After finding these two best values, the particle updates its velocity and position at the next iteration are calculated according to the following equations:

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1r_1 \left[p_{\text{best}_{i,j}}(t) - x_{i,j}(t) \right] \quad (4)$$

$$+ c_2r_2 \left[g_{\text{best}_i}(t) - x_{i,j}(t) \right],$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), \quad (5)$$

where c_1 and c_2 are cognitive coefficients, r_1 and r_2 are random real numbers drawn from the interval $(0, 1)$, w is called inertia weight, $0 \leq c_1, c_2 \leq 2$, $0.8 \leq w_1, w_2 \leq 1.2$, and

$$r \in \{r_1, r_2\}, \quad r \sim U(0, 1), \quad j = 1, 2, \dots, D. \quad (6)$$

During the search process for optimum values, it is possible for a particle to escape its search space in any of the dimensions. In each iteration process, when the fitness value of each particle tends to converge or local optimum, it will lead to inertia weight increase; while the fitness value of each particle scattered, it will be easy to make the inertia weight decreases. Therefore, in order to maintain the value of inertia weight range in a reasonable range, (4) should make proper adjustments to the specific implementation is as follows:

$$\begin{aligned} v_{i,j}(t+1) &= w' v_{i,j}(t) + c_1 r_1 [p_{\text{best}_{i,j}}(t) - x_{i,j}(t)] \\ &\quad + c_2 r_2 [g_{\text{best}_t}(t) - x_{i,j}(t)], \\ w' &= \begin{cases} w_{\max} - \frac{w \times (f - f_{\min})}{f_{\max} - f_{\min}}, & f \leq \bar{f}, \\ w_{\min} + \frac{w \times (f - f_{\min})}{f_{\max} - f_{\min}}, & f > \bar{f}. \end{cases} \end{aligned} \quad (7)$$

4.5. Chaotic Local Search in PSO. PSCOMA uses chaotic local search methods during the search process; namely, the chaotic map is used to control the value of parameters in the velocity updating equation. The specific implementations are as follows.

Step 1. Each of the individual x_t^i maps to the interval $[0, 1]$, namely,

$$s_t^j = \frac{x_t^j - x_{\min,j}}{x_t^j - x_{\max,j}}, \quad (8)$$

where $x_{\min,j}$ and $x_{\max,j}$ are, respectively, the lower and upper bounds of the j th dimensional variable, $t = 0, i = 1, 2, \dots, n, j = 1, 2, \dots, D$.

Step 2. Use the logistic equation for chaotic iteration so as to get chaotic gene series:

$$s_{t+1}^j = 4 \times s_t^j \times (1 - s_t^j), \quad j = 1, 2, \dots, D. \quad (9)$$

Step 3. Convert the chaos variables s_{t+1}^j into decision variables x_{t+1}^j :

$$x_{t+1}^j = x_{\min,j} + s_{t+1}^j (x_{\max,j} - x_{\min,j}), \quad j = 1, 2, \dots, D. \quad (10)$$

Step 4. Evaluate the new solution on the basis of decision variables x_{t+1}^j . If the new solution is better than the initial solution, the new solution will be as an optimization result of chaos. Otherwise, go to Step 2, $t = t + 1$.

4.6. The Process of PSCOMA. The process of PSCOMA consists of the following seven steps.

Step 1. Data preparation: training, validation, and test sets are represented, respectively.

Step 2. Particle initialization and PSCOMA parameters setting: the webpages that have higher weight are selected to compose an initial population, which N particles moves around in a D -dimensional search space. Set the PSCOMA parameters including the number of iterations MG , the number of particles N , particle dimension D , inertia weight, cognitive coefficients, and $t = 0$.

Step 3. Randomly generate the position and velocity of particles.

Step 4. Evaluate the fitness of each particle and store the current position of each particle and the adaptation degree of each particle in p_{best} and store the best individual fitness value of the position and fitness value of p_{best} in g_{best} .

Step 5. Update the velocity and position of each particle using (4) and (5).

Step 6. Perform the following chaotic local search for the best particles in population and update its p_{best} and g_{best} .

Step 6.1. Each of the individual x_t^i maps to the interval $[0, 1]$; namely,

$$s_t^j = \frac{x_t^j - x_{\min,j}}{x_t^j - x_{\max,j}}, \quad i = 1, 2, \dots, n, \quad j = 1, 2, \dots, D. \quad (11)$$

Step 6.2. Use the logistic equation for chaotic iteration, so as to get chaotic gene series:

$$s_{t+1}^j = 4 \times s_t^j \times (1 - s_t^j), \quad j = 1, 2, \dots, D. \quad (12)$$

Step 6.3. Convert the chaos variables s_{t+1}^j into decision variables x_{t+1}^j :

$$x_{t+1}^j = x_{\min,j} + s_{t+1}^j (x_{\max,j} - x_{\min,j}), \quad j = 1, 2, \dots, D. \quad (13)$$

Step 6.4. Evaluate the new solution on the basis of decision variables x_{t+1}^j . If fitness of the the new individual is larger than the old one, then the new individual (solution) will be an optimization result of chaos. Otherwise, return Step 6.2.

Step 7. Stop condition checking: if $t > MG$, end the training and testing procedure, otherwise go to Step 2 to begin the next iteration, $t = t + 1$.

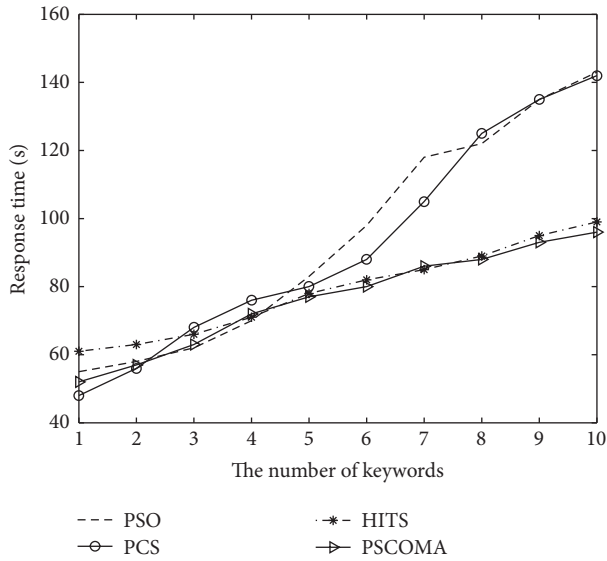


FIGURE 1: The response time in different number of keywords ($m = 10000$).

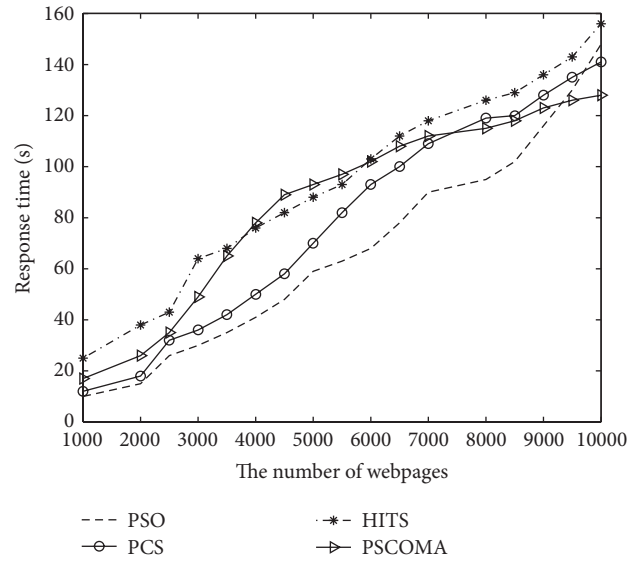


FIGURE 3: The response time in different number of webpages ($n = 5$).

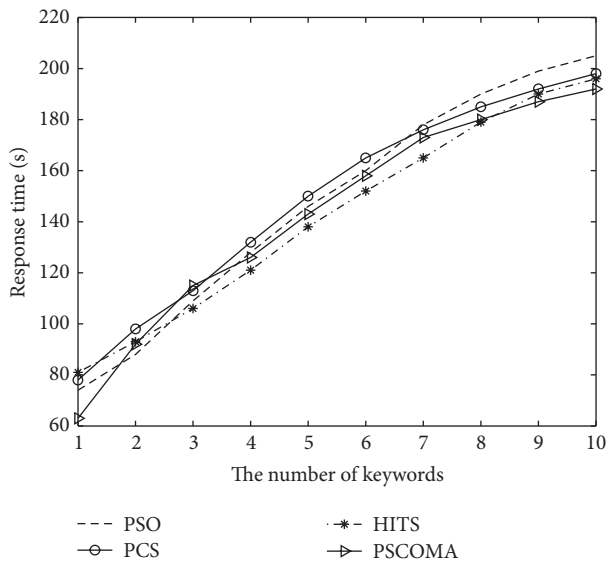


FIGURE 2: The response time in different number of keywords ($m = 20000$).

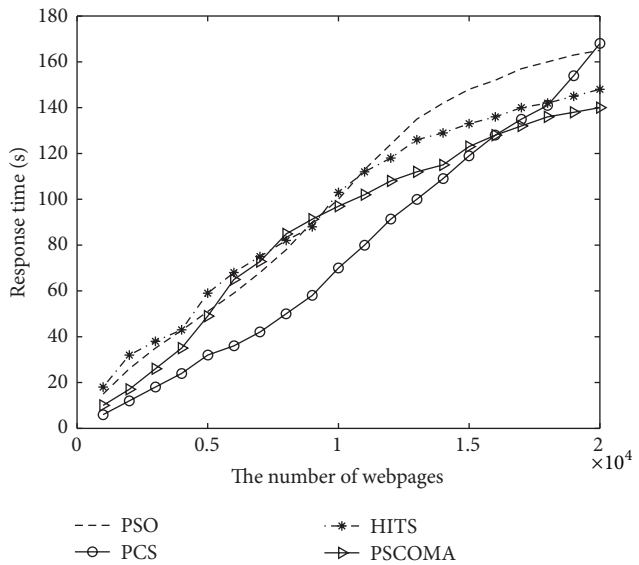


FIGURE 4: The response time in different number of webpages ($n = 6$).

5. Simulation Experiment and Result Analysis

In this section, we present the experimental results which include the algorithm parameter configuration and comparative performances with other algorithms. The platform for conducting the experiments are a PC with Intel Core 2 Duo CPU E6300 processor, 1.86 GHz. All programs are coded in C# language under a Windows NT platform. The numerical results are the means of outcomes from 50 independent runs of the algorithms.

The experimental results compare the PSCOMA with several typical web mining algorithms including the PSO, PCS, and HITS algorithms. We experimented with a few queries

on six popular search engines, namely, AltaVista, Netscape, Excite, Google, Direct Hit, and Yahoo. We denote the number of webpages and keywords in m and n . This paper will be compared from the important aspects of web search such as response time, execution time, precision and recall. Heuristic algorithms have different configuration parameters, which may affect the results. In order to reflect the fairness of algorithms, this paper will take the same configuration

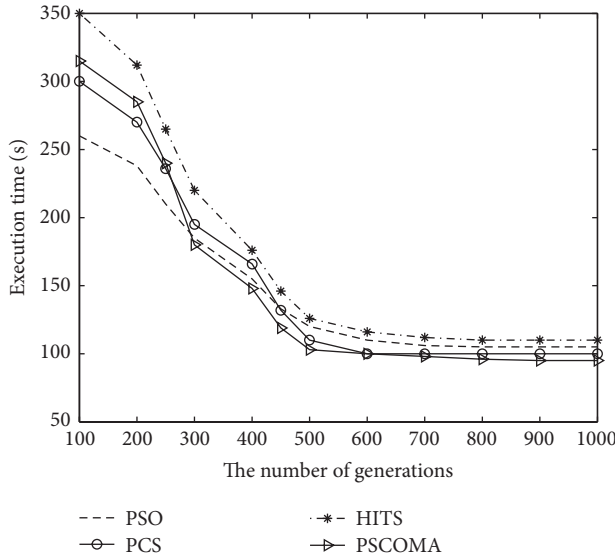


FIGURE 5: The execution time in different number of generations ($m = 10000, n = 5$).

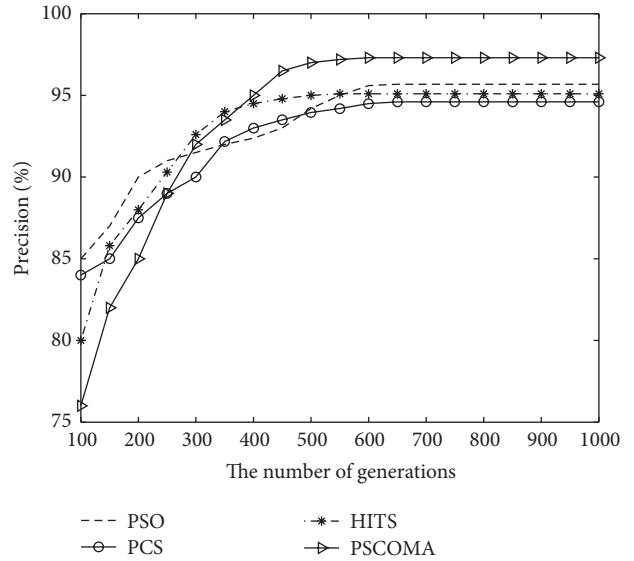


FIGURE 7: The precision in different number of generations ($m = 10000, n = 5$).

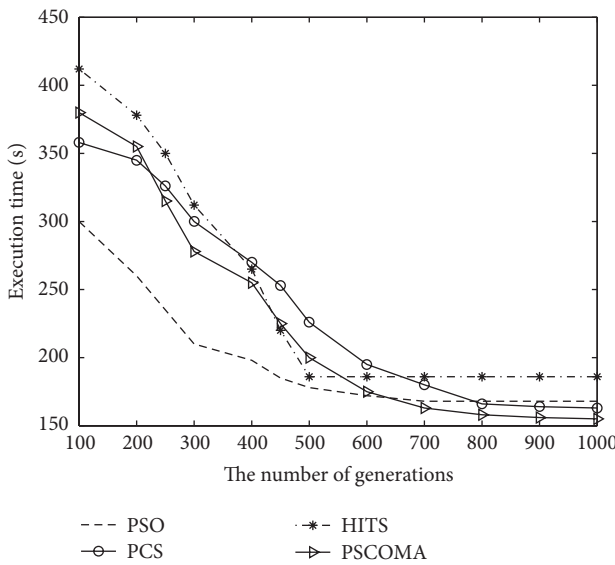


FIGURE 6: The execution time in different number of generations ($m = 20000, n = 6$).

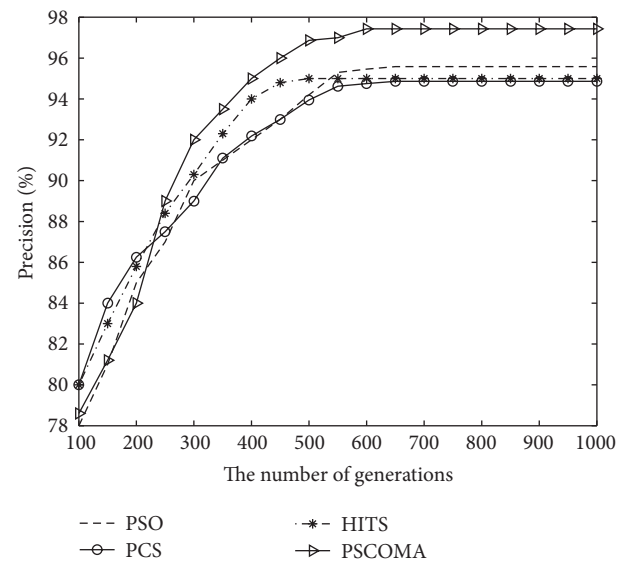


FIGURE 8: The precision in different number of generations ($m = 20000, n = 6$).

parameters in [11]. The specific configuration parameters are as follows:

$$\begin{aligned} N = 40, \quad MG = 1000, \quad r_1 = r_2 = 1, \quad c_1 = c_2 = 2, \\ w_{\max} = 1, \quad w_{\min} = 0.3. \end{aligned} \quad (14)$$

Figures 1 and 2 show the experimental results of the response time in different number of keywords. Figures 3 and 4 show the experimental results of the response time in different number of keywords. Figures 5 and 6 show the execution time in different number of generations. Figures 7 and 8 show the precision in different number of generations. Figures 9 and 10 show the recall in different number of

generations. Figure 11 shows the contrast curve of recall and precision.

As we can see from Figures 1–11, the proposed algorithm PSCOMA outperforms PSO, PCS, and HITS algorithms. In the aspect of response time, PSCOMA is faster than PCS and PSO by 28.6%, indicating that PSCOMA has fast response speed. When increasing the number of keywords, the response time curve of PSCOMA shows to be the most stable. In the aspect of execution time, PSCOMA spends the least time than other algorithms. In addition, when the number of webpages or keywords continually increases, PSCOMA increases the magnitude is the smallest. The simulation results illustrate that the response time and execution

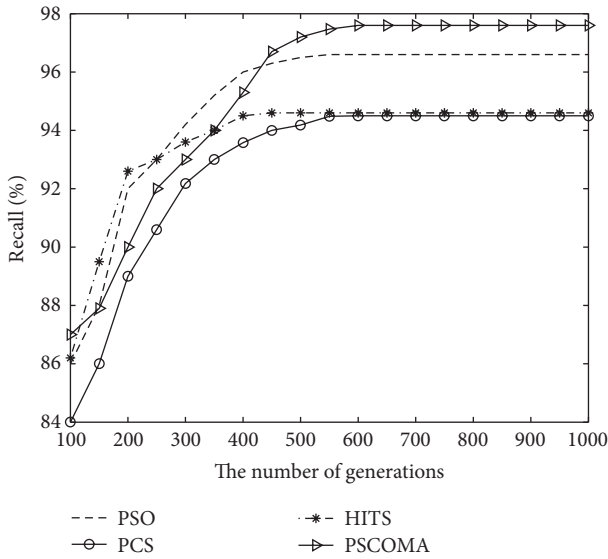


FIGURE 9: The recall in different number of generations ($m = 10000$, $n = 5$).

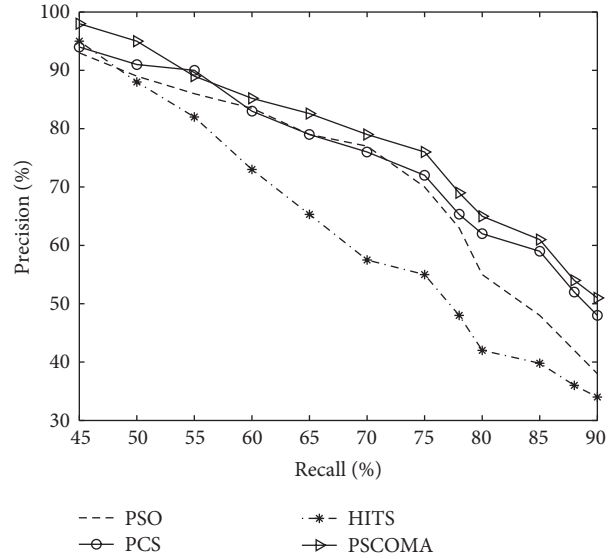


FIGURE 11: The contrast curve of recall and precision ($m = 20000$, $n = 6$).

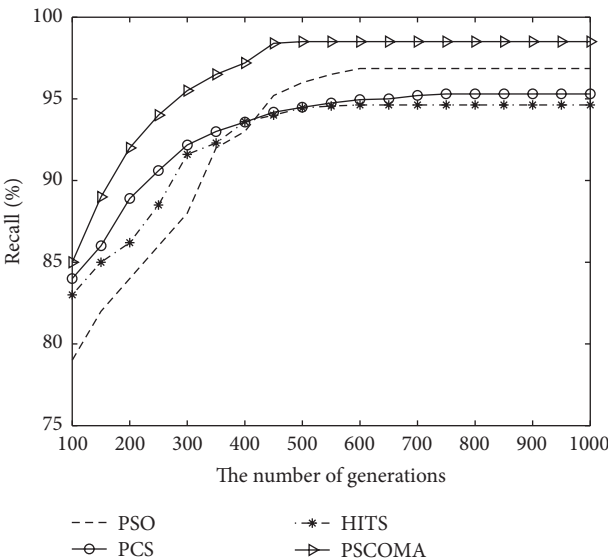


FIGURE 10: The recall in different number of generations ($m = 20000$, $n = 6$).

time of the proposed algorithm are better than those obtained by others. In the aspects of precision and recall, the precision and recall of PSCOMA have performed the best, where the highest precision is close to 97.2% and the highest recall rate is close to 97.8%.

From the experimental results in the aspects of response time, execution time, precision, and recall, we can conclude that PSCOMA is more satisfactory than the PSO, PCS, and HITS algorithms.

6. Conclusions

To prevent the user from being overwhelmed by a large number of redundant and useless or uninteresting information, approaches are needed to provide for data mining. In this paper, we have presented a survey on web mining involving chaos optimization and particle swarm optimization. This paper is the first full use of the strong global search ability of PSO and the strong local search ability of chaos optimization for solving web search and has gained a higher quality solution in the aspects of response time, execution time, precision, and recall. In the future, we will extend the PSCOMA algorithm to other domains of data mining and investigate the possibility of reaching closer optimum by improving chaotic local search.

Acknowledgments

This research work was supported by the Hubei Key Laboratory of Intelligent Wireless Communications (Grant no. IWC2012007) and the Special Fund for Basic Scientific Research of Central Colleges, South-Central University for Nationalities (Grant no. CZY11005). The authors gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

- [1] N. Gandal, "The dynamics of competition in the internet search engine market," *International Journal of Industrial Organization*, vol. 19, no. 7, pp. 1103–1117, 2001.
- [2] Y. Shang and L. Li, "Precision evaluation of search engines," *World Wide Web: Internet and Web Information Systems*, vol. 5, pp. 159–173, 2002.
- [3] M. M. Sufyan Beg and C. P. Ravikumar, "Measuring the Quality of Web Search Results," in *Proceedings of the 6th Joint Conference*

- on *Information Sciences (JCIS '02)*, pp. 324–328, Durham, NC, USA, March 2002.
- [4] M. R. Morris, J. Teevan, and S. Bush, “Enhancing collaborative web search with personalization: groupization, smart splitting, and group hit-highlighting,” in *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '08)*, pp. 481–484, San Diego, Calif, USA, November 2008.
 - [5] J. Sun, H. Zeng, H. Liu, Y. Lu, and Z. Chen, “CubeSVD: a novel approach to personalized web search,” in *Proceedings of the 14th International WorldWide Web Conference*, pp. 382–390, Chiba, Japan, 2005.
 - [6] A. Mahmood, “Dynamic replication of web contents,” *Science in China, Series F*, vol. 50, no. 6, pp. 811–830, 2007.
 - [7] P.-Y. Yin and Y.-M. Guo, “Optimization of multi-criteria website structure based on enhanced tabu search and web usage mining,” *Applied Mathematics and Computation*, vol. 219, pp. 11082–11095, 2013.
 - [8] N. Turajlić and S. Nešković, “Variable Neighborhood Search and Tabu Search for the Web Service Selection Problem,” *Electronic Notes in Discrete Mathematics*, vol. 39, pp. 177–184, 2012.
 - [9] L.-C. Chen, C.-J. Luh, and C. Jou, “Generating page clippings from web search results using a dynamically terminated genetic algorithm,” *Information Systems*, vol. 30, no. 4, pp. 299–316, 2005.
 - [10] A. Spink, B. J. Jansen, C. Blakely, and S. Koshman, “A study of results overlap and uniqueness among major Web search engines,” *Information Processing and Management*, vol. 42, no. 5, pp. 1379–1391, 2006.
 - [11] A. Mahmood, “Replicating web contents using a hybrid particle swarm optimization,” *Information Processing and Management*, vol. 46, no. 2, pp. 170–179, 2010.
 - [12] M. R. Herrera, E. S. de Moura, M. Cristo, T. P. Silva, and A. S. da Silva, “Exploring features for the automatic identification of user goals in web search,” *Information Processing and Management*, vol. 46, no. 2, pp. 131–142, 2010.
 - [13] J. Ling and P. Van Schaik, “The influence of font type and line length on visual search and information retrieval in web pages,” *International Journal of Human Computer Studies*, vol. 64, no. 5, pp. 395–404, 2006.
 - [14] I. F. Moawad, H. Talha, E. Hosny et al., “Agent-based web search personalization approach using dynamic user profile,” *Egyptian Informatics Journal*, vol. 13, pp. 191–198, 2012.
 - [15] D. Zhang and Y. Dong, “A novel Web usage mining approach for search engines,” *Computer Networks*, vol. 39, no. 3, pp. 303–310, 2002.
 - [16] A. H. Keyhanipour, M. Piroozmand, and K. Badie, “A GP-adaptive web ranking discovery framework based on combinative content and context features,” *Journal of Informetrics*, vol. 3, no. 1, pp. 78–89, 2009.
 - [17] J. Bar-Ilan, “Comparing rankings of search results on the Web,” *Information Processing and Management*, vol. 41, no. 6, pp. 1511–1519, 2005.
 - [18] Y. Lv, L. Sun, J. Zhang, J.-Y. Nie, W. Chen, and W. Zhang, “An iterative implicit feedback approach to personalized search,” in *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING/ACL '06)*, pp. 585–592, Sydney, Australia, July 2006.
 - [19] C.-C. Lin, “Optimal Web site reorganization considering information overload and search depth,” *European Journal of Operational Research*, vol. 173, no. 3, pp. 839–848, 2006.
 - [20] H. Qahri Saremi, B. Abedin, and A. Meimand Kermani, “Website structure improvement: quadratic assignment problem approach and ant colony meta-heuristic technique,” *Applied Mathematics and Computation*, vol. 195, no. 1, pp. 285–298, 2008.
 - [21] C.-G. Fei and Z.-Z. Han, “A novel chaotic optimization algorithm and its applications,” *Journal of Harbin Institute of Technology (New Series)*, vol. 17, no. 2, pp. 254–258, 2010.
 - [22] B. Alatas and E. Akin, “Chaotically encoded particle swarm optimization algorithm and its applications,” *Chaos, Solitons and Fractals*, vol. 41, no. 2, pp. 939–950, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

